

## MODELOS HIPERHEURÍSTICOS BASADOS EN RAZONAMIENTO CON PROCESAMIENTO PARALELO Y DOMINIO EN METAHEURÍSTICAS X-PSO Y AFS MULTIOBJETIVO

GUSTAVO SCHWEICKARDT - CARLOS CASANOVA  
CONICET – Universidad Tecnológica Nacional, Concepción del Uruguay  
Concepción del Uruguay - ARGENTINA  
*gustavoschweickardt@conicet.gov.ar - casanovac@frcu.utn.edu.ar*

*Fecha Recepción: Diciembre 2014 - Fecha Aceptación: Abril 2015*

### RESUMEN

En este trabajo se presenta el desarrollo e implementación de una HiperHeurística Basada en Razonamiento, con dominio en MetaHeurísticas variantes de la Optimización Por Enjambre de Partículas, X-FPSO, y Cardumen de Peces Artificiales, FAFS, MultiObjetivo. Como aporte respecto de otras publicaciones en la línea de investigación que los autores han desarrollado, se obtiene un importante avance: la paralelización del algoritmo, reemplazando su modelo secuencial primigenio, empleando la plataforma denominada Interfaz de Paso de Mensajes, MPI, (*Message Passing Interface*). Son propuestas dos estrategias para la implementación del Modelo, sustentadas en la habilidad que las X-Formas del conjunto X-(FPSO-FAFS) exhiben para satisfacer, en cierta instancia de decisión, los cinco Principios de la Inteligencia de Grupo (PIG). La primera, determina el número óptimo de núcleos MPI, y asigna, a cada uno, una subpoblación de la población total, con la misma cantidad invariante de individuos. La Función de Selección, FS, opera identificando la X-Forma más apta. La segunda, asigna un núcleo y la mejor X-Forma conforme cada PIG, modificando, mediante la FS, el tamaño de las subpoblaciones. Se presenta una aplicación de ambas estrategias, en el Problema de Balance de Fases en un Sistema de Distribución Eléctrica de Baja Tensión, comparando los resultados obtenidos.

**PALABRAS CLAVE:** Optimización – MetaHeurísticas – HiperHeurísticas – Computación Paralela – Interfaz de Paso de Mensajes.

### ABSTRACT

In this work the development and implementation of a Case Based Reasoning HyperHeuristic with a domain supported on variants of MultiObjective Particle Swarm Optimization MetaHeuristic, called X-FPSO, and Artificial Fish School (FAFS), is presented. As contribution in the same line of research that the authors had developed, parallel processing, instead of primary sequential processing model, is introduced and implemented. The parallel computing use

the Message Passing Interface System, MPI. Two different strategies in the HyperHeuristic Model are proposed, both of them supported on the ability of X-(FPSO+FAFS) Forms to satisfy the five Swarm Intelligence Principles (SIP) at certain instance of decision: The first, and simplest, determine the optimal number of cores MPI, and assign to each core a subpopulation from whole population, which particles number is invariant and the same for all cores. The Selection Function (SF) choose the MetaHeuristic of maximum fitness from X-(FPSO+FAFS) set. The second strategy, and most complex, assign a core and the X-Form of best performance to each SIP, changing the subpopulation assigned to each core, at certain decision instance. An application of both strategies to the Problem of Phase Balancing in a Low Voltage Electric Distribution System, to compare the results obtained, is presented.

**KEYWORDS:** Optimization – MetaHeuristics – HyperHeuristics – Parallel Computing – Message Passing Interface.

## 1. INTRODUCCIÓN

En trabajos previos (Schweickardt, Casanova, Gimenez, 2013) y (Schweickardt, Casanova, Gimenez, 2014), los autores del presente trabajo describieron la evolución histórica del concepto y aplicación de HiperHeurísticas en problemas de optimización para los que se pretenden soluciones de calidad, obtenidas en un tiempo menor al que proporcionan los Algoritmos Heurísticos o MetaHeurísticos. Asimismo, se desarrolló un Modelo HiperHeurístico Basado en Razonamiento o CBR (*Case Based Reasoning*), cuya Función de Selección (FS) se implementó mediante una Red Neuronal tipo Retropropagación Simple (RNR), aplicada sobre un dominio compuesto por un conjunto de MetaHeurísticas MultiObjetivo, variantes de la PSO (*Particle Swarm Optimization*) Canónica, y referidas como X-FPSO Formas. Como en tales trabajos previos se explica, la letra F en el acrónimo FPSO, significa que las MetaHeurísticas de tal conjunto, han sido dotadas de la capacidad para resolver problemas de optimización MultiObjetivo en el Espacio de Soluciones, recurriendo a la extensión de los Objetivos y Restricciones desde su formulación clásica al dominio de los Conjuntos Difusos (*Fuzzy Sets*). Tal Modelo HiperHeurístico, fue designado, desde estos conceptos, como HY X-FPSO CBR. Si bien presentó excelentes resultados aplicada sobre la Clase de Problemas, caracterizada por su Espacio de Búsqueda, de optimización relativos a la Economía Computacional de Regulación y Análisis de Funcionamiento de Sistemas de Distribución de Energía Eléctrica (SDEE), respecto de los resultados obtenidos por aplicación de diferentes MetaHeurísticas MultiObjetivo (en particular, cualquiera de las X-FPSO Formas), su implementación algorítmica fue secuencial.

Con el objeto de mejorar aún más dos aspectos críticos para los cuales el Modelo HY X-PSO CBR aportó notables mejoras sobre la Clase de Problemas referida (aumento de la Relación Velocidad de Respuesta/Calidad de Soluciones y Autoadaptación en el ajuste de parámetros metaheurísticos),

en este trabajo se presentan dos modelos con procesamiento paralelo incorporando al dominio sobre el que operan la HiperHeurísticas, una MetaHeurística que no es una variante X de la PSO Canónica, pero puede integrarse sin inconvenientes: Cardumen de Peces Artificiales (AFS) (*Artificial Fish School*), extendida, también para dotarla de la capacidad de resolver problemas de optimización MultiObjetivo, al dominio de los Conjuntos Difusos (FAFS). La Computación o Procesamiento Paralelo, es implementada mediante la plataforma *software* denominada Interfaz de Paso de Mensajes, (MPI) (*Message Passing Interface*), la cual, como se explicará, ha sido desarrollada para su empleo en programas de computación que exploten la existencia de múltiples procesadores, de aquí en más referidos como núcleos (*core*), estructura *hardware* muy común, de fácil acceso y relativo bajo costo, observable en las computadoras estándar. De tal modo, el Modelo General HiperHeurístico propuesto será designado como HY X-(FPSO+FAFS) CBR MPI, para el cual se presentan dos estrategias de diseño, ambas sustentadas en la habilidad que las X-Formas del conjunto X-(FPSO-FAFS) exhiben para satisfacer, en cierta instancia de decisión, los cinco Principios de la Inteligencia de Grupo (PIG). La primera, determina el número óptimo de núcleos MPI, y asigna, a cada uno, una subpoblación de la población total, con la misma cantidad invariante de individuos. La Función de Selección, FS, opera identificando la X-Forma más apta para cada instancia de decisión o iteración algorítmica. La segunda, asigna un núcleo y la mejor X-Forma conforme cada PIG (de modo que al ser 5 los PIG, el número de núcleos es predefinido en 5), modificando, mediante la FS, el tamaño de las subpoblaciones para cada instancia de decisión o iteración algorítmica. En las dos estrategias, la FS de la HiperHeurística es una Red Neuronal tipo Retropropagación Simple (RNR), de idéntica estructura pero funcionalmente muy distinta, conforme puede inferirse desde sendas descripciones de la FS. El diseño diferente de la RNR, impactará sobre un diseño completamente diferente para el Modelo HiperHeurístico de cada Estrategia. Para identificarlos, serán referidos, correspondientemente según el orden en que fueron descriptas ambas estrategias como: 1) HY X-(FPSO+FAFS) CBR MPI con Número Constante de Partículas por Núcleo (NCPN) y 2) HY X-(FPSO+FAFS) CBR MPI con Número Variable de Partículas por Núcleo (NVPN).

