

MODELO DE SIMULACIÓN *SOFT-COMPUTING* PARA LA SELECCIÓN DE CONTINGENCIAS CRÍTICAS EN LA SEGURIDAD DINÁMICA DE SISTEMAS DE POTENCIA SOPORTADO EN REDES NEURONALES DE RETROPROPAGACIÓN SIMPLE Y MÚLTIPLE

GUSTAVO SCHWEICKARDT¹ - CARLOS CASANOVA¹ - EDUARDO PÉREZ²

(1) CONICET/Universidad Tecnológica Nacional - FRCU, Entre Ríos, Argentina

(2) UBA/Universidad Tecnológica Nacional - FRCU, Entre Ríos, Argentina
gustavoschweickardt@conicet.gov.ar - casanovac@frcu.utn.edu.ar - perez@frcu.utn.edu.ar

Fecha Recepción: Febrero 2015 - Fecha Aceptación: Julio 2015

RESUMEN

Un importante número de Contingencias simuladas en la Evaluación de la Seguridad Dinámica de un Sistema de Potencia (ESDSP), no afectan de manera significativa las variables de estado. Su exclusión del conjunto que requiere analizarse, permitiría una importante reducción en los tiempos de computación, posibilitando la Evaluación En Línea (Tiempo Real) adoptando acciones correctivas sólo sobre las Contingencias consideradas Críticas. El Método clásico de Selección de Contingencias Críticas resulta de un Análisis Fuera de Línea, que cubre escenarios típicos considerando aspectos tales como: frecuencia del sistema, tensiones de barra y ángulos internos de los generadores. Sobre tal selección, el operador experto implementa las acciones correctivas pertinentes. En este trabajo es propuesto un nuevo Modelo *Soft-Computing*, que identifica Contingencias Críticas en Línea para la ESDSP, soportado en Redes Neuronales de Retropropagación Simple y Múltiple (RNRS/RNRM), Conjuntos Difusos y MatHeurísticas. Las RNRM presentan la capacidad de fraccionar el espacio de entrada, utilizando neuronas selectivas. Se disminuye el tiempo de entrenamiento y se logra aproximar mejor características no lineales en regiones localizadas, así como la interpolación dentro del rango de entrenamiento. Los resultados obtenidos son presentados y discutidos, para lo cual es utilizada una red reducida del Sistema de Interconexión Argentino (SADI).

PALABRAS CLAVE: Simulación – Soft Computing – Seguridad Dinámica – Sistemas de Potencia – Redes Neuronales – Retropropagación Múltiple.

ABSTRACT

An important number of contingencies simulated during Dynamic Security Assessment of a Power System (DSAPS), do not result in unacceptable values of state variables, due to their small influence on system

operation. Their exclusion from the Set of Contingencies to be analyzed, would achieve a significant reduction in computation time. The standard Selection Method results from an Off-Line Dynamical Analysis, which covers typical scenarios and various related aspects like frequency, voltage, and internal angles of generators. In this work, a new Soft Computing-based Critical Contingencies Selection Method for On-Line Dynamic Security Assessment, supported in Simple and Multi-Backpropagation Neural Networks (SBNN/MBNN), Fuzzy Sets and MatHeuristics, is presented. The MBNN have the capability of split the input space, using selective neurons, achieving a better adjust in the no linear characteristics over localized regions, improving the computation time and interpolation into the training range. The results of a simulation Model on the reduced network of Interconnected Power System of Argentina (IPS), are presented.

KEYWORDS: Simulation – Soft Computing – Dynamic Security – Power Systems – Neural Networks – Multiple Backpropagation.

1. INTRODUCCIÓN

La importancia fundamental de la evaluación de la Seguridad Dinámica de Operación de un Sistema de Potencia (ESDSP) estriba en que, si el mismo se encuentra operando en un estado inseguro, debe ser llevado a un estado de operación más seguro. Esto se logra mediante adecuadas acciones de control, las cuales deben ser determinadas y aplicadas. La ESDSP se lleva a cabo simulando Contingencias del Sistema. Por Contingencia, debe entenderse aquel evento que pone en riesgo la operación segura del SP. Las Contingencias típicas a simular son: a) la salida de servicio imprevista de generadores o de líneas de transmisión y b) la pérdida imprevista de grandes cargas. Son, luego, estimadas las consecuencias o severidad de cada Contingencia, observando las Variables de Estado del SP: frecuencia, tensiones de nodo, flujos de potencia/intensidad por las líneas de transmisión, y ángulos internos de los generadores. Se realiza mediante la comparación de sus valores respecto de los correspondientes a la operación normal o segura.

La ESDSP puede implementarse utilizando métodos aproximados o mediante simulaciones completas de todas y cada una de las posibles Contingencias (Morison, Wang y Kundur, 2004). Los métodos aproximados, seleccionan un conjunto de Contingencias tales que, por experiencia del operador del SP, se presume que llevarán al sistema a un estado de operación no seguro. Son empleadas herramientas estacionarias para esta selección, fundamentalmente el Flujo de Potencia AC (en Corriente Alterna) o CC (en Corriente Continua). Constituyen un enfoque poco confiable, y prácticamente no es utilizado, excepto en SPs de pequeño porte o aislados. La simulación completa, en cambio, toma la Lista Inicial de Contingencias a analizar, y emplea herramientas no estacionarias, tales como el Análisis de

Estabilidad del SP y el Flujo de Potencia Óptimo (Morison *et al.*, 2004). Para ambos métodos, resulta una Lista de Contingencias que llevan al SP a un estado de operación no seguro. Estos cálculos son realizados Fuera de Línea, y para cada cambio de importancia en la topología, generación o demanda del SP, deben repetirse. En Línea, sobre tal lista, al presentarse una de tales Contingencias, son adoptadas las acciones correctivas pertinentes en el SP. Por tal motivo reciben el nombre de Contingencias Críticas (Corteggiano y Mercado, 1998).

Si se pretende realizar una ESDSP En Línea, los métodos aproximados quedan excluidos y las simulaciones completas de la Lista Inicial de Contingencias resultan inviables debido a los tiempos que demandan los cálculos, respecto de la ventana temporal de toma de decisión para dar curso a las acciones correctivas sobre el SP (< 20 seg) (Morison, Wang y Kundur, 2005) y (Gimenez Álvarez y Mercado, 2007).

De manera que el problema de la ESDSP En Línea, consiste en identificar las Contingencias Críticas, dentro de la ventana temporal de toma de decisión, más allá de la cual el SP puede perder su sincronismo y colapsar.

Desde estas consideraciones, se ha propuesto la introducción de técnicas basadas en Inteligencia Artificial Computacional (Mc Calley y Krause, 1995), siendo este enfoque el adoptado para el presente trabajo.

Como antecedente, un Modelo *Soft-Computing* para este problema es propuesto en (Schweickardt y Gimenez, 2012), pero utilizando Redes Neuronales de Retropropagación Simple y Metaheurísticas. De manera que el presente trabajo constituye un avance sobre tal propuesta de solución. Adicionalmente, son considerados aquí, Cambios Topológicos en el SP (por caso, se agrega una nueva línea de transmisión).

El trabajo está organizado como sigue: en la Sección 2 se presenta una descripción breve de una Red Neuronal de Retropropagación, primero Simple (RNRS) y luego Múltiple (RNRM). En la Sección 3 se presenta la estrategia seguida para la identificación de la Contingencias Críticas y el desarrollo del Modelo ESDSP En Línea. Se describen los Índices Pre y Pos-Contingencia. Se detallan los pasos para componer los Patrones de Entrada y de Salida para el entrenamiento de las dos redes neuronales que trabajan acopladas en el Modelo, y se propone una heurística para integrar el conocimiento experto en la RNRM, cuya función es específicamente la identificación de Contingencias Críticas. En la Sección 4 se valida el Modelo y se presentan simulaciones sobre el SADI. Finalmente, en la Sección 5, se desarrollan las conclusiones más relevantes del trabajo.

