

Proyecto Fin de Grado

Grado en Ingeniería de las Tecnologías Industriales

Intensificación en Electricidad

Dimensionamiento De Parques Eólicos Basado En Métodos Probabilísticos

Autor: Pedro Antonio Flores Torné

Tutor: Ángel Luis Trigo García

Departamento de Ingeniería Eléctrica
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2014



Proyecto Fin de Grado
Grado en Ingeniería de las Tecnologías Industriales
Intensificación en Electricidad

Dimensionamiento De Parques Eólicos Basado En Métodos Probabilísticos

Autor:

Pedro Antonio Flores Torné

Tutor:

Ángel Luis Trigo García

Profesor Contratado Doctor

Departamento de Ingeniería Eléctrica
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2014

Índice

1. INTRODUCCIÓN -----	6
2. OBJETIVOS Y RESUMEN -----	8
3. GENERACIÓN EN RÉGIMEN ESPECIAL EN ESPAÑA -----	10
4. NORMATIVA Y CRITERIOS -----	12
5. FORMULACIÓN DEL PROBLEMA -----	13
5.1 ANÁLISIS PROBABILÍSTICO EN ESTADO NORMAL DE EXPLOTACIÓN-----	13
5.1.1 Análisis lineal en continua. Planteamiento.-----	13
5.1.2 Análisis probabilístico-----	16
5.2 ANÁLISIS PROBABILÍSTICO EN ESTADO DE CONTINGENCIA SIMPLE-----	18
5.2.1 Análisis lineal en continua-----	18
5.2.2 Formulación probabilística-----	21
5.3 DETERMINACIÓN PROBABILÍSTICA DE LA CAPACIDAD EN ESTADO DE CONTINGENCIA SIMPLE A PARTIR DE LA FORMULACIÓN EN ESTADO NORMAL DE EXPLOTACIÓN-----	23
6. CASO DE ESTUDIO -----	25
7. OBTENCIÓN DE DATOS. SIMULACIÓN Y TRATAMIENTO. -----	27
7.1 SIMULACIÓN-----	27
7.1.1 Obtención de P_{ij0} -----	27
7.1.2 Obtención de p_{ijk} -----	28
7.1.3 Obtención de $P_{ijm\acute{a}x}$ -----	31
7.2 TRATAMIENTO DE DATOS-----	31
7.3 JUSTIFICACIÓN DEL MODO DE OBTENCIÓN DE P_{ijmn} -----	31
7.4 FACTORES DE REPARTO DE LA GENERACIÓN γ_{gs} -----	32
8. OBTENCIÓN DE ΔP_k EN MATLAB -----	35
8.1 DETERMINACIÓN DE LOS FACTORES A_{ijk} MEDIANTE EL PROGRAMA “Mat_A” DE MATLAB-----	35
8.2 OBTENCIÓN DE LAS FUNCIONES DE DISTRIBUCIÓN DE PROBABILIDAD-----	37
8.2.1 Conversión de los flujos P_{ij0} en una función de densidad de probabilidad: $F_{P_{ij}} =$ $P_x = X$ -----	37
8.2.2 Cálculo de la Función de Distribución-----	40
8.3 CÁLCULO DEL FLUJO $P_{ij}\Delta P_k$ FIJADA LA PROBABILIDAD $F(P_{ij0})$ -----	44
8.4 INVERSIÓN DE FLUJO-----	46
9. GENERALIZACIÓN PARA LAS 181 CONTINGENCIAS -----	47
10. ANÁLISIS DE RESULTADOS -----	48
11. CONCLUSIONES Y LÍNEAS FUTURAS DE TRABAJO -----	59
12. BIBLIOGRAFÍA -----	62

ANEXO 1: SIMULACIONES PARA LA OBTENCIÓN DE DATOS	63
1. DEFINICIÓN DE FUNCIONES QUE SE UTILIZARÁN EN PYTHON PARA LA OBTENCIÓN DE DATOS	63
2. PROGRAMAS QUE UTILIZAN LAS FUNCIONES ANTERIORES	67
ANEXO 2: TRATAMIENTO DE LOS DATOS OBTENIDOS	71
1. DEFINICIÓN DE FUNCIONES QUE SE UTILIZARÁN EN PYTHON PARA EL TRATAMIENTO DE DATOS	71
2. PROGRAMA N-10 QUE UTILIZA LAS FUNCIONES ANTERIORES	72
ANEXO 3: CÁLCULO DE ΔP_k CON MATLAB PARA UN ESCENARIO CONCRETO	73
1. DEFINICIÓN DE FUNCIONES QUE SE UTILIZARÁN EN MATLAB PARA LA OBTENCIÓN DE LA CONSTANTE A_{ijk}	75
2. DEFINICIÓN DE FUNCIONES QUE SE UTILIZARÁN EN MATLAB PARA LA OBTENCIÓN DE LOS ΔP_k	78
ANEXO 4: FUNCIONES DE MATLAB PARA CALCULAR LOS 181 ESCENARIOS	84
1. DEFINICIÓN DE LAS FUNCIONES NECESARIAS PARA REALIZAR LOS CÁLCULOS DESCRITOS EN EL PUNTO 8 PARA LOS 181 ESCENARIOS.	84
2. DEFINICIÓN DE LAS FUNCIONES NECESARIAS PARA ANALIZAR LOS DATOS OBTENIDOS	86

1. INTRODUCCIÓN

Los estudios generales de evacuación en estado de contingencia simple, se basan en escenarios muy concretos en los que las condiciones a las que se somete a la red son, a priori, las más exigentes. Estas condiciones se dan tan solo en dos momentos del año (en dos horas concretas), la punta y el valle, obviando escenarios intermedios que pueden ser más restrictivos.

Se plantea la posibilidad de relajar las exigencias que se imponen a la conexión de las plantas de energías renovables, permitiendo una mayor potencia instalada aunque ésta no pueda estar funcionando al 100% durante todas las horas de año. Lo que se busca es, una vez establecida la fracción de horas que la planta no podrá funcionar al 100%, obtener cuánto se puede mayorar la potencia instalada de los parques.

La cuestión anteriormente descrita, fue introducida en dos Proyectos Fin de Máster:

1. Limitación de la generación en Régimen Especial en una red de distribución a partir de funciones de densidad. En adelante PFMN.
2. Análisis probabilístico de la capacidad de evacuación eólica en estado de contingencia simple. En adelante PFMN-1.

El problema de la formulación propuesta en el PFMN-1 fue el fuerte acoplamiento entre variables aleatorias, problema sin solución actualmente por lo que se optó por hacer distintas aproximaciones.

El presente proyecto trata un método basado en el PFMN en el que se obtienen los resultados buscados por el PFMN-1 pero sin el problema del acoplamiento.

Para comenzar se expondrá claramente el objetivo del proyecto.

A continuación, se realizará un análisis de la generación en régimen especial en España debido a la gran importancia que ha adquirido la misma en los últimos años.

En el capítulo 5 se presentará el problema y las fórmulas necesarias para solucionarlo.

En el capítulo 6 se presentarán los distintos datos de carga y generación para las 8784 horas del año y la red de prueba, que consta de 118 nudos y 181 ramas, sobre la que se harán las simulaciones y cálculos pertinentes.

Se continuará explicando todo lo referente a la obtención y tratamiento de los datos para su posterior uso.

En el capítulo 8 se detallarán los cálculos llevados a cabo en MATLAB para la obtención de los ΔP_k .

Se generalizarán dichos cálculos para las 181 contingencias estudiadas.

Para terminar se realizará un análisis de resultados.

Además de lo anterior, se ha incluido un epígrafe correspondiente a la bibliografía donde se listan todos los libros, manuales, artículos y páginas webs que se han consultado para poder llevar a cabo la tarea encomendada.

Para terminar, los anexos, donde se detallan los programas utilizados para la simulación y tratamiento de datos y para la obtención de los resultados obtenidos en el presente estudio.

2. OBJETIVOS Y RESUMEN

El proyecto trata de determinar la capacidad de evacuación nodal a través de métodos probabilístico para un sistema en estado de contingencia simple. Para ello se desarrolla la formulación matemática y la metodología seguida.

Se consigue entonces conocer la cantidad de energía que se es capaz de entregar en un nudo sin tener que realizar estudios de evacuación de generación en régimen especial para cada uno de los escenarios posibles en un año, ya que los estudios basados solamente en los escenarios de punta y valle pueden llevar a ignorar la presencia de situaciones más restrictivas por presentar éstos una demanda media.

Se podrá ver cómo a partir de las funciones de densidad probabilística, obtenidas a su vez de las curvas cronológicas, se pone de manifiesto el fenómeno nombrado en el párrafo anterior.

La metodología para el caso N, será entonces generalizada para las N-1 contingencias del sistema. Éste estudio, se basará en la simulación de la red para los 181 escenarios, obteniendo todos los datos necesarios para cada uno de ellos, además de justificarse el porqué del método seguido.

Para la determinación de la capacidad de evacuación, se tendrán en cuenta la normativa y criterios vigentes los cuales se presentarán más adelante.

Además, el método estadístico expuesto permitirá sobredimensionar la capacidad de inyección nodal ya que se asumirá una probabilidad X de horas al año en las que el sistema se verá sobrecargado, estando el resto del año en estado normal de explotación.

Para la obtención de datos se van a usar dos programas, Python y PSS/E. El primero se usa para realizar el trabajo repetitivo y el segundo se encarga de realizar las simulaciones requeridas por el primero.

Python es un lenguaje de programación de alto nivel ampliamente utilizado en la actualidad por ser software libre y por contar con una enorme cantidad de librerías disponibles.

PSS/E o POWER SYSTEM SIMULATOR for ENGINEERING es un programa de simulación eléctrica desarrollado por SIEMENS para el análisis de Sistemas Eléctricos de Potencia. Éste, está reconocido tanto a nivel nacional (es el estándar en transporte) como internacional.

El trabajo conjunto de estos dos programas permitirá por un lado realizar los repartos de carga necesarios, obtener los factores de distribución de nudos y determinar el "RateA" de las líneas y por otro ordenar dichos datos para ser tratados posteriormente en MATLAB.

Para realizar los cálculos del método probabilístico, se partirá de una serie de scripts en MATLAB que, a partir de los datos obtenidos con las aplicaciones anteriores, obtienen los ΔP por cada nudo para un porcentaje determinado, mediante A_{ij}^k , P_{max} y P_x .

3. GENERACIÓN EN RÉGIMEN ESPECIAL EN ESPAÑA

La generación en régimen especial está cobrando cada vez un papel más importante tanto a nivel nacional como internacional debido a las inversiones realizadas por gobiernos y empresas. Ello conlleva que la potencia instalada de la misma represente un porcentaje significativo de la generación total además de aportar buena parte de la potencia demandada.

La Energía eólica adquirida ha ido evolucionando de forma creciente los últimos años pasando de los 37.800 GWh del año 2009 a los más de 54.300 GWh de 2013, siendo este último el máximo histórico. En la figura 1 se puede observar esta tendencia.

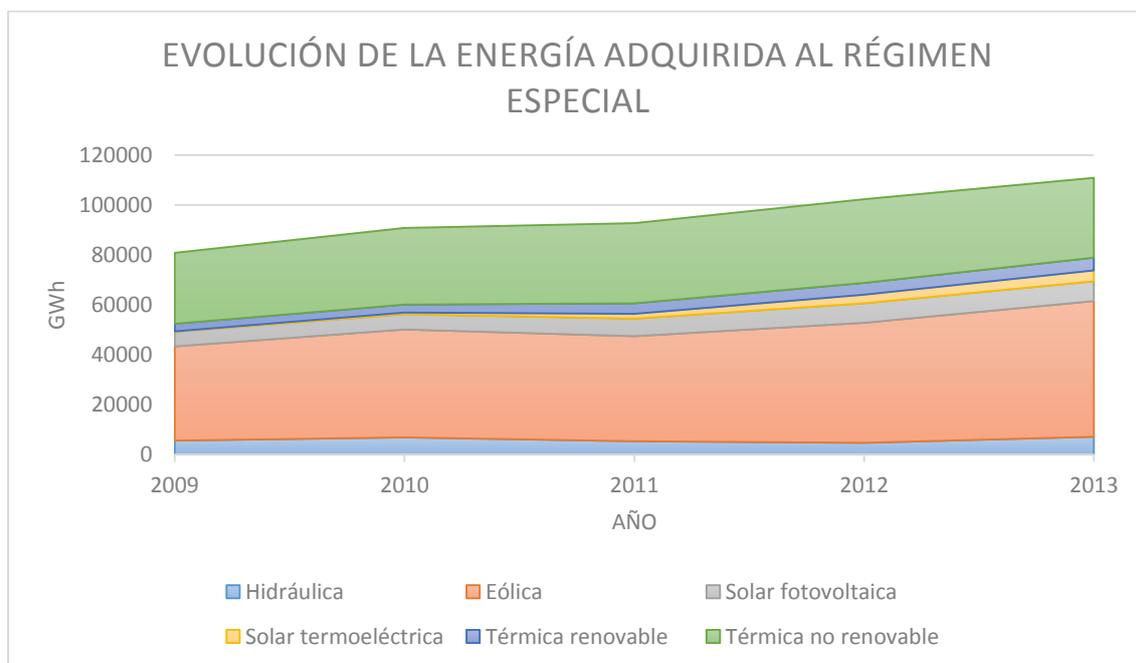


Figura 1: Gráfico de la evolución de la energía adquirida al Régimen Especial

En 2013, se alcanzaron los 32.600 MW de potencia instalada en energías renovables, de los cuales 22.800 MW se correspondieron con la potencia eólica instalada¹. En la figura 2 se puede observar con claridad este hecho.

¹ Fuente: 2013 *El Sistema Eléctrico Español*, informe anual elaborado por REE.

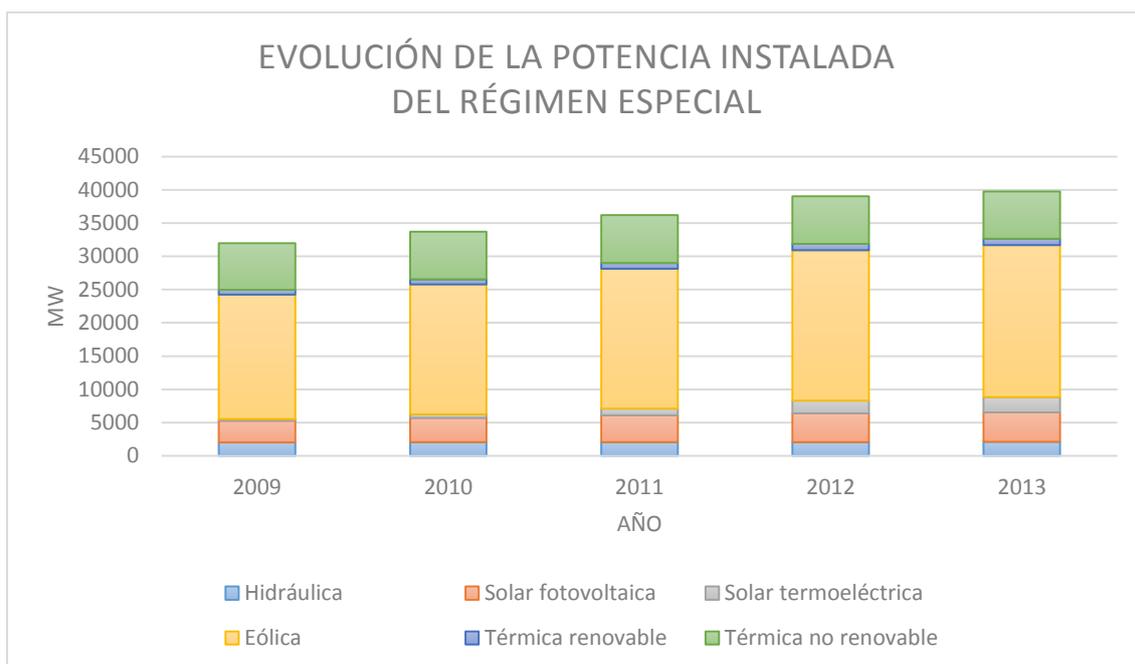


Figura 2: Gráfico de la evolución de la potencia instalada del Régimen Especial

	Hidráulica	Solar fotovoltaica	Solar termoeléctrica	Eólica	Térmica renovable	Térmica no renovable
<i>2009</i>	2022	3250	232	18723	741	7001
<i>2010</i>	2036	3654	532	19569	780	7124
<i>2011</i>	2042	4057	999	21026	884	7200
<i>2012</i>	2042	4320	1950	22617	970	7160
<i>2013</i>	2102	4422	2300	22854	975	7089

Tabla 1: Evolución de la potencia instalada del Régimen Especial

En la tabla 1 se ve como el crecimiento ha sido constante en los últimos años.

Todos estos datos ponen de manifiesto la importancia que la energía en régimen especial tiene en la gestión y planificación de sistemas eléctricos como el español (peninsular).

4. NORMATIVA Y CRITERIOS

La legislación actual (Ley 24/2013 art. 33 y RD1955/2000 art. 60), por el que se regulan las actividades de transporte, distribución, comercialización, suministro y procedimientos de autorización de instalaciones de energía eléctrica, establece el derecho de acceso a las redes de distribución eléctrica por parte de productores y consumidores y establece claramente que este derecho únicamente puede ser restringido en caso de falta de capacidad, la cual se justifica exclusivamente en base a criterios de seguridad y calidad del suministro.

El modelo de fiabilidad más ampliamente aceptado en el sector es el conocido como análisis de contingencias N-1, que consiste en desconectar un elemento cualquiera del sistema (línea/trafo o generador) y verificar que el sistema se mantiene funcionando en un estado normal. Este criterio, es legalmente aplicable en España sólo a la red de Transporte y no a la red de distribución. Sin embargo, será el aplicado en este trabajo ya que el objetivo del mismo es el dimensionado de parques eólicos mediante métodos probabilísticos, hecho que no requiere que se obtengan resultados en tiempo real, por lo que el gran número de horas empleado para las simulaciones y cálculos no serán un problema.

5. FORMULACIÓN DEL PROBLEMA

Como se ha indicado anteriormente, se tratará de calcular la capacidad de evacuación máxima en un nudo con un determinado umbral de confianza en estado de contingencia simple.

Se empezará desarrollando la formulación propuesta en el PFMN que trata de calcular la capacidad antes mencionada para un caso en estado N.

A continuación se formulará el método planteado en el PFMN-1 para analizar dicha capacidad pero en estado de contingencia simple N-1. Dicho método se basa en aproximaciones debido al fuerte acoplamiento entre variables probabilísticas, problema que no tiene solución.

Para terminar, se propondrá un método alternativo que consistirá en analizar todas las redes simulando cada una de ellas por separado.

5.1 ANÁLISIS PROBABILÍSTICO EN ESTADO NORMAL DE EXPLOTACIÓN

5.1.1 Análisis lineal en continua. Planteamiento.

Se pretende calcular la inyección de un generador en régimen especial en un nudo de una red. Para mantener el equilibrio del sistema, los demás generadores tendrán que disminuir su producción para compensar esta nueva inyección, hecho que se puede expresar mediante la fórmula:

$$\Delta P_k = \sum_{s=1}^n \Delta P_{g_s}$$

Se supondrá en buena aproximación que cada generador en régimen ordinario ve disminuida su producción de manera proporcional a la potencia que éste genera respecto al total de la producción de todos los generadores del sistema. Así pues, la variación de la producción en el generador en régimen ordinario del nudo s al inyectar ΔP_k en el nudo k , será:

$$\Delta P_{g_s} = \frac{P_{g_s}^0}{\sum_{s=0}^n P_{g_s}^0} \Delta P_k = \gamma_{g_s}^k \Delta P_k \quad (1)$$

Donde γ_{g_s} son los Factores de reparto de la Generación.

Bajo estas premisas, se puede establecer que el flujo esperado a través de una línea existente entre los nudos $i-j$, debida al incremento de inyección ΔP_k en el nudo k , será igual a la suma del flujo existente antes de la inyección (caso base), el incremento del mismo calculado a partir de los factores de distribución ρ_{ij}^k y el decremento del mismo calculado a partir de los factores de generación ρ_{ij}^s .

$$P_{ij}^{\Delta P_k} = P_{ij}^0 + \rho_{ij}^k \Delta P_k + \sum_{s=0}^n \rho_{ij}^s \Delta P_{g_s} \quad (2)$$

Donde:

- el Factor de Distribución ρ_{ij}^k se define como la variación del flujo a través de la línea existente entre los nudos $i-j$, debida al incremento de inyección en el nudo k .
- el Factor de distribución ρ_{ij}^s se define como la variación del flujo a través de la línea existente entre los nudos $i-j$, debida al decremento de la producción en el generador en régimen ordinario del nudo s .

Sustituyendo (1) en (2) se obtiene:

$$P_{ij}^{\Delta P_k} = P_{ij}^0 + \rho_{ij}^k \Delta P_k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \Delta P_k$$

Y reagrupando términos queda:

$$P_{ij}^{\Delta P_k} = P_{ij}^0 + \rho_{ij}^k \Delta P_k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \Delta P_k = P_{ij}^0 + \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \right) \Delta P_k \quad (3)$$

Llamando A_{ij}^k a la constante:

$$\left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \right) = A_{ij}^k (cte)$$

Ésta, sólo depende de parámetros topológicos de la red y de los factores de reparto de la generación ordinaria. Se puede así expresar la potencia por la rama $i-j$ cuando se inyecta en el nudo k una potencia adicional ΔP_k como:

$$P_{ij}^{\Delta P_k} = P_{ij}^0 + A_{ij}^k \cdot \Delta P_k$$

Y puesto que el objetivo es calcular la inyección máxima en el nudo k sin incurrir en situación de riesgo, entonces el flujo a través de una línea existente entre los nudos $i-j$ debida al incremento de inyección ΔP_k en el nudo k ha de ser inferior a la capacidad nominal máxima de dicha línea $P_{ij}^{máx}$.

