



**Universidad Michoacana de San Nicolás de Hidalgo  
Facultad de Ingeniería Eléctrica  
División de Estudios de Posgrado**

**Aplicación de Cómputo en Paralelo basado en Threads en el  
Cálculo de Métricas usadas en el Análisis Armónico**

**TESIS**

Que para obtener el grado de  
**Maestro en Ciencias de Ingeniería Eléctrica**

presenta

**Oriel Keba Manianga**

Director de Tesis  
**Dr. Antonio Ramos Paz**

Noviembre 2021



**APLICACIÓN DE CÓMPUTO EN PARALELO BASADO EN THREADS  
EN EL CÁLCULO DE MÉTRICAS USADAS EN EL ANÁLISIS  
ARMÓNICO**

Los Miembros del Jurado de Examen de Grado aprueban la Tesis de Maestría en Ciencias en Ingeniería Eléctrica de Oriel Keba Manianga.

Dr. Juan Anzures Marín  
*Presidente del Jurado*

Dr. Antonio Ramos Paz  
*Director de Tesis*

Dr. Roberto Tapia Sánchez  
*Vocal*

Dr. J. Aurelio Medina Ríos  
*Vocal*

Dr. Fernando Martínez Cárdenas  
*Revisor Externo (ITM)*

Dr. Roberto Tapia Sánchez  
*Jefe de la División de Estudios de Posgrado  
de la Facultad de Ingeniería Eléctrica. UMSNH  
(Por reconocimiento de firmas)*

# Dedicatoria

*A mis padres, mis hermanos, hermanas, novia, amigos*

## Agradecimientos

Agradezco a mi Dios, por su amabilidad y misericordia por mantenerme vivo y a un buen estado de salud para continuar haciendo mi investigación hasta el final sin él, no podía llegar allí.

Agradezco a todos los profesores de la División de Estudios de Postgrado de la Facultad de Ingeniería Eléctrica de la Universidad Michoacana de San Nicolás de Hidalgo por sus profesionalismos y amplios conocimientos que me permitió a profundizar mi conocimiento en el área de la ingeniería eléctrica.

A CONACYT por haberme considerado como becario, esto fue de gran ayuda para la realización de los estudios de maestría, por propiciar el desarrollo de posgrados de alta calidad y la formación de especialistas.

Mis profundos agradecimientos al Dr. Antonio Ramos Paz por su incondicional apoyo y guía en el desarrollo de este trabajo. Por su excelente trato y consideración.

Mis profundos agradecimientos al Dr. Roberto Tapia, Dr. Juan Anzures, Dr. Juan Flores y su esposa por sus apoyos y consejos por la finalidad de lograr mi estudio.

Agradezco a toda mi familia por su apoyo y especialmente a mi papá Zephyrin Keba Di-Bantoto, mamá Bienvenue, mamá Aita Mpindi, Thierry Keba, Trésor Manianga, Elliott Keba, Emeurode Lutumba, Naomie Keba, Jean Claude Manunga y Asheya Mbangu Manzombi por sus esfuerzos y apoyos.

Agradezco a mis compañeros de la clase Juan José Avilés, Isadora Bejarano, Dalia Botaka, Gabriel Guzmán, Diego Antonio López, Geovanni Madrigal, José Luis Montufar, Miguel Ángel Pimentel, Miguel Ángel Reynoso, José Andrés Saaravia y Sergio Jonathan Rubio por colaboración y consideración durante este camino.

Gracias a ustedes Charles Mokani Landu, Zoe Mayemba, Marcel Kitenge, Cedre Katambayi, Yan Mbenza, Modesto Sombo, Edo Sakenge, Léviex Mapuno, y Viviane de ser de verdaderos amigos en los momentos más difíciles una vez más gracias por sus amistades y ayudas.

## Resumen

La utilización cada vez mayor de dispositivos de control en las industrias, basados en el uso de electrónica de potencia, permiten realizar una gran cantidad de procesos industriales de manera eficiente, sin embargo, generan la distorsión de las formas de onda de corriente y voltaje. Este fenómeno llamado distorsión armónica está asociado a diversos efectos adversos en las empresas, tales como calentamiento excesivo en las maquinas eléctricas, activación de manera incorrecta de dispositivos de protección, aumento en la facturación, etc. La observación de este fenómeno, por el medio de las métricas usadas en el análisis armónico se hace indispensable en el estudio de la calidad de la energía. Sin embargo, es importante tomar en cuenta el tiempo de cómputo, es deseable que éste sea pequeño, con el propósito de dar información lo más rápido posible para mitigar el efecto de este problema. Para poder optimizar el tiempo se aplica el cómputo en paralelo. En este trabajo se aplica cómputo en paralelo, basado en Threads en el cálculo de tres métricas; la Transformada Rápida de Fourier (FFT), donde la propuesta de cálculo se hace de manera que cada elemento de proceso calcula un conjunto de armónicos, para la propuesta de cálculo de la Distorsión armónica total (THD) se realiza una estrategia de manera a que cada elemento de proceso, calcula un conjunto de variables de estado (VE) y al final para la propuesta de cálculo de la respuesta a la frecuencia de una red eléctrica (DPI), la estrategia se hace de manera que cada elemento de proceso calcula matriz con base a un segmento del vector de frecuencias.

**Palabras clave:** Transformada rápida de Fourier, Distorsión armónica total, Respuesta a la frecuencia de la impedancia, cómputo en paralelo, Threads.