2. REDES NEURONALES DE RETROPROPAGACIÓN (RNR)

2.1. Redes Neuronales de Retropropagación Simple (RNRS)

Las Redes Neuronales de Retropropagación (RNRS) (aquí les es adicionado el calificativo *Simple* a efectos de diferenciarlas del modelo *Múltiple* presentado) son sistemas de regresión no paramétricos que permiten

resolver relaciones complejas de múltiples variables, muy a menudo difíciles de expresar matemáticamente o mediante reglas proposicionales. En general, cualquier problema que requiera de una función que caracterice a un sistema, asociando patrones de entrada con otros de salida, exhibe, como alternativa, una solución en base a RNRS. La limitación en el tipo de función así aproximada, viene impuesta por el llamado Teorema de Funahashi (Funahashi, 1989).

2.2. Aprendizaje por Retropropagación

El algoritmo entrenamiento (o aprendizaje) de (por) retropropagación, es un algoritmo supervisado que ha sido aplicado exitosamente en la resolución de problemas de muy diferentes tipos. Es el más exitoso y conocido en el entrenamiento de Perceptrones Multicapa. Puede encontrarse una explicación muy detallada en (Haykin, 1999). Siguiendo la referencia (Lopes y Ribeiro, 2001), pues estos autores proponen también en ella la retropropagación múltiple, se formula sintéticamente como sigue:

Dado un vector de entradas $[x]$, un vector de salidas $[y]$, y un patrón actual de entrenamiento p , la salida para una determinada neurona k , (y_k^p), con N conexiones de entrada, cuando tal patrón p es presentado a la red, está dada por una función de activación $F(a_k^p)$, siendo a_k^p su activación, expresada mediante:

$$y_k^p = F(a_k^p) = F_k \left(\sum_{j=1}^N \omega_{jk} y_j^p + \theta_k \right) \quad (1)$$

Donde: ω_{jk} representa el peso asociado a la conexión entre la neurona j y la neurona k y θ_k el *bias*. Nótese que en el caso de la capa de entrada, y_k^p es igual a x_k^p . La regla del Gradiente Descendente, propuesta originalmente como método, minimiza el error cuadrático medio dado por la función:

$$E^p = \frac{1}{2} \times \sum_{o=1}^{N_o} (d_o^p - y_o^p)^2 \quad (2)$$

donde d_o^p es la salida deseada para cuando el patrón p es presentado a la red y y_o^p es la salida de la neurona. Siguiendo el proceso de minimización, los pesos de la red son ajustados mediante la expresión:

$$\Delta_p \omega_{jk} = \gamma \times \delta_k^p \times y_j^p + \alpha \times \Delta_q \omega_{jk} \quad (3)$$

Donde: $\Delta_p \omega_{jk}$ es el ajuste resultante sobre el peso ω_{jk} , γ es la *tasa de aprendizaje*, parámetro fijado externamente, δ_k^p es el gradiente local de la neurona k , $\Delta_q \omega_{jk}$ es el cambio logrado en el peso ω_{jk} respecto del anterior patrón, q , presentado a la red, y α se denomina el *factor de momento*.

2.3 Aprendizaje en Redes de Retropropagación Múltiple (RNRM)

Cuando el mapeo de una función f , correspondiente a un determinado dominio de entrada y su imagen, presenta discontinuidades, el algoritmo de retropropagación clásico o simple falla. Este tiene un buen desempeño con funciones suaves. Para generalizarlo a funciones suaves a trozos, es utilizada una partición del espacio en cuyas divisiones la función resulta suave. La idea

propuesta por (Lopes y Ribeiro, 2001), mejora la capacidad predictiva mediante una mejor interpolación (mejora en las capacidades de generación y aproximación de funciones) en el rango de entrenamiento de la red, con un menor tiempo de aprendizaje respecto de la retropropagación simple. Consiste en activar las neuronas selectivamente, a fin de lograr que parte de lo aprendido se comparta entre las particiones del dominio de entrada. La contribución de una neurona genérica k en la salida de la red, cuando cierto patrón p es presentado, se modifica introduciendo un *factor o coeficiente de activación*, $m_k^p \geq 0$, en la ecuación de salida (1):

$$y_k^p = m_k^p \times F(a_k^p) = m_k^p \times F_k \left(\sum_{j=1}^N \omega_{jk} y_j^p + \theta_k \right) \quad (4)$$

Se observa que en la medida que mayor resulte m_k^p , más importante resulta la contribución de la neurona k . En el caso que su valor sea 0, el resultado de la contribución de dicha neurona resultará nulo.

En (Lopes y Ribeiro, 2001) los autores combinan una *red neuronal principal* conteniendo neuronas con activación selectiva y una *red secundaria (space network)* responsable de la determinación de los *factores de activación* m_k^p , permitiendo la división del espacio de entrada.

Estas redes, entrenadas con el algoritmo de Retropropagación Múltiple (RM), muestran tener mejores propiedades de generalización respecto de las redes *feed-forward* (o de propagación hacia adelante clásica, concepto opuesto a *recurrente*, donde la información fluye o se propaga en ambos sentidos), entrenadas con el algoritmo de Retropropagación Simple (RS). El RM ajusta los pesos de ambas redes y, mediante la modificación de los pesos de la red secundaria, cambia la importancia atribuida a cada neurona de la red principal, logrando la activación selectiva de las mismas. Así, las dos redes funcionan complementariamente, componiendo una nueva arquitectura referida como Red de Retropropagación Múltiple (RNRM). La red principal, puede calcular sus salidas una vez calculada la salida de la red secundaria.

Los pesos de la red principal son ajustados mediante (3), pero con formulaciones diferentes para los gradientes de las capas ocultas ($h = \textit{hidden}$) y de salida ($o = \textit{output}$).

Gradiente para la capa de salida:

$$\delta_o^p = (d_o^p - y_o^p) \times m_o^p \times F_o'(a_o^p) \quad (5)$$

Gradiente para las capas ocultas:

$$\delta_h^p = m_h^p \times F_h'(a_h^p) \times \sum_{o=1}^{N_o} \delta_o^p \times \omega_{ho} \quad (6)$$

Los pesos en la red secundaria deben ajustarse a fin de minimizar los errores de salida de la red principal:

$$\Delta_p m_k^p = - \frac{\partial E^p}{\partial m_k^p} \quad (7)$$

$$\Delta_p m_o^p = (d_o^p - y_o^p) \times F_o'(a_o^p) \quad (8)$$

$$\Delta_p m_h^p = \sum_{o=1}^{N_o} \delta_o^p \times \omega_{ho} \times F_h(a_h^p) \tag{9}$$

Basándose en (8) y (9) es posible encontrar la salida deseada para la red secundaria, $m_k^p + \Delta_p m_k^p$.

En las FIGURA 1 y 2 se presentan las estructuras de una Red Neuronal de Retropropagación Simple y Múltiple. Se omite la representación de los *biases* o “pesos propios” de cada neurona.

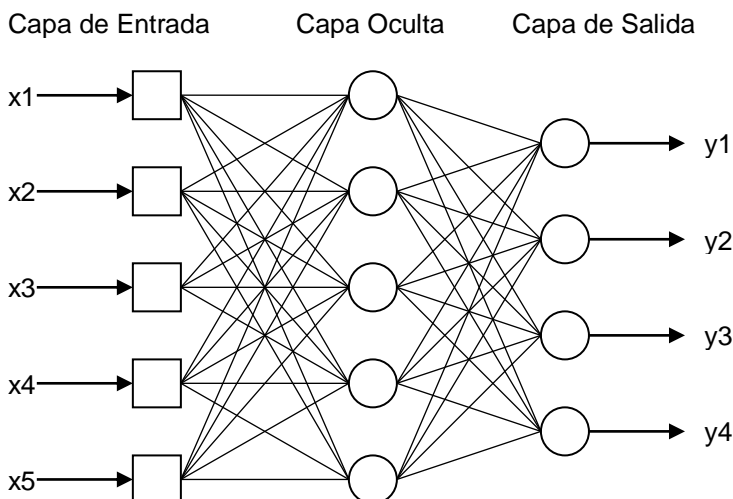


FIGURA 1: Arquitectura de una Red Neuronal de Retropropagación Simple (RNRS) con 5 entradas, 1 capa oculta – 5 neuronas - y 4 salidas.