El trabajo está organizado como sigue: En la Sección 2, se describen los conceptos relativos a una HiperHeurística del tipo CBR, presentados por los autores con mayor detalle en las referencias más arriba citadas, así como las principales características de las MetaHeurísticas MultiObjetivo que integran el conjunto X-(FPSO+FAFS), dominio del Modelo HiperHeurístico aquí propuesto, independientemente de la estrategia adoptada. Se hace especial énfasis en los detalles de la MetaHeurística FAFS, puesto que se agrega al conjunto observado en los trabajos anteriores. También es introducida, como se verá, una X-Forma modificada, denominada FEPSO GStT. Se describe, además y para la Clase de Problemas referida, la aptitud que exhiben cada una de las X-(FPSO+FAFS) Formas para satisfacer los 5 Principios de la Inteligencia de Grupo, elemento vertebrador a los efectos de definir la Función de Selección

FS. En la Sección 3, se presentan los aspectos más relevantes de la Plataforma Interfaz de Intercambio de Mensajes, los cuales permitirán codificar los Modelos propuestos mediante Computación Paralela. En la Sección 4, se aborda el desarrollo de cada una de las estrategias mencionadas, para concebir los Modelos HiperHeurísticos HY X-(FPSO+FAFS) CBR MPI NCPN y HY X-(FPSO+FAFS) CBR MPI NVPN. Son presentados los aspectos relativos al diseño de las Redes Neuronales RNR NCPN y RNR NVPN que operan como FS de cada estrategia, en su correspondiente Modelo HiperHeurístico. En la Sección 5, se presenta la aplicación de ambos Modelos HiperHeurísticos para resolver el Problema de Balance de Fases en un Sistema de Distribución Eléctrica de Baja Tensión, SDEE BT, comparando los resultados obtenidos por ambos, y los obtenidos aplicando la mejor Forma X-FPSO. En la Sección 6 son presentadas las conclusiones más relevantes y en la Sección 7, la bibliografía consultada.

## **2. HIPERHEURÍSTICA CBR Y METAHEURÍSTICAS X-FPSO Y FAFS**

### **2.1 HiperHeurística CBR**

#### 2.1.1 Limitaciones Reconocidas en el Estado del Arte de las MetaHeurísticas

Las MetaHeurísticas constituyen, en general, métodos especializados de búsqueda sobre un espacio de posibles soluciones de un problema, con un grado de especialización mucho menos restringido que el de las Heurísticas, conforme las definiciones adoptadas desde el estado del arte. Sin embargo, exhiben dos limitaciones importantes: a) Se requiere de un ajuste previo de parámetros según el tipo de problema a resolver, limitando su posibilidad de aplicación en otros tipos y, con ello, su concepción y diseño para la búsqueda de soluciones sobre Clases de Problemas. Suponen, en su mejor *performance*, un grado detallado relativo al conocimiento sobre el dominio del problema a resolver, para poder generar resultados de calidad; b) Por lo general el tiempo de cómputo requerido, resulta muy elevado, situación crítica si se pretendiese aplicar a problemas de optimización en tiempo real. Por tales motivos, las MetaHeurísticas son consideradas con frecuencia como técnicas falibles.

Surge, de tal modo, el concepto de solución “suficientemente buena”, la cual permite proponer métodos de mayor generalidad para resolver diferentes tipos de problemas y, a su vez, que las soluciones puedan ser obtenidas requiriendo menores tiempos de cómputo. Este es el concepto que subyace en la idea primigenia de las Hiperheurísticas: dado que diferentes MetaHeurísticas, exhiben diferentes fortalezas y debilidades, una estrategia más eficiente en términos de la relación velocidad de respuesta/calidad de soluciones, la constituye su utilización conjunta, para que la ejecución oportuna de cada una pueda compensar las debilidades de las otras. La HiperHeurística, supondría una función, de diferente complejidad, según el caso, tal que examinando las propiedades de las soluciones obtenidas en el proceso de búsqueda y/o las características del espacio en el cual las mismas van presentándose, conmute, si procede, a una MetaHeurística de mejor aptitud, como nuevo proceso a ejecutar en los siguientes pasos de evolución sobre la estrategia establecida.

### 2.1.2 Definición de HiperHeurística

Una HiperHeurística puede ser definida, entonces, como *una estrategia de búsqueda tal, que, a partir de cierto conjunto pre-establecido de MetaHeurísticas de bajo nivel, reconocidas como eficientes en la resolución de problemas pertenecientes a una misma clase, selecciona la que resulte más apta en cada instancia de decisión, identificada mediante algún proceso de aprendizaje, ejecutándola para el hallazgo de buenas soluciones en el menor tiempo posible*. Tal estrategia continúa hasta que cierta condición de salida o finalización resulte satisfecha. Por tanto, una HiperHeurística constituye un Algoritmo Heurístico de alto nivel, identificado por el proceso de aprendizaje/selección referido. Existe una diferencia muy importante, desde el punto de vista funcional, entre una MetaHeurística y una HiperHeurística: la MetaHeurística, es aplicada en el espacio de soluciones del problema, mientras que la HiperHeurística es aplicada en cierto espacio de MetaHeurísticas de bajo nivel. Es decir: una Hiperheurística trata con métodos de solución, mientras que una MetaHeurística trata con las soluciones de cierto problema. El nivel de abstracción de una HiperHeurística resulta, consecuentemente, mayor que el correspondiente a una MetaHeurística.

### 2.1.3 HiperHeurística CBR

En la referencia (Schweickardt, Casanova y Gimenez, 2013) se presenta una completa descripción sobre los orígenes y desarrollos del concepto de HiperHeurística y su clasificación conforme sea implementada su Función de Selección, FS (también referida como Método de Aprendizaje/Selección), cuyo dominio es el conjunto de métodos de solución metaheurísticos seleccionados. Es de interés en el presente trabajo, resaltar el tipo de HiperHeurística denominada Basada en Razonamiento (Petrovic y Qu, 2002) o *Case Based Reasoning*, CBR. La estrategia que se aplica en este tipo de HiperHeurística, consiste en emplear un conjunto de instancias de decisión/vector de soluciones del problema abordado, para el entrenamiento, por caso, de una Red Neuronal, a los efectos de la detección de patrones de estado/solución u otros elementos pertinentes, que permitan evaluar la aptitud de la MetaHeurística que se está aplicando. Posteriormente, los patrones salida, son considerados en forma exclusiva o complementada con nuevos patrones de entrada, como las instancias anteriores de decisión/vector de soluciones, sobre nuevas instancias de decisión y, con ello, se logra evolucionar sobre el espacio de MetaHeurísticas. Como se explica en (Schweickardt, Casanova y Gimenez, 2014), esta tipificación fue la adoptada para el desarrollo del Modelo HiperHeurístico HY X-FPSO CRB, en la cual la FS (Método de Selección) fue implementado mediante una Red Neuronal tipo Retropropagación Simple (RNR), cuyo diseño se presentará también en este trabajo, por ser tal RNR empleada en la primer estrategia propuesta.

La HiperHeurística HY X-FPSO CBR desarrollada por los autores en (Schweickardt, Casanova y Gimenez, 2014), tiene, como se dijo, dominio en un conjunto de variantes de la MetaHeurística PSO MultiObjetivo (X-FPSO Formas). Al igual que en los dos Modelos HiperHeurísticos que se desarrollarán

en la Sección 4, tal conjunto se integra seleccionando la variantes X-FPSO según su habilidad para satisfacer todos o algunos (mejor que el resto) los Principios de Inteligencia de Grupo (PIG). Adicionalmente, y con el mismo criterio, se incorpora la MetaHeurística FAFS en tal conjunto, definiendo un dominio híbrido para los dos Modelos HiperHeurísticos propuestos. Se tratan, a continuación, estos conceptos y desarrollos.

## **2.2 Las MetaHeurísticas X-FPSO – FAFS y los Principios de Inteligencia de Grupo**

### **2.2.1 Los Principios de la Inteligencia de Grupo**

La Inteligencia de Grupo, IG, (o *Swarm Intelligence*) refiere un tipo de inteligencia artificial, basado en el comportamiento de sistemas colectivos, auto-organizados. La expresión fue introducida en 1989 por Gerardo Beni (Beni y Wang, 1989). El término *swarm* describió, a su entender, el comportamiento de los autómatas celulares, que exhiben características similares a las observables en ciertos sistemas biológicos, tales como los insectos. De esas características, mencionó: la descentralización, no-sincronización y simplicidad en los movimientos de los miembros del grupo. Además este término tiene una significación que lo trasciende: permite analizar el comportamiento social de grupos de individuos (peces, insectos, pájaros, etc.), identificando el tipo de configuración en sus sistemas de cooperación y auto-adaptación, al efecto de encontrar la manera óptima de alcanzar un objetivo. La comunicación, según los biólogos constituye el parámetro clave en este proceso, de modo que a las cualidades mencionadas debe incorporársele la cooperación entre los individuos del *swarm*. Desde tales consideraciones, pueden ser establecidos cinco principios que caracterizan la IG (*Swarm Intelligence Principles*). Estos principios, y su breve enunciación, resultan: **1) Proximidad:** Promueve la habilidad que debe exhibir el grupo, para ejecutar cálculos simples de espacio y tiempo en sus movimientos hacia el objetivo pretendido; **2) Calidad:** Promueve la habilidad del grupo para responder a los factores que induzcan mejoras en la aptitud de sus individuos, en el espacio de soluciones; **3) Diversidad de Respuesta:** Promueve la posibilidad de que los individuos tengan respuestas diferentes ante los mismos estímulos; **4) Estabilidad:** Promueve la habilidad de que el grupo permanezca estable, en ausencia de estímulos que induzcan mejoras en las soluciones alcanzadas; y **5) Adaptación:** Constituye un aspecto complementario de la Estabilidad, puesto que promueve la habilidad de que los individuos reaccionen ante cualquier cambio en la aptitud de las soluciones alcanzadas.