Por tanto:

$$P_{ij}^{\Delta P_k} = P_{ij}^0 + \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \right) \Delta P_k = P_{ij}^0 + A_{ij}^k \cdot \Delta P_k \leq P_{ij}^{máx}$$

O lo que es lo mismo, la inyección en el nudo k ha de ser:

$$\Delta P_k \leq \frac{P_{ij}^{máx} - P_{ij}^0}{A_{ij}^k}, \forall \text{ línea } i - j$$

Luego para calcular la inyección máxima admisible ΔP_k en el nudo k se necesita conocer:

- El flujo de potencia por cada línea antes de la conexión de la generación en régimen especial en estado de contingencia simple, es decir, el flujo de potencia por cada línea en el Caso PSS/E en estudio cuando se desconecta una rama (IEEE118 Test Case) P_{ij}^0 . Se obtendrá en el apartado 7.1.1
- La capacidad máxima de cada línea $P_{ij}^{máx}$: Parámetro conocido = RATEs del Caso PSS/E en estudio (IEEE118 Test Case) de todas las líneas del sistema. Se calcularán en la sección 7.1.3
- La constante A_{ij}^k para el nudo k , líneas $i-j$ del sistema, que viene definida por:

$$A_{ij}^k = \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \right)$$

Que a su vez depende de:

- Los Factores de Generación γ_{g_s} definidos como potencia de cada generador respecto al total de la generada de todos los generadores del sistema.
- Los Factores de Distribución ρ_{ij}^k y ρ_{ij}^s definidos como la variación del flujo a través de la línea existente entre los nudos $i-j$, debida al incremento de inyección en el nudo k .

5.1.2 Análisis probabilístico

Se considera que el flujo de potencia activa por la línea $i-j$ es una variable aleatoria de la que se tiene su Función de Distribución:

$$\mathcal{F}(P_{ij} \leq P_{ij}^{max}) = 1 - X^2$$

El objetivo del análisis probabilístico será determinar la inyección máxima ΔP_k^{max} , en el nudo k , tal que la probabilidad de que el flujo $P_{ij}^{\Delta P_k}$, no rebase la capacidad nominal de dicha rama P_{ij}^{max} , sea igual a un determinado valor $1-X$.

$$\mathcal{F}(P_{ij}^{\Delta P_k} \leq P_{ij}^{max}) = 1 - X$$

El análisis lineal determinista, trabajado en el anterior apartado es fácilmente ampliable al terreno probabilístico. La expresión anteriormente deducida:

² Por el momento se utilizará la notación $1-X$, para mantener coherencia con desarrollos matemáticos de próximos apartados. El valor X está comprendido entre 0 y 1.

$$P_{ij}^{\Delta P_k} = P_{ij}^0 + \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s} \right) \Delta P_k = P_{ij}^0 - A_{ij}^k \cdot \Delta P_k \leq P_{ij}^{m\acute{a}x}$$

Es lineal y puede ser introducida en la función de distribución:

$$\mathcal{F} \left(P_{ij}^0 + \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s} \right) \Delta P_k \leq P_{ij}^{m\acute{a}x} \right) = 1 - X$$

Al ser lineal se puede reescribir la ecuación anterior como:

$$\mathcal{F} \left(P_{ij}^0 \leq P_{ij}^{m\acute{a}x} - \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s} \right) \Delta P_k \right) = 1 - X$$

Luego,

$$\mathcal{F}(P_{ij}^0 \leq P_{ij}^{m\acute{a}x} - A_{ij}^k \cdot \Delta P_k) = 1 - X$$

La expresión anterior equivale a considerar que el límite térmico de la línea i - j es:

$$P_{ij}^{m\acute{a}x} - A_{ij}^k \cdot \Delta P_k$$

Así si se admite que la línea esté saturada p (%) horas al año, se tendrá que:

$$P_x = P_{ij}^{m\acute{a}x} - A_{ij}^k \cdot \Delta P_k$$

Luego:

$$\Delta P_k = \frac{P_{ij}^{m\acute{a}x} - P_x}{A_{ij}^k}$$

Se tiene pues que calcular el ΔP_k respecto a cada una de las ramas $i-j$ del sistema y tomar el mínimo como la potencia evacuable.

De esta forma, para obtener la capacidad de evacuación en un nudo a partir de esta expresión, se deben tener como datos previos:

1. La constante A_{ij}^k . Para lo cual es necesario:
 - (a) Los factores de distribución ρ_{ij}^k
 - (b) Los factores de reparto de la generación γ_{g_s}
2. La función de distribución de probabilidad del flujo por cada línea $i-j$ en estado normal de explotación $\mathcal{F}(P_{ij}^0)$.

5.2 ANÁLISIS PROBABILÍSTICO EN ESTADO DE CONTINGENCIA SIMPLE

A continuación se expone el planteamiento del PFMN-1 para calcular la inyección máxima en el nudo k , sin incurrir en situación de riesgo en el estado N-1.

5.2.1 Análisis lineal en continua

Se considera la siguiente secuencia de eventos:

- Incrementar la generación en el nudo k en un valor ΔP_k (potencia evacuada)
- Desconexión de la línea m-n.

Evidentemente, la secuencia de eventos no influirá en el resultado final, pero considerarla de esta manera facilitará la formulación matemática, ya que todos los factores de distribución se referirán al caso base (misma topología) y por tanto serán constantes.

En primer lugar, la conexión de la nueva generación en el nudo k , originará una variación de flujo por la línea $i-j$ dada por la expresión ya conocida:

$$P_{ij}^k = P_{ij}^0 + \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \right) \Delta P_k$$

En este estado, se produce la desconexión del elemento m-n, que provoca una variación de flujo por el elemento i - j expresada en función de los factores de distribución de:

$$P_{ij}^{mnk} = P_{ij}^k + \rho_{ij}^{mn} \cdot P_{mn}^k \quad (4)$$

Donde,

- P_{ij}^{mnk} , es el flujo final por la rama i - j cuando se inyecta en el nudo k una potencia y la rama m - n se encuentra fuera de servicio.
- P_{ij}^k , es la potencia que circula por la rama i - j cuando se inyecta en el nudo k una potencia, en estado normal de explotación.
- P_{mn}^k , es la potencia que circula por la rama m - n cuando se inyecta en el nudo k una potencia, en estado normal de explotación.
- ρ_{ij}^{mn} , es el factor de distribución de la línea i - j frente a la contingencia m - n . Es decir, representa la variación de flujo por la línea i - j cuando dispara la línea m - n por cada megavatio que circulaba por la línea m - n .

Se puede expresar P_{mn}^k en función del flujo inicial de la línea m - n y de los factores de distribución como:

$$P_{mn}^k = P_{mn}^0 + \left(\rho_{mn}^k + \sum_{s=0}^n \rho_{mn}^s \rho_{g_s}^k \right) \Delta P_k \quad (5)$$

De esta forma sustituyendo (3) y (5) en la fórmula (4) queda:

$$P_{ij}^{mnk} = P_{ij}^0 + \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \right) \Delta P_k + \rho_{ij}^{mn} \cdot \left\{ P_{mn}^0 + \left(\rho_{mn}^k + \sum_{s=0}^n \rho_{mn}^s \rho_{g_s}^k \right) \Delta P_k \right\}$$

Reordenando términos, y sacando factor común la potencia generada ΔP_k :

$$P_{ij}^{mnk} = P_{ij}^0 + \rho_{ij}^{mn} \cdot P_{mn}^0 + \left\{ \rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k + \rho_{ij}^{mn} \cdot \left(\rho_{mn}^k + \sum_{s=0}^n \rho_{mn}^s \rho_{g_s}^k \right) \right\} \Delta P_k$$

Se define la constante B_{ij}^{mnk} como:

$$B_{ij}^{mnk} = \left\{ \rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k + \rho_{ij}^{mn} \cdot \left(\rho_{mn}^k + \sum_{s=0}^n \rho_{mn}^s \rho_{g_s}^k \right) \right\}$$

Que únicamente depende de la topología del sistema en estado normal de explotación y de los factores de reparto de la generación.

Así, se puede expresar la potencia que circula por la línea $i-j$, cuando falla la línea m y se inyecta en el nudo k una potencia ΔP_k , como:

$$P_{ij}^{mnk} = P_{ij}^0 + \rho_{ij}^{mn} \cdot P_{mn}^0 + B_{ij}^{mnk} \cdot \Delta P_k$$

Y puesto que el objetivo es calcular la inyección máxima en el nudo k sin incurrir en situación de riesgo en estado de contingencia simple N-1, entonces el flujo a través de una línea existente entre los nudos $i-j$, debida al incremento de inyección ΔP_k en el nudo k ha de ser inferior a la capacidad nominal máxima de dicha línea $P_{ij}^{m\acute{a}x}$.

Por tanto:

$$P_{ij}^{mnk} = P_{ij}^0 + \rho_{ij}^{mn} \cdot P_{mn}^0 + B_{ij}^{mnk} \cdot \Delta P_k \leq P_{ij}^{m\acute{a}x}$$

O lo que es lo mismo, la inyección en el nudo k ha de ser:

$$\Delta P_k \leq \frac{P_{ij}^{m\acute{a}x} - P_{ij}^0 - \rho_{ij}^{mn} \cdot P_{mn}^0}{B_{ij}^{mnk}(cte)}$$

Luego para calcular la inyección máxima admisible en el nudo k se necesita conocer:

- La capacidad máxima de cada línea $P_{ij}^{m\acute{a}x}$: Parámetro conocido = RATEs del Caso PSS/E en estudio (IEEE118 Test Case)
- El flujo de potencia por cada línea sin presencia de generación en régimen especial, es decir, el flujo de potencia por cada línea en el Caso PSS/E en

estudio (IEEE118 Test Case) P_{ij}^0 : flujo de potencia por el elemento i-j en el estado inicial.

- El flujo de potencia por cada línea a fallar sin presencia de generación en régimen especial es decir, el flujo de potencia por cada línea en el Caso PSS/E en estudio (IEEE118 Test Case) P_{mn}^0 : flujo de potencia por el elemento m-n en el estado inicial.
- Los Factores de distribución de la línea i-j frente a la contingencia m-n. Es decir, la variación de flujo por la línea i-j cuando dispara la línea m-n: ρ_{ij}^{mn}
- La constante B_{ij}^{mnk} antes definida:

$$B_{ij}^{mnk} = \left\{ \rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k + \rho_{ij}^{mn} \cdot \left(\rho_{mn}^k + \sum_{s=0}^n \rho_{mn}^s \rho_{g_s}^k \right) \right\}$$

Que a su vez depende de:

- Los Factores de Participación $\gamma_{g_s}^k$ definidos como potencia de cada generador respecto al total de la generada de todos los generadores del sistema.
- Los Factores de Distribución ρ_{ij}^k definidos como la variación del flujo a través de la línea existente entre los nudos i – j, debida al incremento de inyección en el nudo k.

5.2.2 Formulación probabilística

Considerando que el flujo de potencia activa por la línea i-j es una variable aleatoria de la que se tiene su Función de Distribución:

$$\mathcal{F}(P_{ij}^0)$$

En este caso, el objetivo del análisis probabilístico es determinar la inyección máxima $\Delta P_k^{m\acute{a}x}$ en el nudo k, tal que la probabilidad de que el flujo P_{ij}^{mnk} por cualquier rama i-j no rebase la capacidad nominal de dicha rama $P_{ij}^{m\acute{a}x}$ en estado de red de contingencia simple N-1, sea igual a un determinado valor (por ejemplo el 90%).

$$\mathcal{F}(P_{ij}^{\Delta P_k} \leq P_{ij}^{m\acute{a}x}) = 90\%$$

El análisis lineal determinista, trabajado en el apartado 5.2.1 es ampliable al terreno probabilístico.

$$\mathcal{F}(P_{ij}^{mnk}) = \mathcal{F}(P_{ij}^0 + \rho_{ij}^{mn} \cdot P_{mn}^0 + B_{ij}^{mnk} \cdot \Delta P_k)$$

Dado que el objetivo es determinar la inyección máxima $\Delta P_k^{m\acute{a}x}$ en el nudo k , tal que la probabilidad de que el flujo P_{ij}^{mnk} por cualquier rama $i-j$ ante la contingencia de la rama $m-n$, no rebase la capacidad nominal de dicha rama $P_{ij}^{m\acute{a}x}$, sea igual a un determinado valor (p ejem: 80%).

$$\mathcal{F}(P_{ij}^{mnk} \leq P_{ij}^{m\acute{a}x}) = 80\%$$

Se tendrá que:

$$\mathcal{F}([P_{ij}^0 + \rho_{ij}^{mn} \cdot P_{mn}^0 + B_{ij}^{mnk} \cdot \Delta P_k] \leq P_{ij}^{m\acute{a}x}) = 80\%$$

Extrayendo de la función de distribución la constante B_{ij}^{mnk} y el incremento de potencia ΔP_k , se tendrá:

$$\mathcal{F}(P_{ij}^0 + \rho_{ij}^{mn} \cdot P_{mn}^0 \leq P_{ij}^{m\acute{a}x} - B_{ij}^{mnk} \cdot \Delta P_k) = 80\%$$

Y para para poder resolver la ecuación anterior se necesita la función de densidad de la suma de variables aleatorias:

$$P_{ij}^0 + \rho_{ij}^{mn} \cdot P_{mn}^0$$

Estas variables son variables aleatorias **dependientes**. La función de dependencia de ambas variables las impone la topología de la red. Es decir, dado un estado de red y una distribución de carga y generación (variables aleatorias independientes) tenemos determinado el flujo por todas y cada una de las líneas del sistema y en particular por P_{ij}^0 y P_{mn}^0 . Así pues y dado que ambas variables son función de las mismas variables independientes, se puede decir que existe una fuerte dependencia entre los valores de ambas variables.

La teoría de la probabilidad no tiene una solución clara para la resolución de este problema. *El teorema central del límite* permite obtener la función de densidad de la suma de variables aleatorias **independientes** a partir de la media y la varianza de cada una de las

poblaciones suma. Sin embargo, existe una fuerte dependencia entre las diferentes variables por lo que este teorema resulta inaplicable.

Llegados a este punto se tiene la función de distribución $\mathcal{F}(P_{ij}^{mn}) = \mathcal{F}(P_{ij}^0 + \rho_{ij}^{mn} \cdot P_{mn}^0)$, por lo que bien se puede determinar la función de distribución de cada P_{ij}^{mn} (lo que en la red de testeo de 181 ramas supone 32.761 funciones de distribución) o bien realizar una aproximación sobre el valor de $\rho_{ij}^{mn} \cdot P_{mn}^0$, de manera que se puedan determinar los ΔP_k únicamente en función de las 181 funciones de distribución correspondientes a los flujos por la ramas en el estado normal de explotación P_{ij}^0 . Luego se suponen distintas aproximaciones para el valor de $\rho_{ij}^{mn} \cdot P_{mn}^0$ y se analizan los ΔP_k para el valor tomado.

5.3 DETERMINACIÓN PROBABILÍSTICA DE LA CAPACIDAD EN ESTADO DE CONTINGENCIA SIMPLE A PARTIR DE LA FORMULACIÓN EN ESTADO NORMAL DE EXPLOTACIÓN

La solución para el caso N-1 de suponer distintas aproximaciones basadas en el valor esperado de P_{ij}^0 para obtener el valor de $\rho_{ij}^{mn} \cdot P_{mn}^0$, conlleva enmascarar soluciones ocultas debido a estar trabajando con probabilidades.

Cuando se realiza un reparto de cargas de un sistema con un estado de generación y cargas determinado, la solución que se obtiene de los flujos por las líneas es única. Entonces, si se quiere conocer el valor real de éstos en un análisis N-1, habrá que realizar 181 repartos de carga correspondientes con los 181 escenarios distintos resultantes del fallo de una rama en cada uno de ellos. Es decir, en vez de realizar un reparto de cargas para el caso N y a partir de ahí calcular para los casos N-1 los valores esperados de los flujos por las líneas, se propone simular 181 veces la misma red para obtener los valores **reales** de P_{ij}^0 para cada contingencia.

Esto, que a priori parece una cuestión sencilla, conlleva un gran coste computacional pues por cada uno de los 181 escenarios distintos hay que simular las 8784 horas del año, cada una de ellas con una generación y carga distinta. Es decir, 181 escenarios x 8784 horas distintas = 1.589.904 repartos de carga a realizar para obtener los P_{ij}^0 reales.

El enfoque será el de utilizar la formulación deducida en el apartado 5.1 referente al estado normal de explotación para analizar los 181 escenarios del estado de contingencia simple. Esto se puede llevar a cabo porque cada escenario es independiente y el problema se puede ver desde el punto de vista del análisis de 181 casos N distintos.

La fórmula a utilizar será entonces:

$$\Delta P_k = \frac{P_{ij}^{m\acute{a}x} - P_x}{A_{ij}^k}$$

Y para calcularla será necesario disponer de los siguientes datos previos:

1. La constante A_{ij}^k , que se calculará en la sección 8.1, y depende de:
 - (a) Los factores de distribución ρ_{ij}^k que se obtendrán en el punto 7.1.2
 - (b) Los factores de reparto de la generación γ_{gs} . Se verán en la sección 7.4
2. El valor de P_x , es la potencia tal que la probabilidad de que el flujo por la línea $i-j$ sea mayor que P_x es igual a X , siendo X el umbral de confianza que se decida. Su cálculo se llevará a cabo en el apartado 8.3

6. CASO DE ESTUDIO

Se desarrollará el trabajo sobre una red estándar de testeo del IEEE (Institute of Electrical and Electronics Engineers) ampliamente utilizada en toda la literatura del sector. De entre las diversas redes que esta organización pone a disposición de estudiantes e investigadores, se elige la red de 118 nudos ya que con un número tal de subestaciones nos permite seleccionar diferentes áreas y modelar situaciones realistas (p. ej. zonas costeras, de interior, industriales, etc.) algo que con un número inferior de nudos no sería posible.

Esta red está basada en una existente de EE.UU., por lo que las tensiones nominales que utiliza son las propias de dicho país: 230 y 138 kV. La red se adaptó en los proyectos PFMN y PFMN-1 a las tensiones más próximas normalizadas en España: 220 y 132kV, por lo que no habrá que realizar ninguna modificación sobre la misma. Se muestra a continuación el esquema unifilar del sistema empleado:

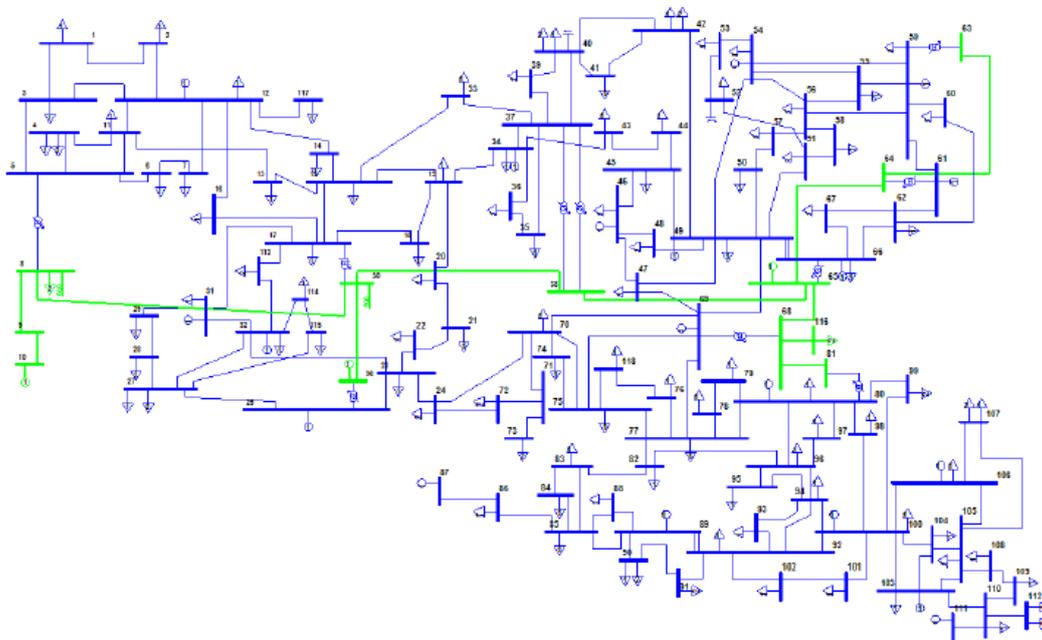


Figura 3: Esquema Unifilar red de 118 nudos IEEE

Los colores normalizados en el esquema anterior representan los siguientes niveles de tensión: Azul para 132 kV, Verde para 220 kV.

Se pretende que el sistema anterior represente a pequeña escala una red similar a la de la zona Sur de España.

Además, se partirá de dos archivos:

1. Archivo de cargas
2. Archivo de generación

Ambos con datos distintos para cada una de las 8784 horas de un año. Éstos, fueron calculados en los proyectos anteriormente mencionados, PFMN y PFMN-1, tomando 118 curvas de demanda diferentes, cada una de ellas a su vez construidas tomando como base curvas tipo reales que se le asignaron a los diferentes nudos del sistema, generando zonas en el mismo de diferente comportamiento.