En la FIGURA 2, se observa una red principal (superior) y una secundaria (inferior). En la red principal existen dos contribuciones para el error de salida: (i) los pesos de las conexiones en la red principal; y (ii) los *factores de activación* de cada neurona, desde las conexiones de la red secundaria, m_k^p . Los círculos con X, representan *multiplicadores* de m_k^p (según (4)).

El algoritmo de Retropropagación Múltiple (MBP) empleado en el presente trabajo, se encuentra disponible como una librería *Open Source*, para ser utilizado con procesamiento paralelo masivo (Lopes y Ribeiro, 2011). El *Hardware* necesario para correr, es una PC de escritorio (o *notebook* de características avanzadas), que cuente con una placa gráfica que brinde la capacidad de procesamiento de su GPU. Actualmente esta librería, llamada GPULib, implementa los algoritmos de Retropropagación Simple (BP), Retropropagación Múltiple (MBP), Factorización Matricial No Negativa (NMF) y Redes de Base Radial (RBF).

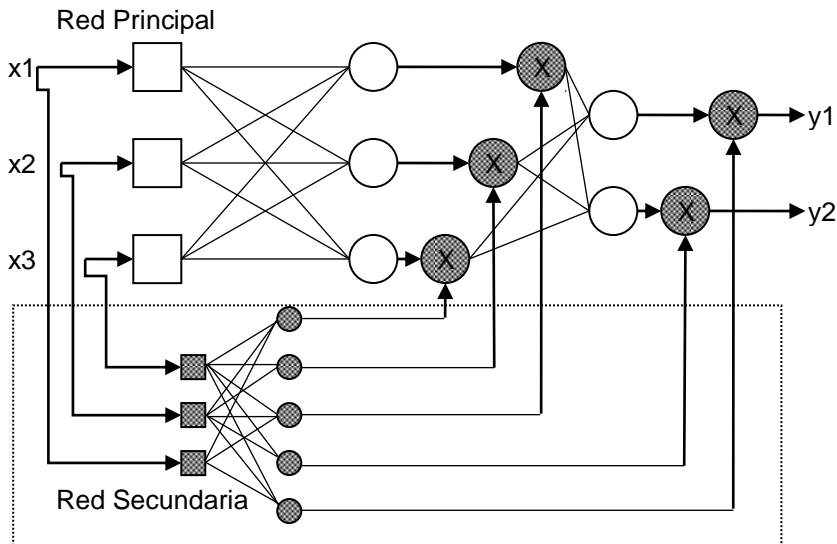


FIGURA 2: Arquitectura de Retropropagación Múltiple (RNRM).

3. MODELO DE SIMULACIÓN PROPUESTO

El enfoque adoptado en este trabajo consiste en obtener un índice que estime el valor de la Función de Seguridad, FS, del SP ante una Contingencia. Dependiendo del valor de FS, una Contingencia será identificada (o no) como Crítica. Para ello se requieren los desarrollos presentados a continuación.

3.1. Indicadores de Estado Pre-Contingencia

Los indicadores del estado *Pre-Contingencia* expresan el estado del SP antes de la Contingencia, o bien ocurrida la misma pero evaluándolo sin emplear simulaciones transitorias. Se dividen en tres grupos:

1) Índice de Margen de Flujo de Intensidad (MI) e Índice de Margen de Tensión (MTI): el índice MI proporciona un valor de medida respecto del margen de capacidad del flujo de intensidad. El índice MTI mide el margen de cambio de tensión global. Sus expresiones son las siguientes:

$$MI = \left(\sum_{i=1}^{NL} w_i \times \left(\frac{l_i^{pre}}{l_i^{lim}} \right)^2 \right) / \sum_{i=1}^{NL} w_i \quad (10)$$

donde: NL es el Número de líneas, \$l_i^{pre}\$ es la Intensidad en la línea \$i\$, \$l_i^{lim}\$ es el Límite de Intensidad en la línea \$i\$ y \$w_i\$ es un Factor de Peso de Línea, variable en \$[0, 1]\$, y definido por el operador del SP.

$$MTI = \frac{\sum_{i=1}^{NN} \left(wn_i \times \left(\frac{V_i^{pre} - V_i^n}{\Delta V_i^{lim}} \right)^2 \right)}{\sum_1^{NN} wn_i} \quad (11)$$

donde: NN es el Número de Nodos, V_i^{pre} es la Tensión en el Nodo i antes de la contingencia, V_i^n es la Tensión Nominal en el Nodo i, ΔV_i^{lim} es el Límite de Desviación de la Tensión en el Nodo i (Ejemplo: 5% V_i^n) y wn_i es un Factor de Peso del Nodo, variable en [0, 1], también definido por el operador del SP.

2) Índice de Flujo de Intensidad Cuasi-Estacionario (MQI) e Índice de Tensión Cuasi-Estacionario (MTQI): evalúan el sistema considerando una Contingencia, pero *calculando el estado del sistema usando herramientas estacionarias* (Flujo de Potencia AC). Sus expresiones son las siguientes:

$$MQI = \frac{\sum_1^{NL} \left(wl_i \times \left(\frac{I_i^{Cuasi}}{I_i^{lim}} \right)^2 \right)}{\sum_1^{NL} wl_i} \quad (12)$$

donde: I_i^{Cuasi} es la Intensidad en la línea i después de la Contingencia.

$$MTQI = \frac{\sum_1^{NN} \left(wn_i \times \left(\frac{V_i^{Cuasi} - V_i^n}{\Delta V_i^{lim}} \right)^2 \right)}{\sum_1^{NN} wn_i} \quad (13)$$

donde: V_i^{Cuasi} es la Tensión en el nodo i después de la Contingencia.

3) Índice de Generación Desconectada (GI): evalúa la magnitud de la Contingencia comparando la generación desconectada a causa de la misma, con la generación total del sistema. Su expresión es la siguiente:

$$GI = (G_{out} / G_{total}) \times Fp \quad (14)$$

donde: G_{out} es la Potencia del generador desconectado, G_{total} es la Generación total del SP y Fp es un Factor de Peso, variable en [0, 1].

3.2. Indicadores de Estado Pos-Contingencia

Estos indicadores definen el estado del SP *luego de una Contingencia*. Requieren, así, de *simulaciones dinámicas*. Se dividen en *cuatro grupos*:

1) Índice Flujo de Intensidad (SI): se calcula computando un promedio ponderado de diferencias entre los flujos después de la Contingencia, I_i^{pos} , y los límites de flujo de intensidad en cada línea, I_i^{lim} . I_i^{pos} es *el valor final de la intensidad obtenida luego de la simulación dinámica*. Se le asigna un peso a cada línea de transmisión, de acuerdo a su importancia en el sistema de potencia, wl_i , el mismo valor empleado en (10) y en (12). La expresión es la siguiente:

$$SI = \frac{\sum_1^{NL} \left(w_i \times \left(\frac{I_i^{pos} - I_i^{lim}}{I_i^{lim}} \right)^2 \right)}{\sum_1^{NL} w_i} \quad (15)$$

2) Índice de Tensión (STI): se calcula en base a los promedios ponderados de las diferencias entre las desviaciones de la tensión, antes y después de la Contingencia y el valor de tensión límite aceptable en cada nodo. El ponderador w_i es el mismo que en (11) y (13). La expresión resulta:

$$STI = \frac{\sum_{i=1}^{NN} \left(w_i \times \left(\frac{V_i^{pos} - V_i^n}{\Delta V_i^{lim}} \right)^2 \right)}{\sum_{i=1}^{NN} w_i} \quad (16)$$

donde: V_i^{pos} es Tensión en el Nodo i después de la Contingencia.