Procede, seguidamente, realizar una síntesis de las Metaheurísticas X-FPSO+FAFS, apoyándose en la bibliografía pertinente para no abundar, a efectos de establecer un vínculo entre las mismas y los Principios IG.

### **2.2.2 MetaHeurísticas X-PSO y AFS**

Como se ha planteado, fundamentalmente por el primer autor del presente artículo, en colaboración con distintos investigadores, por caso en (Schweickardt y Miranda, 2010), tanto la formulación primigenia de la

MetaHeurística PSO, llamada forma Canónica, como muchas de sus variantes entre las cuales se encuentran las propuestas en diferentes trabajos publicados por el primer autor, son aplicables para problemas de Optimización MonoObjetivo. Parte de los desarrollos que han sido aportados desde tales trabajos estriban en haber dotado a las X-PSO Formas (MonoObjetivo) de la capacidad para resolver problemas MultiObjetivo, refiriéndolas así como X-FFPSO Formas. Lo mismo ocurre con la MetaHeurística AFS, que será extendida de idéntica manera al dominio MultiObjetivo, refiriéndola como FAFS. De modo que se hará, a continuación, referencia a la X-PSO y AFS Formas, para explicar luego como son extendidas al dominio MultiObjetivo.

**A) X0: Forma Canónica de la MetaHeurística PSO:** Desde una instancia de decisión en el Espacio de Soluciones (o iteración algorítmica) a la siguiente, cada partícula se mueve conforme a cierta regla de movimiento. La misma depende de tres factores, que se explican a continuación. Se indicará mediante  $[p]$  el vector de partículas en movimiento, de modo que  $p_i$  resultará una partícula individual de las  $n$  que pertenecen al enjambre o Población. Adicionalmente, se indicará mediante  $[b]$  el vector de las mejores posiciones (cada posición es, a su vez, un vector) que las partículas han alcanzado individualmente en las iteraciones anteriores (aspecto referido en el modelo como vida pasada de la partícula); entonces  $b_i$  se corresponderá con el óptimo individual de la partícula  $i$  en su vida pasada. Del mismo modo, se indicará mediante  $[b]_G$  el vector cuyos elementos son las mejores posiciones globalmente alcanzadas por el conjunto de partículas en las iteraciones anteriores; entonces se indicará mediante  $b_G$  al óptimo global alcanzado por la Población hasta la iteración presente. Dado el vector que indica la posición de las partículas en cierta iteración  $k$ ,  $X^k$ , el cambio de posición en la iteración siguiente,  $k+1$ , para la partícula  $i$ -ésima, resultará de la siguiente regla de movimiento:

$$X^{[k+1]}_i = X^{[k]}_i + V^{[k+1]}_i \times \Delta t \quad (1)$$

donde el término  $V^{[k+1]}_i$  es referido como velocidad de la partícula  $i$ -ésima;  $\Delta t$  es el paso de iteración e igual a la unidad. De modo que es más frecuente encontrar la expresión:

$$X^{[k+1]}_i = X^{[k]}_i + V^{[k+1]}_i \quad (2)$$

El vector velocidad para la partícula  $i$ -ésima, se expresa como sigue:

$$V^{[k+1]}_i = V^{[k]}_i + w_C \times (r_1^{[k]}) \times [b^{[k]}_i - X^{[k]}_i] + w_S \times (r_2^{[k]}) \times [b^{[k]}_G - X^{[k]}_i] \quad (3)$$

Aquí se tienen los tres factores referidos que definen cualitativamente la regla PSO de movimiento, cada uno se corresponde con un término de la ecuación (3). El primer término representa la inercia o hábito de la partícula  $i$ ; tiende a mantener su movimiento, para la iteración  $k+1$ , en la dirección en la que se movía en la iteración  $k$ . El segundo término representa la memoria o capacidad cognitiva de la partícula  $i$ , la cual es "atraída" por el mejor punto del espacio de búsqueda alcanzado individualmente en su vida pasada. Finalmente, el tercer término representa la cooperación entre el conjunto, o capacidad social, de la partícula  $i$  respecto del enjambre: las partículas comparten información sobre la mejor posición globalmente alcanzada por el enjambre. La incidencia de estos factores sobre cada partícula, está dada por

las constantes o parámetros del modelo,  $w_{c,s}$ . El parámetro  $w_c$  recibe el nombre de constante cognitiva y el parámetro  $w_s$  se denomina constante social del enjambre. Intervienen los parámetros  $(r_1)$  y  $(r_2)$ , los cuales son números aleatorios uniformemente distribuidos en  $[0,1]$ ,  $U [0,1]$ , y cuyo objetivo es emular el comportamiento estocástico que exhibe la Población, en cada iteración  $k$ .

**B) X1: PSO con Función de Decaimiento Inercial:** En esta forma, el operador velocidad canónico, dado por la expresión (3), es modificado mediante la introducción de una función decreciente con el número de iteraciones,  $k$ , denominada Función de Inercia o de Decaimiento Inercial,  $\delta(k)$ . Su objeto es reducir, progresivamente, la importancia del término de inercia. La expresión (3) es modificada sustituyendo el primer término por el producto entre una nueva constante,  $w_i$ , denominada *constante de inercia* y tal función  $\delta(k)$ :

$$\mathbf{V}^{[k+1]}_i = \delta(k) \times w_i \times \mathbf{V}^{[k]}_i + w_c \times (r_1^{[k]}) \times [\mathbf{b}^{[k]}_i - \mathbf{X}^{[k]}_i] + w_s \times (r_2^{[k]}) \times [\mathbf{b}^{[k]}_G - \mathbf{X}^{[k]}_i] \quad (4)$$

**C) X2: PSO con Factor de Constricción:** En este caso, el operador velocidad canónico se modifica como sigue:

$$\mathbf{V}^{[k+1]}_i = \chi \times \{ \mathbf{V}^{[k]}_i + \varphi_M \times (r_1^{[k]}) \times [\mathbf{b}^{[k]}_i - \mathbf{X}^{[k]}_i] + \varphi_C \times (r_2^{[k]}) \times [\mathbf{b}^{[k]}_G - \mathbf{X}^{[k]}_i] \} \quad (5)$$

donde  $\chi$  se denomina Factor de Constricción, y se obtiene desde la siguiente expresión condicionada:

$$\chi = 2 \times \kappa / \left| 2 - \varphi - \sqrt{\varphi^2 - 4 \times \varphi} \right| \quad (6)$$

$$\text{con: } \varphi_M + \varphi_C = \varphi; \varphi > 4 \text{ y } 0 < \kappa \leq 1 \quad (7)$$

**D) X3: EPSO (Evolutionary Particle Swarm Optimization):** Esta X-PSO Forma, combina la programación evolucionaria, mediante operadores evolutivos similares a los establecidos para los Algoritmos Genéticos (particularmente: mutación, selección, combinación), con el movimiento PSO, aplicándolos sobre sus parámetros y óptimo global en cada instancia de iteración  $k$ -ésima. El operador velocidad canónico se modifica, dando lugar a una ecuación evolutiva para la regla del movimiento EPSO. Se tiene así, para la partícula  $i$ -ésima:

$$\mathbf{V}^{[k+1]}_i = w_{i1}^{[*k]} \times (r_1^{[k]}) \times \mathbf{V}^{[k]}_i + w_{iC}^{[*k]} \times (r_2^{[k]}) \times [\mathbf{b}^{[k]}_i - \mathbf{X}^{[k]}_i] + w_{iS}^{[*k]} \times (r_3^{[k]}) \times [\mathbf{b}^{[k]*}_G - \mathbf{X}^{[k]}_i] \quad (8)$$

donde: el superíndice (\*), significa que los parámetros son evolutivos, producto de la mutación. La regla de mutación aplicable a las constantes  $w_{[i], c, s]}$  tiene, como expresión general:

$$w_{[i], c, s]}^{[*k+1]} = w_{[i], c, s]}^{[k]} \times [1 + \sigma \times N(0,1)] \quad (9)$$

en la cual:  $\sigma$  es un parámetro de aprendizaje, externamente fijado, que controla la amplitud de las mutaciones;  $N(0,1)$  es una variable aleatoria con distribución gaussiana de media 0 y varianza 1. Adicionalmente, el óptimo global también es perturbado en cada iteración, según la regla:

$$\mathbf{b}^{[k+1]*}_G = \mathbf{b}^{[k]*}_G + w_{iN}^{[*k]} \times N(0,1) \quad (10)$$

donde  $w_{iN}^{[*k]}$ , es un parámetro estratégico de la regla de movimiento, que controla la amplitud del vecindario de  $\mathbf{b}^{[k]*}_G$  donde es más probable localizar la mejor solución global (entendida, al menos, como una solución mejor que la  $\mathbf{b}^{[k]}_G$  actual, para la iteración  $k$ ). El nuevo parámetro, como se indica con el

superíndice (\*), es también mutado según la regla (9).