Se generaron siguiendo los siguientes pasos:

1. **Se partió de un catálogo de curvas.** Dicho catálogo contaba con toda la tipología de curvas de demanda que se da en la realidad como por ejemplo:
 - a. Curva tipo Bombeo
 - b. Consumo doméstico (tipología plano). Predominio sector servicio
 - c. Curva tipo Rural interior
 - d. Tipo Costa
 - e. Curva industrial

Ésta biblioteca de curvas de carga, era similar a las encontradas en subestaciones de demanda reales. A continuación se procedió a la **asignación de dichas curvas de demanda a los 118 nudos.**

2. **Se calcularon las puntas locales.**
3. **Se calcularon la Punta y Valle simultáneos.**
4. **Se continuó calculando la demanda simultánea para una Fecha/Hora.**
5. **Seguidamente se hallaron las desviaciones entre Puntas Locales y Simultáneas.** Dichas desviaciones son propias de redes con zonas bien diferenciadas de demanda, como es el caso de la red eléctrica andaluza, donde la punta simultánea de todo el territorio no tiene por qué coincidir con puntas locales en zonas costeras o de interior.
6. **Se generaron los escenarios de demanda “.csv” .**
7. **Para terminar se distribuyeron las zonas de demanda.** Para ello se utilizaron curvas de demanda anuales semejantes a las encontradas en subestaciones reales, que se distribuyeron sobre el “Caso PSS/E ajustado” (IEEE118 Test Case) de manera que se simularan las condiciones eléctricas reales encontradas típicamente en una Red de Transporte y Distribución.

7. OBTENCIÓN DE DATOS. SIMULACIÓN Y TRATAMIENTO.

Como se comentó anteriormente, para la simulación y tratamiento de datos se utilizarán PSS/E y Python respectivamente.

Para ello, PSS/E cuenta con una API (Application Program Interface) que externaliza todas las funciones del programa para ser usadas por un lenguaje de programación como es Python.

7.1 SIMULACIÓN

Los datos necesarios para obtener los ΔP_k son:

1. Flujos por las 180 líneas para las 8.784 horas por cada una de las 181 contingencias (180 ramas x 8784 horas x 181 contingencias = 286.182.720 P_{ij}^0).
2. Factores de distribución de nudos para las 181 contingencias (181 contingencias x 180 ramas x 118 nudos = 3.844.440 ρ_{ij}^k)
3. La capacidad máxima de cada rama (líneas y transformadores) del sistema (nº de ramas = 181 $P_{ij}^{máx}$).

7.1.1 Obtención de P_{ij}^0

- Se copian las cargas y la generación de la hora n de los ficheros "Cargas.csv" y "Generación.csv" a dos ficheros temporales.
- Se cargan los datos de los ficheros temporales en la red de 118 nudos mediante dos comandos:
 1. "Psspy.load_data_3"
 2. "Psspy.machine_data_2"
- Se crea un bucle que se repite tantas veces como ramas tiene el sistema (181) y que realiza lo siguiente:
 1. Desconecta la rama correspondiente con los comandos, "psspy.branch_data" o "psspy.two_winding_data_3" según sea una línea o un transformador.
 2. Realiza un reparto de cargas.

3. Se obtienen los flujos por las ramas mediante el comando "p=psspy.aflowreal" y se crea un fichero, "Flujos_lin_tra.csv", en el que se copian los mismos.
 4. Conecta la rama que había desconectado anteriormente.
- Se vuelve al punto uno hasta que se hayan completado las 8784³ horas.

A continuación se muestra un extracto del fichero de salida para el caso de desconectar la línea 1-2 a la izquierda y el fichero del caso N a la derecha:

Case	Line Configuration	Flow Value
F1-000 (Left)	-1,1,1,3,1	-7.963294
	1,1,2,12,1	-8.755441
	1,1,3,5,1	-10.165982
	1,1,3,12,1	-6.038517
	1,1,4,5,1	-15.624973
	1,1,4,11,1	3.665661
	1,1,5,6,1	9.439848
	1,1,5,8,1	-40.809067
	1,1,5,11,1	5.543446
	1,1,6,7,1	-0.919368
	1,1,7,12,1	-11.564000
	1,1,8,9,1	-28.592787
	1,1,8,30,1	-13.482820
	1,1,9,10,1	-28.659353
	1,1,11,12,1	-12.227709
	FL1 (Right)	-1,1,2,1
1,1,3,1		-5.223975
1,2,12,1		-11.497545
1,3,5,1		-8.690148
1,3,12,1		-4.770481
1,4,5,1		-16.030468
1,4,11,1		4.071157
1,5,6,1		9.889312
1,5,8,1		-40.638248
1,5,11,1		5.999938
1,6,7,1		-0.470948
1,7,12,1	-11.115560	
1,8,9,1	-28.592813	
1,8,30,1	-13.311779	
1,9,10,1	-28.659454	

Figura 4: Extracto de los ficheros "F1-000.csv" y "FL1.csv"

Se puede ver en el fichero "F1-000" de la izquierda, que el flujo de la línea 1-2 no aparece, a diferencia del fichero de la derecha, en el que dicho flujo toma el valor -2,739319. Esto ocurre porque PSS/E sólo devuelve el valor de los flujos de las ramas conectadas, ya que el valor de las desconectadas es obviamente cero. Por tanto, como el fichero de la izquierda muestra los flujos cuando se desconecta la rama 1-2, dicho valor será cero y no aparecerá.

Para los cálculos que se llevarán a cabo será conveniente que dichos flujos nulos queden reflejados como tal, por lo que se definirá la función "Convertir_F (FN, caso)" para tal propósito.

7.1.2 Obtención de ρ_{ij}^k

Se utilizará la actividad DFAX para generar un fichero con los factores de distribución de todas las líneas del sistema respecto a cada uno de los nudos del mismo (excepto el nudo de referencia) para cada una de las contingencias. Para ello, el fichero

³ El programa se simuló en bloques de 500 horas debido a que para más de 650 horas el ordenador se quedaba sin memoria

de contingencias que se le proporciona a la entrada se encargará de indicarle que el efecto que tiene que modelar es la de fijar la carga (equivalente a generación) en 10 MW. La estructura de este fichero es la que sigue:

```
CONTINGENCY 'BUS 1 '  
SET BUS 1 LOAD TO 10 MW  
END  
CONTINGENCY 'BUS 2 '  
SET BUS 2 LOAD TO 10 MW  
END  
CONTINGENCY 'BUS 3 '  
SET BUS 3 LOAD TO 10 MW  
END
```

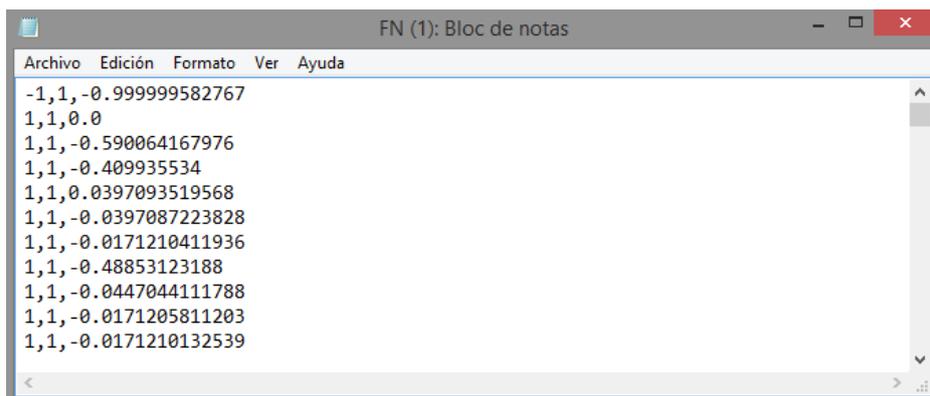
7.1.2.1 Pseudocódigo obtención de ρ_{ij}^k

- Se crea un bucle que se repite 181 veces y que realiza los siguientes pasos:
 1. Desconecta la rama n .
 2. Calcula el archivo ".dfax" mediante la instrucción "psspy.dfax" con los siguientes ficheros de entrada:
 - (a) "subsistema.sub"
 - (b) "monitorizacion.mon"
 - (c) "contingencias.con"y se obtiene como salida el fichero "dfax_nudos.dfx" .
 3. Obtiene los ρ_{ij}^k gracias a "pssarrays.otdf_factors"
 4. Conecta la rama que había desconectado anteriormente.
 5. Escribe los valores en un fichero de salida: "FN.csv"
- Se vuelve al principio hasta que se completa el bucle.

7.1.2.2 Resultados ρ_{ij}^k

El fichero "FN.csv" contienen todos los factores de distribución del sistema necesarios para realizar los cálculos.

A continuación se muestran un extracto del fichero de salida del caso uno resultante de desconectar la rama 1-2 y la tabla correspondiente en MATLAB:

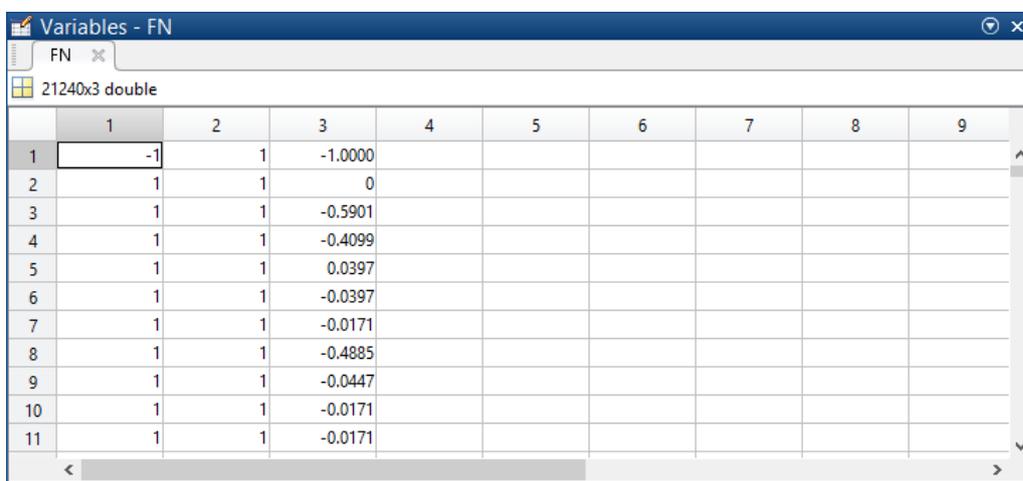


```

-1,1,-0.999999582767
1,1,0.0
1,1,-0.590064167976
1,1,-0.409935534
1,1,0.0397093519568
1,1,-0.0397087223828
1,1,-0.0171210411936
1,1,-0.48853123188
1,1,-0.0447044111788
1,1,-0.0171205811203
1,1,-0.0171210132539

```

Figura 5: Extracto del fichero "FN (1).csv"



	1	2	3	4	5	6	7	8	9
1	-1	1	-1.0000						
2	1	1	0						
3	1	1	-0.5901						
4	1	1	-0.4099						
5	1	1	0.0397						
6	1	1	-0.0397						
7	1	1	-0.0171						
8	1	1	-0.4885						
9	1	1	-0.0447						
10	1	1	-0.0171						
11	1	1	-0.0171						

Figura 6: Extracto de la tabla "FN.csv"

Se puede observar en la tabla anterior que se cuenta con 21.240 factores distintos cuando cabría esperar 21358 valores (118 nudos x 181 ramas). Esto, al igual que se comentó en el apartado 7.1.1, se debe a que PSS/E sólo obtiene los factores para las ramas conectadas y como en este caso la primera está desconectada, todos los valores referentes a ésta no

aparecen. Para cálculos posteriores será más sencillo trabajar con ficheros completos, es decir, que aparezca el valor de ρ_{ij}^k de la rama desconectada aunque éste sea cero, por lo que se utilizará la función "Convertir_FactN (FN, caso)" para tal propósito.

7.1.3 Obtención de $P_{ij}^{máx}$

Es con diferencia el valor más sencillo de obtener. Tan solo hay que utilizar el comando "RateA=psspy.abrreal" y copiar el resultado en el fichero "RateA.csv" .

7.2 TRATAMIENTO DE DATOS

Por la forma de simular los distintos casos para la obtención de los flujos P_{ij}^0 , (en el apartado 7.3 se detallará el porqué del método escogido) los datos obtenidos necesitaron de un procesado posterior antes de poder realizar cálculos. Para ello se creó un programa en Python, "N-10.py" , que copiaba los flujos P_{ij}^0 de cada contingencia en un fichero aparte en el que se reunían para las 8784 horas, obteniéndose 181 ficheros distintos nombrados de "F1.csv" a "F181.csv" .

Debido a los más de 280 millones de registros a procesar, se tuvo que optimizar el algoritmo para que el programa mantuviese abierto el fichero a ordenar y fuese actualizando el puntero con cada iteración, en vez de abrir el fichero y buscar los datos con cada iteración.

7.3 JUSTIFICACIÓN DEL MODO DE OBTENCIÓN DE P_{ij}^0

Para conseguir los P_{ij}^0 había dos formas de proceder:

1. Desconectar una rama, obtener los P_{ij}^0 para las 8784 horas cargando la generación y la carga para cada hora, volver a conectar la rama y repetir el proceso 181 veces.
2. Cargar los datos de carga y generación para una hora, desconectar y conectar iterativamente las 181 ramas obteniendo P_{ij}^0 y repetir el bucle para todos los escenarios.

El problema surge porque la carga de datos para cada hora es un proceso "lento" en PSS/E. Con el primer método se producen $181 \times 8784 = 1.589.904$ cargas de datos y se obtiene un fichero de salida ordenado por caso, mientras que con el segundo se producen sólo 8784 cargas de datos aunque requiera un procesado posterior el fichero de salida. Esa diferencia de cargas de datos se traduce en 30 días de simulación con el primer método frente a las poco más de 3 días para el segundo incluyendo el tratamiento posterior.

Es por ello que se optó por el método que requería menos tiempo de simulación, el segundo.

7.4 FACTORES DE REPARTO DE LA GENERACIÓN γ_{g_s}

Los factores que se utilizarán para las simulaciones son los mismos que los obtenidos en el PFMN, por lo que no se entrará en detalle sobre su obtención, simplemente se hará una introducción y se expondrá una tabla con los valores buscados.

Los factores de reparto de la generación, que se han designado como γ_{g_s} representan como se reparte la generación conectada que se está estudiando entre el resto de generadores del sistema. En efecto, para mantener el equilibrio demanda-generación necesario en todo sistema eléctrico el incremento de generación ΔP_k que hemos conectado en el nudo k debe reducirse en el resto de nudos del sistema. Como hipótesis de partida, se supondrá que esta generación se reducirá en cada generador de forma proporcional a la potencia que inicialmente producía. Así la variación de potencia del generador r vendrá dada por:

$$\Delta P_r = \Delta P_k \frac{P_{gr}}{\sum_{s=1}^{n \neq k} P_{g_s}} = \Delta P_k \gamma_{g_s}$$

Donde

$$\sum_{r=1}^n \gamma_{g_s} = 1$$

Se debe tener en cuenta que se van a utilizar para un cálculo probabilístico que involucra 8784 escenarios de generación, sin embargo se utilizará el valor promedio de estos factores a lo largo del año, valor que por otra parte es bastante estable ya que igualmente se supone un reparto de la generación en cada hora proporcional a la punta, y solo las pérdidas tiene una mínima influencia en los mismos.

Los factores de reparto de Generación $\gamma_{g_s}^k$, promediados, son representados en la siguiente tabla:

<i>NUDO MÁQUINA</i>	$\gamma_{g_s}^k$
100	4,0%
10	2,0%
103	3,7%
106	3,1%
111	1,0%
12	4,5%
25	2,7%
26	6,2%
31	0,2%
32	4,7%
34	2,7%
46	0,5%
49	6,7%
54	4,0%
59	4,1%
61	4,3%
65	8,0%
66	10,4%
(69)	-0,1%
80	12,7%
87	0,1%
89	14,5%
TOTAL	100%

Tabla 2: Factores de Participación

El factor de participación del nudo de referencia (69) oscila en valores muy bajos por lo que la media se acerca a cero como se puede ver en la tabla 2.

Se muestra a continuación el gráfico de la participación de cada máquina en el total de generación del sistema:

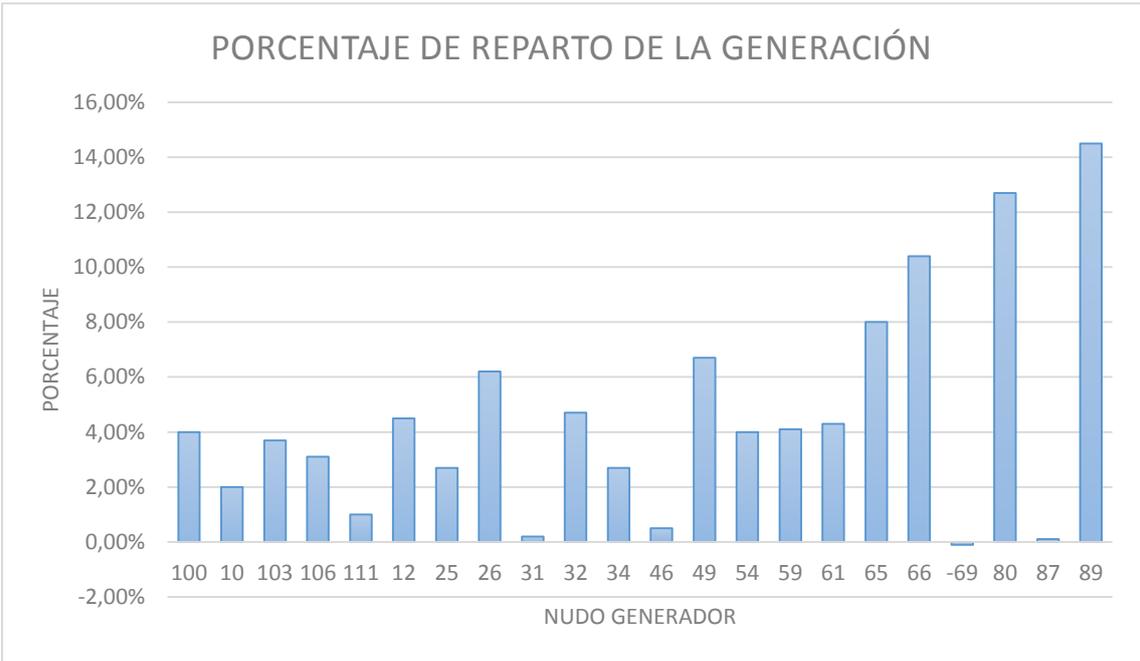


Figura 7: Gráfico del porcentaje de reparto de la generación

8. OBTENCIÓN DE ΔP_k EN MATLAB

8.1 DETERMINACIÓN DE LOS FACTORES A_{ij}^k MEDIANTE EL PROGRAMA "Mat_A" DE MATLAB

Una vez se disponen de todos los datos necesarios, se procede al cálculo de la expresión:

$$A_{ij}^k = \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \right)$$

Esta constante se calculará para las 181 contingencias estudiadas.

Como se comentó en el apartado 7.1.2.2, cuando se calculan los factores de distribución mediante PSS/E, se obtiene una lista con N-1 factores (180) por cada uno de los 118 nudos, es decir, el programa obvia el valor correspondiente a la rama que está fuera de servicio, que es cero. Ello puede llevar a error en cálculos que involucran a dichos factores por lo que se utiliza un script de MATLAB "Convertir_FactN" para añadir un cero en la fila correspondiente según el caso que se esté estudiando, es decir, si la línea uno está fuera de servicio, se agrega un cero en la primera fila por cada nudo. Una vez hecho esto se puede proceder con los cálculos.

El procedimiento seguido se expone a continuación:

- Los Factores de Generación $\gamma_{g_s}^k$ están definidos como un vector "RG" (Reparto de Generación) en el script de MATLAB "SIMULA_N"
- A continuación, se calcula el término del sumatorio $\sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k$ de cada constante A_{ij}^k . Primero se utilizará el script "SUMGEN" que obtiene el sumatorio para una rama determinada multiplicando el vector "RG" por los factores ρ_{ij}^s . Seguidamente se usará "SUMGEN_TOTAL" que calcula el sumatorio para las 181 ramas del sistema:

	1	2	3	4	5	6	7	8	9
1	0								
2	8.9116e-08								
3	0								
4	-0.0021								
5	0.0021								
6	-0.0033								
7	0.0033								
8	0.0051								
9	-0.0141								
10	0.0037								
11	0.0051								
12	0.0051								
13	0.0200								

Figura 8: Resultados obtenidos con la función "SUMGEN TOTAL"

- Una vez calculado el sumatorio $\sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k$ se puede cruzar el resultado con la matriz "FactN" que contiene los factores de distribución de cada rama cuando se varia la inyección en cada uno de los nudos del sistema:

	1	2	3	4	5	6	7	8	9
1	1	1	2	0	0	0			
2	1	1	3	-1.0000	8.9116e-08	-1.0000			
3	1	2	12	0	0	0			
4	1	3	5	-0.5901	-0.0021	-0.5921			
5	1	3	12	-0.4099	0.0021	-0.4079			
6	1	4	5	0.0397	-0.0033	0.0364			
7	1	4	11	-0.0397	0.0033	-0.0364			
8	1	5	6	-0.0171	0.0051	-0.0120			
9	1	5	8	-0.4885	-0.0141	-0.5026			
10	1	5	11	-0.0447	0.0037	-0.0410			
11	1	6	7	-0.0171	0.0051	-0.0120			
12	1	7	12	-0.0171	0.0051	-0.0120			
13	1	8	9	0	0.0200	0.0200			

Figura 9: Obtención del factor A mediante "Mat_A"

En la figura 9 cada columna representa respectivamente:

1. Nudo
2. Línea "From"
3. Línea "To"
4. Factor de distribución de nudos
5. Sumatorio $\sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k$
6. Factor A_{ij}^k

De esta manera se obtienen los 21358 Factores A_{ij}^k del caso, correspondientes a un A_{ij}^k por cada una de las 181 ramas cuando se inyecta potencia en cada uno de los 118 nudos (118 nudos x 181 ramas = 21358 A_{ij}^k).

$$A_{ij}^k = \left(\rho_{ij}^k + \sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k \right)$$

8.2 OBTENCIÓN DE LAS FUNCIONES DE DISTRIBUCIÓN DE PROBABILIDAD

El procedimiento para determinar las funciones de Distribución de probabilidad consta de las siguientes etapas:

- Conversión de los flujos obtenidos en el apartado 7.1.1 P_{ij}^0 en una función de densidad de probabilidad: $\mathcal{F}(P_{ij} = P_x) = X$ (histograma)
- Conversión de la curva anterior en una Función de Distribución: $\mathcal{F}(P_{ij} \leq P_x) = X$

Se describen a continuación cada una de éstas etapas.

