

## Partición de Sistemas Eléctricos en Subsistemas menores para su Resolución Distribuida

B. Barán<sup>+</sup>, D. Benítez y R. Ramos  
 Centro Nacional de Computación  
 Universidad Nacional de Asunción  
 Paraguay

### RESUMEN

El presente trabajo propone un método para descomponer sistemas eléctricos, de manera a facilitar su resolución utilizando ambientes distribuidos heterogéneos como computadores paralelos y/o redes de computadores. La presente propuesta asigna a los procesadores de un sistema distribuido heterogéneo distintos números de ecuaciones según sea la performance relativa de los procesadores a ser utilizados. Asimismo, el método presentado permite utilizar la técnica de solapamiento parcial (*Partial Overlapping*) para mejorar la convergencia. Además, se introduce un parámetro de selección para escoger la mejor partición de un conjunto dado. Se presentan resultados experimentales que avalan la presente propuesta.

**Palabras clave:** *Partición de sistemas - Paralelismo - Método iterativo.*

### 1. INTRODUCCIÓN

Las implementaciones paralelas de algoritmos iterativos son una opción importante dentro del contexto de la computación distribuida [1]. Para aprovechar sus ventajas en la resolución de sistemas eléctricos de gran porte es necesario descomponer la red eléctrica en subredes menores, las cuales pueden ser asignadas a los distintos procesadores de un sistema distribuido para su resolución paralela. Esta descomposición estará condicionada por dos factores:

- a) la capacidad de procesamiento relativo de cada máquina del sistema distribuido, factor que determina las dimensiones de las subredes asignadas a cada procesador;
- b) el grado de acoplamiento que puedan tener las subredes entre sí, lo que determina la dependencia entre las variables y por consiguiente, influye en la convergencia del algoritmo [2].

Así, el sistema debe descomponerse de forma tal que a cada procesador se le asigne una subred de dimensión proporcional a su performance, y que la dependencia entre las variables actualizadas por cada procesador facilite la convergencia del algoritmo iterativo a ser utilizado.

Desde la década del 60[3] fueron publicados diversos métodos de descomposición, en su mayoría diseñados para procesadores, problemas o algoritmos bien específicos [4-5]. Así, Vale et al. [6] formuló una propuesta para la resolución de sistemas de ecuaciones lineales utilizando computadores paralelos. Sin embargo, la Descomposición  $\epsilon$  [7] que desprecia ligaciones débiles, es la más utilizada en la actualidad, por su simplicidad.

Inspirada en el trabajo de Vale et al. [6], se desarrolla la presente propuesta que permite generar particiones en número y tamaño controlables por el operador, válida para cualquier sistema eléctrico. Además, se introduce un parámetro que permite escoger una buena partición de un conjunto de particiones y se sugiere la utilización de solapamiento parcial (*partial overlapping*) [8].

La sección 2 formula matemáticamente el problema; mientras la sección 3 presenta el método propuesto. Resultados experimentales se presentan en la sección 4 y las conclusiones se resumen en la sección 5.

<sup>+</sup>Centro Nacional de Computación - Campus Universitario - San Lorenzo, Paraguay  
 Casilla de Correos 1439; teléfonos (595-21)585.550; E-mail: bbaran@una.py

## 2. FORMULACIÓN DEL PROBLEMA

Considerando un sistema de  $n$  ecuaciones algebraicas con  $n$  incógnitas dado por:

$$\Phi(x)=0, \quad x \in \mathfrak{R}^n, \quad \Phi : \mathfrak{R}^n \rightarrow \mathfrak{R}^n \quad (2.1)$$

definido en un dominio  $D \subset \mathfrak{R}^n$ ; se desea resolver (2.1) utilizando un sistema distribuido con  $p$  procesadores,  $n \geq p$ . Para esto, se debe descomponer el problema de forma tal que cada procesador *resuelva* solamente una parte del sistema, comunicando sus resultados parciales a los demás procesadores del sistema computacional.