**E) X4: EPSO GIST (Global Individual Stochastic/Topology):** Esta X-PSO Forma evolucionaria se introduce en el presente trabajo, desde una propuesta presentada en (Schweickardt, Miranda y Gimenez, 2010), basada en una variante, denominada EPSO GIST, en la cual el acrónimo GIST, significa Topología Estrella Global/Individual (*Global Individual Star Topology*). Por Topología de Comunicación, se entiende la manera en que se establece la comunicación entre las partículas del enjambre. En las X-PSO Formas presentadas, el óptimo global alcanzado en cierta instancia de decisión o iteración algorítmica,  $k$ , es comunicado a la Población completa. Desde allí que se hable de Topología Estrella, pues podría asimilarse a una forma de comunicación en la que  $\mathbf{b}^{[k]}_G$  ( $\mathbf{b}^{[k]*}_G$ , en la X-Forma EPSO), “irradia” información hacia todas las direcciones, hasta las posiciones del Espacio de Búsqueda donde se encuentran las partículas, con excepción de aquella/s cuyo óptimo individual,  $\mathbf{b}^{[k]}_i$ , coincide con el global. La MetaHeurística EPSO GIST, introduce, respecto de la EPSO, dos modificaciones a los efectos de favorecer el 3er Principio IG, la Diversidad de Respuesta, propiciando evitar convergencias prematuras en óptimos locales, sobre Espacios de Búsqueda que exhiben soluciones muy aisladas, tales como los observados en la Clase de Problemas referida y que aquí sigue considerándose. Para ello, la primera modificación consiste en asignar una probabilidad de comunicación del óptimo global al resto de las partículas, y la segunda, consiste en permitir, además, que cada partícula comunique, con cierta probabilidad, la información relativa a su óptimo individual al resto de la Población. De tal modo, la X-Forma EPSO GIST, tiene por

ecuación evolutiva de movimiento, la expresión:

$$\mathbf{V}^{[k+1]}_i = w_{i1}^{*[k]} \times (r_1^{[k]}) \times \mathbf{V}^{[k]}_i + w_{iC}^{*[k]} \times (r_2^{[k]}) \times [\mathbf{b}^{[k]}_i - \mathbf{X}^{[k]}_i] + w_{iS}^{*[k]} \times (r_3^{[k]}) \times [\mathbf{b}^{[k]*}_G - \mathbf{X}^{[k]}_i] \times \alpha OG^{[k]}_i + \sum_{j=1, j \neq i}^n (\delta(k) \times w_{iC}^{*[k]} \times [\mathbf{b}^{[k]}_j - \mathbf{X}^{[k]}_i] \times \alpha OI^{[k]}_{ij}) \quad (11)$$

donde:  $\alpha OG^{[k]}_i$ : es la probabilidad de que el óptimo global (OG) afecte al movimiento de la partícula  $i$ , en la iteración  $k$  y  $\alpha OI^{[k]}_{ij}$  es la probabilidad de que el óptimo individual (OI) de la partícula  $j$ , afecte al movimiento de la partícula  $i$ ,  $j \neq i$ ;  $\delta(k)$  es el mismo factor multiplicativo empleado en (4), el cual va haciendo “decaer” la influencia de los OI, con probabilidades distintas de 0, conforme aumenta el número de iteraciones.

Si bien esta X-PSO Forma mostró una muy buena aptitud para “escapar” de una solución sub-óptima (local), propiciando claramente la Diversidad de Respuesta, en algunas situaciones, por caso al adicionar una variable control discreta un problema de optimización de la Clase considerada, cuyo impacto aísla aún más las buenas soluciones en el Espacio de Búsqueda, o bien cuando se restringe con menos margen de factibilidad una variable de estado (ejemplo del primer caso, incorporar regulación de tensión a la salida del transformador vinculado a la red de Baja Tensión en estudio, con pasos de a +/- 1%, hasta +/- 20%, de la tensión nominal; ejemplo del segundo caso, limitar el rango de caídas de tensión en los nodos terminales de la red, menos de un 5%, por exigencias regulatorias restricta), la EPSO GIST puede resolver sub-óptimamente el Problema (de Balance de Cargas/Fases, en estos ejemplos), quedando “atrapada” en un óptimo local.

Luego de estudiar las respuestas, se obtuvo una sensible mejora al eliminar el factor decaimiento,  $\delta(k)$ , que atenúa la comunicación entre óptimos individuales, enfatizando el carácter estocástico esta nueva X-Forma, EPSto GIST, cuya ecuación evolutiva de movimiento queda expresada como:

$$\mathbf{V}_i^{[k+1]} = w_{i1}^{[k]} \times (r_1^{[k]}) \times \mathbf{V}_i^{[k]} + w_{iC}^{[k]} \times (r_2^{[k]}) \times [\mathbf{b}_i^{[k]} - \mathbf{X}_i^{[k]}] + w_{iS}^{[k]} \times (r_3^{[k]}) \times [\mathbf{b}_G^{[k]} - \mathbf{X}_i^{[k]}] \times \alpha_{OG}^{[k]} + \sum_{j=1, j \neq i}^n (w_{iC}^{[k]} \times [\mathbf{b}_j^{[k]} - \mathbf{X}_i^{[k]}] \times \alpha_{OI}^{[k]}_{ij}) \quad (12)$$

La EPSto GIST, reemplazará en los Modelos HiperHeurísticos aquí propuestos, a la X-PSO Forma, extendida al dominio MultiObjetivo, EPSto GIST, empleada en el Modelo HY X-PSO CBR presentado en (Schweickardt, Casanova y Gimenez, 2013).

**F) X5: AFS (Artificial Fish School):** Esta X-Forma es la que confiere un carácter híbrido al conjunto de MetaHeurísticas que componen el dominio de los Modelos HiperHeurísticos presentados en este trabajo. También referida en el estado del arte como Algoritmo de Cardumen de Peces Artificiales. Se sustenta en dos tareas básicas: primero, construye un modelo simple de Peces Artificiales (AF) y segundo, implementa una estrategia de búsqueda del óptimo global, basada en el comportamiento que caracteriza la búsqueda de alimento de los individuos (AF) del grupo o Cardumen.

El Algoritmo intenta imitar el siguiente proceso bio-inspirado: En el agua, considerada como hábitat natural, cualquier pez puede localizar alimento siguiendo a otro pez, pues donde hay más concentración de alimento, hay mayor cantidad de peces. Para emular este proceso, son establecidos comportamientos que identifican al Cardumen de Peces Artificiales, y definidas algunas variables y parámetros. Se tipifican cinco comportamientos en el Cardumen: 1) De Ataque (Prey): El pez percibe la concentración de alimento en el agua, y escoge, por medio de su visión el movimiento a seguir; 2) Grupal (Swarm): En el movimiento del Cardumen, cada pez integra el grupo como un hábitat en el cual supervive y es garantizada una mayor seguridad de no ser presa de otro pez, en general; 3) De Seguimiento (Follow): En el movimiento del grupo de peces, cuando un pez individualmente, o un grupo de ellos, detecta alimento, la vecindad (sub-grupo próximo de peces) de los mismos, se alineará y seguirá su movimiento con el objeto de alcanzar rápidamente el alimento; 4) Registro (Bulletin Update): Se asemeja a la Memoria Autobiográfica que exhiben las partículas del PSO. La posición o estado de un pez artificial que ha alcanzado la máxima concentración de alimento, individualmente, es registrado en un boletín. Luego de cada acción o movimiento (iteración), el nuevo estado de cada pez artificial, es comparado con su registro en el Boletín, el cual, si dicho estado tiene mayor mérito que el registrado, se actualiza por reemplazo; y 5) Evaluación del Movimiento (Moving Evaluation): Los comportamientos Grupal y De Seguimiento, tendrán lugar en el Cardumen cuando exhiban pertinencia, efecto que será observable en el algoritmo, al describir sus procedimientos respectivos. En tal caso, el comportamiento del Cardumen seleccionado entre ambos, será aquel que conduzca a la mayor consistencia alimenticia (*food consistence*), representada por la posición de mayor aptitud. En caso contrario, no existe tal pertinencia, el comportamiento que tiene lugar es el De Ataque. En lo que respecta a la definición de los parámetros y