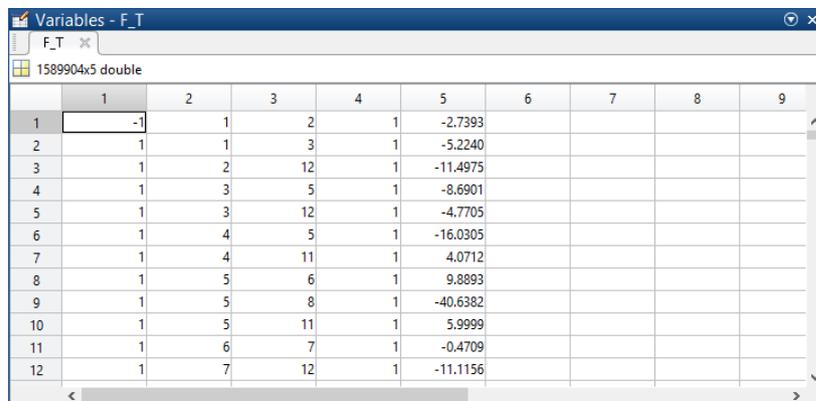
8.2.1 Conversión de los flujos P_{ij}^0 en una función de densidad de probabilidad: $\mathcal{F}(P_{ij} = P_x) = X$

La función de probabilidad $\mathcal{F}(P_{ij}^0)$ será determinada en base a los flujos P_{ij}^0 calculados para los 8784 escenarios horarios de demanda del año eléctrico, por cada contingencia, contenidos en la matriz "F_T".

Para ello se realizará, en primer lugar, el histograma del flujo de potencia por cada rama del sistema a intervalos de 1MW, donde se llevará a cabo el conteo del número de veces, a lo largo del año eléctrico, que el flujo por una determinada rama se ha situado entre un determinado valor (X MW) y el siguiente valor en orden creciente (X+1 MW).

El procedimiento a seguir para este paso consistirá en la definición de una función en MATLAB, "Programa_Final" que realiza los siguientes pasos:

Primero se obtienen los flujos de una línea determinada de la matriz "F_T" que tiene la siguiente estructura:



	1	2	3	4	5	6	7	8	9
1	-1								
2	1	1							
3	1	2	12						
4	1	3	5	1					
5	1	3	12	1	-4.7705				
6	1	4	5	1	-16.0305				
7	1	4	11	1	4.0712				
8	1	5	6	1	9.8893				
9	1	5	8	1	-40.6382				
10	1	5	11	1	5.9999				
11	1	6	7	1	-0.4709				
12	1	7	12	1	-11.1156				

Figura 10: Obtención del Histograma. Matriz de flujos "F_T"

Seguidamente, se utiliza la función "repeticiones=histc (x,binranges)" que cuenta el número de veces que se repite un determinado valor del vector "x" dentro de unos límites definidos por el vector "binranges". Éste contiene valores desde cero hasta el valor máximo de "x" en intervalos de 1MW. El resultado se copia en el vector "repeticiones"

	1	2	3	4	5	6	7	8	9
1	0	0	0	374	1862	1846	1728	1290	909
2									
3									
4									

Figura 11: Obtención del Histograma. Vector "repeticiones"

A continuación, se calcula la probabilidad de que el flujo que atraviesa cada una de las 181 ramas durante la totalidad de los 8784 escenarios horarios de demanda del año eléctrico, se sitúe en un determinado número de MW, entendiendo por probabilidad el número de veces que el flujo por una rama, toma un valor comprendido entre X y X+1 MW a lo largo del año eléctrico, dividido por el número de muestras total de flujo, es decir, dividido entre el número de escenarios horarios de demanda (8784). La probabilidad de cada rama viene recogida en el vector "vect_prob"

	1	2	3	4	5	6	7	8	9
1	0	0	0	4.2577	21.1976	21.0155	19.6721	14.6858	10.3484
2									
3									
4									
5									
6									

Figura 12: Obtención del Histograma. Vector "vect_prob"

Con estos pasos se ha determinado la función de probabilidad de que el flujo de potencia que atraviesa una rama $i-j$ del sistema tome un determinado valor (entre X y X+1) a lo largo de la totalidad del año eléctrico.

A modo de comprobación, se sumarán todas las probabilidades de cada valor en MW del espectro de potencia agrupando por rama para asegurar que se obtiene un 100%.

Mediante la función "plot" de MATLAB, se pueden visualizar las distintas distribuciones de probabilidad calculadas:

Para una línea ejemplo:

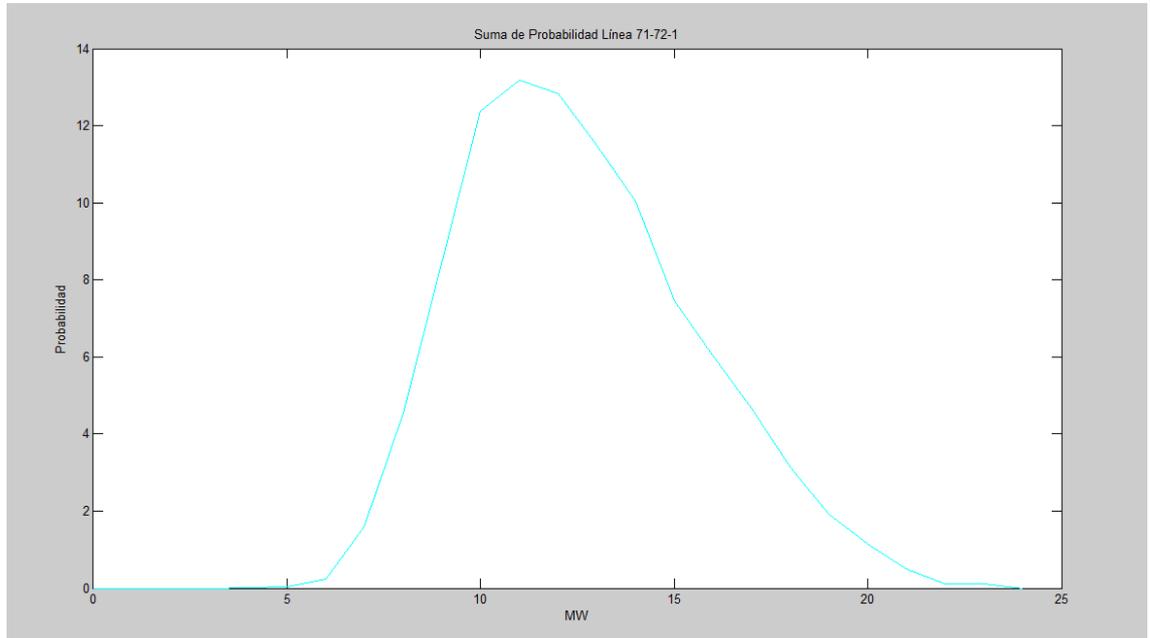


Figura 13: Función de densidad de una línea

Para un transformador ejemplo:

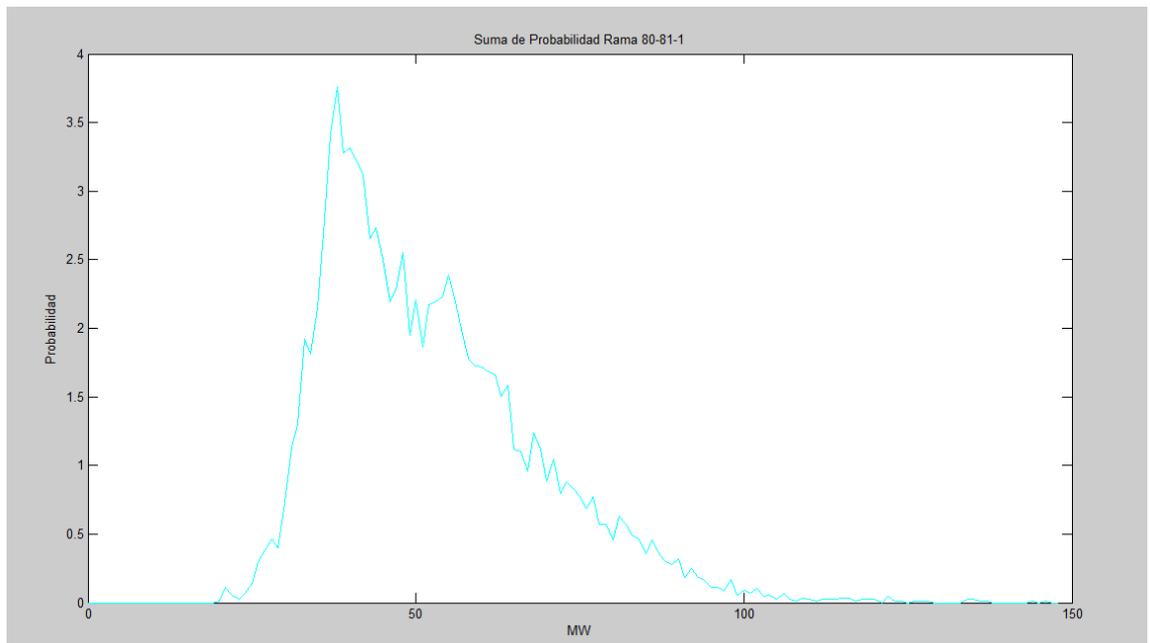


Figura 14: Función de densidad de un trafo

8.2.2 Cálculo de la Función de Distribución

El objetivo del análisis probabilístico será determinar la inyección máxima $\Delta P_k^{m\acute{a}x}$ en el nudo k , tal que la probabilidad de que el flujo $P_{ij}^{\Delta P_k}$ por cualquier rama $i - j$ no rebase la capacidad nominal de dicha rama $P_{ij}^{m\acute{a}x}$, sea igual a un determinado valor $1-X$.

$$\mathcal{F}(P_{ij}^{\Delta P_k} \leq P_{ij}^{m\acute{a}x}) = 1 - X$$

O bien:

$$\mathcal{F}(P_{ij}^0 \leq P_{ij}^{m\acute{a}x} - A_{ij}^k \cdot \Delta P_k) = 1 - X$$

La desigualdad de la anterior ecuación implica que para cada valor de flujo de potencia por una rama del sistema, no basta con contabilizar la probabilidad de que fluya dicha potencia:

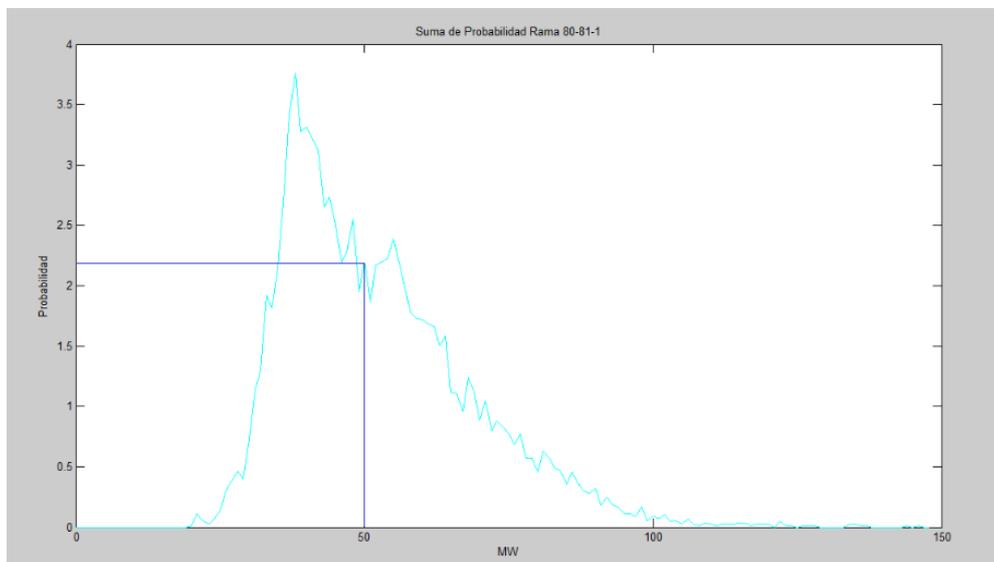


Figura 15: Función de densidad

La figura muestra que la probabilidad de que por la rama circulen entre 49 y 50 MW es del 2,3%

Si no que, por el contrario, se ha de contabilizar la probabilidad de que fluya por cada rama un determinado valor de potencia y cualquiera de las potencias inferiores. Es decir, se necesita calcular la Función de Distribución de Probabilidad.

Un planteamiento alternativo al anterior sería considerar exactamente lo contrario, es decir, en lugar de buscar la probabilidad de que el flujo por una determinada rama se sitúe por debajo de determinado valor, se calculará la probabilidad de que el mencionado flujo se sitúe por encima del valor buscado o probabilidad de sobrecarga por encima del umbral que se fije.

Expresado matemáticamente, lo mismo es realizar el cálculo de la función de probabilidad:

$$\mathcal{F}(P_{ij}^{\Delta P_k} \leq P_{ij}^{m\acute{a}x}) = 1 - X \quad / \quad \mathcal{F}(P_{ij}^0 \leq P_{ij}^{m\acute{a}x} - A_{ij}^k \cdot \Delta P_k) = 1 - X$$

Que realizar el cálculo de:

$$\mathcal{F}(P_{ij}^{\Delta P_k} \geq P_{ij}^{m\acute{a}x}) = X \quad / \quad \mathcal{F}(P_{ij}^0 \geq P_{ij}^{m\acute{a}x} - A_{ij}^k \cdot \Delta P_k) = X$$

Que es la función complementaria a la función de Distribución de probabilidad, y que se denominará Función de Distribución Acumulada.

La probabilidad de que el flujo que circula por la rama sea superior a 50 MW es de aproximadamente el 47% = Prob (P=1) + Prob (P=2) + Prob (P=3)... + Prob (P=50) como se muestra en la figura 16.

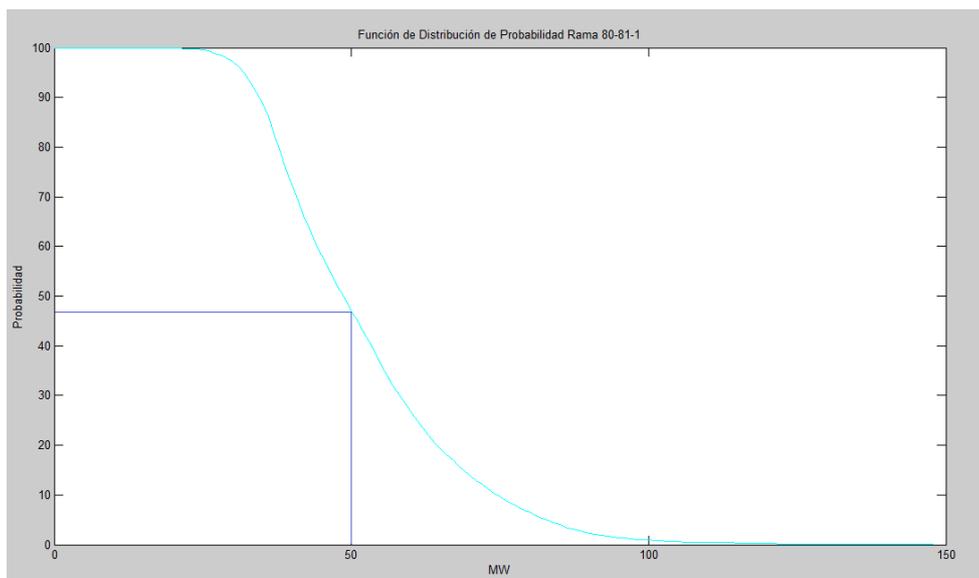
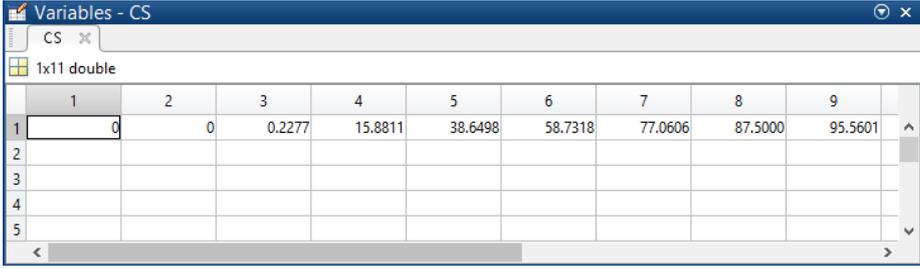


Figura 16: Función complementaria a la función de distribución

Se adoptará el segundo criterio utilizando por tanto la función complementaria a la función de distribución de probabilidad.

El procedimiento a seguir para este paso consistirá en utilizar la función "cumsum" de MATLAB con el vector "vect_prob" que va sumando los valores del vector conforme lo recorre y obtener "CS" :

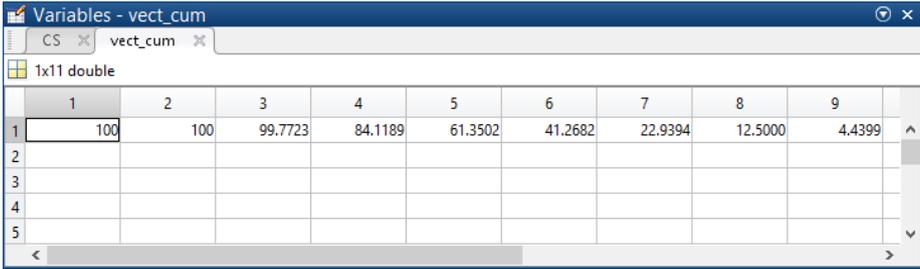


The screenshot shows a MATLAB window titled "Variables - CS". It displays a 1x11 double array. The first row contains the following values: 0, 0, 0.2277, 15.8811, 38.6498, 58.7318, 77.0606, 87.5000, 95.5601. The rest of the array is empty.

	1	2	3	4	5	6	7	8	9	
1	0	0	0.2277	15.8811	38.6498	58.7318	77.0606	87.5000	95.5601	
2										
3										
4										
5										

Figura 17: Vector "CS"

A continuación se calcula "vect_cum" restándole al 100% el valor de la probabilidad acumulada, es decir, el vector "CS" :



The screenshot shows a MATLAB window titled "Variables - vect_cum". It displays a 1x11 double array. The first row contains the following values: 100, 100, 99.7723, 84.1189, 61.3502, 41.2682, 22.9394, 12.5000, 4.4399. The rest of the array is empty.

	1	2	3	4	5	6	7	8	9	
1	100	100	99.7723	84.1189	61.3502	41.2682	22.9394	12.5000	4.4399	
2										
3										
4										
5										

Figura 18: Vector "vect_cum"

Con lo que se habrá calculado la Función de Distribución Acumulada.