## Abstract

The increasing use of control devices in industries, based on the use of power electronics, allow a large number of industrial processes to be carried out efficiently, however, they generate distortion of current and voltage waveforms. This phenomenon called harmonic distortion is associated with various adverse effects in companies, such as excessive heating in electrical machines, incorrect activation of protection devices, increased billing, etc. The observation of this phenomenon, by means of the metrics used in harmonic analysis, is essential in the study of power quality. However, it is important to take into account the computation time, it is desirable that it be small, in order to provide information as quickly as possible to mitigate the effect of this problem. In order to optimize time, parallel computing is applied. In this work, it is applied computed in parallel, based on Threads in the calculation of three metrics; the Fast Fourier Transform (FFT), where the calculation proposal is made in such a way that each process element calculates a set of harmonics, for the Total Harmonic Distortion (THD) calculation proposal a strategy is carried out so that Each process element calculates a set of state variables (VE) and at the end for the proposed calculation of the response to the frequency of an electrical network (DPI),the strategy is made in such a way that each process element computes matrix based on a segment of the frequency vector.

**Keywords:** Fast Fourier Transform, Total Harmonic Distortion, Driving Point Impedance, Parallel Processing, Threads

# Contenido

|   |      |
|---|------|
| Dedicatoria . . . . .                       | III  |
| Agradecimientos . . . . .                   | IV   |
| Resumen . . . . .                           | V    |
| Abstract . . . . .                          | VI   |
| Contenido . . . . .                         | VII  |
| Lista de figuras . . . . .                  | XI   |
| Lista de tablas . . . . .                   | XV   |
| Lista de símbolos . . . . .                 | XVII |
| Lista de abreviaturas . . . . .             | XIX  |
| <br>  |      |
| 1. Introducción . . . . .                   | 1    |
| 1.1. Planteamiento del Problema . . . . .   | 1    |
| 1.2. Antecedentes . . . . .                 | 2    |
| 1.3. Objetivos de la Tesis . . . . .        | 6    |
| 1.3.1. Objetivo General . . . . .           | 6    |
| 1.3.2. Objetivos particulares . . . . .     | 6    |
| 1.4. Justificación de la Tesis . . . . .    | 7    |
| 1.5. Metodología . . . . .                  | 8    |
| 1.6. Descripción de los Capítulos . . . . . | 8    |

---

|  |    |
|--|----|
| 2. Calidad de la energía en los sistemas eléctricos                              | 11 |
| 2.1. Introducción . . . . .  | 11 |
| 2.2. Fenómenos adversos de la calidad de la energía . . . . .                    | 12 |
| 2.2.1. Transitorios . . . . .  | 12 |
| 2.2.2. Variaciones rms de corta duración . . . . .                               | 13 |
| 2.2.3. Variaciones rms de larga duración . . . . .                               | 13 |
| 2.2.4. Distorsión de forma de onda . . . . .                                     | 14 |
| 2.2.5. Fluctuaciones de voltaje . . . . .  | 15 |
| 2.2.6. Variaciones de frecuencia de potencia . . . . .                           | 15 |
| 2.3. Causas y efectos de los armónicos en los sistemas de potencia . . . . .     | 15 |
| 2.3.1. Causas . . . . .  | 16 |
| 2.3.2. Efectos de armónicos . . . . .  | 17 |
| 2.4. Recomendaciones de los límites de armónicos en los sistemas de potencia . . | 21 |
| 2.5. Serie de Fourier . . . . .  | 22 |
| 2.6. Cantidades para la cuantificación de los armónicos . . . . .                | 23 |
| 2.6.1. Transformada rápida de Fourier . . . . .                                  | 23 |
| 2.6.2. Distorsión armónica total . . . . .                                       | 24 |
| 2.7. Conclusiones . . . . .  | 25 |
| 3. Cómputo en paralelo basado en threads   | 27 |
| 3.1. Introducción . . . . .  | 27 |
| 3.2. Definiciones . . . . .  | 28 |
| 3.2.1. Cómputo en paralelo . . . . .   | 28 |
| 3.2.2. Computadora paralela . . . . .  | 28 |
| 3.2.3. Programación en paralelo . . . . .  | 30 |
| 3.3. Plataforma Threads . . . . .  | 30 |
| 3.3.1. Concepto de Thread . . . . .  | 30 |

---

|        |  |    |
|--------|--|----|
| 3.3.2. | Comandos principales de la plataforma Thread en Lenguaje C . . . .   | 32 |
| 3.4.   | Diseño de algoritmos en paralelo por la Metodología de Ian Foster . . . . .                                  | 34 |
| 3.4.1. | Particionamiento . . . . .   | 35 |
| 3.4.2. | Comunicación . . . . .   | 35 |
| 3.4.3. | Aglomeración . . . . .   | 36 |
| 3.4.4. | Mapeo . . . . .  | 36 |
| 3.5.   | Métricas para el procesamiento en paralelo . . . . .   | 36 |
| 3.5.1. | <i>Speed-up</i> . . . . .  | 36 |
| 3.5.2. | Eficiencia . . . . .   | 37 |
| 3.5.3. | Ley de Amdahl . . . . .  | 37 |
| 3.6.   | Conclusiones . . . . .   | 37 |
| 4.     | Propuesta de paralelización de algoritmos usados para el cálculo de métricas del análisis armónico . . . . . | 39 |
| 4.1.   | Introducción . . . . .   | 39 |
| 4.2.   | Propuesta de paralelización del cálculo de la FFT . . . . .  | 39 |
| 4.3.   | Propuesta de paralelización del cálculo del THD . . . . .  | 45 |
| 4.4.   | Propuesta de paralelización del cálculo de la DPI de una red eléctrica . . . .                               | 50 |
| 4.5.   | Conclusiones . . . . .   | 56 |
| 5.     | Casos de Estudio . . . . .   | 59 |
| 5.1.   | Introducción . . . . .   | 59 |
| 5.2.   | Características de la computadora utilizada para el cómputo en paralelo . . .                                | 59 |
| 5.3.   | Caso de Estudio No. 1: Cálculo en paralelo de la FFT . . . . .   | 61 |
| 5.3.1. | Medición de 40 armónicos . . . . .   | 62 |
| 5.3.2. | Medición de 80 armónicos . . . . .   | 65 |
| 5.4.   | Caso de Estudio No. 2: Cálculo en paralelo de la THD . . . . .   | 69 |
| 5.5.   | Caso de Estudio No. 3: Cálculo en paralelo de la DPI . . . . .   | 76 |