Sea la Descomposición Cartesiana:

$$\mathfrak{R}^n = \mathfrak{R}^{n_1} \times \dots \times \mathfrak{R}^{n_p}, \quad n_1 + \dots + n_p = n \quad (2.2)$$

con  $D \subset \mathfrak{R}^n$  un dominio tal que

$$D = D_1 \times \dots \times D_p, \quad D_i \subset \mathfrak{R}^{n_i}, \quad \forall i \in \{1, \dots, p\} \quad (2.3)$$

Entonces:

$$x = \begin{bmatrix} x_1^T & \dots & x_p^T \end{bmatrix}^T, \quad x_i \in D_i, \quad \forall i \in \{1, \dots, p\} \quad (2.4)$$

Análogamente,

$$\Phi(x) = \begin{bmatrix} \Phi_1^T(x) & \dots & \Phi_p^T(x) \end{bmatrix}^T, \quad \Phi_i : D \rightarrow \mathfrak{R}^{n_i} \quad (2.5)$$

Consecuentemente, la ecuación a ser resuelta en cada procesador  $i$  (problema local), estará dada por:

$$\Phi_i(x) = 0, \quad \forall i \in \{1, \dots, p\} \quad (2.6)$$

**Nota 1:** La partición adecuada de la función  $\Phi(x)$  en  $p$  funciones  $\Phi_i(x)$ , es el objetivo central del presente trabajo.

Para resolver el problema será utilizado un algoritmo iterativo de la forma:

$$x^{k+1} = G(x^k) \quad (2.7)$$

donde  $k$  representa la iteración. Por lo tanto, en forma análoga a (2.4) y (2.5):

$$G(x) = \begin{bmatrix} G_1^T(x) & \dots & G_p^T(x) \end{bmatrix}^T, \quad G_i(x) : D \rightarrow D_i \quad (2.8)$$

La implementación síncrona de (2.7) para el procesador  $i$  será entonces:

$$x_i^{k+1} = G_i(x^k) \quad (2.9)$$

Considerando un contexto asíncrono, se denota como  $x_j(d_j^i(k))$  al valor de  $x_j$  enviado desde el procesador  $j$  y disponible en el procesador  $i$  al iniciar su iteración  $k$ . Denotamos así como  $x^i(k)$  al vector  $x$

disponible en el procesador  $i$  en el momento de su iteración  $k$ , esto es:

$$x^i(k) = \begin{bmatrix} x_1^T(d_1^i(k)) & \dots & x_p^T(d_p^i(k)) \end{bmatrix}^T \quad (2.10)$$

Usando esta notación dada en [1], podemos escribir el algoritmo asíncrono basado en (2.8) en la forma :

$$x_i(k+1) = G_i(x^i(k)), \quad \forall i \in \{1, \dots, p\} \quad (2.11)$$

En [9] se deriva una condición suficiente de convergencia del algoritmo, demostrando el siguiente teorema:

### **Teorema 1 (Condición suficiente de convergencia)**

*Bajo las hipótesis de: unicidad de la solución en el dominio  $D$ , asincronismo parcial y operador bloque-Lipschitz, el algoritmo asíncrono representado por (2.11) converge a un único punto fijo  $x^*$  en  $D$  si  $\rho(H) < 1$  donde  $\rho$  es el radio de la matriz de comparación  $H$ .*

En resumen, el radio espectral de la matriz de comparación es un parámetro que nos permitiría asegurar a priori que el algoritmo converge [10], y por consiguiente podría ser utilizado para escoger buenas particiones.

La técnica de solapamiento parcial (*partial overlapping*) fue introducida para mejorar la convergencia de los métodos de resolución en bloque, y consiste en resolver una misma ecuación repetidamente en varios bloques (o procesadores de una implementación paralela) [8].

## 3. PROPUESTA DEL TRABAJO

El método de descomposición propuesto utiliza la matriz simétrica de admitancias  $Y$  ( $y_{ij} = y_{ji}$ ) de dimensión  $n \times n$  y elementos diagonales no nulos ( $y_{ii} \neq 0$ ), cuyos elementos  $y_{ij}$  ( $i \neq j$ ) representan el acoplamiento existente entre las barras  $i$  y  $j$  respectivamente.

Decimos que las barras no son adyacentes si  $y_{ij} = 0$ ; caso contrario las barras son adyacentes. Se dice que barras adyacentes están débilmente acopladas si  $y_{ij} \rightarrow 0$  y que están fuertemente acopladas si  $y_{ij}$  es muy grande.