Para una línea ejemplo:

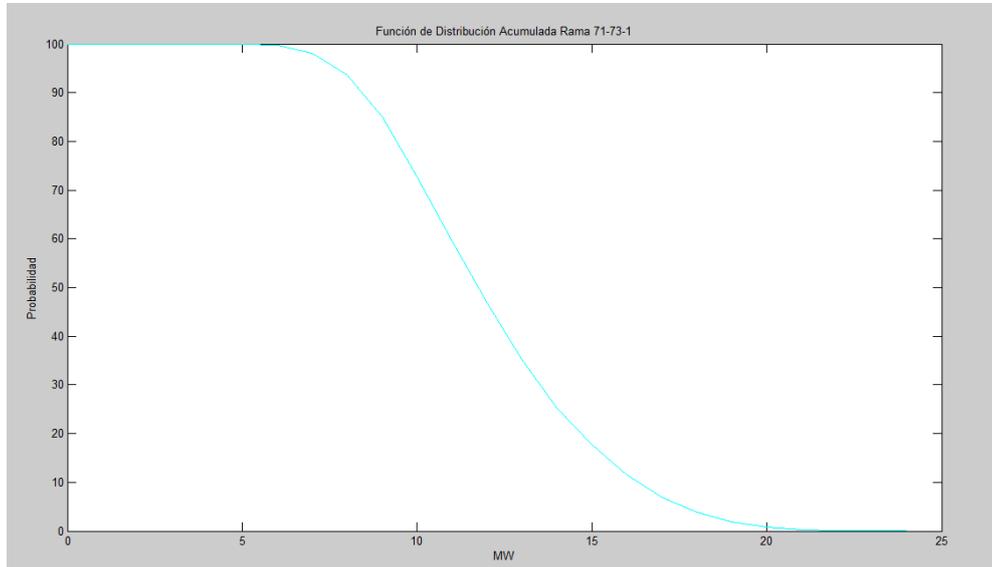


Figura 19: Función de Distribución para una línea

Para un transformador ejemplo:

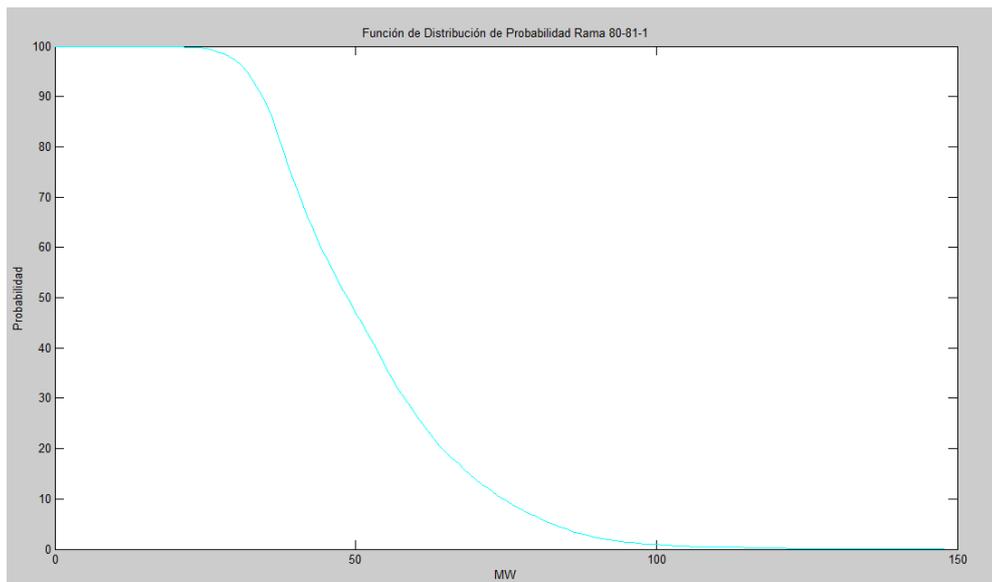


Figura 20: Función de Distribución para un transformador

8.3 CÁLCULO DEL FLUJO $P_{ij}^{\Delta P_k}$ FIJADA LA PROBABILIDAD $\mathcal{F}(P_{ij}^0)$

Una vez conocidos todos los parámetros en juego se está en disposición de calcular, no únicamente el Flujo de potencia P_{ij}^0 umbral para una probabilidad de sobrecarga dada, sino que también podemos calcular el Flujo de potencia $P_{ij}^{\Delta P_k}$ umbral, para esa misma probabilidad de sobrecarga fijada.

Es decir hasta ahora se ha calculado $\mathcal{F}(P_{ij}^0)$, pero se puede conocer para una probabilidad de sobrecarga fijada, la potencia umbral en un escenario en el que exista inyección ΔP_k en algún nudo k del sistema.

Para un umbral de probabilidad fijado X;

$$\mathcal{F}(P_{ij}^{\Delta P_k} \geq P_{ij}^{m\acute{a}x}) = X \quad / \quad \mathcal{F}(P_{ij}^0 \geq P_{ij}^{m\acute{a}x} - A_{ij}^k \cdot \Delta P_k) = X$$

Donde,

$$A_{ij}^k \cdot \Delta P_k = P_{ij}^{m\acute{a}x} - P_x$$

Despejando ΔP_k se tiene:

$$\Delta P_k = \frac{P_{ij}^{m\acute{a}x} - P_x}{A_{ij}^k}$$

Y dado que se conocen los $P_{ij}^{m\acute{a}x}$ y se han calculado anteriormente los A_{ij}^k , se está en disposición de poder calcular los ΔP_k en cuanto se calculen los valores de P_x .

Valores pueden calcularse a partir de la función de probabilidad: $\mathcal{F}(P_{ij}^0)$ obtenida en el apartado anterior. Es decir, según la expresión anterior, se puede calcular la capacidad de evacuación en un nudo sin resolver repartos de carga conectando dicha generación. Bastaría conocer la función de densidad de los flujos por las líneas antes de conectar la generación en régimen especial, los factores de distribución y los factores de participación de la generación.

El procedimiento a seguir en este paso consistirá en la definición de la función "P_x_positivo" que calcula los valores de P_x para una probabilidad determinada.

Mediante los vectores "vec_cum" y "binranges", que contienen los valores de Probabilidad acumulada y Potencia respectivamente, se obtienen los valores de Probabilidad anterior y posterior (superior/inferior) a la Probabilidad de sobrecarga fijada (por ejemplo: 10%) con objeto de interpolar y obtener el valor más aproximado posible de P_x .

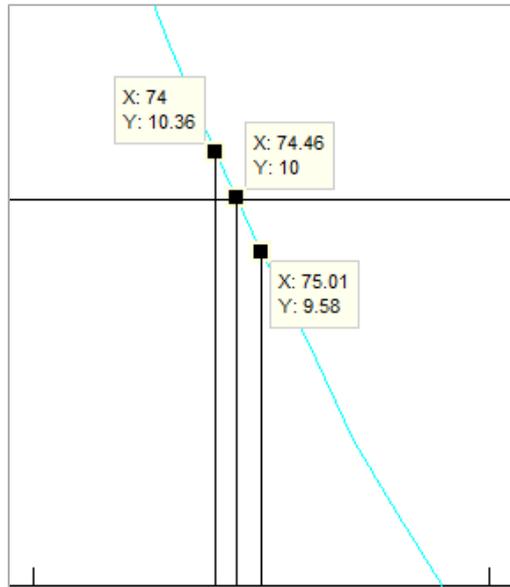


Figura 21: Interpolación de valores

1. Para la obtención de dichos valores de probabilidad (anterior y posterior), utilizaremos el polinomio de interpolación de Newton:

$$\mathcal{F}(x|x_1; x_2) = \mathcal{F}(x_1) + \frac{\mathcal{F}(x_2) - \mathcal{F}(x_1)}{(x_2 - x_1)}(x - x_1)$$

que será implementado en la función "P_x_positivo" de MATLAB para una probabilidad fijada (por ejemplo: 10%).

2. Ésta misma función, se encargará de calcular el valor de P_x para probabilidades fijadas de 5%, 10%, 15%, 20% y 25%.

De este modo se obtiene el valor de P_x para todas las ramas por cada porcentaje con un total de 905 registros (5 porcentajes x 181 ramas).

8.4 INVERSIÓN DE FLUJO

Como se explica en el PFMN, cuando hay inversión de flujo el modo de calcular el apartado 8.3 es mediante la fórmula:

$$\mathcal{F}(P_{ij}^0 - A_{ij}^k \cdot \Delta P_k \leq -P_{ij}^{máx}) = X$$

Que supone irnos al cuadrante izquierdo del eje de abscisas:

$$\mathcal{F}(P_{ij}^0 \leq -P_{ij}^{máx} + A_{ij}^k \cdot \Delta P_k) = X$$

Las ecuaciones a resolver son entonces del tipo:

$$A_{ij}^k \cdot \Delta P_k - P_{ij}^{máx} = P_y$$

Luego:

$$\Delta P_k = \frac{P_y - P_{ij}^{máx}}{A_{ij}^k}$$

Para calcular el valor de P_y , se utilizará la función de MATLAB "P_x_negativo" que realiza los mismos pasos que "P_x_positivo" pero para valores de flujo negativos.

En el caso de una rama que presenta tanto valores positivos como negativos de flujo, se procede calculando P_x con los valores positivos y P_y con los valores negativos con las fórmulas

$$\Delta P_k = \frac{P_{ij}^{máx} - P_x}{A_{ij}^k} / \Delta P_k = \frac{P_y - P_{ij}^{máx}}{A_{ij}^k}$$

y tomando el mínimo positivo de los resultados obtenidos.

9. GENERALIZACIÓN PARA LAS 181 CONTINGENCIAS

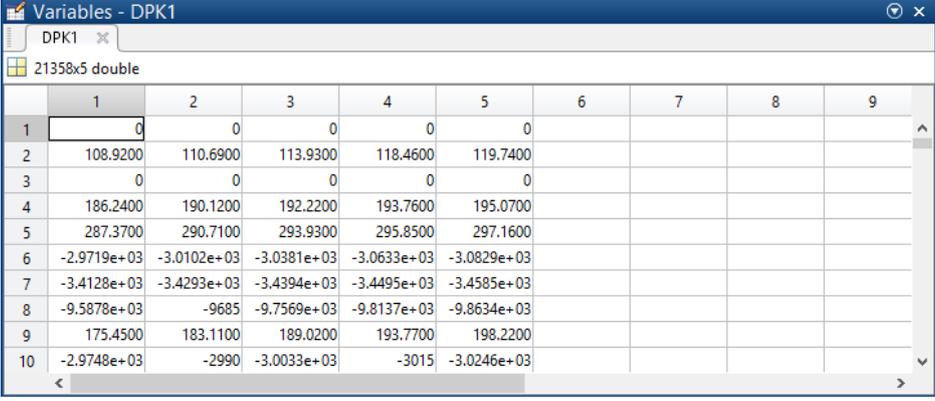
Todo el procedimiento descrito en los anteriores apartados estaba referido al caso de eliminar la línea 1-2 de la red de 118 nudos. Como el objeto de estudio de este trabajo es el análisis del sistema para las 181 contingencias resultantes de ir desconectando y conectando las ramas del mismo, se ha creado un programa en MATLAB "SIMULA_TODO" que simula todos los casos. En cada uno de ellos toma los siguientes datos de partida:

- Fichero de flujos por las ramas para las 8748 horas con la rama $m-n$ desconectada: "Fn.csv" siendo n la rama desconectada.
- Fichero con los factores de distribución de nudos con la rama $m-n$ desconectada: "FN (n).csv" siendo n la rama desconectada.
- Máxima capacidad de las ramas del sistema: "RateA.csv"

Y obtiene los ΔP_k para cada porcentaje definido (5%, 10%, 15%, 20%, 25%). Seguidamente copia éstos en un fichero "DPKn.csv" , siendo n el caso estudiado, para su posterior análisis.

10. ANÁLISIS DE RESULTADOS

Los archivos "DPK.csv" obtenidos tienen una estructura como la mostrada en la figura 22.



The screenshot shows a MATLAB window titled 'Variables - DPK1' with a tab for 'DPK1'. Below the tab, it indicates '21358x5 double'. The main area displays a grid of data for the first 10 rows and 9 columns. The values in the first row are all 0. The second row shows values increasing from left to right: 108.9200, 110.6900, 113.9300, 118.4600, 119.7400. The third row returns to 0. The fourth row shows values increasing from 186.2400 to 195.0700. The fifth row shows values increasing from 287.3700 to 297.1600. The sixth row shows negative values in scientific notation, increasing in magnitude from left to right. The seventh row shows negative values in scientific notation, increasing in magnitude from left to right. The eighth row shows negative values, increasing in magnitude from left to right. The ninth row shows values increasing from 175.4500 to 198.2200. The tenth row shows negative values, increasing in magnitude from left to right.

	1	2	3	4	5	6	7	8	9
1	0	0	0	0	0				
2	108.9200	110.6900	113.9300	118.4600	119.7400				
3	0	0	0	0	0				
4	186.2400	190.1200	192.2200	193.7600	195.0700				
5	287.3700	290.7100	293.9300	295.8500	297.1600				
6	-2.9719e+03	-3.0102e+03	-3.0381e+03	-3.0633e+03	-3.0829e+03				
7	-3.4128e+03	-3.4293e+03	-3.4394e+03	-3.4495e+03	-3.4585e+03				
8	-9.5878e+03	-9685	-9.7569e+03	-9.8137e+03	-9.8634e+03				
9	175.4500	183.1100	189.0200	193.7700	198.2200				
10	-2.9748e+03	-2990	-3.0033e+03	-3015	-3.0246e+03				

Figura 22: Fichero "DPK1.csv" cargado en MATLAB

Cada columna representa los ΔP_k para un porcentaje determinado (5%, 10%, 15%, 20%, 25%) mientras que las filas muestran los ΔP_k por rama y nudo, empezando con la rama 1 nudo 1 de la fila 1 y terminando con la rama 181 nudo 118 de la fila 21358.

Esta forma de ver los resultados es muy gráfica porque permite comparar un ΔP_k a simple vista para los distintos porcentajes. Por ejemplo se puede observar en la figura 22, para la fila 2, que conforme se observan los valores a la derecha, éstos aumentan. Esto tiene sentido porque se está aumentando el porcentaje de horas al año que la rama está sobrecargada, luego a mayor porcentaje, mayor valor de ΔP_k .

Sin embargo, para trabajar con los datos y buscar elementos limitantes, es conveniente ordenar los mismos. Para ello se usa el programa MATRIZ_CUBICA creado en MATLAB, que crea una matriz cúbica DPKT con las siguientes dimensiones:

- Fila, coordenada X, donde se especifica para una línea el ΔP_k por cada nudo para los 5 porcentajes distintos (5%, 10%, 15%, 20%, 25%).
- Columna, coordenada Y, donde se tienen los ΔP_k para todas las líneas por cada nudo.
- Página, que se definirá como la coordenada Z de la matriz cúbica. En cada página se representa un caso de los 181 distintos.

A continuación se muestra la página 1 de dicha matriz cubica DPKT:

	1	2	3	4	5	6	7	8	9
1	0	0	0	0	0	0	0	0	0
2	108.9200	0	0	0	0	0	0	0	0
3	0	115.5900	0	0	0	0	0	0	0
4	186.2400	1.8334e+03	186.2400	-2.6598e+03	-2.2611e+03	20851	4.2232e+03	-4.6863e+03	-4.6863e+03
5	287.3700	-1.9485e+03	287.3700	2.8268e+03	2.4031e+03	0	-4.4884e+03	4.9806e+03	4.9806e+03
6	-2.9719e+03	983.5100	-2.9719e+03	136.4600	-800.0500	-8.0054e+03	3.2423e+03	-1618	-1618
7	-3.4128e+03	1.1294e+03	-3.4128e+03	-603.3300	-918.7400	-9.1933e+03	3.7233e+03	-1.8581e+03	-1.8581e+03
8	-9.5878e+03	777.4800	-9.5879e+03	-1128	-958.8800	222.9500	305.7100	-1.9873e+03	-1.9873e+03
9	175.4500	199.3200	175.4500	164.8000	162.3300	178.8000	186.0700	-394.0200	-394.0200
10	-2.9748e+03	984.4800	-2.9748e+03	-1.0605e+03	-800.8300	-8.0134e+03	3.2455e+03	-1.6196e+03	-1.6196e+03

Figura 23: Página 1 de la matriz tridimensional DPKT

Se puede observar en la figura 23 como en cada columna está representando el nudo y en cada fila la rama correspondiente de las 181 que hay.

Teniendo entonces DPKT se está en posición de analizar los resultados obtenidos.

Para facilitar la búsqueda de máximos y mínimos se seguirán los siguientes pasos:

- Se eliminarán los valores negativos puesto que no son relevantes para el cálculo de la potencia evacuable.
- Se utilizará la función "round ()" de MATLAB que redondea los valores contenidos en una matriz cualquiera (para facilitar la búsqueda de valores).

A continuación se muestra la matriz para las ramas de la 14 a la 23 antes y después de realizar las operaciones mencionadas, en las figuras 24 y 25 respectivamente:

	1	2	3	4	5	6	7	8	9
14	537.7200	607.7000	537.7200	506.2500	498.9400	547.5800	568.9400	352.9700	352.9700
15	0	0	0	0	0	0	0	0	0
16	-1.4577e+03	-283.6200	-1.4577e+03	666.4900	812.4600	-884.9500	-490.3500	1.8635e+03	1.8635e+03
17	-731.8300	-716.2500	-731.8300	-725.6500	-742.7300	-729.3500	-724.3200	-1.3573e+03	-1.3573e+03
18	-566.8300	-495.5000	-566.8300	-620.3900	-627.8400	-554.3500	-530.4400	-1.1337e+03	-1.1337e+03
19	641.1600	558.3300	641.1600	702.6200	712.5100	626.6200	598.8200	1.3997e+03	1.3997e+03
20	0	0	0	0	0	0	0	0	0
21	798.9200	781.9100	798.9200	792.1800	810.8200	796.2100	790.7200	1.4817e+03	1.4817e+03
22	641.9400	561.1600	641.9400	702.6000	711.0300	627.8100	600.7300	1284	1284
23	2.0554e+03	1.6551e+03	2.0554e+03	2.3143e+03	2.4563e+03	1.9805e+03	1843	-4.7566e+03	-4.7566e+03

Figura 24: Página 1 de la matriz tridimensional DPKT

Figura 25: Matriz DPKTp resultante de realizar las operaciones descritas sobre DPKT

Se empezará analizando el caso 1 de desconectar la rama 1-2 que es, como se ha comentado, la Página 1 de la matriz DPKT mostrada en la figura 23. Se buscarán los valores limitantes por nudo (columna de la matriz).

Haciendo una comparación entre nudos tipo se tiene:

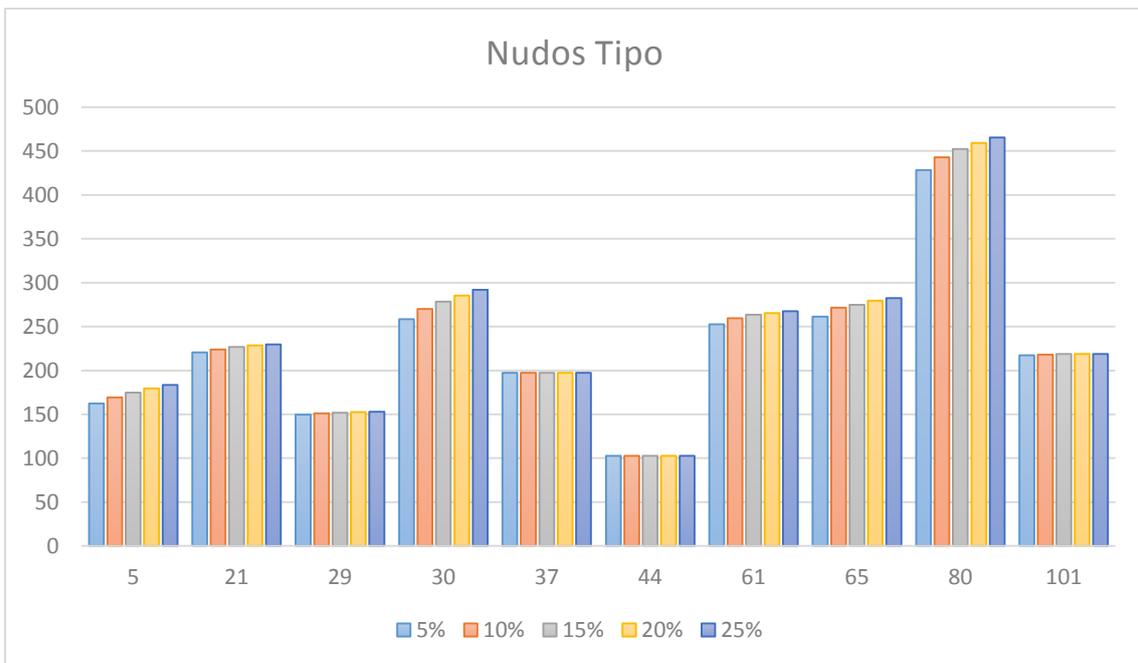


Figura 26: Resultados del análisis estadístico. Nudos tipo

Lo primero que se observa en la figura 25 es que en los nudos, al crecer X , la capacidad de evacuación calculada por métodos probabilísticos crece.

Recordando la expresión utilizada $\mathcal{F}(P_{ij}^0 > P_x) = X$, el valor de P_x es tal que éste se supera en el escenario sin generación durante X veces (X en tanto por ciento), por tanto, se obtiene que a mayor valor de X la potencia evacuada es mayor, ya que se está permitiendo que el generador se encuentre apagado o a un régimen menor durante más horas.

Seguidamente, se puede ver cuál es la máxima capacidad de evacuación por nudo calculando el valor mínimo de cada columna de la matriz "DPKtp" .