3) Índices de Desviación de Frecuencia (FI) y (FT): la desviación de la frecuencia es un claro indicador de la evolución dinámica del sistema después de una contingencia. Para la frecuencia se definen *dos índices*, el *índice de desviación máxima de la frecuencia (FI)* y el *índice de desviación total de frecuencia (FT)*. Sus expresiones son las siguientes:

$$FI = \Delta F_{Max} / \Delta F_{MaxAdmisible} \quad (17)$$

$$FT = \int_0^{ts} \Delta F(t) dt / (\Delta F_{MaxAdmisible} \times ts) \quad (18)$$

donde: ΔF_{Max} es la Desviación Máxima de la Frecuencia y $\Delta F_{MaxAdmisible}$ es la Desviación Máxima de la Frecuencia Admisible. ts es el tiempo hasta que tienen lugar las acciones correctivas. La integral en (18) es aproximada por sumatoria, con valores discretos de ΔF y Δt , en el intervalo $[0, ts]$.

4) Índice de Desconexión de Carga (PDI): este índice considera la cantidad de carga desconectada por el subsistema de desconexión de carga automática, que opera sobre el SP, luego de la Contingencia. Su expresión es la siguiente:

$$PDI = \Delta P_{Desconectada} / P_{Total} \quad (19)$$

donde: $\Delta P_{Desconectada}$ es la Carga Desconectada y P_{Total} es la Carga Total después de la Contingencia.

3.3. Estrategia y Pasos de Ejecución del Modelo Propuesto

El Modelo requiere, como se dijo, de dos RNR. Se describen, a continuación, la estrategia y los pasos de cómputo. Luego se analizan los detalles de las diferentes herramientas que lo componen.

3.3.1. Etapa Fuera de Línea

1) La RNRS:

Al igual que en la solución propuesta en (Schweickardt y Gimenez, 2012), esta red estima los Índices de Pos-Contingencia En Línea. Esto es así

porque el cálculo directo de los mismos requiere de simulaciones dinámicas, cuyo tiempo de cómputo excede por mucho la ventana temporal de respuesta En Línea, del orden de los 20 [seg]. Se tiene un Escenario de Generación/Demanda (de aquí en adelante EG/D) Base, designado como EG/D A. También se dispone de la Lista Inicial de Contingencias, integrada por NC Contingencias. Se establece un número NCT de Cambios Topológicos en el SP, y un número Ne de EG/Ds.

Cada Patrón de Entrada presentado a la RNRS durante su aprendizaje, tiene dos componentes: a) los Índices de Pre-Contingencia, calculados en cada Topología y EG/D del SP y b) los Índices de Pos-Contingencia, calculados en cada Topología en el EG/D A; ambos para cada Contingencia de la Lista Inicial. En Schweickardt y Gimenez (2012) se mostró que los Índices de Pos-Contingencia calculados para el EG/D Base, A, aportan información sobre el efecto de una Contingencia en diferentes escenarios, al considerarlos en conjunto con los Índices de Pre-Contingencia resultantes para componer los Patrones de Entrada.

Cada Patrón de salida se compone de los Índices de Pos-Contingencia calculados en cada Topología y EG/D del SP, para cada Contingencia de la Lista Inicial.

Los pasos de para generar ambos patrones, resultan:

Los Patrones de Entrada: siguiendo dos etapas:

A) Índices de Pos-Contingencia:

1) Para cada Topología j del SP:

2) Para cada Contingencia c de la Lista Inicial:

3) Se calculan mediante simulación dinámica completa los Índices de Pos-Contingencia, $I\text{Pos}^{k, c, j}$ (ecuaciones (15)-(19)) para el EG/D A. k indica el tipo de índice ($k = 1 \rightarrow \text{SI}$; $k = 2 \rightarrow \text{STI}$; $k = 3 \rightarrow \text{FI}$; $k = 4 \rightarrow \text{FT}$ y $k = 5 \rightarrow \text{PDI}$).

B) Índices de Pre-Contingencia:

1) Para cada Topología j del SP:

2) Se compone el EG/D i. Para ello, se considera un incremento (+/-) en el Vector Demanda, $[\Delta D^i_j]$ respecto del EG/D A, según las variaciones típicas de la Curva de Carga disponible y el acopio de datos correspondiente a cada barra de carga. $[\Delta D^i_j]$ es un vector, cuyo número de componentes es el número de barras de carga.

3) Se corre un Flujo de Potencia AC (simulación necesaria, pues las pérdidas en las líneas no son conocidas), identificando los i incrementos en cada barra de generación según el Vector de Generación resultante, $[\Delta G^i_j]$, cuyo número de componentes es el número de tales barras. Los vectores $[\Delta D^i_j]$ y $[\Delta G^i_j]$, definen el EG/D i.

4) Para cada Contingencia c de la Lista Inicial:

5) Se calculan los Índices de Pre-Contingencia en el EG/D i, $I\text{Pre}^{k, c, j}$ (ecuaciones (10)-(14)). k indica el tipo de índice ($k = 1 \rightarrow \text{MI}$; $k = 2 \rightarrow \text{MTI}$; $k = 3 \rightarrow \text{MQI}$; $k = 3 \rightarrow \text{MQTI}$ y $k = 5 \rightarrow \text{GI}$). Para los Índices Pre-Contingencia cuasi-estacionarios, MQI y MQTI y GI, se requiere simular la Contingencia c de la Lista Inicial, como se explicó en 3.1.

6) Se determina, desde el Conjunto Difuso asociado a Cada Índice de Pre-Contingencia, el valor de su Función de Pertenencia para cada valor obtenido, $\mu(\text{IPre}^{k,c,j_i})$. Este valor se utilizará para componer los Patrones de Salida de la segunda red neuronal (RNRN), que estima la Función de Seguridad En Línea.

Cada Patrón de Entrada, pe^{c,j_i} , quedará definido, entonces, por una Contingencia c , ocurrida en un EG/D i , con una Topología j del SP. Estará compuesto por 5 Índices de Pos-Contingencia, IPos^{k,c,j_i} (calculados en el EG/D A -por ello no dependen de i - con k en [1..5]), 2 Índices de Pre-Contingencia estacionarios IPre^{k,c,j_i} (k en [1..2]), y 3 Índices de Pre-Contingencia cuasi-estacionarios IPre^{k,c,j_i} (k en [3..5]). De modo que el número de entradas de la RNRN, será $NE = 5 + 2 + 3 = 10$.

Los Patrones de Salida:

1) Para cada Topología j del SP:

2) Para cada EG/D i :

3) Para cada Contingencia c de la Lista Inicial:

4) Se calculan, mediante simulación dinámica completa, los Índices de Pos-Contingencia, IPos^{k,c,j_i} (ecuaciones (15)-(19)).

5) Se determina, desde el Conjunto Difuso asociado a Cada Índice de Pos-Contingencia, el valor de su Función de Pertenencia para cada valor obtenido, $\mu(\text{IPos}^{k,c,j_i})$. Este valor también se utilizará para componer los Patrones de Salida de la segunda red neuronal.

Cada Patrón de Salida, ps^{c,j_i} , estará definido, entonces, por una Contingencia c , ocurrida en un EG/D i , con una Topología j del SP. Se compone de 5 Índices de Pos-Contingencia, IPos^{k,c,j_i} (k en [1..5]). De modo que el número de salidas de la RNRN será $NS = 5$. Su diseño en el Modelo propuesto, se completa con 2 Capas Ocultas, $NCO = 2$, de 10 neuronas.