Básicamente, el método propuesto consiste en la formación de  $p$  subredes a partir de  $p$  barras iniciales, llamadas *semillas*, donde  $p$  representa también el número de procesadores. A estas barras semillas se van agregando las demás barras del sistema hasta que este quede totalmente descompuesto en  $p$  subsistemas, de dimensiones relativas especificadas.

Basados en el principio descrito, se propone un método compuesto por cuatro etapas [11]: clasificación de barras, selección de semillas, formación de particiones y selección de la mejor partición, inspirado en el trabajo de Vale et al. [6].

### 3.1. Clasificación de barras:

Esta se lleva a cabo estableciendo un ranking de las barras en base al *peso* relativo de cada una de ellas, calculado según:

$$P_i = \sum_{m_{ij} \neq 0} (m_{ij})^{z_{ij}} \quad \forall i \in \{1, \dots, n\}, \quad (3.1)$$

donde

$$m_{ij} = |y_{ij}| / |y_{ii}|,$$

$$z_{ij} = m_{ij} / D$$

$$D = \frac{1}{nc} \sum_{i=1}^n \sum_{j=1, j \neq i}^n |y_{ij}|,$$

$$nc = \text{número de } y_{ij} \neq 0$$

### 3.2. Selección de semillas

Deben seleccionarse como semillas a aquellas barras que sean centro de aglutinamiento y que no se encuentren fuertemente acopladas entre sí. Para ello, se consideran como candidatas a semillas sólo aquellas barras que posean un valor de peso significativo, es decir, que cumplan con la condición  $P_i > vlim$ , donde  $vlim$  es un parámetro pre-definido. Alrededor de cada una de estas barras candidatas se agrupan  $ngrup$  barras adyacentes, calculándose posteriormente la sumatoria de los pesos de las barras agrupadas para cada candidata. Se seleccionan como semillas aquellas barras que posean los mayores valores de sumatorias, cuidando que ninguna semilla se encuentre entre las  $nvec$  primeras barras agrupadas alrededor de las demás semillas.

### 3.3. Formación de particiones

Una vez determinadas las semillas para cada subsistema, es necesario adicionar las demás barras a las subredes adecuadas. Esto se realiza de la siguiente manera: a cada paso del proceso, la barra a ser agregada a una subred es la adyacente a ésta que posea el mayor peso. La determinación de las barras a ser anexadas a una dada semilla depende del ranking de barras, establecido en la etapa 1 (sección 3.1). En caso de que dos o más subredes seleccionen a la misma barra como candidata a ser incluida, el método la asignará a aquella subred con la cual posea un mayor acoplamiento. Si no es posible desempatar y la barra en disputa posee un peso significativo, realizar solapamiento parcial sería una opción recomendable. Para esto se asigna la barra a todas las subredes que la hayan seleccionado como candidata. A medida que las subredes van siendo formadas, debe verificarse que el tamaño de las mismas se encuentre en proporción a la performance relativa  $w$  de los procesadores a los cuales serán asignadas. Para esto, se anexan barras sólo a aquellas subredes que aun no alcanzaron la proporción de barras sugerida por su performance relativa. El proceso de agrupamiento de barras continua hasta que todas las barras hayan sido asignadas a alguna subred. En caso de obtenerse mas de una descomposición, se utiliza un criterio de selección para elegir la partición mas conveniente.

### 3.4. Selección de la mejor partición

Partiendo de distintas ternas de parámetros  $vlim$ ,  $ngrup$  y  $nvec$  pueden ser determinados varios conjuntos diferentes de semillas que a su vez servirán de pie para generar diferentes particiones. Para identificar a la mejor partición de todas las obtenidas, se impone aquí una selección de particiones utilizando para ello un parámetro de selección.

En [2] se proponen varios parámetros posibles de selección, indicándose como óptimo al Parámetro de Acoplamiento "Par\_A", dado por:

$$Par\_A = \sum |y_{ij}|, \quad \forall \text{rama cortada por la particion}$$

esto es, la sumatoria de los valores absolutos de los acoplamientos entre barras separadas por la partición.