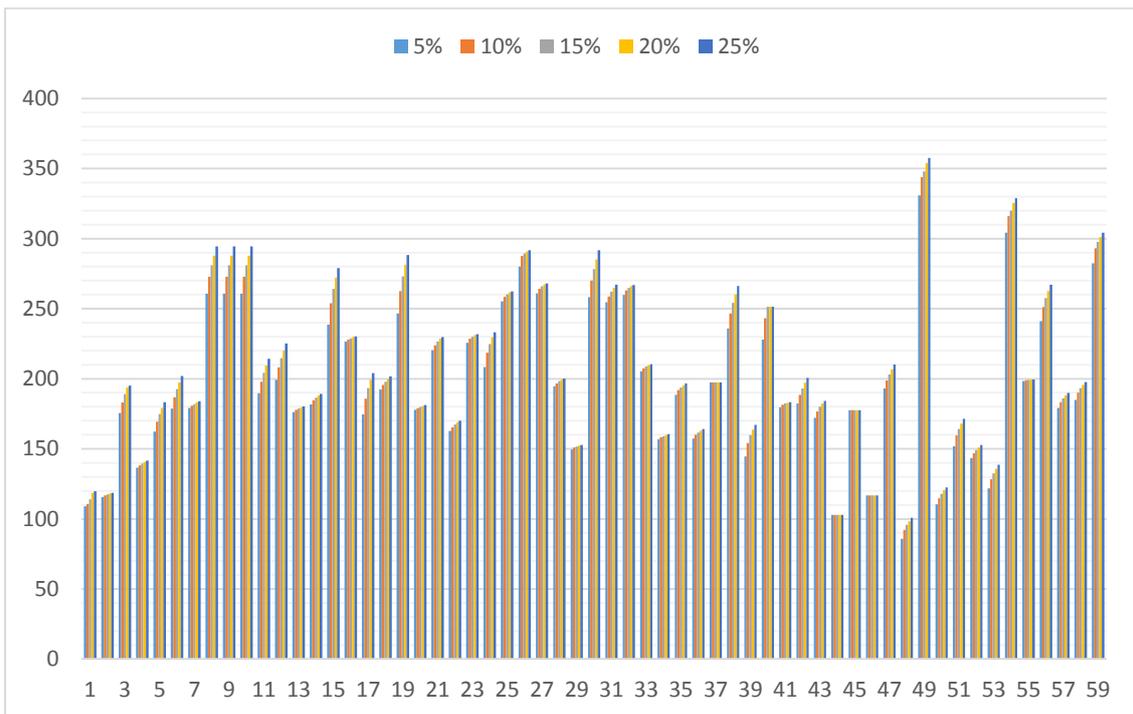


Figura 27: Resultado análisis probabilístico (1/2). Nudos 1 a 59

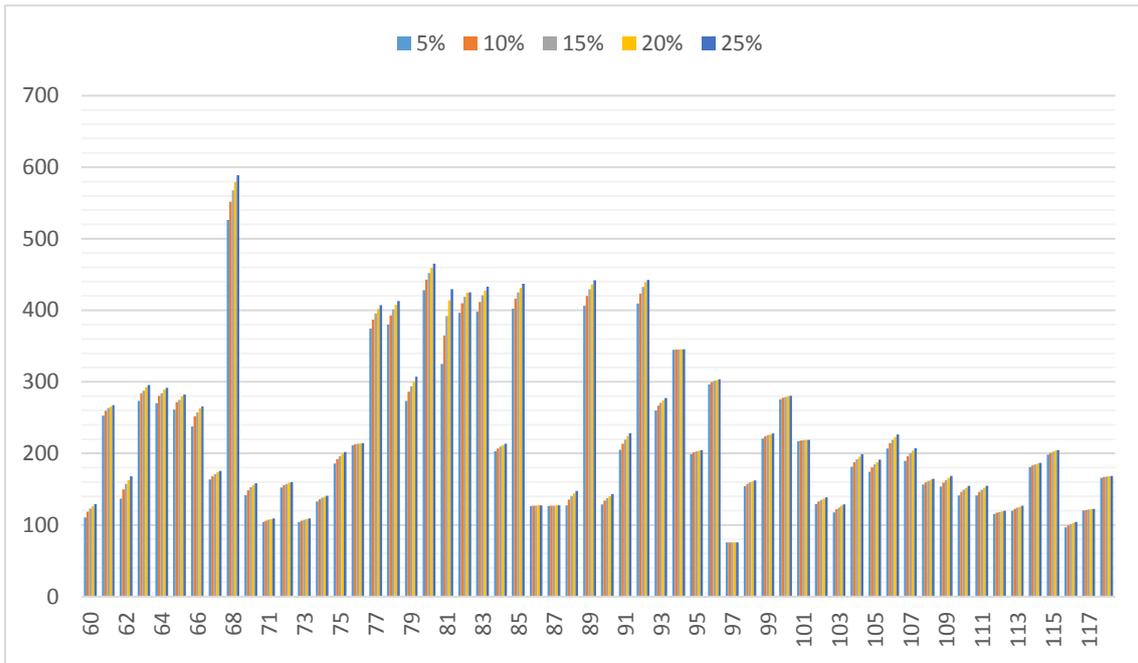


Figura 28: Resultado análisis probabilístico (2/2). Nudos 60 a 118

A continuación se puede obtener el nudo en el que se puede inyectar más potencia para una probabilidad dada sin más que coger el valor máximo de los recogidos en las figuras 27 y 28. Así, se tiene que el nudo 68 es el más favorable.

Se representa a continuación la máxima capacidad evacuable para una planta instalada en el nudo 68 dependiendo del porcentaje:

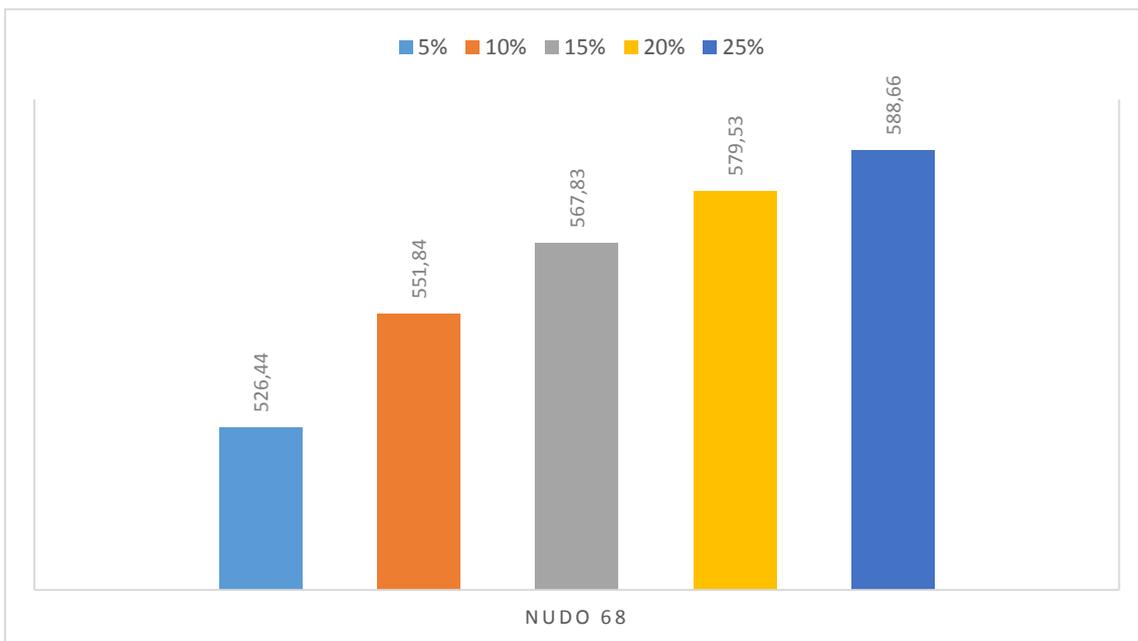


Figura 29: Potencia evacuable en el nudo 68 para distintos porcentajes

Luego dependiendo del número de horas al año en el que la planta se encuentre por debajo de su régimen nominal o desconectada, se podrán instalar desde 526 MW hasta 588 MW.

Por otro lado, si se toman los valores máximos por cada fila de la matriz DPKTp se obtiene un vector con 181 valores, uno por cada rama. Si de este vector se calcula el valor mínimo, se estará obteniendo el máximo tamaño que puede tener la planta sin que se sobrecargue ninguna rama. Se obtendrá así el nudo dónde estará la planta y la línea limitante. Para el caso que se está analizando, en el que la línea 1-2 se desconecta, el tamaño máximo de la planta se tiene en el nudo 116 con el límite impuesto por la saturación de la línea 37-40.

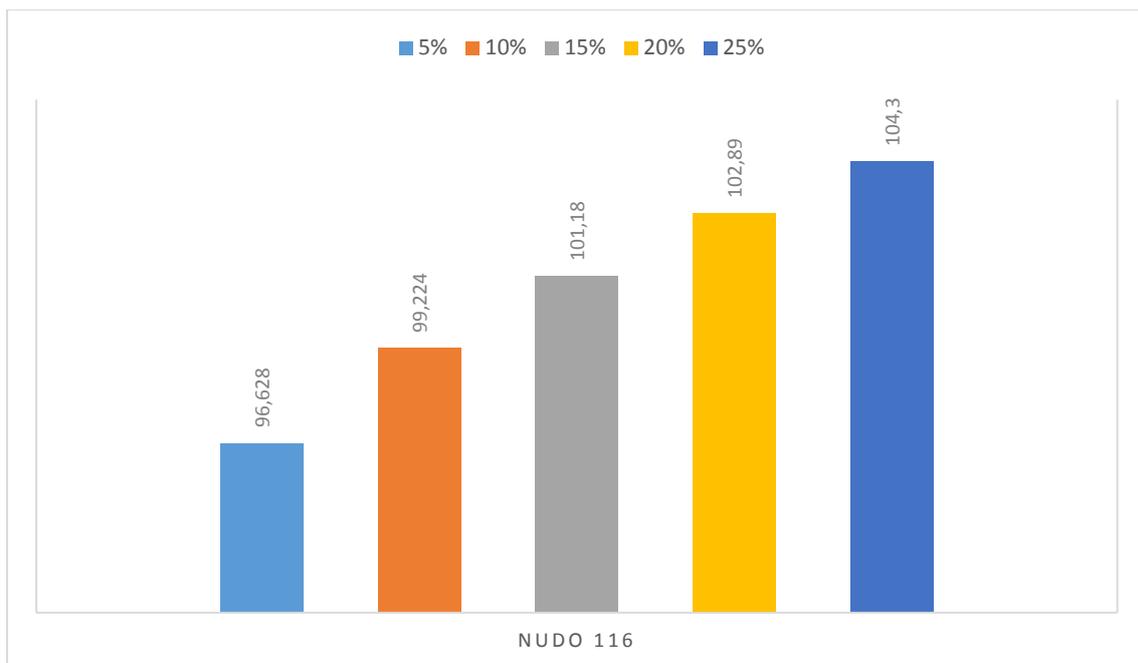


Figura 30: Nudo dónde se inyecta la máxima potencia sin sobrecargar ramas

La figura 30 muestra el nudo dónde se puede evacuar mayor potencia de todo el sistema sin sobrecargar ninguna rama para el caso 1.

Se pasa ahora a estudiar los resultados obtenidos para los 181 casos. La operación realizada anteriormente con el caso 1 de buscar máximo por fila y coger el menor de ellos, se realizará para todos los casos mediante la función "Maximo_Casos" creada en MATLAB. Se tendrá, por cada caso, la potencia máxima evacuable limitada por la primera línea que se satura del sistema.

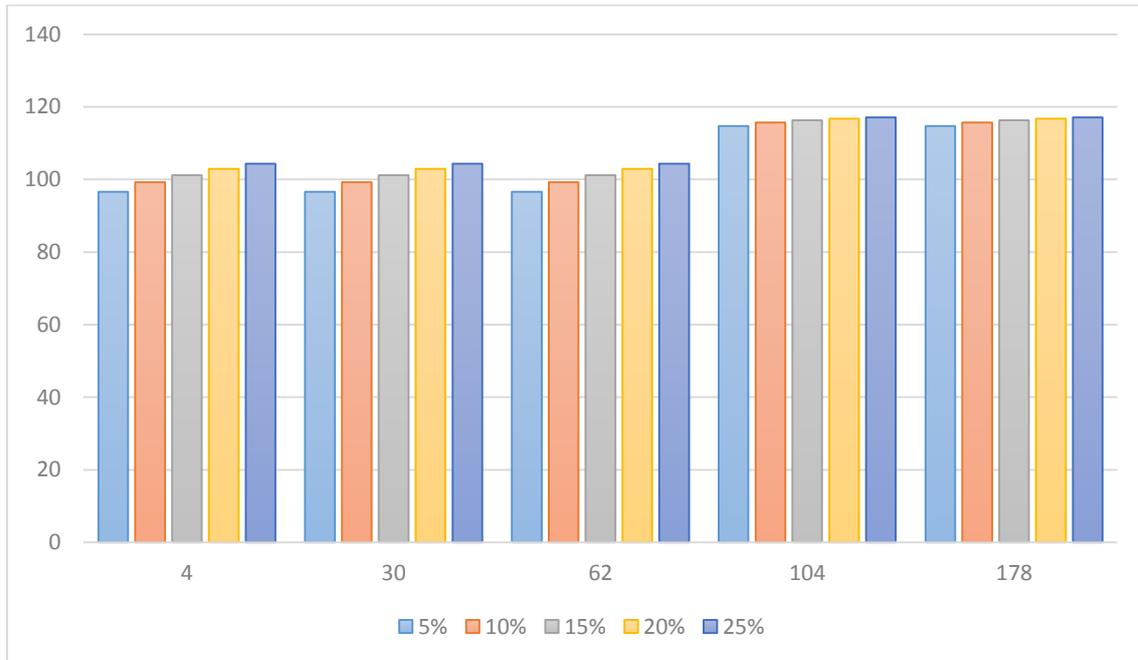


Figura 31: Potencias limitantes para distintos casos

En la figura 31 se representan las potencias limitantes para 5 casos ejemplo. Los tres primeros se repiten para la mayoría de los 181 casos mientras que los 2 últimos se obtienen en menos ocasiones. Luego se tiene que para un 5%, por ejemplo, el flujo limitante de todos los casos será el mínimo global, 96,628 MW. Este valor, aunque se refiere al 5%, se puede comparar con el resultado que se obtiene haciendo el estudio con la función TLTG de PSS/E para ver si la cifra obtenida mediante el método probabilístico aquí expuesto es muy distinta de la obtenida con una herramienta comercial. La salida proporcionada por la actividad TLTG es de 92,3 MW, una cifra que no es muy dispar con la obtenida para el 5%.

Continuando con el análisis de los 181 escenarios, si se estudian más detenidamente los casos en los que la potencia limitante coincide con el mínimo global, se tiene que la línea limitante es la 37-40 y el nudo el 116.

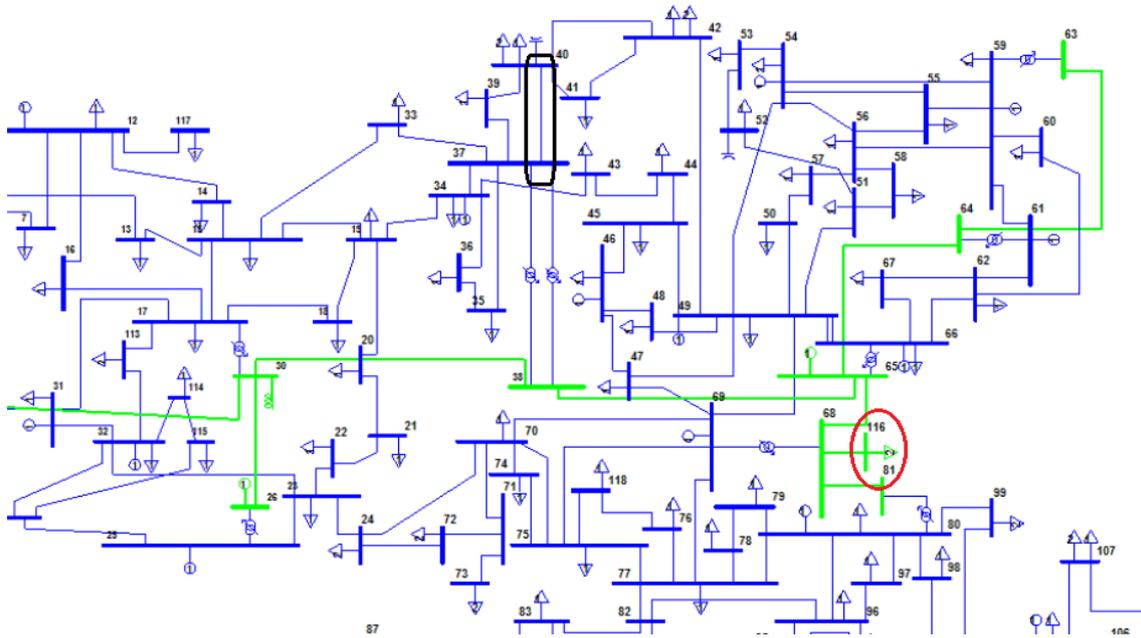


Figura 32: Línea limitante en la mayoría de los casos

Efectivamente, en la figura 32, la línea rodeada en negro es la 37-40. Como la mayoría de los casos están limitados por dicha rama, se podría plantear reforzar esta conexión para incrementar la potencia evacuable. Para ello habría que realizar un estudio económico que determinase si es rentable dicha inversión.

Siguiendo con el análisis, se puede observar que el valor más grande se da, entre otros, en el caso 104 que es cuando se desconecta la línea 68-116:

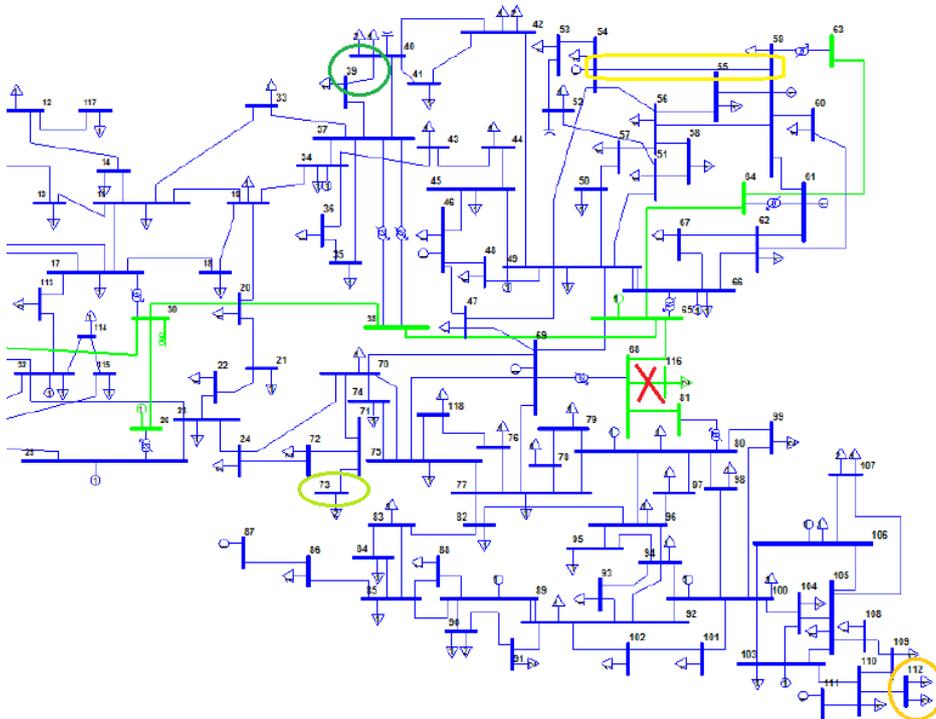


Figura 33: Líneas limitantes en el caso 104

Como muestra la figura 33 hay dos escenarios limitantes:

1. Cuando se inyecta en el nudo 73 la línea limitante es la 40-39 (Color verde).
2. Cuando se inyecta en el nudo 112 la línea limitante es la 59-54 (Color amarillo).

El hecho de tener dos líneas limitantes para dos nudos hace que el límite sea mayor que el mínimo global.

Retomando los casos donde la línea limitante era la 37-40, si se quita esta restricción, y se vuelve a buscar el mínimo global, el valor limitante es ahora el mismo que para el caso 104 antes analizado como se observa en la figura 34.