variables en AFS, se tiene: Sea  $[AF] = \{af_1, af_2, \dots, af_{NF}\}$  el conjunto de peces artificiales, referido como Cardumen o Población; cada  $af_i$ , ocupará, en el Espacio de Búsqueda y para cierta instancia de decisión (iteración del algoritmo), una posición  $\mathbf{x}(af_i)$ , la cual resulta un vector cuyas componentes son cada una de las variables de decisión del problema. La aptitud que exhibe el pez artificial  $af_i$ , en términos de localización de máximo alimento, ocupando la posición  $\mathbf{x}(af_i)$ , será indicada como  $f(\mathbf{x}(af_i))$ , siendo  $f$  la función de aptitud establecida para la estrategia de optimización. Simplificadamente, puede emplearse la notación:  $\mathbf{x}_i \equiv \mathbf{x}(af_i)$  y  $f(\mathbf{x}_i) \equiv f(\mathbf{x}(af_i))$ , porque todo pez artificial, siempre ocupa una posición, y siempre tiene una aptitud, tal y como se hace en el PSO. Sean, entonces, dos peces artificiales  $\mathbf{x}_i$  y  $\mathbf{x}_j$ , la distancia entre ambos resultará:  $d_{ij} = |\mathbf{x}_i - \mathbf{x}_j|$ . Además,  $visual$ , es el campo visual o alcance de un  $af_i$  y  $step$ , la distancia que puede moverse un  $af_i$  en cada paso de iteración;  $\delta$  es el grado de congestión e indica, en el comportamiento de un  $af_i$ , si existe alimento en la vecindad de cierto punto o concentrado en él. Su valor pertenece a  $[0, 1]$ .  $visual$ ,  $step$  y  $\delta$  son parámetros. Además se tienen los siguientes:  $NF$ : Número de peces artificiales o Tamaño de la Población/Cardumen,  $TryNum$ : es el máximo número de intentos de un pez artificial para encontrar una mejor posición,  $\mathbf{x}_{iBest}$ , y será aplicado en el comportamiento Prey o de Ataque. Adicionalmente  $MaxIter$ : es el número máximo de iteraciones del algoritmo, condición Estricta de salida del mismo, y  $NumEstable$  es el número fijado como límite de iteraciones en los que el Registro no es actualizado, por no existir cambios en las aptitudes (y, consecuentemente, posiciones de los  $af_i$  en el Cardumen), y supone una condición No Estricta de salida del Algoritmo. Los comportamientos De Ataque, Grupal o De Seguimiento, se definen en el Modelo, como sigue:

1) De Ataque: si, en cierta instancia de decisión,  $\mathbf{x}_i$  es el estado de  $af_i$  y  $\mathbf{x}_j$  es el estado de  $af_j$ , estando  $af_i$  en el campo de visión de  $af_j$ , esto es:  $|d_{ij}| \leq visual$ , entonces se procede a comparar las aptitudes  $f(\mathbf{x}_i)$  y  $f(\mathbf{x}_j)$ . Si  $f(\mathbf{x}_j) > f(\mathbf{x}_i)$ , entonces  $af_i$  se moverá  $Random(step)$  (función aleatoria de  $step$ ) hacia y en la dirección de  $af_j$ , dentro de su campo de visión. En caso contrario, intentará un número  $TryNum$  de veces encontrar una mejor posición.  $\mathbf{x}_{iBest}$ , que la que ocupa, dentro de su campo de visión, y, si falla,  $af_i$  se moverá aleatoriamente  $Random(step)$  dentro de su campo de visión. Matemáticamente y en pseudo-código:

Si  $f(\mathbf{x}_j) > f(\mathbf{x}_i)$  entonces

$$\mathbf{x}_{i\ next} = \mathbf{x}_i + \text{Random}(\text{Step}) \times [(\mathbf{x}_j - \mathbf{x}_i) / |\mathbf{x}_j - \mathbf{x}_i|]$$

Sino

Para  $k=1$  hasta  $TryNum$

$$\mathbf{x}_{i\ next} = \mathbf{x}_i + \text{Random}(\text{Step})$$

Si  $f(\mathbf{x}_{i\ next}) > f(\mathbf{x}_i)$  entonces

FinPara

FinSi

FinPara

$$\mathbf{x}_{i\ next} = \mathbf{x}_i + \text{Random}(\text{Step})$$

FinSi

Siendo  $Random(Step)$  una función que genera números aleatorios en el

intervalo  $[0, \text{step}]$  con distribución uniforme de probabilidad.

**2) De Grupo:** si, en cierta instancia de decisión,  $\mathbf{x}_i$  es el estado/posición de  $a_i$ ,  $nf$  es el número de  $a_f$  en el campo visual de  $a_i$ , dado por la función  $N$  definida mediante  $nf = N(\mathbf{x}_i, \text{visual}) = \{ \mathbf{x}_j / |d_{ij}| \leq \text{visual} \}$ , y  $\mathbf{x}_c$  es la posición centro de la vecindad de  $a_i$  y definida por  $\mathbf{x}_c = \sum_{j=1}^{nf} (\mathbf{x}_j/nf)$ , se tiene, en pseudo-código:

Si  $f(\mathbf{x}_c) > f(\mathbf{x}_i)$  y  $(nf/NF) < \delta$  entonces

$$\mathbf{x}_{i \text{ next}} = \mathbf{x}_i \times \text{Random}(\text{Step}) \times [(\mathbf{x}_i - \mathbf{x}_c)/|\mathbf{x}_i - \mathbf{x}_c|]$$

Sino

Comportamiento De Ataque

FinSi

**3) De Seguimiento:** si, en cierta instancia de decisión,  $\mathbf{x}_i$  es el estado/posición de  $a_i$ , y  $a_{i_{\text{Max}}}$  es el  $a_f$  en el campo visual de  $a_i$ , que ha alcanzado la máxima consistencia alimenticia de la vecindad  $nf$ ,  $\mathbf{x}_{\text{Max}}^{nf}$ , siendo  $f(\mathbf{x}_{\text{Max}}^{nf})$  su aptitud, se tiene en pseudo-código:

Si  $f(\mathbf{x}_{\text{Max}}^{nf}) > f(\mathbf{x}_i)$  y  $(nf/NF) < \delta$  entonces

$$\mathbf{x}_{i \text{ next}} = \mathbf{x}_i \times \text{Random}(\text{Step}) \times [(\mathbf{x}_i - \mathbf{x}_{\text{Max}}^{nf})/|\mathbf{x}_i - \mathbf{x}_{\text{Max}}^{nf}|]$$

Sino

Comportamiento De Ataque

FinSi

De modo que la pertinencia referida para los comportamientos De Grupo y De Seguimiento, está dada por la misma condición de congestión en la vecindad  $nf$ :  $(nf/NF) < \delta$ . Si esta condición no se satisface, no son pertinentes los comportamientos De Grupo o De Seguimiento, y tiene lugar el De Ataque.

Con estos desarrollos, puede ser definido el Algoritmo AFS, en forma de pseudo-código y por pasos: **Paso 1):** Inicialización de los parámetros  $NF$ ,  $\text{visual}$ ,  $\text{TryNum}$ ,  $\delta$ ,  $\text{step}$ ,  $\text{MaxItera}$  y  $\text{NumEstable}$ ; **Paso 2):** Inicialización de la Población o Cardumen; **Paso 3):** Inicialización del Registro (*Bulletin*) para evaluar los nuevos estados de cada  $a_f$  y, si procede, actualizarlos, durante el procedimiento iterativo, con los mejores estados alcanzados. En el comienzo, el Registro contiene posiciones/estados y aptitudes coincidentes con la inicialización de la Población/Cardumen; **Paso 4):** Se aplica el comportamiento normativo del Cardumen, que consiste en la Evaluación del Movimiento; esto supone, evaluar la pertinencia de los comportamientos De Grupo y De Seguimiento, observando si se satisface la condición  $(nf/NF) < \delta$ . De ser pertinentes, proceden ambas, y la posición y aptitud para un nuevo estado, proviene del comportamiento que arroje la mayor aptitud. Si no son pertinentes estos comportamientos, el Cardumen adopta el comportamiento De Ataque. Es claro que las evaluaciones y cálculos correspondientes, se ejecutan para cada  $a_f$  del Cardumen; **Paso 5):** Actualización del Registro (*Bulletin Update*): Si procede, se registran los nuevos valores de posición y aptitud para cada  $a_f$  del Cardumen, y se actualiza el número de iteración del proceso. Se evalúa si tiene lugar la condición de parada, consistente en la permanencia del Registro sin modificaciones durante el número  $\text{NumEstable}$  de iteraciones, o bien en alcanzar el número  $\text{MaxItera}$  de iteraciones. Si una de estas condiciones tiene lugar, se sigue con el **Paso 6)**, en contrario, se regresa al **Paso 4)**, incrementando, previamente, en 1 el contador de iteraciones; **Paso 6) Condición de Salida Satisfecha:** se almacenan

Posiciones y Aptitudes de cada af del Cardumen, así como otros parámetros de interés registrados. Desde allí se obtiene el  $\mathbf{x}_{Max}$  y su  $f(\mathbf{x}_{Max})$ .