En base a lo expuesto en la sección 2, este trabajo propone como parámetro de selección al radio espectral de la matriz de comparación  $H$ , siendo elegida como óptima aquella partición que presente el mínimo  $\rho(H)$ . La efectividad de este parámetro frente al parámetro Par\_A quedará demostrada por los resultados experimentales presentados en la sección 4.

## 4. RESULTADOS EXPERIMENTALES

La eficiencia del método descrito fue verificada resolviendo el problema del Flujo de Potencia para dos sistemas tipos de la IEEE: el sistema IEEE-14 y el sistema IEEE-118. Dicho problema puede ser formulado como un sistema cuasi-lineal de ecuaciones [10] de la forma:

$$Yx = I(x), \quad Y \in C^{n \times n}, \quad x \in C^n, \quad I(x) \in C^n \quad (4.1)$$

donde  $Y$  es la matriz de admitancias de la red,  $x$  el vector de voltajes en las barras e  $I(x)$  el vector de corrientes inyectadas (función de la incógnita  $x$ ).

La matriz de acoplamientos a ser utilizada será la matriz de admitancias  $Y$ , teniendo en cuenta que ésta representa en forma efectiva al acoplamiento entre los voltajes en las barras.

Las diferentes particiones generadas por el método propuesto fueron comparadas con otras particiones determinadas arbitrariamente y con particiones generadas por la Descomposición  $\epsilon$  [7]. Además se realizó un estudio comparativo entre los parámetros de selección  $\rho(H)$  y Par\_A.

### 4.1. Sistema IEEE de 14 barras

Se aplicó la metodología propuesta a la descomposición del sistema IEEE de 14 barras para la resolución del problema de flujo de potencia.

Este sistema fue resuelto en forma síncrona (2.9) y asíncrona (2.16), utilizando un sistema distribuido constituido por dos workstations: una SUN SPARC y una DEC 3000, con performance relativa de  $w = [4 \ 1]^T$ . Para esta relación de carga computacional, se analizaron todas las 286 particiones posibles del sistema.

Las Tablas I y II muestran los resultados experimentales. Se observan las posiciones de las 2 particiones (56 y 59 respectivamente) generadas por el método propuesto en los diversos *ranking*, tanto para el caso síncrono como para el asíncrono. La partición 59 es la seleccionada como mejor partición por el método propuesto cuando se utiliza  $\rho(H)$  como parámetro de selección. Inclusive, en el caso asíncrono esta partición es la óptima. Nótese además que si se utilizara Par\_A como parámetro de selección, la partición 56 sería elegida como la mejor, siendo que ésta es en realidad inferior a la partición 59 seleccionada por el parámetro  $\rho(H)$ .

Resolución síncrona del sistema IEEE-14						
Part.	Posición respecto a:					
	iter.	% sup.	T. Real	% sup.	T. CPU	% sup.
59	3 <sup>ra</sup>	1.04	4 <sup>ta</sup>	1.39	3 <sup>ra</sup>	1.04
56	47 <sup>a</sup>	16.43	34 <sup>a</sup>	11.88	43 <sup>a</sup>	15.03

Tabla I: Resolución síncrona

Resolución asíncrona del sistema IEEE-14						
Part.	Posición respecto a:					
	iter.	% sup.	T. Real	% sup.	T. CPU	% sup.
59	7 <sup>ma</sup>	2.44	1 <sup>ra</sup>	0.35	1 <sup>ra</sup>	0.35
56	82 <sup>a</sup>	28.67	45 <sup>a</sup>	15.73	56 <sup>a</sup>	19.58

Tabla II: Resolución asíncrona

La Tabla III indica las correlaciones entre los valores medidos y los parámetros de selección a comparar, tanto en el caso síncrono como en el asíncrono. Se verifica en ambos casos que el radio espectral  $\rho(H)$  presenta una mejor correlación que el parámetro de selección Par\_A.