---

|  |    |
|--|----|
| 5.5.1. Caso de Estudio 3.1. Red de 3 nodos con 2 líneas de transmisión . . . | 76 |
| 5.5.2. Caso de Estudio 3.2. Red de 5 nodos con 7 líneas de transmisión . . . | 84 |
| 5.6. Conclusiones . . . . .  | 93 |
| 6. Conclusiones y recomendaciones para trabajos futuros                      | 95 |
| 6.1. Conclusiones Generales . . . . .  | 95 |
| 6.2. Trabajos Futuros . . . . .  | 96 |
| Referencias  | 99 |

# Lista de figuras

|  |    |
|--|----|
| 3.1. Red de cómputo en paralelo . . . . .  | 29 |
| 3.2. Esquema de un sistema multiprocesador . . . . .                               | 29 |
| 3.3. Programación en paralelo . . . . .  | 30 |
| 3.4. Concepto de simple thread y varias threads . . . . .                          | 31 |
| 3.5. Esquema de threads . . . . .  | 32 |
| 3.6. Metodología de diseño de algoritmos paralelos de Ian Foster . . . . .         | 35 |
| 4.1. Distribución de los armónicos en los procesadores . . . . .                   | 40 |
| 4.2. Distribución de los armónicos en 8 procesadores . . . . .                     | 41 |
| 4.3. Diagrama de flujo de la FFT . . . . .   | 42 |
| 4.4. Código de cálculo de un rango de armónico . . . . .                           | 43 |
| 4.5. Código de repartición de los armónicos por cada elemento de proceso . . . . . | 44 |
| 4.6. Sección de actividades de procesadores . . . . .                              | 44 |
| 4.7. Código de verificación de la finalización de las tareas . . . . .             | 44 |
| 4.8. Distribución de los armónicos en los 8 procesadores . . . . .                 | 46 |
| 4.9. Diagrama de flujo del THD . . . . .   | 47 |
| 4.10. Código de cálculo de la THD a un rango de armónico . . . . .                 | 48 |
| 4.11. Código de cálculo de un rango de armónico . . . . .                          | 49 |
| 4.12. Sección de actividades de procesadores . . . . .                             | 49 |

|  |    |
|--|----|
| 4.13. Código de verificación de la finalización de las tareas . . . . .                    | 50 |
| 4.14. Distribución de las frecuencia en $P$ procesadores . . . . .                         | 52 |
| 4.15. Diagrama de flujo del DPI . . . . .  | 53 |
| 4.16. Código de cálculo de la DPI en rango de frecuencia . . . . .                         | 54 |
| 4.17. Código de repartición de rango de frecuencia a cada procesadores . . . . .           | 55 |
| 4.18. Sección de actividades de procesadores . . . . .                                     | 55 |
| 4.19. Código de verificación de la finalización de las tareas . . . . .                    | 56 |
| 5.1. Presentación de la computadora . . . . .  | 60 |
| 5.2. Presentación interna de la Computadora . . . . .                                      | 60 |
| 5.3. Voltaje en el nodo 10 . . . . .   | 62 |
| 5.4. Gráfica de Tiempo de la Paralelización de la FFT con 40 armónicos . . . . .           | 65 |
| 5.5. Gráfica del <i>Speed-up</i> de la Paralelización de la FFT con 40 armónicos . . . . . | 65 |
| 5.6. Gráfica de Tiempo de la Paralelización de la FFT con 80 armónicos . . . . .           | 68 |
| 5.7. Gráfica del <i>Speed-up</i> de la Paralelización de la FFT con 80 armónicos . . . . . | 68 |
| 5.8. Gráfica de Tiempo de la Paralelización del THD . . . . .                              | 71 |
| 5.9. Gráfica del <i>speed-up</i> de la Paralelización del THD . . . . .                    | 72 |
| 5.10. Voltaje en el nodo 11 . . . . .  | 73 |
| 5.11. Corriente del nodo 6 al nodo 12 . . . . .  | 73 |
| 5.12. Corriente entregada por el generador conectado al nodo 1 . . . . .                   | 74 |
| 5.13. Esquema del sistema de 3 nodos . . . . .   | 76 |
| 5.14. Gráfica de Tiempo de la Paralelización del DPI con 40 armónicos . . . . .            | 79 |
| 5.15. Gráfica del <i>speed-up</i> de la Paralelización del DPI con 40 armónicos . . . . .  | 80 |
| 5.16. Gráfica de Tiempo de la Paralelización del DPI con 80 armónicos . . . . .            | 82 |
| 5.17. Gráfica del <i>speed-up</i> de la Paralelización del DPI con 80 armónicos . . . . .  | 83 |
| 5.18. Esquema del sistema de 5 nodos . . . . .   | 84 |
| 5.19. Gráfica de Tiempo de la Paralelización de la DPI con 40 armónicos . . . . .          | 87 |