Como se mencionó en el epígrafe 2, el algoritmo de entrenamiento clásico de retropropagación, se basa en el método del Gradiente Descendente. Se tienen algunas variantes. Una muy utilizada es con Momento y Tasa de Aprendizaje Variable. Se observa en las ecuaciones (8) y (9). El Factor de Momento actúa como un filtro pasa-bajos (Moallem y Ayoughi, 2010), que permite a la red ignorar puntos de la superficie de la función error cuadrático medio global, en las que el aprendizaje podría quedar “atrapado”, arrojando un mínimo local (Haykin, 1999).

Sin embargo, aquí intervienen algunas consideraciones de tipo general y de aplicación para la RNRN propuesta:

a) Cualquiera de estos métodos, supone la solución de un problema de optimización no lineal. Como es sabido, por medio de métodos clásicos, no se garantiza la convergencia hacia un mínimo minimorum del error cuadrático medio global. Siempre existe el riesgo de que el algoritmo de entrenamiento clásico, quede “atrapado” en un mínimo local;

b) Este aspecto impacta sobre la capacidad predictiva de la RNRN. Por capacidad predictiva, debe entenderse que no falle en los patrones de salida cuando el patrón presentado en la entrada fue observado en su entrenamiento, y que logre resolver satisfactoriamente patrones de salida ante patrones de entrada no observados durante su entrenamiento, dentro del

rango de aprendizaje. Tal capacidad depende de dos factores, excluyendo el diseño (que es un problema en sí mismo): 1ro) del número de patrones de entrada y sus correspondientes patrones de salida, empleados durante el entrenamiento de la RNRS: *a mayor número de patrones, mayor capacidad predictiva*; y 2do) de que el algoritmo de entrenamiento converja en el mínimo error cuadrático medio global o en un valor próximo: *a menor error, mayor capacidad predictiva*. Estos dos factores actúan inter-dependientemente.

En tal sentido, existen propuestas para el entrenamiento de Redes Neuronales, específicamente para el Perceptrón Multicapa, mediante metaheurísticas tales como la PSO (Kawam y Manssur, 2012), con cuyo empleo pueden obtenerse mejores resultados respecto del algoritmo RS.

Por este motivo, siguiendo el modelo presentado en (Schweickardt y Gimenez, 2012) se probó una solución MatHeurística (combinación de programación matemática y metaheurísticas), de dos tipos.

En la primera, se hibridiza el Algoritmo de Entrenamiento de la RNRS, aplicando, en iteraciones tempranas, la variante con Gradiente Descendente y Momento. Luego, en un punto de corte convenientemente definido (para el caso, cuando se alcanza cierto error), es aplicada una MetaHeurística EPSO (MonoObjetivo) (Schweickardt y Miranda, 2010). A tal estrategia se la refiere como MatHeurística GDM/EPSO (Gradiente Descendente con Momento/MetaHeurística EPSO). Se obtuvieron, resultados muy satisfactorios, *cuando la topología del SP, no se modifica*, hipótesis asumida en (Schweickardt y Gimenez, 2012).

Pero en el presente trabajo, *se consideran cambios topológicos* en el SP y fueron observadas dos limitaciones en el entrenamiento empleando la MatHeurística referida:

1ra) Al existir un cambio topológico, los patrones de entrada-salida se modifican y la RNRS en cuestión requiere un nuevo entrenamiento. Comparando resultados de una simulación dinámica completa con la estimación de la RNRS, se detectó una importante disminución de la capacidad predictiva ante patrones no observados. La red entrenada primero sin cambios topológicos y posteriormente reentrenada con cambios topológicos en el SP, arrojaba resultados diferentes respecto de la red entrenada considerando las diferentes topologías previstas.

2da) El tiempo requerido para el entrenamiento resultaba elevado, si bien esto depende mucho del *Hardware*, cuando el procesamiento es *secuencial*, tal y como se implementó en el trabajo previo.

Por tales razones se introdujo el segundo tipo de MatHeurística. Se propuso *paralelizar* el algoritmo de aprendizaje para esta RNRS, sustituyendo la MatHeurística GDM/EPSO, de procesamiento *secuencial*, por una estrategia aquí referida como MatHeurística GDM/HY X-PSO MPI: Gradiente Descendente con Momento/HiperHeurística soportada X-PSO Formas con procesamiento *paralelo*, utilizando la plataforma denominada Interfaz de Paso de Mensajes, MPI (*Message Passing Interface*). El empleo de una HiperHeurística soportada en X-PSO Formas MultiObjetivo, mediante computación paralela utilizando la plataforma MPI, es presentado por los

autores en (Schweickardt y Casanova, 2015). A diferencia de la aplicación en la referencia citada, la optimización no es MultiObjetivo sino MonoObjetivo.

Con esta MatHeurística GDM/HY X-PSO MPI, fue posible incorporar los cambios topológicos en el entrenamiento de una única RNRS, con muy buenos resultados.

II) RNRM:

La segunda red neuronal, estima la denominada Función de Seguridad, $f(\mathbf{x})$, siendo \mathbf{x} el vector de estado del SP. Esta función, no se conoce. Sólo se tienen rasgos de tipo cualitativo. Se trata de una Hiper-Superficie cuyos puntos corresponden a estados totalmente seguros del sistema (Mc Calley y Krause, 1995). Conforme lo explicado en 2.2, es por tal caracterización de la Función $f(\mathbf{x})$ que se optó por una RNRM para estimarla.

Cada Patrón de Entrada presentado a la RNRM durante su aprendizaje, incluye los Índices de Pre y Pos-Contingencia, calculados sobre cada Topología y EG/D del SP, y para cada Contingencia de la Lista Inicial.

Los Patrones de Entrada se componen como sigue:

- 1) Para cada Topología j del SP:
- 2) Para cada EG/D i :
- 3) Para cada Contingencia c de la Lista Inicial:
- 4) Se calculan los Índices de Pre-Contingencia, $I\text{Pre}^{k, c, j}_i$ (ecuaciones (10)-(14)). Paso 5) para componer los patrones de entrada de la RNRS.
- 5) Se calculan, mediante simulación dinámica completa, los Índices de Pos-Contingencia, $I\text{Pos}^{k, c, j}_i$ (ecuaciones (15)-(19)). Paso 4) para componer los patrones de salida de la RNRS.

Cada Patrón de Entrada, $pe^{c, j}_i$, de la RNRM, quedará definido, entonces, por una Contingencia c , ocurrida en un EG/D i , con una Topología j del SP. Estará compuesto por 5 Índices de Pos-Contingencia, $I\text{Pos}^{k, c, j}_i$ (k en $[1..5]$), 2 Índices de Pre-Contingencia estacionarios $I\text{Pre}^{k, c, j}_i$ (k en $[1..2]$), y 3 Índices de Pre-Contingencia cuasi-estacionarios $I\text{Pre}^{k, c, j}_i$ (k en $[3..5]$). De modo que el número de entradas de la RNRM, será $NE = 5 + 2 + 3 = 10$.

Para comprender cómo se componen los Patrones de Salida de la RNRM durante su aprendizaje, se requiere de la consideración previa de algunos conceptos.

Por una parte intervienen los valores de la función de pertenencia de cada Conjunto Difuso solidario a cada Índice de Pre y Pos-Contingencia, calculados Fuera de Línea, siguiendo también el trabajo previo (Schweickardt y Gimenez, 2012). Sus cálculos se detallan más abajo.