	Iteraciones		Tiempo Real		Tiempo de CPU	
	sin.	asin.	sin.	asin.	sin.	asin.
$\rho(H)$	0.41	0.87	0.37	0.34	0.41	0.39
Par_A	0.03	-0.05	0.01	0.08	0.01	0.01

Tabla III: Correlaciones

4

## 4.2. Sistema IEEE de 118 barras

El procesamiento paralelo muestra todo su potencial en la resolución de problemas de grandes dimensiones. Teniendo en cuenta la necesidad de evaluar el

comportamiento del método propuesto con sistemas eléctricos de mayor porte, el problema del Flujo de Potencia fue resuelto sobre el Sistema IEEE-118 de forma a utilizar 4 procesadores de igual performance relativa. Fueron implementadas las particiones generadas por el método propuesto, utilizando semillas seleccionadas tanto de forma automática como manual. Se analizó además la partición generada por la Descomposición  $\epsilon$ , así como particiones realizadas manualmente sobre el grafo del sistema. Estas últimas fueron organizadas tomando en cuenta criterios empíricos y la experiencia del operador sobre el problema. Debido a la gran dimensión del sistema, no es posible un estudio exhaustivo sobre todas las particiones posibles. Como ejemplo, cabría considerar todas las formas posibles de partir el sistema en dos subsistemas iguales. Tendríamos así:

$$C_{59}^{118} = \frac{118!}{(118-54)! \times 54!} = 1,6 \times 10^{34} \text{ particiones}$$

Por este motivo, y adicionalmente a las particiones antes citadas, fueron analizadas 100 particiones realizadas al azar en forma aleatoria.

El parámetro de comparación en este caso fue solamente el tiempo real utilizado por el sistema distribuido en llegar a la solución.

En las tablas IV y V se muestran los resultados de estos estudios, indicando el posicionamiento de las distintas particiones evaluadas.

Puede verificarse fácilmente que las particiones generadas por el método propuesto se encuentran entre las mejores posibles, especialmente en el caso de la resolución asíncrona. Este último caso es de especial interés, al ser la implementación asíncrona la que en general ofrece un mejor aprovechamiento de las capacidades de procesamiento de las máquinas, al no existir tiempos muertos de sincronización en el proceso de resolución.

Posición	Tipo de partición	Semillas	T.(s)
1 <sup>ra</sup>	Método propuesto	man.	57
2 <sup>da</sup>	Método propuesto	man.	58
3 <sup>ra</sup>	Método propuesto	man.	62
3 <sup>ra</sup>	Descomposición $\epsilon$	---	62
4 <sup>ta</sup>	Método propuesto	man.	87
5 <sup>ta</sup>	Método propuesto	autom.	95
6 <sup>ta</sup>	Partición manual	---	139
7 <sup>ma</sup>	Partición manual	---	149
8 <sup>a</sup>	Partición manual	---	289

Tabla IV: Resolución síncrona

Posición	Tipo de partición	Semillas	T.(s)
1 <sup>ra</sup>	Método propuesto	autom.	30
2 <sup>da</sup>	Método propuesto	man.	31
3 <sup>ra</sup>	Método propuesto	man.	43
4 <sup>ta</sup>	Método propuesto	man	44
5 <sup>ta</sup>	Descomposición $\epsilon$	---	189
6 <sup>ta</sup>	Método propuesto	man.	N.C. <sup>+</sup>
6 <sup>ta</sup>	Método propuesto	man	N.C. <sup>+</sup>
6 <sup>ta</sup>	Partición manual	---	N.C. <sup>+</sup>
6 <sup>ta</sup>	Partición manual	---	N.C. <sup>+</sup>
6 <sup>ta</sup>	Partición manual	---	N.C. <sup>+</sup>

+ N.C. : No converge

Tabla V: Resolución síncrona

Al analizar la resolución del problema utilizando las particiones generadas aleatoriamente, ninguna de ellas pudo llegar a la solución, tanto para el caso síncrono como para el asíncrono. Esto enfatiza una vez mas la importancia de una buena partición en la resolución paralela de este tipo de problemas.

## 5. CONCLUSIONES

Los métodos de descomposición de sistemas eléctricos tradicionalmente utilizados no poseen dominio simultáneo sobre la cantidad de subredes en la cual el sistema en estudio es descompuesto y el tamaño relativo de los subproblemas, o consideran soluciones muy particulares. Esta carencia impide lograr un aprovechamiento óptimo de los procesadores que componen un sistema distribuido, generalmente heterogéneo.