---

|   |    |
|---|----|
| 5.20. Gráfica del <i>speed-up</i> de la Paralelización del DPI con 40 armónicos . . . . . | 88 |
| 5.21. Gráfica de Tiempo de la Paralelización del DPI con 80 armónicos . . . . .           | 90 |
| 5.22. Gráfica del <i>speed-up</i> de la Paralelización del DPI con 80 armónicos . . . . . | 91 |
| 5.23. Respuesta de la impedancia a la frecuencia visto del nodo 3 . . . . .               | 92 |
| 5.24. Respuesta de la impedancia a la frecuencia visto del nodo 4 . . . . .               | 92 |
| 5.25. Respuesta de la impedancia a la frecuencia visto del nodo 5 . . . . .               | 93 |



# Lista de tablas

|  |    |
|--|----|
| 2.1. Indicaciones sobre el espectro armónico inyectado por diferentes cargas . . .                               | 16 |
| 2.2. Resumen de causas y efectos de los armónicos . . . . .  | 20 |
| 2.3. Límites de distorsión de voltaje . . . . .  | 21 |
| 2.4. Límites de distorsión de corriente para sistemas clasificados de 120 V a 69 kV                              | 21 |
| 2.5. Límites de distorsión de corriente para sistemas clasificados por encima de<br>69 kV hasta 161 kV . . . . . | 22 |
| 2.6. Límites de distorsión de corriente para sistemas clasificados > 161 kV . . .                                | 22 |
|  |    |
| 4.1. Algoritmo de cálculo en paralelo de la FFT . . . . .  | 42 |
| 4.2. Algoritmo de cálculo en paralelo de la THD . . . . .  | 47 |
| 4.3. Algoritmo de cálculo en paralelo de la DPI . . . . .  | 52 |
|  |    |
| 5.1. Resultados de Mediciones de Tiempo [s] de 40 armónicos . . . . .  | 63 |
| 5.2. Resultados de Cálculo del <i>speed-up</i> de 40 armónicos . . . . .   | 64 |
| 5.3. Resultados de Mediciones de Tiempo [s] de 80 armónicos . . . . .  | 66 |
| 5.4. Resultados de Cálculo del <i>speed-up</i> de 80 armónicos . . . . .   | 67 |
| 5.5. Resultados de Mediciones de Tiempo [s] cálculo del THD . . . . .  | 69 |
| 5.6. Resultados de Cálculo del <i>speed-up</i> del THD . . . . .   | 70 |
| 5.7. Variables de estado del sistema de 14 nodos con sus THD . . . . .   | 75 |
| 5.8. Resultados de Mediciones de Tiempo [s] de 40 armónicos . . . . .  | 77 |

|  |    |
|--|----|
| 5.9. Resultados del cálculo del <i>speed-up</i> de 40 armónicos . . . . .  | 78 |
| 5.10. Resultados de Mediciones de Tiempo [s] de 80 armónicos . . . . .     | 81 |
| 5.11. Resultados del cálculo del <i>speed-up</i> de 80 armónicos . . . . . | 81 |
| 5.12. Resultados de Mediciones de Tiempo [s] de 40 armónicos . . . . .     | 85 |
| 5.13. Resultados del cálculo del <i>speed-up</i> de 40 armónicos . . . . . | 86 |
| 5.14. Resultados de Mediciones de Tiempo [s] de 80 armónicos . . . . .     | 89 |
| 5.15. Resultados del cálculo del <i>speed-up</i> de 80 armónicos . . . . . | 89 |

# Lista de símbolos

|          |                                      |
|----------|--------------------------------------|
| $I$      | Intensidad de la corriente eléctrica |
| $V$      | Voltaje                              |
| $R$      | Resistencia                          |
| $L$      | Inductancia                          |
| $C$      | Capacitancia                         |
| $Z$      | Impedancia                           |
| $Y$      | Admitancia                           |
| $t$      | Tiempo                               |
| $T$      | Periodo                              |
| $\omega$ | Velocidad angular                    |
| $A$      | Amperios                             |
| $V$      | Voltios                              |
| $s$      | Segundo                              |
| $\Omega$ | Ohmios                               |
| $H$      | Henrios                              |
| $F$      | Faradios                             |
| $W$      | Watt                                 |
| VAR      | Volt Ampere reactivo                 |
| RMS      | Valor eficaz                         |
| $f$      | Frecuencia                           |
| $h$      | Orden de armónico                    |
| $P$      | Número de procesadores               |