Por la otra, se propone captar, como ocurre en la práctica, la opinión sobre la severidad de una contingencia por parte de un operador experto (Gimenez Álvarez y Mercado, 2007). Es muy frecuente, como se indica en tal referencia, que, no obstante uno o varios Índices Pre-Contingencia y/o Pos-Contingencia arrojen valores que identifiquen la Contingencia ocurrida como Crítica, no lo sea en realidad. Esta condición, referida como Falso Positivo, proviene de la inexactitud de métodos de cálculo (particularmente en las simulaciones dinámicas) en conjunción con los errores de medida en las variables de estado del SP. El operador experto está al tanto de tal situación,

y puede decidir no implementar ninguna acción correctiva frente a una Contingencia identificada como Crítica a partir de los cálculos realizados. O a la inversa (por oposición, ante un Falso Negativo), implementando acciones correctivas al ocurrir una Contingencia identificada como No Crítica.

Los Conjuntos Difusos solidarios a cada Índice Pre/Pos-Contingencia, se introducen, al igual que en (Gimenez Álvarez y Mercado, 2007) y (Schweickard y Gimenez, 2012), a los efectos de captar sus incertidumbres de valor. Se emplean funciones de pertenencia exponenciales, definidas en general como:

Sea I un Índice de Pre/Pos Contingencia en el grupo correspondiente, e IRef su *valor de referencia admisible*, entonces:

$$\mu(I) = 1, \text{ si } I < IRef \quad (20)$$

$$\mu(I) = e^{-(I-IRef)/IRef}, \text{ si } I \geq IRef \quad (21)$$

En la medida que el valor del índice genérico, I, se aparte de su valor de referencia, IRef, su aceptación, $\mu(I)$, disminuye, tendiendo la Contingencia evaluada a estar más cerca de resultar Crítica.

La composición de tales funciones de pertenencia en un único Índice, se realiza mediante la operación de Confluencia de Conjuntos Difusos. Para ello, siguiendo a Bellman y Zadeh (1970), se adopta un operador de confluencia, cuyos operandos son los valores de las funciones de pertenencia $\mu(I)$. Estos son los calculados en el paso 6) de la composición de los Patrones de Entrada, $\mu(IPre^{k,c,i_j})$, y en el paso 5) de los Patrones de Salida, $\mu(IPos^{k,c,i_j})$, para el entrenamiento de la RNRS. El operador de confluencia es como una *t-norma*, y la utilizada en este trabajo se denomina Producto de Einstein (Schweickardt y Miranda, 2010). Para dos valores de Índices de Pre/Pos Contingencia, I1 e I2, cuyas funciones de pertenencia de los Conjuntos Difusos solidarios arrojan valores $\mu(I1)$ y $\mu(I2)$, respectivamente, el valor de la t-norma Producto de Einstein, t_{PE} , está dado por la expresión:

$$t_{PE}(1,2) = \mu(I1) \times \mu(I2) / (2 - (\mu(I1) + \mu(I2) - \mu(I1) \times \mu(I2))) \quad (22)$$

Para integrar en la confluencia el valor de un tercer Índice, I3, cuyo valor de su función de pertenencia resulta $\mu(I3)$, se opera como sigue:

$$t_{PE}(1,2,3) = t_{PE}(1,2) \times \mu(I3) / (2 - (t_{PE}(1,2) + \mu(I3) - t_{PE}(1,2) \times \mu(I3))) \quad (23)$$

Se continúa de este modo hasta obtener el valor de la función de pertenencia, $\mu(C)$, del Conjunto Difuso Confluencia de todos los Conjuntos Difusos asociados a los diferentes Índices:

$$\mu(C) = t_{PE}(1,2,3...n) \quad (24)$$

Para introducir la opinión del operador experto, se emplea una regla heurística que considera los siguientes aspectos:

a) Cada valor de la función de pertenencia solidaria a cada Índice de Pre/Pos-Contingencia, $\mu(IPre^{k,c,i_j})$ y $\mu(IPos^{k,c,i_j})$, se afecta de un *ponderador exponencial*. Los ponderadores son números, pOp , reales positivos. Su efecto es alterar la función de pertenencia dada por (21), del siguiente modo:

$$\mu(I) = e^{-pOp \times (I-IRef)/IRef}, \text{ si } I \geq IRef \quad (25)$$

Si $pOp < 1$, el Conjunto Difuso se *dilata*, y tiene *menor impacto* en la Confluencia. En cambio si $pOp > 1$, el Conjunto Difuso se *contrae*, y tiene *mayor impacto* en la confluencia (Yager, 1977). Con ello el operador experto proporciona menor o mayor importancia a cada Índice calculado.

b) Definidos los ponderadores para cada valor $\mu(IPre^{k, c, i})$ y de $\mu(IPos^{k, c, i})$, se calculan sus confluencias según (24), utilizando (25). Así se obtienen los valores $\mu(C(IPre^{c, i}))$ y $\mu(C(IPos^{c, i}))$ de las funciones de pertenencias de los Conjuntos Difusos Confluencia de los Índices Pre y Pos-Contingencia, respectivamente. El valor estimado de la Función de Seguridad, $f(\mathbf{x})^{c, i}$, resultará entonces, de la aplicación de la t-norma t_{PE} , aplicada sobre los valores $\mu(C(IPre^{c, i}))$ y $\mu(C(IPos^{c, i}))$. $f(\mathbf{x})$ resulta ser, operacionalmente, también un Conjunto Difuso.

d) El operador experto, define, además, un valor μOp en $[0, 1]$, que se comporta como un α -corte de $f(\mathbf{x})$ por debajo del cual el SP se considera *inseguro*. Para una Topología j , EG/D i , si ocurre una Contingencia c , la condición $f(\mathbf{x})^{c, i} < \mu Op$ resulta ser la *condición de Contingencia Crítica*.

De modo que los Patrones de Salida para la RNRM se componen como sigue:

- 1) Para cada Topología j del SP:
- 2) Para cada EG/D i :
- 3) Para cada Contingencia c de la Lista Inicial:
- 4) Considerando la regla heurística descripta, se calcula $f(\mathbf{x})^{c, i}$

Cada Patrón de Salida, $ps^{c, i}$, estará definido por una Contingencia c , ocurrida en un EG/D i , con una Topología j del SP. Se compone del valor estimado de la Función de Seguridad, $f(\mathbf{x})^{c, i}$. De modo que el número de salidas de la RNRM será $NS = 1$. Su diseño en el Modelo, tiene un Número de Capas Ocultas, $NCO = 1$, con 10 Neuronas (Red Principal).

Puede ser definido un Nivel de Severidad para una Contingencia, como el complemento a 1 del valor estimado de $f(\mathbf{x})^{c, i}$. Es decir:

$$NSE^{c, i} = 1 - f(\mathbf{x})^{c, i} \quad (26)$$

3.3.2 Etapa En Línea

En Línea, ocurrida una contingencia c , son calculados los correspondientes Índices de Contingencia según se explicó, pues los de Pos-Contingencia para el Escenario Base, A, han sido calculados. Se presenta el Patrón de Entrada compuesto por ambos grupos de Índices a la RNRS, y se obtienen los Índices de Pos-Contingencia estimados, como Patrón de Salida. Estos son presentados a la RNRM ahora como Patrón de Entrada, obteniéndose, como Patrón de Salida, el valor de la Función de Seguridad estimada. Fijado el valor (operador experto), μOp , si $f(\mathbf{x})^{c, i} < \mu Op$, la Contingencia c resulta identificada como Crítica.