### 2.2.3 Extensión Multiobjetivo de X-Formas PSO y AFS: X-(FPSO+FAFS)

Para no redundar y pretendiendo brindar una síntesis desde múltiples publicaciones en las que se presentan los aspectos conceptuales para extender estas X-Formas al dominio MultiObjetivo (conociendo el conjunto X-(FPSO+FAFS)), se sigue en este epígrafe la referencia (Schweickardt y Miranda, 2010). Son así destacados y sintetizados dos aspectos: los conceptuales o metodológicos y los operacionales o de implementación.

**1) Aspectos Metodológicos:** Al efecto de concebir una función que pondere la aptitud de las soluciones en un contexto MultiObjetivo de optimización, deben satisfacerse dos requerimientos:

**A) Pareto-Dominancia** en las soluciones: las soluciones igualmente óptimas deben satisfacer una escala pareto-dominate, lo cual supone que si dos soluciones componen un frente pareto-óptimo, ninguna de ellas puede ser mejorada en cierto objetivo, sin empeorar, al menos, en uno de los restantes (soluciones no dominadas), y resultan igualmente preferibles o aptas;

**B) Compatibilidad Métrica** respecto del Espacio de Búsqueda: El Espacio de Búsqueda tiene una influencia sustancial en la definición de la función de aptitud para una MetaHeurística MultiObjetivo. Las múltiples variables, discretas, con intervalos estrechos en sus valores factibles, y buenas soluciones muy dispersas en tal espacio, impactan fuertemente sobre la compatibilidad que debe guardar la métrica de la función de aptitud, respecto de los estímulos que induce sobre las partículas, para explorarlo eficientemente. Para las MetaHeurísticas MultiObjetivo sustentadas en la IG, una función de aptitud resultará métricamente compatible con el Espacio de Búsqueda, en la medida que pueda propiciar el cumplimiento de los cinco principios enunciados. Este es uno de los aspectos de mayor cuidado.

**C) Capacidad de Captar las Incertidumbres de Valor:** Se refiere a la posibilidad de modelar las incertidumbres asociándoles un grado de satisfacción respecto del cumplimiento de un objetivo/restricción individual del problema de optimización abordado.

**2) Aspectos Operacionales:** Para formalizar la extensión X-(PSO+AFS) al dominio MultiObjetivo, se aplica el Principio de Toma de Decisión en Ambientes Difusos:

a) Para cada Objetivo/Restricción, se asocia un Conjunto Difuso, solidario a la variable que lo representa. Para ello se construye una variable adimensional, denominada variable de apartamiento, y sobre la misma cada Conjunto Difuso es construido. Esto permite que tales conjuntos Objetivos/Restricciones, **O/R**, puedan mapearse por igual en la misma imagen, denominada Conjunto Difuso de Decisión, **D**, a través de un operador, **<C>**, referido como Confluencia, y que se corresponde típicamente con el operador Intersección. Para N Objetivos y H Restricciones Difusas vale la expresión:

$$\mathbf{D} = \mathbf{O}_1 \langle \mathbf{C} \rangle \mathbf{O}_2 \langle \mathbf{C} \rangle \dots \langle \mathbf{C} \rangle \mathbf{O}_N \langle \mathbf{C} \rangle \mathbf{R}_1 \langle \mathbf{C} \rangle \mathbf{R}_2 \langle \mathbf{C} \rangle \dots \langle \mathbf{C} \rangle \mathbf{R}_H \quad (13)$$

b) Asociado al operador **<C>** entre los Conjuntos Difusos, existe un

operador t (por t-norma) entre sus funciones de pertenencia, que genera, desde (13), el valor de pertenencia en el Conjunto Difuso de Decisión, **D**, conforme los valores individuales de las funciones de pertenencia del segundo miembro,  $\mu_o/\mu_R$ . Es decir:

$$\mu_D = \mu_{O1} t \mu_{O2} t \dots t \mu_{On} t \mu_{R1} t \mu_{R2} t \dots t \mu_{RH} \quad (14)$$

Luego, para la Población, P, o para una partícula individual, i, se define como decisión maximizante de Bellman y Zadeh (Schweickardt y Miranda, 2010), al valor de la función de pertenencia en el conjunto **D**, dado por:

$$\mu_D^{Max} = \text{MAX}^{[P/i]} \{ \mu_{O1} C \mu_{O2} C \dots C \mu_{On} C \mu_{R1} C \mu_{R2} C \dots C \mu_{RH} \} \quad (15)$$

c) con  $\mu_D^{Max} \equiv \text{fapD}$ , función de aptitud difusa, el conjunto X-(PSO+AFS), con capacidad de solución MonoObjetivo, pasa a transformarse en el conjunto X-(FPSO+FAFS), con capacidad de solución en problemas de optimización MultiObjetivo.

d) Los requerimientos **A**) (Pareto Dominancia) y **B**) (Compatibilidad Métrica) y **C**) (Captación de Incertidumbres de Valor), para el Espacio de Búsqueda que caracteriza a la Clase de Problemas considerada, la t-norma más apropiada, se denomina Producto de Einstein, y se define como:

~~$$\mu_{m1} t \mu_{m2} t \dots t \mu_{mM} \quad (16)$$~~

e) Sobre cada Objetivo/Restricción, O/R, los Conjuntos Difusos solidarios a sus variables de apartamiento, se definen del siguiente modo (no excluyente): Considérense, dos límites, superior e inferior, en los valores posibles de la variable correspondiente a cierto O/R, m, vm. Se referirán como vMax<sub>m</sub> y vMin<sub>m</sub>, respectivamente. Adicionalmente, sea  $p_\mu^m$  su ponderador exponencial, cuyo efecto sobre el conjunto difuso solidario, es su contracción ( $p_\mu^m > 1$ ), aumentando la importancia del objetivo/restricción en la confluencia <C>, o dilatación ( $p_\mu^m < 1$ ), disminuyendo dicha importancia. Entonces, la función de pertenencia adoptada, de característica base lineal (este es el caso cuando  $p_\mu^m = 1$ ), genérica para el O/R m-ésimo, resultará de la expresión condicional:  $\forall m$  en [1..M], con M variables O/R:

$$\mu_m = 1 ; \text{ si } vMin_m \geq vm \quad (17)$$

$$\mu_m = \left( \frac{(vMax_m - vm)}{(vMax_m - vMin_m)} \right)^{p_\mu^m} ; \text{ si } vMin_m \leq vm \leq vMax_m \quad (18)$$

$$\mu_m = 0 ; \text{ si } vMax_m \leq vm \quad (19)$$

Luego, la función de aptitud difusa fapD resultará:

$$\text{fapD} = \mu_D^{Max} = t_{PE}^{Max} \{ \mu_{m1}; \mu_{m2}; \dots; \mu_{mM} \} \quad (20)$$

### 2.3 Las X-(FPSO+ FAFS) y su habilidad para satisfacer los Principios IG

Habiendo realizando una buena cantidad de pruebas, fue posible validar la correlación, cualitativamente esperada desde las estrategias de movimientos formuladas, entre cada X-Forma del conjunto X-(FPSO+FAFS), y la habilidad para satisfacer predominantemente cierto Principio IG. Excluyendo la X-Forma X0, la FPSO Canónica, las cinco restantes han sido escogidas precisamente por tal habilidad. Se profundizan las conclusiones al respecto presentadas en

(Schweickardt, Casanova y Gimenez, 2013) respecto a Modelo HiperHeurístico HY X-FPSO CBR, con la diferencia que en este trabajo, como se verá, para la primer estrategia, Modelo HY X-(FPSO+AFS) CBR MPI NCPN, su dominio tiene cuatro X-Formas (mismo número) para componerlo, mientras que para la segunda estrategia, Modelo HY X-(FPSO+FAFS) CBR MPI NVPN, su dominio tiene cinco X-Formas, si bien dos de ellas resultarán ser la misma. En síntesis: **X1**: Habilidad específica en la Estabilidad; **X2**: Habilidad específica para la Proximidad, predominantemente, y Estabilidad, en iteraciones próximas a la convergencia del algoritmo metaheurístico; **X3**: Habilidad general en los cinco principios, fundamentalmente en Calidad y Adaptación. Constituye, por ello, la X forma de inicio en instancias tempranas de decisión; **X4**: Habilidad general en los cinco principios, pero fundamental o predominantemente en la Diversidad de Respuesta, aún en iteraciones tempranas; **X5**: la FAFS, en Espacios de Búsqueda como los que caracterizan a la Clase de Problemas aquí considerados, es la X-Forma más conservadora, con habilidad predominante sobre la Estabilidad, en cualquier instancia de decisión o iteración algorítmica. Por ello, se complementa muy bien con **X4** (FPSO GISiT), y los Modelos HiperHeurísticos propuestos en el presente trabajo, sacarán provecho de tal condición.