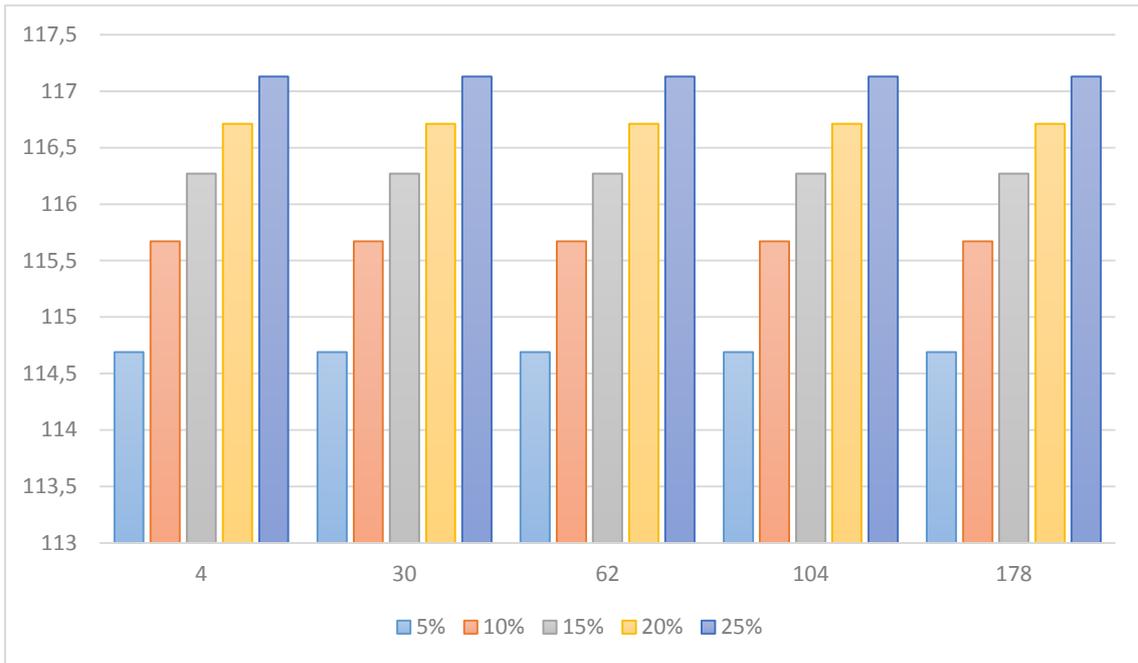


Figura 34: Potencia limitante tras eliminar la limitación de la rama 37-40

Por otro lado, si se realizan los cálculos para un 0%, y se compara con los demás porcentajes y el resultado de la actividad TLTG de PSS/E en el caso 4 por ejemplo, se tiene lo siguiente:

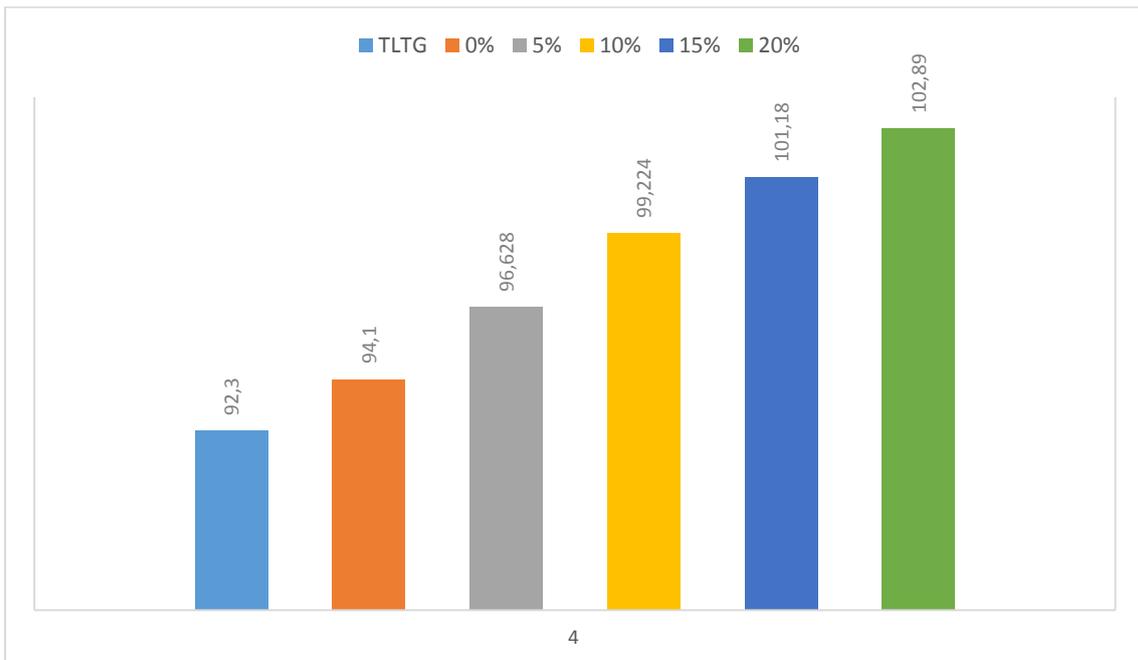


Figura 35: Potencia limitante en el caso 4 incluyendo el 0% y TLTG

INCR.	PRE-RATING		DISTR.		SHIFT BAS/CNT		CONTINGENCY DESCRIPTION	
TRANS	-----	LIMITING ELEMENT	-----	>				
CAPAB	-----	FROM	-----	<	-----	TO	-----	-----
					CKT	FACTOR	MW	A/A
359.6	37	39 NWLIBRTY	132	1	0.10493	92.3	130.0	OPEN
								37 [] TO 40 [WEST END 132] CKT 1

Figura 36: Resultado de la actividad TLTG

Se tiene en la figura 36 que la línea que dispara es la 37-40, la misma obtenida por métodos probabilísticos. Sin embargo no se obtiene el mismo valor limitante.

En el análisis probabilístico se utiliza la expresión basada en factores de distribución, que proporciona resultados menos exactos que los de la actividad TLTG, por lo que en determinados casos puede existir un ligero desequilibrio entre los datos obtenidos por métodos estadísticos y deterministas.

11. CONCLUSIONES Y LÍNEAS FUTURAS DE TRABAJO

- Para el análisis de la capacidad de evacuación en los nudos de una red de distribución en estado de contingencia simple N-1, es conveniente la realización de un estudio más allá de la punta y el valle. Con la formulación y metodología expuesta basta con calcular las funciones de distribución de probabilidad por cada línea, resolviendo repartos de carga para cada hora y escenario, y haciendo uso de la expresión:

$$\Delta P_k = \frac{P_{ij}^{m\acute{a}x} - P_x}{A_{ij}^k}$$

Donde P_x tal que $\mathcal{F}(P_{ij}^0 > P_y) = X$

Determinar ΔP_k para cada línea del sistema y seleccionar la más restrictiva.

- La aplicación de ésta metodología requiere comprobar si existe inversión de flujo. Es decir, se debe comprobar si la generación inyectada en el nudo k origina un flujo en la línea $i-j$ contrario al inicial. En la práctica se deben resolver dos inecuaciones:

$$\Delta P_k = \frac{P_{ij}^{m\acute{a}x} - P_x}{A_{ij}^k} \quad y \quad \Delta P_k = -\frac{P_y + P_{ij}^{m\acute{a}x}}{A_{ij}^k}$$

Donde P_y tal que $\mathcal{F}(P_{ij}^0 > P_y) = 1 - X = Y$

De las cuales se obtiene la potencia máxima evacuable y la demanda máxima conectable. Se tienen que comprobar los signos ya que la expresión que determina cada una depende de la dualidad nudo (k)-elemento delimitador ($i-j$).

- El análisis determinista realizado habitualmente (estudio clásico en punta y valle) puede no ser suficientemente restrictivo en determinados casos. En ocasiones existen escenarios en los que la carga por determinadas líneas supera el valor de la punta o el valle, por lo que si este elemento resulta ser limitante para la capacidad de evacuación no se estará considerando el escenario más desfavorable provocando situaciones de emergencia en periodos entre la punta y el valle que tendrán que ser gestionados con medidas de operación no previstas.

- La situación descrita anteriormente puede ser evitada mediante la realización de un análisis de probabilidad, ya que en estos casos no se toma el flujo en un determinado escenario, sino el valor que se supera un determinado número de horas al año. De esta forma basta con tomar el valor P_{ij}^0 que se supera, por ejemplo, 1% de las horas, evitando de ésta forma las sobrecargas descritas anteriormente.
- En los casos en los que no se presenta la situación anterior, el análisis de la evacuación desde el punto de vista probabilístico conduce a un incremento de la potencia evacuable respecto a la calculada en escenarios deterministas. Este aumento en ocasiones es moderado y en otras ocasiones es muy alto. Esto último sucede cuando el elemento limitador tiene una distribución de potencia muy aguda, es decir, con una punta o valle muy marcado, ya que esto conduce a un tramo final de la curva de probabilidad muy plano, que implica que asumiendo pocas horas de sobrecarga, se incrementa la potencia evacuable en una significativa cantidad. Teniendo en cuenta que a efectos prácticos tal aumento se debe a que el promotor de la planta asuma que durante unas determinadas horas al año (X % utilizado como umbral de la función de probabilidad) la planta de energía renovable no va a funcionar (o al menos no lo hará a plena potencia para no provocar sobrecargas en la red de distribución) cabe plantearse cuál sería el umbral para que desde el punto de vista económico al promotor le pueda resultar interesante aumentar la inversión inicial en potencia instalada, pese a que esta no pueda estar trabajando el 100% de las horas.
- El coste en tiempo de computación que ha supuesto todo el análisis es de unas 88 horas, repartidas entre:
 - 54 horas para las simulaciones
 - 24 horas para el tratamiento de los ficheros obtenidos en las simulaciones
 - 10 horas en el cálculo probabilístico
- La aplicación de esta metodología requiere el cálculo de numerosos datos. En el caso que nos ocupa:
 - Flujos P_{ij}^0 , 180 ramas x 8784 horas x 181 contingencias = 286.182.720 datos
 - Factores de distribución ρ_{ij}^k , 181 contingencias x 180 ramas x 118 nudos = 3.844.440 datos
 - La capacidad máxima de cada rama del sistema $P_{ij}^{máx}$, 181 datos

Esta cantidad de datos es manejable sin excesiva dificultad con una herramienta como MATLAB. Sin embargo para aplicar esta metodología a casos reales sería necesario desarrollar herramientas más potentes en la gestión de datos.

- Se puede plantear para futuros trabajos la posibilidad de optimizar el programa para poder manejar redes con un mayor número de nudos sin que ello conlleve un tiempo de computación excesivo. Además sería interesante implementar las simulaciones, el tratamiento y los cálculos en un mismo programa que evitase la intervención del usuario e hiciera todo el proceso automáticamente con solo introducirle la red a calcular y los estados de carga y generación para cada hora del año. También se podría hacer un análisis económico evaluando el beneficio de estar un porcentaje de horas al año a menor régimen o desconectado pero el resto producir mayor energía que el caso de no desconectar nunca la planta. También se podría plantear el refuerzo de una línea que limite mucho la potencia evacuable.

12. BIBLIOGRAFÍA

- Proyecto Fin de Máster: “LIMITACIÓN DE LA GENERACIÓN EN RÉGIMEN ESPECIAL EN UNA RED DE DISTRIBUCIÓN A PARTIR DE FUNCIONES DE DENSIDAD” . Autor: Rafael Cano Marín.
- Proyecto Fin de Máster: “ANÁLISIS PROBABILÍSTICO DE LA CAPACIDAD DE EVACUACIÓN EÓLICA EN ESTADO DE CONTINGENCIA SIMPLE” . Autor: Luis Chofre García.
- Program Operation Manual Vol. 1. PSS/E™ 33
- Program Application Guide Vol. 1. PSS/E™ 33
- Análisis y operación de sistemas de energía eléctrica. Antonio Gómez Expósito. Ed. Mc Graw Hill. ISBN:94-481-3592-X
- <https://psspy.org/psse-help-forum/questions/>
- <https://docs.python.org/2/tutorial/index.html>
- <http://librosweb.es/libro/python/>
- Proyecto Fin de Carrera: “CAPACIDAD DE EVACUACIÓN Y SUMINISTRO NODAL EN UN SISTEMA ELÉCTRICO” . Autor: Rubén del Caño Rojo. Universidad Carlos III.
- https://www.boe.es/diario_boe/txt.php?id=BOE-A-2013-13645
- <http://www.boe.es/buscar/doc.php?id=BOE-A-2000-24019>

ANEXO 1: SIMULACIONES PARA LA OBTENCIÓN DE DATOS

1. DEFINICIÓN DE FUNCIONES QUE SE UTILIZARÁN EN PYTHON PARA LA OBTENCIÓN DE DATOS

“Copiar_cargas (a, b)”

Descripción: Copia los valores de carga comprendidos entre las línea a y b del fichero original que se corresponden con los valores de cargas del sistema para la hora n .

```
def copiar_cargas(a,b):  
    i, j = a, b  
    g= open ('C: \Documents and  
Settings\profesor\Escritorio\PssDep\Reparto\Cargas.csv', 'rt')  
    f= open ('cargas_hora_n.csv','w')  
    spamreader = csv.reader (g, delimiter=' ', quotechar='|')  
    for row in itertools.islice (spamreader, i, j+1):  
        cs = ('', '.join (row))  
        f.write (cs)  
        f.writelines ('\n')
```

“Copiar_gen (c,d)”

Descripción: Misma función que “Copiar_cargas (a,b)” pero para la generación. Copia los valores de generación comprendidos entre las línea c y d del fichero original que se corresponden con los valores de generación del sistema para la hora n .

```
def copiar_gen(c,d):  
    i, j = c, d  
  
    #A partir de aquí la función es la misma que “Copiar_cargas (a,b)”
```

“Cargar_nueva_gen_cargas ()”

Descripción: Lee los archivos de carga y generación creados con las funciones

“Copiar_cargas (a,b)” y “Copiar_gen (c,d)” y carga los datos en el archivo

“118Nudos.raw” de PSS para simular la hora n .

```
def cargar_nueva_gen_cargas ():
    LOAD_DATAFILE = 'C:\Documents and
    Settings\profesor\Escritorio\PssDep\Reparto\cargas_hora_n.csv'

    GEN_DATAFILE = 'C:\Documents and
    Settings\profesor\Escritorio\PssDep\Reparto\generacion_hora_n.csv'

    redirect.psse2py()

    psspy.psseinit(10000)

    psspy.readrawversion(numnam=0, # treat bus numbers as unique
                        vernum='29.4', # read in version 29.4
                        ifile=r'C:\Documents and
    Settings\profesor\Escritorio\PssDep\118 Nudos.raw')

    data = list(csv.reader(open(LOAD_DATAFILE)))
    for isload, bus, _id, p, q, in data:
        p = float(p)
        q = float(q)
        bus = int(bus)
        if isload.isdigit() and int(isload):
            psspy.load_data_3(bus, _id, realar1=p, realar2=q)

    data1 = list(csv.reader(open(GEN_DATAFILE)))
    for isload, bus, _id, p, q, in data1:
        p = float(p)
        q = float(q)
        bus = int(bus)
        if isload.isdigit() and int(isload):
            0
        else:
            psspy.machine_data_2(bus, _id, realar1=p, realar2=q)
```

“Obtener_flujos(cont,i)”

Descripción: Copia en el fichero de salida “flujos_lin_tra.csv” los valores de los flujos después de realizar un reparto de cargas en la hora n . La variable “cont” especifica la hora y la “j” la rama que se está desconectando.

```
def obtener_flujos (cont,i):
    ierr, p = psspy.aflowreal (-1, 2, 1, 1, 'P')
    ierr, toBus = psspy.aflowint (-1, 2, 1, 1, 'TONUMBER')
    ierr, fromBus = psspy.aflowint (-1, 2, 1, 1, 'FROMNUMBER')
    ierr, brnID = psspy.aflowchar (-1, 2, 1, 1, 'ID')
    cont=str (cont)
    branches = []
    j=i+1
    j=str (j)
    csv = '-'
    for i in range (len(p[0])):
        if not (fromBus[0][i],toBus[0][i],brnID[0][i]) in branches:
            csv +=cont + ',' + j + ',' + '%i,%i,%s,%f\n' %
                (fromBus[0][i],toBus[0][i], brnID[0][i],p[0][i])
            branches.append ((toBus[0][i],
                fromBus[0][i],brnID[0][i]))

    print csv
    f= open ('flujos_lin_tra.csv','a')
    f.write (csv)
```

“Desc_lineas (j)”

Descripción: Desconecta la línea especificada por la variable “j” , realiza un reparto de cargas, llama a la función “obtener_flujos (cont,j)” para escribir el resultado y finalmente vuelve a conectar la línea inicialmente desconectada.

```
def desc_lineas (j)
    psspy.branch_data (fr[j],to[j],intgar1=0)
    psspy.fns1 ([0,0,0,1,1,0,99,0])
    obtener_flujos (cont,j)
    psspy.branch_data (fr[j],to[j],intgar1=1)
```

“Desc_trafos (l)”

Descripción: Realiza la misma función que la función “desc_lineas (j)” pero para los transformadores.

```
Def desc_trafos (l):
    psspy.two_winding_data_3 (frt[l],tot[l],intgar1=0)
    psspy.fns1 ([0,0,0,1,1,0,99,0])
    obtener_flujos (cont,l+171)
    psspy.two_winding_data_3 (frt[l],tot[l],intgar1=1)
```

2. PROGRAMAS QUE UTILIZAN LAS FUNCIONES ANTERIORES

Python Cabecera de programas

Descripción: conjunto de comandos necesarios para poder utilizar las funciones que se especifican a continuación.

```
import csv
import itertools
import os,sys

PSSE_LOCATION = r"C:\Archivos de programa\PTI\PSSE32\PSSBIN"
sys.path.append (PSSE_LOCATION)
os.environ ['PATH'] = os.environ ['PATH'] + ';' + PSSE_LOCATION

import psspy
import csv
import redirect
```

Python "FN-1"

Descripción: Determina los flujos por la líneas para las 8784 horas del año en cada uno de los 181 escenarios distintos.

```
cont = 1
a = 0
b = 105
c = 0
d = 21
j = 0
l = 0
while (cont<500): #numero de horas + 1 --> 8785
    copiar_cargas (a,b)
```

```

copiar_gen (c,d)
cargar_nueva_gen_cargas ()
for j in range (171): #numero de lineas
    desc_lineas (j)
for l in range(10): #numero de trafos
    desc_trafos (l)

cont=cont+1
a = a + 106
b = b + 106
c = c + 22
d = d + 22

```

Python "FactN-1"

Descripción: Obtiene los factores de nudos para los 118 casos distintos y los copia en el fichero de salida "FN.csv" .

```

redirect.psse2py ()
psspy.psseinit (10000)
psspy.readrawversion (numnam=0,
    vernum='29.4',
    ifile=r'C:\Documents and Settings\profesor\Escritorio\PssDep\118
    Nudos.raw')

l=0
b=0

while l<171:
    psspy.branch_data (fr[l],to[l],intgarl=0)
    psspy.dfax ([l,l],r""C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\subsistema.sub"",r""
    "C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\monitorizacion.mon""
    , r""C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\contingencias.con"",
    r""C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\dfax_nudos.dfx""")
    rlst = pssarrays.otdf_factors ('C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\dfax_nudos.dfx')

```

```

fd = rlst.factor
psspy.branch_data (fr[l],to[l],intgarl=1)
i=0
a=l+1
while (i<len (fd)):
    csv = '-'
    for fields in zip (fd[i]):
        j=i+1
        a=str (a)
        j=str (j)
        csv +=a + ',' + j + ',' + ','.join(map(str,fields)) +
        '\n'
    f= open ('FN.csv','a')
    f.write (csv)
    i=i+1
l=l+1

while b<10:
    psspy.two_winding_data_3 (frt[b],tot[b],intgarl=0)
    psspy.dfax ([1,1],r""C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\subsistema.sub"",r""
    "C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\monitorizacion.mon""
    , r""C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\contingencias.con"",
    r""C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\dfax_nudos.dfx""")
    rlst = pssarrays.otdf_factors ('C:\Documents and
    Settings\profesor\Escritorio\FactoresTodos\dfax_nudos.dfx')
    fd = rlst.factor
    psspy.two_winding_data_3 (frt[b],tot[b],intgarl=1)
    i=0
    a=l+1
    while (i<len (fd)):
        csv = '-'
        for fields in zip (fd[i]):
            j=i+1

```

```
        a=str (a)
        j=str (j)
        csv +=a + ',' + j + ',' + ','.join (map(str,fields))
        + '\n'
    f= open ('FN.csv','a')
    f.write (csv)
    i=i+1
l=l+1
b=b+1
```

Python "RateA"

Descripción: Calcula los $P_{ij}^{\text{máx}}$ para todas las ramas del sistema.

```
redirect.psse2py ()
psspy.psseinit (10000)
psspy.readrawversion (numnam=0, # treat bus numbers as unique
    vernum='29.4', # read in version 29.4
    ifile=r'C:\Documents and Settings\profesor\Escritorio\PssDep\118
    Nudos.raw')
ierr, RateA = psspy.abrreal (-1,1,1,4,1,'RATEA')
ierr, toBus = psspy.aflowint (-1, 2, 1, 1, 'TONUMBER')
ierr, fromBus = psspy.aflowint (-1, 2, 1, 1, 'FROMNUMBER')
print RateA
csv = '-'

for fields in zip (fromBus[0], toBus[0],RateA[0]):
    csv +=','.join (map(str,fields)) + '\n'
print csv
f = open ('RateA.csv','a')
f.write(csv)
```

ANEXO 2: TRATAMIENTO DE LOS DATOS OBTENIDOS

1. DEFINICIÓN DE FUNCIONES QUE SE UTILIZARÁN EN PYTHON PARA EL TRATAMIENTO DE DATOS

“Copiar_flujos (x,m,RUTA_ACTUAL,R_S,num_horas)”

Descripción: Abre el fichero obtenido con la función “FN-1” llamado “flujos_lin_tra.csv” que se especifica en la variable “RUTA_ACTUAL” . Va recorriendo el fichero en busca de los flujos del caso n y los va copiando en los ficheros de salida especificados por la variable “R_S” .

```
def copiar_flujos (x,m,RUTA_ACTUAL,R_S,num_horas):
    f=open (RUTA_ACTUAL,'r')
    g=open (R_S,'a')
    a=0
    i=0
    j=0
    y=0
    while i<num_horas:
        while y<m:
            f.readline ()
            y=y+1
        while j<180:
            pos=f.readline ()
            g.writelines (pos)
            j=j+1
        if j==180:
            j=0
        while a<(32580-180*(x+1)):
            f.readline ()
            a=a+1
        print i
        i=i+1
        a=0
        y=0
```

“Crear_fichero (R_S)”

Descripción: Crea un fichero con el nombre especificado por la variable “R_S”

```
def crear_fichero (R_S):  
    f = open (R_S,'w')  
    csv = ' '  
    f.write (csv)
```

2. PROGRAMA N-10 QUE UTILIZA LAS FUNCIONES ANTERIORES

“N-10”

Descripción: Programa que ordena los flujos de los ficheros “flujos_lin_tra_n.csv” y copia los resultados en 181 ficheros de salida del tipo “.csv” .

```
A = 'C:\Users\Pedro\Desktop\ResultadosFlujos\Flujos_lin_tra_1.csv'  
B = 'C:\Users\Pedro\Desktop\ResultadosFlujos\Flujos_lin_tra_2.csv'  
..  
RUTAS = [A, B,...]  
SALIDAS = ['F1.csv',... 'F181.csv']  
cont = 1  
a = 0  
i=0  
j=0  
num_horas=8784  
while j<181:  
    R_S = SALIDAS[j]  
    crear_fichero (R_S)  
    while i<len (RUTAS):  
        RUTA_ACTUAL = RUTAS[i]  
        copiar_flujos (j,a,RUTA_ACTUAL,R_S,num_horas)  
        i=i+1  
  
    i = 0  
    j = j + 1  
    a = 180*j
```