# Lista de abreviaturas

**CA** Corriente Alterna.

**CD** Corriente Directa.

**CPU** Unidad central de proceso.

**DPI** Respuesta a la frecuencia de la impedancia.

**FFT** Transformada Rápida de Fourier.

**IEEE** Instituto de Ingenieros Eléctricos y Electrónicos.

**PCC** Punto de acoplamiento común.

**SEP** Sistemas Eléctricos de Potencia.

**TDD** Distorsión total de demanda.

**THD** Distorsión armónica total.

**VE** Variable de estado.



# Capítulo 1

## Introducción

### 1.1. Planteamiento del Problema

Los tiristores y los interruptores semiconductores se utilizan en los sistemas eléctricos para controlar las cargas eléctricas y así manejar la velocidad de motores, manejar la energía que va a los hornos eléctricos, la conversión de la Corriente Alterna (CA) en Corriente Directa (CD) o CD en CA [Singh et al., 1999]. El control de la energía en las redes eléctricas se realiza en los enlaces con el uso de los dispositivos de Sistemas de Transmisión Flexibles de Corriente Alterna (FACTS por sus siglas en Inglés). Sin embargo, este control causa problemas en la red eléctrica con respecto a la calidad de energía [Long et al., 2013], [Suslov et al., 2018]. Los problemas de la calidad de energía se conocen a través de los fenómenos tales como sags de voltaje, swells de voltaje, flickers, transitorios, armónicos, [IEEE Std 1159, 2019], [C.Sankaran, 2002], [Cobben et al., 2007]. En los sistemas trifásicos pueden provocar un desbalance de fase [Wagner et al., 1993].

Con el desarrollo de la tecnología, los convertidores de frecuencia, los convertidores de corriente y voltaje, los sistemas de alimentación ininterrumpida (UPS), las fuentes de alimentación conmutadas, las lámparas LED y fluorescentes, etc., son cada vez más utilizados. Estos dispositivos constituyen cargas no lineales que modifican la forma de onda de voltaje y

corriente. Los armónicos provocan sobrecalentamiento en los devanados de transformadores y máquinas rotatorias, sobrecarga de conductores neutros, falla en los bancos de capacitores, costos operativos más altos, envejecimiento prematuro de los equipos, deteriora la calidad del servicio. Además, debido al cambio en la estructura de la producción de energía con el uso de las energías renovables, los sistemas eólicos y fotovoltaicos generan armónicos debido al uso de dispositivos electrónicos para los ajustes de la velocidad de las turbinas eólicas y la regulación de voltaje en los sistemas fotovoltaicos [Motta and Faúndes, 2016].

Existen varios fenómenos adversos de calidad de la energía. El enfoque de esta tesis se base en los armónicos que, son realmente un problema para los equipos eléctricos como líneas de transmisión, líneas de distribución, motores, bancos de capacitores, transformadores, líneas telefónicas, conductores neutro y conductores de fases [Kennedy, 2000], [Atanasoae and Pentiu, 2014]. Debido a los efectos negativos causados por los armónicos en los Sistemas Eléctricos de Potencia (SEP) se hace necesario el análisis armónico. El análisis armónico de una señal se realiza a través de la Transformada Rápida de Fourier (FFT), lo cual permite conocer el contenido armónico de una forma de onda de corriente o de voltaje. Por medio de la Distorsión armónica total (THD) se mide el grado de contaminación de la señal. La conexión de cargas no-lineales extraen corrientes armónicas que se inyectan en los SEP, estas corrientes tienen ciertas frecuencias que pueden corresponder a uno o más frecuencia natural del sistema provocado una resonancia en el sistema. [Eghtedarpour et al., 2014]. Por esta razón se hace un estudio por el medio de la Respuesta a la frecuencia de la impedancia (DPI) para poder identificar las diferentes frecuencias a riesgo de resonancia.

## 1.2. Antecedentes

Actualmente con el crecimiento de las aplicaciones de la electrónica de potencia en la generación de energía eléctrica, su distribución y consumo, la electrónica de potencia está cada vez más presente en estos sectores [Kumar and Zare, 2014]. La preocupación a tener en cuenta son las implicaciones de su operación con la calidad de la energía.

---

De acuerdo con Barry W. Kennedy en [Kennedy, 2000], la calidad de la energía se puede definir desde dos perspectivas diferentes, dependiendo de si se suministra o se consume electricidad. Por lo tanto, a continuación, daremos definiciones de la calidad de la energía de acuerdo con Gerald Thomas Heydt, Roger Dugan, Mark McGranaghan y Wayne Beast, normas, estándares y compañías.

Gerald Thomas Heydt en [Kennedy, 2000] define la calidad de la energía como: La medida, el Análisis y la mejora del voltaje del bus, generalmente un voltaje y frecuencia nominales.

Roger Dugan, Mark McGranaghan y Wayne Beat definen un problema de la calidad de la energía como: cualquier problema de energía manifestada en desviaciones de voltaje, corriente o frecuencia que resulte en fallas o fallas en la operación de la utilidad o del equipo del usuario final [Dugan et al., 2002].

El Instituto de Investigación de Energía Eléctrica (EPRI) de los Estados Unidos, por ejemplo, define la calidad de la Energía Eléctrica (Power Quality) como cualquier problema de energía suministrada manifestada en la desviación de la tensión, de la corriente o de la frecuencia, de sus valores ideales que ocasione falla o mala operación del equipo de un usuario [Sundaram et al., 1994].

La norma de la Comisión Electrotécnica Internacional (IEC) (61000-2-2/4) y la norma de la Comisión Europea de Normalización Electrotécnica (CENELEC) (50160) definen la Calidad de la Energía Eléctrica como: una característica física del suministro de electricidad, la cual debe llegar al cliente en condiciones normales, sin producir perturbaciones ni interrupciones en los procesos del mismo [Std EN 50160, 1995].

En la norma de la IEEE 1159 de 2019 se define a la calidad de la energía como: La amplia variedad de fenómenos electromagnéticos que caracterizan la tensión y la corriente eléctricas, en un tiempo dado y en una ubicación dada en el sistema de potencia [IEEE Std 1159, 2019].