### 3. PLATAFORMA INTERFAZ DE PASO DE MENSAJES (MPI)

En general, y particularmente en los problemas de optimización, se pretenden y requieren nuevas aplicaciones, con mayores costos computacionales, para cubrir nuevas necesidades. Sin embargo, el rendimiento de las computadoras dentro del paradigma secuencial clásico está comenzando a saturarse: existen límites físicos asociados al incremento en la frecuencia del reloj. Por tal razón que equipos multi-procesadores y/o multi-núcleo han surgido, para resolver estos límites físicos, no intentando extenderlos, sino agregando más unidades de cómputo para que, en conjunto, brinden una mejor *performance* respecto del procesamiento secuencial. Se torna, por tanto, primordial, aprovechar al máximo este esquema de *hardware* mediante un nuevo concepto de *software*: la Computación Paralela.

La plataforma MPI (*Message Passing Interface*) o Interfaz de Paso de Mensajes, constituye un estándar que define la sintaxis y la semántica de las funciones contenidas en una biblioteca de paso de mensajes diseñada para ser usada en programas que exploten la existencia de múltiples procesadores (núcleos). Su principal característica es que no requiere de memoria compartida, por lo que es muy importante en la programación de sistemas distribuidos. Los elementos principales que intervienen en el paso de mensajes son el proceso que envía, el que recibe y el mensaje. Dependiendo de si el proceso que envía el mensaje espera a que el mensaje sea recibido, se puede hablar de paso de mensajes síncrono o asíncrono. En el paso de mensajes asíncrono, el proceso que envía, no espera a que el mensaje sea recibido, y continúa su ejecución, siendo posible que vuelva a generar un nuevo mensaje y enviarlo antes de que se haya recibido el anterior. Por este motivo se suelen

emplear buzones, en los que se almacenan los mensajes a espera de que un proceso los reciba. Generalmente empleando este sistema, el proceso que envía mensajes sólo se bloquea o detiene, cuando finaliza su ejecución, o si el buzón está lleno. En el paso de mensajes síncrono, el proceso que envía el mensaje espera a que un proceso lo reciba para continuar su ejecución. Por esto esta técnica se suele llamar encuentro, o *rendezvous*.

Si bien la MPI es un protocolo de comunicación entre computadoras, es el estándar para la comunicación entre los nodos que ejecutan un programa en un sistema de memoria distribuida. De manera que tales nodos pueden ser procesadores o núcleos que integran el *hardware* de cómputo en una misma computadora, bajo un sistema operativo capaz de explotar esta cualidad eficientemente (tal como Linux). Las implementaciones en MPI consisten en un conjunto de bibliotecas de rutinas que pueden ser utilizadas en programas escritos en los lenguajes de programación C, C++, Fortran y Ada. La ventaja de MPI sobre otras bibliotecas de paso de mensajes, es que los programas que utilizan la biblioteca son portables (dado que MPI ha sido implementado para casi toda arquitectura de memoria distribuida), y muy rápidos, (porque cada implementación de la biblioteca ha sido optimizada para el *hardware* en la cual se ejecuta). El sistema operativo que soporta la plataforma MPI para implementar los Modelos HiperHeurísticos propuestos paralelizados, y resolver el problema de Balance de Carga o de Fases en un SDEE BT (PBC SDEE BT), es Linux, y el lenguaje de programación escogido, ha sido C++.

#### 4. DESARROLLO DE LOS MODELOS HIPERHEURÍSTICOS PROPUESTOS BAJO COMPUTACIÓN PARALELA

##### 4.1 Modelo HiperHeurístico HY X-(FPSO+FAFS) CBR MIP NCPN

Como se ha dicho, la FS de este Modelo Hiperheurístico Basado en Razonamiento, CBR, es implementada mediante una RNR, cuya estructura resulta idéntica a las presentada en (Schweickardt, Casanova y Gimenez, 2014), tanto en términos de diseño (número de entradas, número de capas ocultas, cantidad de neuronas por capa oculta, y número de salidas), como patrones de entrada/salida para su entrenamiento. Por tal razón, no se redundará sobre sus detalles en este epígrafe, sino que se describirán las diferencias, habida cuenta que el dominio de MetaHeurísticas de la FS es diferente, y se describirá la estrategia seguida para su implementación soportada bajo Computación Paralela mediante la plataforma MPI.

La primer estrategia, que conduce al Modelo HiperHeurístico aquí abordado, primero determina, operando sobre, tal y como lo permite la plataforma MIP, la definición de un número variable de núcleos, reales o virtuales. Este enfoque, como se dijo, el más simple, consiste en asignar a cada uno de los procesadores o núcleos, un proceso secuencial idéntico, pero que trate con un número de partículas resultante de dividir el número poblacional, NP, por el número de núcleos empleados, Nn. De modo que cada procesador o núcleo ejecutará el mismo Modelo HiperHeurístico, con  $np = NP/Nn$  partículas. Para obtener los mejores resultados, es entonces necesario especificar a MPI la

cantidad de núcleos que tiene cada computadora. Esto se realiza mediante la especificación de un archivo llamado *machinefile*. En este caso, el archivo resultante constó de una sola línea: *localhost: Nn*. Esta línea se corresponde con el nombre del equipo, junto a la cantidad de núcleos del mismo. En el problema abordado PBC SDEE BT, definiendo un dominio compuesto por las MetaHeurísticas {X2, X3, X4 y X5}, ensayando con Nn en [2..5], y para NP = 300, resultó Nn = 4, ser el número de núcleos con mejor performance para este problema (óptimo), debido a que, a mayor cantidad de núcleos, se tendrá mayor costo computacional de comunicación entre los procesos secuenciales ejecutados por cada núcleo individualmente.

La implementación del Modelo HiperHeurístico bajo este esquema, resulta bastante directa. Sólo serán destacados los detalles principales de las operaciones concernientes a MPI, la cual, vale decir, es una plataforma de libre acceso. Estos detalles son: a) la comunicación de la mejor aptitud nuclear,  $\mu_n^{Max}$ , reteniendo además, la aptitud nuclear media,  $\mu_n^{Med}$ , y la aptitud nuclear mínima,  $\mu_n^{Min}$ , con el objeto de que pueda operar la RNR sobre la que se implementa la FS del Modelo HiperHeurístico. Estos tres valores serán referidos, en este contexto, como Vector de Aptitud Nuclear [ $\mu_n^{Max}$ ,  $\mu_n^{Med}$ ,  $\mu_n^{Min}$ ], y b) Para comunicar el Vector de Aptitud Nuclear a la RNR, así como la mejor Aptitud Global,  $\mu_n^{MaxG}$ , a cada núcleo se utilizó la función *MPI\_Allreduce*, con operación *MPI\_MAXLOC*. La ejecución de esta instrucción, en cada proceso, devuelve, entonces el Vector de Aptitud Nuclear para cada uno de los Nn = 4 núcleos, junto a su rango o número de proceso, en una estructura *MPI\_FLOAT\_INT*. De este modo, al igual que en la RNR del Modelo HiperHeurístico HY F-PSO CBR propuesto en (Schweickardt, Casanova y Gimenez, 2014), en cada iteración se recopilan los Vectores de Aptitudes Nucleares, desde cada núcleo, para la MetaHeurística del dominio {X2, X3, X4 y X5} en curso de aplicación. Como la RNR, fue entrenada en base a un gran número de Patrones de Entrada capaces de indicar, en base al Mejor Vector de Aptitud Nuclear y su estado de habilidad para satisfacer cada Principio IG, en el Patrón de Salida correspondiente, cuál es la MetaHeurística más apta de tal dominio, a ser aplicada en la próxima instancia de decisión, la RNR es idéntica a la del Modelo en tal referencia presentado (HY X-FPSO CBR), con Patrones Entrada/Salidas compuestos por números binarios: se tienen 15 entradas en la Capa de Entrada (3 componentes del Mejor Vector de Aptitud Nuclear x 5 Principios IG), 1 Capa Oculta con 10 neuronas, y 7 salidas en la Capa de Salida. Las 5 primeras salidas, establecen con un bit en 1 o en 0, si el Principio IG, en el orden definido aquí, se satisface o no; mientras que las 2 salidas restantes, componen un número binario de 0 a 3, cuatro estados, el cual indica qué MetaHeurística es la más apta (sumando un 2 al equivalente decimal) para ser aplicada en la próxima instancia de decisión, desde el conjunto {X2, X3, X4, X5}. Por ser, como se dijo, constante el número de partículas por núcleo, y tratarse de un Modelo HiperHeurístico CBR paralelizado mediante la Plataforma MPI, se adopta el nombre HY X-(FPSO+FAFS) CBR MPI NCPN.