ANEXO 3: CÁLCULO DE ΔP_k CON MATLAB PARA UN ESCENARIO CONCRETO

"Convertir_FactN (FN, caso)"

Descripción: Lee el fichero "FN.csv" que contiene los factores de distribución del caso n definido por la variable "caso" e introduce un cero en la rama correspondiente a la contingencia m-n.

```
function FactNnew = Convertir_FactN(FN, caso)

FactNnew = ones (21358,2);
j = 1;
i = 1;
a = 1;
while j<21359

    if a == caso

        FactNnew(j,1) = i;
        FactNnew(j,2) = 0;

        i = i + 1;
        j = j + 1;

        caso = caso + 180;

    end

    FactNnew(j,1) = FN(a,2);
    FactNnew(j,2) = FN(a,3);

    j = j + 1;
    a = a + 1;

end

end
```

"Convertir_F (F_T, caso)"

Descripción: Lee la variable "F_T" que contiene los flujos por las ramas para el caso n definido por la variable "caso" e introduce un cero en la rama correspondiente a la contingencia m-n.

```
function FL_new = Convertir_F(F_T, caso)

    b = length(F_T) + (length(F_T)/180);
    FL_new = ones(b, 1);
    j = 1;
    i = 1;
    a = 1;
    while j < b

        if a == caso

            FL_new(j, 1) = 0;

            j = j + 1;

            caso = caso + 180;

        end

        FL_new(j, 1) = F_T(a, 6);

        j = j + 1;
        a = a + 1;

    end

end
```

1. DEFINICIÓN DE FUNCIONES QUE SE UTILIZARÁN EN MATLAB PARA LA OBTENCIÓN DE LA CONSTANTE A_{ij}^k

“Obtener (nudo, línea, FactN)”

Descripción: Devuelve el valor del factor correspondiente a la línea y nudo definidos por dichas variables de la lista de factores de distribución “FactN” .

```
function fact = obtener(nudo, linea, FactN)

    a = 181 * (nudo-1) + linea;

    fact = FactN(a,2);

end
```

“SUMGEN (línea,FactN,bo,RG)”

Descripción: Obtiene el sumatorio $\sum_{s=0}^n \rho_{ij}^s \gamma_{g_s}^k$ para una línea determinada.

```
function A = SUMGEN(linea, FactN, bo, RG)

    v = ones (1,22);

    for i = 1:22

        nudo = bo(i);

        v(1,i) = obtener(nudo, linea, FactN);

    end

    v = v';

    A = sum(v.*(RG));

end
```

“SUMGEN_TOTAL (FactN, bo, RG)”

Descripción: Calcula $\sum_{s=0}^n \rho_{ij}^s \gamma_{gs}^k$ para todas las líneas.

```
function [SGT] = SUMGEN_TOTAL (FactN,bo, RG)

    SGT = ones (181,1);

    for lin = 1:181

        SGT (lin,1) = SUMGEN (lin,FactN,bo, RG);

    end

end
```

“Mat_A (from2,to2,FactN,RG,bo)”

Descripción: Calcula la constante A_{ij}^k a partir de las funciones antes definidas.

```
function [MA] = Mat_A (from2,to2,FactN, RG,bo)

    SGT=SUMGEN_TOTAL (FactN,bo, RG);

    MA = ones (21358,6);

    a = 1;

    for i = 1:118

        j = 1;

        while j<182
            a = a + 1;
            MA (a-1,1) = i;
            MA (a-1,2) = from2 (j);
            MA (a-1,3) = to2 (j);
            MA (a-1,4) = FactN (a-1,2);
            MA (a-1,5) = SGT (j,1);

            if FactN (a-1,2) ==0

                MA (a-1,6) = 0;

            end

        end

    end

end
```

```
        else
            MA(a-1,6) = (FactN(a-1,2)+SGT(j,1));
        end
        j = j + 1;
    end
end
end
```

"vectA_n (from2,to2, FN, RG, bo)"

Descripción: Copia la última fila de la matriz A_{ij}^k antes calculada en un vector "vectA_n" para facilitar cálculos posteriores.

```
function vectA_n = vectA_n (from2, to2, FN, RG, bo)
    [MA] = Mat_A(from2, to2, FN, RG, bo);
    a = length(MA);
    vectA_n = ones(a,1);
    j = 1;
    while j<a+1
        vectA_n(j,1) = MA(j,6);
        j = j + 1;
    end
end
```

```
end
```

2. DEFINICIÓN DE FUNCIONES QUE SE UTILIZARÁN EN MATLAB PARA LA OBTENCIÓN DE LOS ΔP_k

"P_x_positivo (vect_cum,binranges,p)"

Descripción: Calcula el valor de P_x a partir del vector de probabilidades acumuladas "vect_cum" para valores positivos de los flujos por las líneas.

```
function [vq,index]=P_x_positivo(vect_cum,binranges,p)

    [c index] = min(abs(vect_cum-p));

    l=ones(1,2);
    m=ones(1,2);

    if min(vect_cum)>p

        vq = min(vect_cum);

    elseif max(vect_cum)<p

        vq = 0;

    elseif p>vect_cum(index)

        l(1,1) = vect_cum(index-1);
        l(1,2) = vect_cum(index);

        m(1,1) = binranges(index-1);
        m(1,2) = binranges(index);

        a = ((l(1,2)-l(1,1))/(m(1,2)-m(1,1)));

        vq = ((p-l(1,1)+a*m(1,1))/(a));

    else

        l(1,1) = vect_cum(index+1);
        l(1,2) = vect_cum(index);

        m(1,1) = binranges(index+1);
        m(1,2) = binranges(index);

        a = ((l(1,2)-l(1,1))/(m(1,2)-m(1,1)));

        vq = ((p-l(1,1)+a*m(1,1))/(a));

    end
```

"P_x_negativo (vect_cum,binranges,p)"

Descripción: Calcula el valor de P_x a partir del vector de probabilidades acumuladas "vect_cum" para los casos de inversión de flujos.

```
function [vq,index]=P_x_negativo(vect_cum,binranges,p)

    [c index] = min(abs(p-vect_cum));

    l=ones(1,2);
    m=ones(1,2);

    if min(vect_cum)>p

        vq = min(vect_cum);

    elseif p<vect_cum(index)

        l(1,1) = vect_cum(index-1);
        l(1,2) = vect_cum(index);

        m(1,1) = binranges(index-1);
        m(1,2) = binranges(index);

        a = ((l(1,2)-l(1,1))/(m(1,2)-m(1,1)));

        vq = ((p-l(1,1)+a*m(1,1))/(a));

    else

        l(1,1) = vect_cum(index+1);
        l(1,2) = vect_cum(index);

        m(1,1) = binranges(index+1);
        m(1,2) = binranges(index);

        a = ((l(1,2)-l(1,1))/(m(1,2)-m(1,1)));

        vq = ((p-l(1,1)+a*m(1,1))/(a));

    if vq == inf || vq == -inf

        l(1,1) = vect_cum(index+2);
        l(1,2) = vect_cum(index+1);

        m(1,1) = binranges(index+2);
        m(1,2) = binranges(index+1);

        a = ((l(1,2)-l(1,1))/(m(1,2)-m(1,1)));

        vq = ((p-l(1,1)+a*m(1,1))/(a));
    end
end
```

```
end
```

```
end
```

"Calculo_Delta_P (a, p, F_T, A, Pmax)"

Descripción: Calcula el valor de ΔP_k para una línea "a" y un porcentaje "p" .

```
function [DP] = Programa_Final(a,p,F_T,A,Pmax)

    X = Copiar_P_a_vector(a,F_T);

    XD1 = X(find(X>0));

    XD2 = X(find(X<0));

    fd1 = max(XD1);
    gd2 = round(min(XD2))-1;

    binrangesD1 = (0:fd1);
    binrangesD2 = (gd2:0);

    if isempty(XD2) || length(binrangesD2)==1

        binranges = (0:fd1);

        [repeticiones,posicion] = histc(XD1,binranges);

        vect_prob=((repeticiones/length(XD1))*100);

        vect_cum = 100 - cumsum(vect_prob);

        [vq,index]=P_x_positivo(vect_cum,binranges,p);

        if A == 0

            DP = 0;

        else

            DP = (Pmax-vq)/A;

        end
    end
```

```

elseif isempty(XD1) || length(binrangesD1)==1

    binranges = (gd2:0);

    [repeticiones, posicion] = histc(XD2, binranges);

    vect_prob = ((repeticiones/length(XD2))*100);

    vect_cum = cumsum(vect_prob);

    [vq, index] = P_x_negativo(vect_cum, binranges, p);

    if A == 0

        DP = 0;

    else

        DP = (-Pmax - vq) / A;

    end

else

    [repeticionesD1, posicion] = histc(XD1, binrangesD1);
    [repeticionesD2, posicion] = histc(XD2, binrangesD2);

    vect_probD1 = ((repeticionesD1/length(XD1))*100);
    vect_probD2 = ((repeticionesD2/length(XD2))*100);

    vect_cumD1 = 100 - cumsum(vect_probD1);
    vect_cumD2 = cumsum(vect_probD2);

    [vqD1, index] = P_x_positivo(vect_cumD1, binrangesD1, p);
    [vqD2, index] = P_x_negativo(vect_cumD2, binrangesD2, p);

    if A == 0

        DP = 0;

    else

        DP1 = (Pmax + vqD1) / A;
        DP2 = (-Pmax - vqD2) / A;

        if DP1 > 0 && DP2 < 0

            DP = DP1;

        end

    end

end

```

```

elseif DP2>0 && DP1<0
    DP = DP2;
elseif DP1>0 && DP2>0
    DP = min(DP1,DP2);

else
    DP=0;
end
end
end
end
end

```

"DeltaP_Final (vectA_n,p,F_T,Pmax, caso)"

Descripción: Calcula el valor de ΔP_k para todas las líneas y un porcentaje de confianza "p" .

```

function DP = DeltaP_Final (vectA_n,p,F_T,Pmax,caso)
    j = 1;
    DP = ones (1,21358);
    while j<21358
        i = 1;
        while i<182
            if i == caso
                DP(1,j) = 0;
                i = i + 1;
                j = j + 1;
            end
        end
        A = vectA_n(j,1);
    end
end

```

```

Pm = Pmax(i, 3);

DP(1, j) = Calculo_Delta_P(i, p, F_T, A, Pm);

i = i + 1;
j = j + 1;

end

end

end

```

"DeltaP_Porcentajes (vectA_n, F_T, Pmax, caso)"

Descripción: Calcula el valor de ΔP_k para todas las líneas y para los porcentajes "p" que se definan. En el presente estudio 5%, 10%, 15%, 20% y 25%.

```

function DP_T = DeltaP_Porcentajes (vectA_n, F_T, Pmax, caso)

DP5 = DeltaP_Final (vectA_n, 5, F_T, Pmax, caso);
DP10 = DeltaP_Final (vectA_n, 10, F_T, Pmax, caso);
DP15 = DeltaP_Final (vectA_n, 15, F_T, Pmax, caso);
DP20 = DeltaP_Final (vectA_n, 20, F_T, Pmax, caso);
DP25 = DeltaP_Final (vectA_n, 25, F_T, Pmax, caso);

DP_T = [DP5; DP10; DP15; DP20; DP25];

end

```

ANEXO 4: FUNCIONES DE MATLAB PARA CALCULAR LOS 181 ESCENARIOS

1. DEFINICIÓN DE LAS FUNCIONES NECESARIAS PARA REALIZAR LOS CÁLCULOS DESCRITOS EN EL PUNTO 8 PARA LOS 181 ESCENARIOS.

"SIMULA_N (caso)"

Descripción: Programa que realiza lo siguiente:

- Carga todos los ficheros necesarios referentes al caso definido por dicha variable
- Define las variables de entrada
- Ejecuta las funciones necesarias para calcular ΔP_k
- Genera un fichero de salida con el formato "DPKn.csv" donde se guardan los ΔP_k del caso n

```
function SIMULA_N(caso)

    %Carga de Archivos
    FILE1 = sprintf('F%d-000.csv', caso);
    FILE2 = sprintf('F%d-001.csv', caso);
    FILE3 = sprintf('FN (%d).csv', caso);

    FN = csvread(FILE3);
    FactN = Convertir_FactN(FN, caso);

    FL1_1 = csvread(FILE1);
    FL1_2 = csvread(FILE2);
    FL_new = [FL1_1; FL1_2];
    F_T = Convertir_F(FL_new, caso);

    Pmax = csvread('C:\Users\Pedro\Desktop\DatosFlujos\RateA.csv');

    %Carga de Constantes
    bus = [100, 10, 103, 106, 111, 12, 25, 26, 31, 32, 34, ...
          46, 49, 54, 59, 61, 65, 66, 69, 80, 87, 89];

    bo = sort(transpose(bus))';

    RG =
    [0.0200; 0.0450; 0.0270; 0.0620; 0.0020; 0.0470; 0.0270; 0.0050; 0.0670;
     0.0400; 0.0410; 0.0430; 0.0800; 0.1040; -
     0.0010; 0.1270; 0.0010; 0.1450;
     0.0400; 0.0370; 0.0310; 0.0100];

    from2 %Vector con el "from" de las ramas
    to2   %Vector con el "to" de las ramas
```

```
%Inicio de los Cálculos
vectA_ = vectA_n (from2,to2,FactN,RG,bo);

DP_T = DeltaP_Total(vectA_,F_T,Pmax,caso);

%Escritura de Resultados en un Fichero Externo

FILE4 = sprintf('DPK%d.csv', caso);

csvwrite(FILE4,DP_T');

clc
clear all

end
```

“SIMULA_TODO”

Descripción: Programa que cuenta con un bucle que se repite 181 veces (el número de casos) para simular en cada repetición la función “SIMULA_N” antes definida.

```
function SIMULA_TODO

    i = 1;

    while i<182

        SIMULA_N(i);

        i = i + 1;

    end
```

2. DEFINICIÓN DE LAS FUNCIONES NECESARIAS PARA ANALIZAR LOS DATOS OBTENIDOS

"CARGA_DELTA_P_K (caso)"

Descripción: Carga todos los archivos generados de todos los casos.

```
function MAT_CUB_TOTAL=CARGA_DELTA_P_K(caso)
```

```
    FILE1 = sprintf('DPK%d.csv', caso);
```

```
    DPK = csvread(FILE1);
```

```
end
```

"MATRIZ_CUBICA"

Descripción: Copia los archivos cargados en una matriz cúbica para su análisis.

```
function DPKT=MATRIZ_CUBICA
```

```
    i=1;
```

```
    DPKT = ones (181,590,181);
```

```
    while i<182
```

```
        DPK=CARGA_DELTA_P_K(i);
```

```
        DPKT(:, :, i) = DPK;
```

```
        i = i + 1;
```

```
end
```

calcula_minimos (MAT_CUB_TOTAL)

Descripción: calcula los valores mínimos de ΔP_k para cada nudo y caso.

```
function Mini = calcula_minimos (MAT_CUB_TOTAL)

M = (MAT_CUB_TOTAL);

Mini5 = ones (1,118);
Mini10 = ones (1,118);
Mini15 = ones (1,118);
Mini20 = ones (1,118);
Mini25 = ones (1,118);

i=1;

while i<119

    C = M(:,i);
    m = min(C(C>0));
    if isempty(m)

        Mini5(i)=0;
        i = i + 1;

    else

        Mini5(i) = m;
        i = i + 1;

    end

end

i = 119;
j = 1;

while i<237

    C = M(:,i);
    m = min(C(C>0));
    if isempty(m)

        Mini10(j)=0;
        i = i + 1;
        j = j + 1;

    else

        Mini10(j) = m;
        i = i + 1;
        j = j + 1;

    end

end
```

```
i = 237;
j = 1;

while i<355

    C = M(:,i);
    m = min(C(C>0));
    if isempty(m)

        Mini15(j)=0;
        i = i + 1;
        j = j + 1;

    else

        Mini15(j) = m;
        i = i + 1;
        j = j + 1;

    end

end

i = 355;
j = 1;

while i<473

    C = M(:,i);
    m = min(C(C>0));
    if isempty(m)

        Mini20(j)=0;
        i = i + 1;
        j = j + 1;

    else

        Mini20(j) = m;
        i = i + 1;
        j = j + 1;

    end

end

i = 473;
j = 1;

while i<591

    C = M(:,i);
    m = min(C(C>0));
    if isempty(m)
```

```
        Mini25(j)=0;
        i = i + 1;
        j = j + 1;

    else

        Mini25(j) = m;
        i = i + 1;
        j = j + 1;

    end

end

Mini = [Mini5;Mini10;Mini15;Mini20;Mini25];

end
```

"ordena_resultados_sin_ceros(DPK)"

Descripción: ordena los archivos cargados para su posterior tratamiento.

```
function [MAT_CUB_5, MAT_CUB_10, MAT_CUB_15, MAT_CUB_20, MAT_CUB_25] =
ordena_resultados_sin_ceros(DPK)

    MAT_CUB_5 = ones(181,118);

    a = 1;
    j = 1;
    i = 1;
    b=DPK(:,1);

    while j<21358

        i = 1;

        while i<182

            if b(j,1)<0

                MAT_CUB_5(i,a) = 0;

                i = i + 1;
                j = j + 1;

            else
```

```
MAT_CUB_5(i,a)=b(j,1);

    i = i + 1;
    j = j + 1;

end

end

a = a + 1;

end

MAT_CUB_10 = ones(181,118);
b=DPK(:,2);
a = 1;
j = 1;
i = 1;

while j<21358

    i = 1;

    while i<182

        if b(j,1)<0

            MAT_CUB_10(i,a) = 0;

            i = i + 1;
            j = j + 1;

        else

            MAT_CUB_10(i,a)=b(j,1);

            i = i + 1;
            j = j + 1;

        end

    end

    a = a + 1;

end

MAT_CUB_15 = ones(181,118);
b=DPK(:,3);
a = 1;
j = 1;
i = 1;

while j<21358
```

```
i = 1;

while i<182

    if b(j,1)<0

        MAT_CUB_15(i,a) = 0;

        i = i + 1;
        j = j + 1;

    else

        MAT_CUB_15(i,a)=b(j,1);

        i = i + 1;
        j = j + 1;

    end

end

a = a + 1;

end

MAT_CUB_20 = ones(181,118);
b=DPK(:,4);
a = 1;
j = 1;
i = 1;

while j<21358

    i = 1;

    while i<182

        if b(j,1)<0

            MAT_CUB_20(i,a) = 0;

            i = i + 1;
            j = j + 1;

        else

            MAT_CUB_20(i,a)=b(j,1);

            i = i + 1;
            j = j + 1;

        end

    end

end
```

```
        end

    end

    a = a + 1;

end

MAT_CUB_25 = ones(181,118);
b=DPK(:,5);
a = 1;
j = 1;
i = 1;

while j<21358

    i = 1;

    while i<182

        if b(j,1)<0

            MAT_CUB_25(i,a) = 0;

            i = i + 1;
            j = j + 1;

        else

            MAT_CUB_25(i,a)=b(j,1);

            i = i + 1;
            j = j + 1;

        end

    end

    a = a + 1;

end

end
```

“max_p (MAT_CUB_5,MAT_CUB_10,MAT_CUB_15,MAT_CUB_20,MAT_CUB_25)”

Descripción: busca los máximos por fila y los guarda en un vector para cada porcentaje.

```
function [Max5,Max10,Max15,Max20,Max25]=
max_p (MAT_CUB_5,MAT_CUB_10,MAT_CUB_15,MAT_CUB_20,MAT_CUB_25)

    Max5=calcula_max (MAT_CUB_5);
    Max10=calcula_max (MAT_CUB_10);
    Max15=calcula_max (MAT_CUB_15);
    Max20=calcula_max (MAT_CUB_20);
    Max25=calcula_max (MAT_CUB_25);

end
```

“Min_Glob_un_porcentaje (Max_Porcentajes)”

Descripción: De los máximos por fila, obtiene el mínimo de todos ellos

```
function Min_porcentaje = Min_Glob_un_porcentaje (Max_Porcentajes)

    Min_porcentaje = ones (1);

    C=Max_Porcentajes (:,1);
    Min_porcentaje (1) = min (C (C>0));

end
```

“Maximo_Casos”

Descripción: A partir de las funciones definidas en el ANEXO 4, calcula el valor máximo de evacuación para los 181 casos y los 5 porcentajes.

```
function Max_N_1=Maximo_Casos

    i=1;
    Max_N_1 = ones (5,181);

    while i<181

        DPK=CARGA_DELTA_P_K (i);
```

```
[MAT_CUB_5, MAT_CUB_10, MAT_CUB_15, MAT_CUB_20, MAT_CUB_25] =  
ordena_resultados_sin_ceros(DPK);
```

```
[Max5, Max10, Max15, Max20, Max25] =  
max_p(MAT_CUB_5, MAT_CUB_10, MAT_CUB_15, MAT_CUB_20, MAT_CUB_25);
```

```
Min_5 = Min_Glob_un_porcentaje(Max5);  
Min_10 = Min_Glob_un_porcentaje(Max10);  
Min_15 = Min_Glob_un_porcentaje(Max15);  
Min_20 = Min_Glob_un_porcentaje(Max20);  
Min_25 = Min_Glob_un_porcentaje(Max25);
```

```
Max_N_1(1, i) = Min_5;  
Max_N_1(2, i) = Min_10;  
Max_N_1(3, i) = Min_15;  
Max_N_1(4, i) = Min_20;  
Max_N_1(5, i) = Min_25;
```

```
i = i + 1;
```

```
end
```