4. SIMULACIONES

El Sistema Argentino de Interconexión (SADI) está conformado por un conjunto de 8 regiones las cuales poseen características propias. Estas áreas son: NOA (Noroeste), NEA (Noreste), CENTRO, LITORAL, GRAN BS. AS. (Capital y Gran Buenos Aires), BS. AS. (Provincia de Buenos Aires), CUYO y COMAHUE. Dichas áreas se encuentran interconectadas por una red de alta tensión en 500 [kV] y por algunos vínculos en 220 [kV]. En la FIGURA 3, se representa el esquema unifilar del SADI para dos topologías. La primera es considerada en (Schweickardt y Gimenez, 2012). La segunda incorpora un cambio topológico, incluyendo una línea de interconexión entre las regiones CUYO-COMAHUE (línea continua), conectando las barras 36 y 41. El EG/D Base (A) tiene una demanda de 5517 [MW] (5650 [MW] de generación). El incremento acumulativo de demanda para crear el conjunto de EG/Ds, se estableció en ± 10 [MW] uniformemente distribuida en las barras de carga del SP, generando 280 EG/Ds considerando ambas topologías. La Lista Inicial de Contingencias Inicial sólo incluye la pérdida de generación; la pérdida de una línea es considerada, generalmente, como una Contingencia Crítica.

Por razones de espacio es imposible presentar la totalidad de los cálculos realizados. Los resultados se centran en la Validación del Modelo. Fue utilizado un Método de Validación Cruzada (*Cross Validation*), específicamente el *Multifold Cross Validation* (Haykin, 1999), o Método *K-fold*, para la validación de las dos redes neuronales. Consiste en evaluar los resultados de diferentes entrenamientos y garantizar que son independientes de la partición entre patrones de entrenamiento y validación. Para evaluar esta aptitud, se calcula la media aritmética obtenida de los errores obtenidos en cada partición de validación.

Desde los EG/Ds establecidos, se seleccionaron $K = 7$ particiones, cada una con 40 EG/D. Ambas redes fueron entrenadas K veces, sin incluir los patrones correspondientes a la K -ésima partición, presentando los mismos como patrones de validación. Se calculó el error obtenido en los patrones de salida. En la TABLA 1, se presentan los resultados de los Errores Promedio *sin* (EM%) y *con* (EM%T) el cambio topológico considerado, para cada partición, $K [1..7]$, y para cada red neuronal. El Error Promedio no superó el 5%. En la TABLA 2, se presenta una Validación adicional del Modelo, mediante los resultados de la Función de Seguridad, $f(x)$, *con* (FS) y *sin* (FST) el cambio topológico definido, para la Lista de Contingencia Inicial.

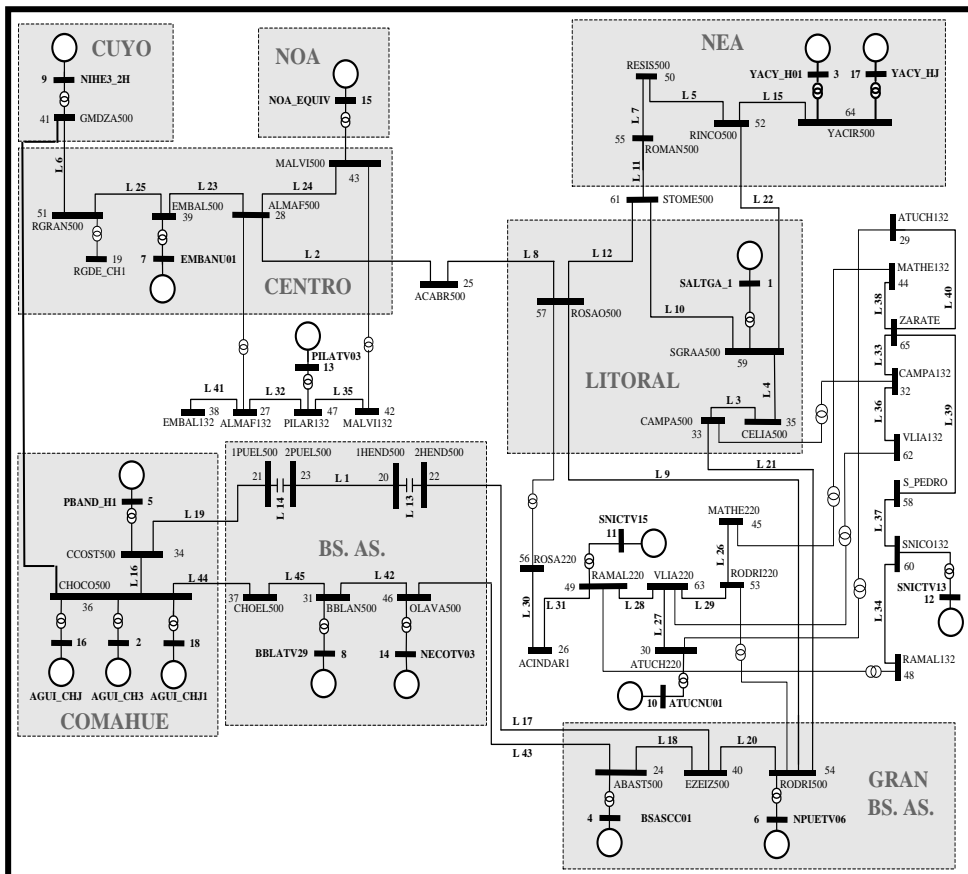


FIGURA 3: Esquema Unifilar SADI empleado en las Simulaciones.

TABLA 1: Validación Cruzada en el Entrenamiento de las RNRs y RNRm

| RNRs | | RNRm | |
|-------------|----------|-------------|----------|
| Partición K | EM%/EM%T | Partición K | EM%/EM%T |
| 1 | 4.5/3.7 | 1 | 4.1/2.9 |
| 2 | 2.6/4.1 | 2 | 3.3/4.7 |
| 3 | 5.1/4.3 | 3 | 4.9/3.6 |
| 4 | 2.1/1.9 | 4 | 1.7/2.5 |
| 5 | 2.5/2.2 | 5 | 2.8/4.8 |
| 6 | 3.8/1.9 | 6 | 4.6/2.0 |
| 7 | 2.7/5.0 | 7 | 3.1/4.3 |

Se comparan los resultados estimados para dos EG/Ds observados en la Etapa Fuera de Línea, con los obtenidos mediante simulaciones completas.

TABLA 2: Función de Seguridad Simulada y Estimada en EG/D B y EG/D C

| Simulación | | Estimación EG/D B | Estimación EG/D C |
|-------------------|-----------------|-------------------|-------------------|
| Cont | EG/D B FS/C FST | FS | FST |
| YACIR500-RINCO500 | 0.653/0.551 | 0.657 | 0.524 |
| CHOCO500 | 0.055/0.061 | 0.054 | 0.060 |
| EMBA500 | 0.046/0.040 | 0.044 | 0.042 |
| CCOST500 | 0.036/0.026 | 0.038 | 0.024 |
| BBLA500 | 0.025/0.030 | 0.024 | 0.030 |
| GMZA500-RGRAN500 | 0.023/0.026 | 0.022 | 0.025 |
| RAMA220 | 0.026/0.029 | 0.026 | 0.027 |
| ABAST500 | 0.025/0.026 | 0.026 | 0.025 |
| ATUCH220 | 0.025/0.028 | 0.025 | 0.028 |
| PILAR132 | 0.020/0.025 | 0.021 | 0.024 |
| OLAVA500 | 0.023/0.024 | 0.022 | 0.023 |
| SNICO132 | 0.025/0.027 | 0.025 | 0.026 |
| RODRI500 | 0.021/0.026 | 0.020 | 0.025 |

TABLA 3: Contingencias Críticas en EG/D D con $\mu Op = 0.03$

| EG/D D (Simulado) | | EG/D D (Estimado) | |
|-------------------|-------|-------------------|-------|
| Contingencia | FST | Contingencia | FST |
| YACIR500-RINCO500 | 0.651 | YACIR500-RINCO500 | 0.644 |
| CHOCO500 | 0.065 | CHOCO500 | 0.067 |
| EMBA500 | 0.042 | EMBA500 | 0.040 |
| CCOST500 | 0.034 | CCOST500 | 0.033 |
| BBLA500 | 0.034 | BBLA500 | 0.032 |
| GMZA500-RGRAN500 | 0.031 | GMZA500-RGRAN500 | 0.030 |
| RAMA220 | 0.025 | RAMA220 | 0.024 |
| PILAR132 | 0.025 | PILAR132 | 0.025 |
| ABAST500 | 0.023 | ABAST500 | 0.021 |
| OLAVA500 | 0.022 | OLAVA500 | 0.021 |
| SNICO132 | 0.020 | SNICO132 | 0.020 |
| RODRI500 | 0.019 | RODRI500 | 0.020 |

| | | | |
|----------|-------|----------|-------|
| ATUCH220 | 0.021 | ATUCH220 | 0.022 |
|----------|-------|----------|-------|

Los EG/Ds son referidos como B (Base + 500 [MW]) y C (Base - 500 [MW]). En el caso del EG/D B se compara la Función de Seguridad simulada con la estimada por el Modelo sin considerar el cambio topológico (FS). En el EG/D C, se considera tal cambio (FST). El error no superó el 7%.