Este trabajo presentó un método que permite descomponer un sistema eléctrico conforme a un desbalanceamiento deseado. Se introdujo además un parámetro de selección de particiones que mostró mejor comportamiento experimental que los presentados en otros trabajos.

De los resultados experimentales se puede concluir que:

- El método propuesto genera mejores particiones que una selección aleatoria.
- La mejor partición generada por el método propuesto es sin duda una buena partición, inclusive en varios casos la óptima, según sea el criterio de optimización.
- Las particiones generadas por el método propuesto son mejores ó iguales que las generadas por la Descomposición- $\epsilon$ , con la ventaja adicional de poder ejercer control sobre el número de subredes y el tamaño relativo de las mismas.
- El radio espectral de la matriz de comparación es un mejor parámetro de selección que el parámetro Par\_A,

como lo demuestran las correlaciones presentadas en la sección 4.2.

Otros trabajos [12] corroboran la utilidad del método propuesto en diversas aplicaciones.

En resumen, la presente propuesta nos brinda una solución superior a las hoy existentes para la descomposición de sistemas eléctricos con el objeto de resolverlos en un sistema distribuido heterogéneo.

**Agradecimientos especiales:** a los profesores Eugenius Kaszkurewicz y Djalma Mosqueira Falcão, de la Universidad Federal de Rio de Janeiro - Brasil.

## BIBLIOGRAFÍA

- [1] Barán B., Kaszkurewicz E. y Bhaya A., "Parallel Asynchronous Team Algorithms: Convergence and Performance Analysis". *IEEE Transactions on Parallel & Distributed Systems*. Julio de 1996.
- [2] Vale M.H., *Descomposição de Redes Eléctricas para Procesamiento Paralelo*. Tesis Doctoral COPPE/UFRJ. Rio de Janeiro, Brasil, 1995.
- [3] Carré B.A., "Solution of Load-Flow by Partitioning Systems into Trees", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-88, pp. 1931-1938, noviembre 1968.
- [4] Irving M.R. y Sterling M.J.H., "Optical Network Tearing Using Simulated Annealing", *IEEE Proceedings*, vol. 137, no. 1, pp. 69-72, enero 1990.
- [5] Gupta M. y Banerjee P., "Demonstration of Automatic Data Partitioning Techniques for Parallelizing Compilers on Multicomputers", *IEEE Transactions on Parallel and Distributed Systems*, vol. 3, no.2, pp. 179-193, marzo 1992.
- [6] Vale M.H., Falcão D.M. y Kaszkurewicz E., "Electrical Power Network Decomposition for Parallel Computations". *IEEE International Symposium on Circuits and Systems*, (ISCAS 92). San Diego, California, 1992.
- [7] Sezer M. y Šiljak D.D., "Nested epsilon decompositions of complex systems". *IFAC 9<sup>th</sup> World Congress*, Budapest, Hungría.
- [8] Ikeda M. y Šiljak D.D., "Overlapping decomposition, expansions and contractions of dynamic systems". *Large Scale System 1*, North-Holland Publishing Co., pp.29-38, 1980.
- [9] Barán B., *Estudio de Algoritmos Combinados Paralelos Asíncronos*. Tesis Doctoral COPPE/UFRJ. Río de Janeiro, Brasil. Octubre 1993.

- [10] Barán B., Kaszkurewicz E. y Falcão D.M., "Team Algorithm in Distributed Load Flow Computations", *IEE Proceeding on Generation, Transmission and Distribution*, vol. 142, no. 6, pp. 583-588, noviembre 1995.
- [11] Barán B., Benítez D., Ramos R., "Partición de Sistemas de Ecuaciones para su Resolución Distribuida", *XXII Conf .Latinoamericana de Informática CLEI 96*, Bogotá, Colombia. Junio 1996.
- [12] B. Barán, F. Cardozo, J. Atlasovich y C. Schaerer, "Solving the Point of Collapse Problem using a Heterogeneous Computer Network". *International Conference on Information Systems, Analysis and Synthesis - ISAS '96*. Orlando - Florida, Estados Unidos.