La calidad de la energía designa más concretamente la calidad del suministro de

electricidad. Esto depende de tres factores: continuidad del suministro, calidad de la onda de voltaje y calidad del servicio [Ise et al., 2000].

Después de definir la calidad de la energía, también es esencial conocer la importancia de la calidad de la energía eléctrica debido a que es un factor muy importante para el desarrollo de un país, un estado y ciudad debido a que vivimos en un mundo eminentemente eléctrico. Recordando que nos encontramos en la era de la cibernética y muchos equipos tienen en su composición los elementos electrónicos que necesitan una energía suministrada de buena calidad. La calidad de la energía es la responsabilidad de todo así que la generación, la distribución y el consumidor deben cumplir con las normas.

Del lado de la generación; La energía eléctrica no es una energía primaria que se puede usar directamente de hecho es necesario generar la. La generación tiene unas ciertas especificaciones propias en términos de nivel del voltaje, forma de onda, frecuencia etc. Un cambio de uno o más de sus especificaciones se considera como un problema de la calidad de la energía. Por esta razón, es importante que los equipos que se conectan a la fuente de generación no deben cambiar una de sus especificaciones y también, La generación, siempre debe cumplir con sus especificaciones de manera a no dañar los equipos que se conectan.

Por el lado de la distribución; La generación de energía eléctrica es función de la demanda de energía. Debido a que los sistemas de generación no se ubica cerca los clientes, por lo que es necesario realizar la transmisión desde el punto de generación al punto de consumo a través de redes de distribución que desempeñan el papel de interfaz entre la generación y la demanda (usuarios finales). El punto de conexión de los equipos de usuarios finales debe ser, como tal que no causa ningún problema a los equipos eléctricos.

Del lado de los usuarios finales (consumidores); los equipos de consumo de los clientes se conectan a la red sin avisar al servicio de suministración de la energía. La conexión a la red debe ser de tal manera que no provoque fenómenos anormales.

Los principales fenómenos adversos de la calidad energía pueden clasificarse según los siguientes criterios [Dugan et al., 2002]:

- Fenómenos conducidos a baja frecuencia, que se divide de esta manera; armónicos e inter-armónicos, sistemas de señal (portador de línea de alimentación), fluctuaciones de voltaje (parpadeo), caída e interrupción de voltaje, desequilibrio de voltaje (desequilibrio), variaciones de frecuencia de potencia, voltajes inducidos a baja frecuencia, CD en redes de CA
- Fenómenos a baja frecuencia, dependiendo de la naturaleza del campo considerado, a saber, campos magnéticos o campos eléctricos.
- Los fenómenos de alta frecuencia provocan voltajes o corrientes inducidas por onda continua, transitorios impulsivos y transitorios oscilatorios.
- Fenómenos radiantes de alta frecuencia resultantes en campos magnéticos, campos eléctricos, campos electromagnéticos, Ondas continuas, transitorias
- Fenómenos de descarga electrostática (ESD) que se pueden traducir mediante un pulso electromagnético nuclear (NEMP).

En este trabajo de investigación nos centraremos en un fenómeno particular de la calidad de la energía eléctrica, la distorsión de forma de onda o armónicos, hablaremos sobre sus efectos en los equipos eléctricos, las causas de su producción, normas y límites de la distorsión armónica.

Los equipos que son susceptibles a producir los armónicos son: Los convertidores, los equipos electrónicos, las lámparas a descarga, los bancos de capacitores, los FACTS, los enlaces de corriente directa en alto voltaje (HVDC por sus siglas en inglés), Todas las cargas no lineales [Wagner et al., 1993].

Los problemas que surgen del flujo de corriente armónica son directos y simples. Los dos problemas principales que se encuentran, son el calentamiento anormal de los componentes y la interrupción de la operación [Smith, 2002]. En los conductores, el calentamiento anormal es el principal efecto perjudicial de los armónicos. Además del calentamiento por el efecto piel, los transformadores también sufren pérdidas por corrientes parásitas. Las corrientes de Foucault son creadas por campos magnéticos de CA y circulan dentro del núcleo

del transformador para oponerse al flujo magnético del núcleo. i Bancos de capacitores: una de las razones por las que los armónicos son un problema para los capacitores es porque la impedancia capacitiva disminuye a medida que aumenta la frecuencia. Por lo tanto, a medida las corrientes aumentan en orden armónico, los condensadores actuarán cada vez más como un corto circuito. Esto básicamente significa que los condensadores atraen los armónicos de alta frecuencia. El resultado es que los condensadores se sobrecargarán debido al aumento del flujo de corriente [Smith, 2002].

Equipo electrónico: El equipo que emplea electrónica de potencia puede ser muy sensible a la distorsión armónica debido a la dependencia de la forma de onda de voltaje para la información de sincronización. Los componentes de electrónica de potencia utilizan el cruce por cero de la señal de voltaje para determinar los tiempos de disparo. Si se desplaza el cruce por cero de la forma de onda, puede ocurrir una operación incorrecta. Además, muchos interruptores semiconductores funcionan a un voltaje nulo para reducir la interferencia electromagnética y la corriente de entrada [Smith, 2002].

## **1.3. Objetivos de la Tesis**

### **1.3.1. Objetivo General**

Aplicar el cómputo en paralelo basado en Threads o hilos en el cálculo de métricas usadas en el análisis armónico.