A fin de validar el desempeño del Modelo para identificar Contingencias Críticas En Línea, se simuló la Lista de Contingencias Inicial para un escenario D (EG/D A + 227 MW), no observado en la Etapa Fuera de Línea. Adicionalmente, se estableció un valor de $\mu_{Op} = 0.03$. Se realizaron los cálculos completos de la simulación dinámica para determinar la FST (con el cambio topológico definido). Luego se aplicó el Modelo propuesto para cada Contingencia, dentro de la ventana temporal admisible (< 20 [seg]). En la TABLA 3 se presentan los resultados. Se observó una variación menor al 7% de error entre los valores de FST, y la misma identificación de Contingencias Críticas (resaltadas en negrita), con condición $FST < \mu_{Op}$.

5. CONCLUSIONES

1ra) Se han presentado los conceptos y desarrollos correspondientes de un novedoso Modelo de Simulación *Soft Computing* para Clasificar las Contingencias Críticas, en la Evaluación de la Seguridad Dinámica de un Sistema de Potencia En Línea. El Modelo permite estimar la Función de Seguridad del Sistema de Potencia e Identificar una Contingencia Crítica, dentro de un margen temporal aceptable (< 20 [seg]).

2da) Los resultados obtenidos, particularmente los presentados en la TABLA 3, son coherentes respecto de los obtenidos en trabajos aquí referidos, que emplean procesamiento paralelo, mediante un *cluster* de computadoras (PC), e inferencia difusa para estimar la Función de Seguridad. La diferencia es que en este trabajo se empleó una *notebook* con 5 núcleos y placa de video que soporta la librería GPULib.

3ra) La inclusión del cambio topológico en el entrenamiento de las redes neuronales tiene un doble propósito: a) que el Modelo pueda emplearse cuando una línea está fuera de servicio por razones de mantenimiento, sin que se requiera re-entrenarlas y b) que pueda incluirse la salida de una línea ante una falla como una Contingencia más, aspecto de investigación actual.

6. REFERENCIAS

BELLMAN, R.; ZADEH, L. (1970): "DECISION-MAKING IN A FUZZY ENVIRONMENT". Management Science. Vol. 17 – pgs. 141 - 164.

CORTEGGIANO, F.; MERCADO, P. (1998): "DETERMINATION OF CRITICAL CONTINGENCIES FOR DYNAMIC OPERATION SECURITY

EVALUATION USING NEURAL NETWORKS”. VI SEPOPE. Salvador, Brazil. Paper SP-108 – pgs. 112 - 122.

FUNAHASHI, K. (1989): “ON THE APPROXIMATE REALIZATION OF CONTINUOUS MAPPINGS BY NEURAL NETWORKS”. Neural Networks. Vol. 2 - pgs. 183 - 192.

GIMENEZ ÁLVAREZ, J.; MERCADO, P. (2007): “ONLINE INFERENCE OF THE DYNAMIC SECURITY LEVEL OF POWER SYSTEMS USING FUZZY TECHNIQUES”. IEEE Transactions On Power Systems. Vol. 22, No. 2 – pgs. 717 - 726.

HAYKIN, S. (1999): “NEURAL NETWORKS AND LEARNING MACHINE”. Third Edition. Prentice Hall.

KAWAM, A.; MANSOUR, L. (2012): “METAHEURISTIC OPTIMIZATION ALGORITHMS FOR TRAINING ARTIFICIAL NEURAL NETWORKS”. International Journal of Computer and Information Technology. Vol. Issue 02 - pgs. 156 - 161.

LOPES, N.; RIBEIRO, B. (2001): “HYBRID LEARNING IN A MULTI NEURAL NETWORK ARCHITECTURE”. IEEE Neural Networks Proceedings. IJCNN '01. International Joint Conference. Vol. 4 – pgs. 2788 - 2793.

LOPES, N.; RIBEIRO, B. (2011): “GPULIB: AN EFFICIENT OPEN-SOURCE GPU MACHINE LEARNING LIBRARY”. International Journal of Computer Information Systems and Industrial Management Applications. Volume 3 – pgs. 355 - 362.

MC CALLEY, J.; KRAUSE, B. (1995): “RAPID TRANSMISSION CAPACITY MARGIN DETERMINATION FOR DYNAMIC SECURITY ASSESSMENT USING ARTIFICIAL NEURAL NETWORK”. Electric Power Systems Research. Vol 34 - Issue 1- pgs. 37 – 45.

MOALLEM, P.; AYOUGHI, S. (2010): “IMPROVING BACK PROPAGATION VIA AN EFFICIENT COMBINATION OF A SATURATION SUPPRESSION METHOD AND MOMENTUM TERM”. Neural Network World. 2/10 – pgs. 207 - 222.

MORISON, K.; WANG, L.; KUNDUR, P. (2004): “POWER SYSTEM SECURITY ASSESSMENT”. IEEE Power and Energy Magazine - Vol. 2 - Issue 5 - pgs. 54 - 61.

MORISON, K.; WANG, L.; KUNDUR, P. (2005): “ADVANCED CONTINGENCY SELECTION METHODOLOGY”. Proceedings of the 37th Annual North American Power Symposium - pgs. 67 - 73.

SCHWEICKARDT, G.; MIRANDA, V. (2010): “METAHEURÍSTICA FEPSO APLICADA A PROBLEMAS DE OPTIMIZACIÓN COMBINATORIA: BALANCE DE FASES EN SISTEMAS DE DISTRIBUCIÓN ELÉCTRICA”. Revista Ciencia Docencia y Tecnología, Universidad Nacional de Entre Ríos - Nro. 40 – pgs. 133 - 163.

SCHWEICKARDT, G.; CASANOVA, C. (2015): “MODELOS HIPERHEURÍSTICOS BASADOS EN RAZONAMIENTO CON PROCESAMIENTO PARALELO Y DOMINIO EN METAHEURÍSTICAS X-PSO Y AFS MULTIOBJETIVO”. Revista de la Escuela de Perfeccionamiento en Investigación Operativa – Nro. 37 – pgs. 7 - 26.

SCHWEICKARDT, G.; GIMENEZ, J. (2012): “MODELO DE SIMULACIÓN SOFT-COMPUTING PARA LA SELECCIÓN DE CONTINGENCIAS CRÍTICAS EN LA EVALUACIÓN DE LA SEGURIDAD DINÁMICA EN TIEMPO REAL DE SISTEMAS DE POTENCIA”. Anales del Encuentro XXIII ENDIO - XXIV ENDIO - XXIII EPIO, Buenos Aires, Argentina. – pgs. 724 - 744.

YAGER, R. (1977): “MULTIPLE OBJECTIVE DECISION MAKING USING FUZZY SETS”. Intl. J. Man-Machine Studies. Vol. 9 – pgs. 53 - 64.