#### 4.2 Modelo HiperHeurístico HY X-(FPSO+FAFS) CBR MIP NVPN

Esta segunda estrategia, conduce a un Modelo HiperHeurístico más complejo de implementar, pero, tomando los elementos del Modelo HiperHeurístico anterior, más sencillo de explicar. **1)** En este caso, el número de núcleos es igual al número de PIG. En cada núcleo se evalúa, entonces un Principio IG. De modo que se tendrán 5 núcleos. **2)** Cada núcleo ejecuta la MetaHeurística más apta para satisfacer el PIG asignado, desde el dominio {X2, X3, X4, X5}, considerando que, luego de diversas pruebas, resultó conveniente bajo esta estrategia, seleccionar dos núcleos que ejecuten el algoritmo correspondiente a la MetaHeurística X4, FEPSO GStT, habida cuenta de que manifiesta muy buenas aptitudes para los cinco PIG, con predominancia, como se dijo, sobre la Diversidad de Respuesta. **3)** La FS se implementa con una RNR de idéntico diseño al anterior, mismo número de entradas en la capa de Entrada, una Capa Oculta con 10 neuronas, y 7 salidas en la Capas de Salida. Pero difiere en el tipo de Patrones de Entrada/Salida empleados tanto para su entrenamiento como para su simulación con fines predictivos (Función de Selección de la HiperHeurística). La Capa de entrada es idéntica a la diseñada para la estrategia anterior: se tienen 15 entradas (3 componentes del Mejor Vector de Aptitud Nuclear x 5 Principios IG) empleando números binarios, y con la misma significación. En cambio, como para esta segunda estrategia lo que se modifica es el número de partículas asignadas por núcleo, las salidas se codifican de un modo diferente: las primeras 5 son números enteros, que corresponden al tamaño de las subpoblaciones asignadas para cada núcleo-PIG en la siguiente iteración algorítmica. Las 2 restantes, representan un efecto de que aquí se refiere como Saturación de X-Forma. Se dice que un X-Forma se Satura, cuando de Vector de Aptitud Nuclear arriba a la condición:  $\mu_n^{\text{Max}} = \mu_n^{\text{Med}} = \mu_n^{\text{Min}} = \mu_n^{\text{Sat}}$ . En tal condición la MetaHeurística asignada a SIP-núcleo, no puede producir mejoras, y, llevando un registro de los núcleos que van saturándose, las salidas codifican con un 1 la MetaHeurística Saturada, y con un 0 la que no lo está. **4)** Frente a una situación de Saturación de X-Forma, se asigna al núcleo que la estaba ejecutando, mediante una función auxiliar que toma el registro referido, y los Vectores de Aptitud Nuclear, la X-Forma que propicia la mayor Estabilidad: la FAFS (excepto que se sature esta X-Forma, en el núcleo en la cual fue asignada, en cuyo caso, no hay cambios). Frente a la Saturación de una X-Forma, no hay reasignación de partículas para cada núcleo. **5)** En todos los casos simulados, El Modelo HiperHeurístico concebido bajo esta estrategia, termina con la X-Forma FAFS Saturada en los 5 núcleos, ejecutando la misma X-Forma, hacia una convergencia estable y con soluciones de calidad, requiriendo un tiempo inferior que el Modelo HiperHeurístico basado en la primera estrategia. Por ser variable el número de partículas asignadas por núcleo, hasta la Saturación de todo el conjunto de X-Formas, es que este Modelo HiperHeurístico se designa HY X-(FPSO+AFS) CBR MPI NVPN.

## 5. SIMULACIONES Y RESULTADOS

Los datos, esquema trifilar y resultados de la reconexión de cargas óptima o más satisfactoria, obtenidos por aplicación de ambos Modelos HiperHeurísticos, son los mismos que pueden ser consultados en múltiples referencias, por caso (Schweickardt y Miranda, 2010). Lo que se modifica sensiblemente, es el tiempo de ejecución, que, a los efectos de los objetivos de este trabajo, y por cuestiones de espacio, es el resultado más indicativo para corroborar el aporte realizado. Utilizando, en todos los casos el Sistema Operativo Linux, codificando las implementaciones en C++, se obtuvieron los siguientes tiempos de ejecución: 1) Caso base, aplicación de la MetaHeurística FEPSO GIST: **35 min**; 2) HY X-(FPSO+FAFS) CBR MPI NCPN: **12 min**; 3) HY X-(FPSO+FAFS) CBR MPI NVPN: **7 min**.

## 6. CONCLUSIONES

Más allá de los desarrollos presentados, los novedosos aportes logrados mediante estos dos Modelos HiperHeurísticos soportados en Computación Paralela bajo la Plataforma MPI, se dejan dos conclusiones: 1ra) Poco existe en el estado del arte sobre la temática de las HiperHeurísticas, y no se tienen antecedentes sobre un enfoque de Procesamiento Paralelo como el aquí presentado y 2da) Existe una clara línea de investigación que debería profundizarse, mejorando estas dos estrategias, las cuales resultan ser sólo un comienzo.

## 7. REFERENCIAS

BENI, G.; WANG, J. (1989): "SWARM INTELLIGENCE IN CELLULAR ROBOTIC SYSTEMS". Proceedings NATO Advanced Workshop on Robots and Biological Systems - Tuscany, Italy - June 1989.

PETROVIC, S.; QU, R. (2002): "CASE-BASED REASONING AS A HEURISTIC SELECTOR IN A HYPER-HEURISTIC FOR COURSE TIMETABLING PROBLEMS". Knowledge-Based Intelligent Information Engineering Systems and Allied Technologies, Proceedings of KES'02 - Vol 82 - Crema, Italy - pgs. 336, 340.

SCHWEICKARDT, G.; MIRANDA, V. (2010): "METAHEURISTICA 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.; GIMENEZ, J. (2013): "HIPERHEURÍSTICA BASADA EN RAZONAMIENTO CON DOMINIO EN

METAHEURÍSTICAS X-FPSO MULTIOBJETIVO. APLICACIÓN SOBRE UN OPTIMIZACIÓN DINÁMICA POSIBILÍSTICA. PARTE 1”. Revista de la Escuela de Perfeccionamiento en Investigación Operativa – Nro. 34 – pgs. 8, 29.

SCHWEICKARDT, G.; CASANOVA, C.; GIMENEZ, J. (2014): “HIPERHEURÍSTICA BASADA EN RAZONAMIENTO CON DOMINIO EN METAHEURÍSTICAS X-FPSO MULTIOBJETIVO. APLICACIÓN SOBRE UN OPTIMIZACIÓN DINÁMICA POSIBILÍSTICA. PARTE 2”. Revista de la Escuela de Perfeccionamiento en Investigación Operativa – Nro. 34 – pgs. 128, 148.

SCHWEICKARDT, G.; MIRANDA, V.; GIMENEZ, J. (2010): “DOS ENFOQUES METAHEURÍSTICOS PARA RESOLVER PROBLEMAS DE OPTIMIZACIÓN COMBINATORIA MULTICRITERIO: FUZZY EVOLUTIONARY PARTICLE SWARM OPTIMIZATION CON TOPOLOGÍA ESTRELLA GLOBAL/INDIVIDUAL (FEPSO GIST) Y FUZZY SIMULATED ANNEALING (FSA)”. Anales del Encuentro XXIII ENDIO - XXI EPIO - II ERABIO, Tandil, Argentina.