### **1.3.2. Objetivos particulares**

- Aplicar el cómputo en paralelo basado en threads para el cálculo de la transformada rápida de Fourier.
- Aplicar el cómputo en paralelo basado en threads para el cálculo de distorsión armónica total para redes eléctrica.

- Aplicar el cómputo en paralelo basado en threads para el cálculo de la respuesta de la impedancia a la frecuencia para redes eléctrica.

#### 1.4. Justificación de la Tesis

La inclusión de nuevas tecnologías en los sistemas eléctricos de potencia, tales como los dispositivos FACTS, dispositivos DFACTS, fuentes no convencionales de generación, cargas controladas por electrónica de potencia, generan grandes retos en su diseño y operación. Desde el punto de vista del diseño, se tienen que realizar una gran cantidad de simulaciones computacionales, antes de poder insertar en forma segura algún tipo de dispositivo en la red eléctrica. Desde el punto de vista de operación, el conocimiento de una forma rápida de alguna variable puede permitir su control, y en consecuencia una operación segura, confiable y económica del sistema.

Muchas ocasiones, estas simulaciones requieren un gran esfuerzo computacional, el cual en gran medida, está asociado a un modelado detallado de los diversos elementos que forman la red eléctrica, así como a los diversos escenarios operativos en los que puede operar el sistema eléctrico. La interpretación de los resultados, obtenidos a través de los procesos de simulación, genera conocimiento sobre el sistema, el cual se usa para mejorar el funcionamiento del mismo o para evitar fallas en el mismo.

Realizar un análisis exhaustivo de los diversos escenarios operativos de la red eléctrica, puede convertirse en una tarea que requiere un gran tiempo de cómputo, aun con el uso de computadoras modernas. Esta desventaja puede subsanarse con el uso del cómputo en paralelo basado en hilos. Con el uso de los hilos, se pueden optimizar las diversas rutinas asociadas a la determinación en el estado estacionario periódico de las diversas variables del sistema eléctrico. Entre las herramientas matemáticas utilizadas, para conocer el grado de contaminación armónica asociada a variables como voltajes nodales y corrientes de línea, se encuentran: la THD, la FFT y la DPI. Los algoritmos de estas herramientas matemáticas, al optimizarse con el uso de cómputo en paralelo basado en hilos, se convertirán en elementos

indispensables en la simulación, con propósitos de análisis y mitigación armónica, de los sistemas eléctricos de potencia.

## 1.5. Metodología

Esta tesis se basa en la metodología que se describe con las etapas siguientes: En primera etapa se hace la revisión documental sobre; la calidad de la energía, del cómputo en paralelo basada en threads. En la Segunda etapa se realiza la propuesta de paralelización de los algoritmos usados para el cálculo de; La Transformada Rápida de Fourier (FFT), el Distorsión armónica total (THD) y la Respuesta a la frecuencia de la impedancia (DPI). En la tercera etapa se realiza los casos de estudios que consiste al cálculo de las métricas que se utilizan por el análisis armónicos aplicado el cómputo en paralelo. Se graficará las curvas de tiempo y *speed-up* de cada métrica. En la cuarta etapa se realiza la escritura de la Tesis.

## 1.6. Descripción de los Capítulos

Este trabajo se estructura en torno a los siguientes capítulos:

En el Capítulo No. 2 se describen los conceptos básicos sobre la calidad de la energía, los armónicos en la red eléctrica, normas y estándares.

En el Capítulo No. 3 se hace una descripción del cómputo en paralelo, del diseño de algoritmos en paralelo, la métrica para el procesamiento en paralelo y la plataforma de cómputo en paralelo de los threads.

En el Capítulo No. 4 se describe el cálculo en paralelo del THD, FFT y DPI en sistemas eléctricos.

En el Capítulo No. 5 se presenta 3 casos de estudios en donde se muestra la ventaja del uso del cómputo en paralelo basado en threads, en el cálculo de la FFT, el THD y la DPI en sistemas eléctricos, con respecto al cómputo secuencial.

En el Capítulo No. 6 se presentarán las conclusiones de esta investigación y se proponen algunos trabajos futuros que pueden realizarse como continuación de este trabajo de investigación.



## Capítulo 2

# Calidad de la energía en los sistemas eléctricos

### 2.1. Introducción

Hace ya varias décadas que los equipos electrónicos están muy extendidos en todos los ámbitos, ya sea en empresas con equipamiento informático, en nuestros hogares con equipamiento doméstico, en redes eléctricas de transmisión, en el control de procesos de generación. En la red de distribución de las compañías suministradoras se conectan los aparatos de los usuarios finales, estos se conocen como cargas que pueden ser lineales o no lineales dependiendo de la forma en que la carga extrae la corriente.

En [IEEE Std 519, 1993], [Mora-Barajas and Banuelos-Sanchez, 2010] establecen que las cargas no lineales son aquellas, que cuando son conectadas a una fuente de voltaje sinusoidal, extraen corriente no sinusoidal que causan problemas en la red de distribución en el punto de acoplamiento común. Este problema de no linealidad está asociado al concepto de calidad de energía. El tema de la calidad de la energía es un concepto que no se puede ignorar en esta era debido a que puede provocar muchas pérdidas de dinero dependiendo del sector. Este tema debe ser tomando en serio por las empresas suministradoras, así como

también por los usuarios finales. En este capítulo se presentan los fenómenos adversos de la calidad de la energía, enfocándonos en las causas y efectos de los armónicos en los sistemas de potencia, recomendaciones de acuerdo con [IEEE Std 519, 2014], la Transformada Rápida de Fourier (FFT) y Distorsión armónica total (THD).

## 2.2. Fenómenos adversos de la calidad de la energía

De acuerdo con [IEEE Std 1159, 2019], los fenómenos adversos de la calidad de la energía se clasifica en siete tipos:

- Transitorios (transitorios impulsivos y transitorios oscilatorios),
- Variaciones rms de corta duración (interrupciones instantáneas, momentáneas y temporales, *sags* y *swells*),
- Variaciones rms de larga duración (Sobretensión, Subtensión y interrupciones sostenidas),
- Desequilibrio de voltaje, distorsión de forma de onda (Desplazamiento de CD, Armónicos, Interarmónicos, Muestras, Sonido de muesca de voltaje y Ruido),
- Fluctuaciones de voltaje,
- Variaciones de frecuencia de potencia.

### 2.2.1. Transitorios

Los transitorios se dividen en dos tipos: Transitorio impulsivo y transitorio oscilatorio:

- Transitorio impulsivo: un transitorio impulsivo se conoce como un cambio de frecuencia repentino e imponente de la condición nominal de voltaje, corriente o ambos, que es unidireccional en polaridad. Los transitorios impulsivos se caracterizan típicamente por sus tiempos de pico, subida y bajada o duración [IEEE Std 1159, 2019].

- Transitorio oscilatorio: un transitorio oscilatorio consiste en un voltaje o corriente cuyo valor instantáneo cambia la polaridad rápidamente varias veces y normalmente decae dentro de un ciclo de frecuencia fundamental [IEEE Std 1159, 2019].

### 2.2.2. Variaciones rms de corta duración

Las variaciones rms de corta duración se dividen en tres tipos: Interrupciones instantáneas, momentáneas y temporales; los *Sags* y los *Swells*.

- Interrupciones instantáneas, momentáneas y temporales: Estos fenómenos ocurren cuando el voltaje de suministro o la corriente de carga disminuye a menos de 0.1 *p.u.* durante un período de tiempo que no excede 1 minuto [IEEE Std 1159, 2019].
- *Sags*: los *sags* ocurren cuando el voltaje rms disminuye entre 0.1 *p.u.* y 0.9 *p.u.* durante períodos de 0.5 ciclos a 1 minuto. Esta disminución del voltaje están entre 0.1 *p.u.* y 0.9 *p.u.* [IEEE Std 1159, 2019].
- *Swells*: los *swells* ocurren cuando el voltaje rms aumenta por encima de 1.1 *p.u.* durante periodos de 0.5 ciclos a 1 minuto. Las magnitudes típicas se encuentran entre 1.1 *p.u.* y 1.2 *p.u.* [IEEE Std 1159, 2019].

### 2.2.3. Variaciones rms de larga duración

Las variaciones rms de larga duración se dividen en dos tipos: Sobrevoltaje, subvoltaje e interrupciones sostenidas.

- Sobrevoltaje: un sobrevoltaje ocurre cuando el voltaje rms aumenta rms a un valor superior a 1.1 *p.u.* durante un período superior a 1 minuto. Los aumentos del voltaje son un intervalo de 1.1 *p.u.* a 1.2 *p.u.* [IEEE Std 1159, 2019].
- Subvoltaje: un subvoltaje ocurre cuando el voltaje rms inferior a 0.9 *p.u.* durante un período superior a 1 minuto. En general estas disminuciones de voltaje están entre 0.8

*p.u.* y *0.9 p.u.* [IEEE Std 1159, 2019].

- Interrupciones sostenidas: Las interrupciones sostenidas son fenómenos que ocurren cuando el voltaje de alimentación baja a menos de 10 % del valor nominal durante un período de tiempo superior a 1 minuto [IEEE Std 1159, 2019].
- Desequilibrio de voltaje (o corriente): el desequilibrio de voltaje (o corriente) de un sistema trifásico se define como la relación entre la amplitud del componente de secuencia negativa y la amplitud del componente de secuencia positiva, expresada como porcentaje. Normalmente, el desequilibrio de voltaje del servicio trifásico es inferior al 5 % [IEEE Std 1159, 2019].

#### 2.2.4. Distorsión de forma de onda

Distorsión de forma de onda se dividen en seis tipos: *DC offset*, armónicos, interarmónicos, muescas, sonido de muescas de voltaje y ruido eléctrico.

- *DC Offset*: el desplazamiento de CD (*DC Offset*) es el fenómeno que ocurre cuando un voltaje o corriente CD se encuentra en un sistema de CA. Este fenómeno es causado para la presencia de rectificador de media onda [IEEE Std 1159, 2019].
- Armónicos: Los armónicos son frecuencias múltiples entero de la fundamental. Este fenómeno es causado por la presencia de cargas no lineales en los sistemas eléctricos de potencia [IEEE Std 519, 1993], [IEEE Std 1159, 2019], [Smith, 2002].
- Interarmónicos: Los interarmónicos se conocen como componentes de frecuencia múltiple no entera de la fundamental de los voltajes o corrientes del sistema de potencia. Pueden aparecer como frecuencias discretas o como un espectro de banda ancha [IEEE Std 519, 1993], [IEEE Std 1159, 2019].
- Muecas: El muesca es el fenómeno producido por una perturbación periódica de la tensión provocada por los dispositivos electrónicos de potencia debido a su funcionamiento normal, cuando la corriente se conmuta de una fase a otra [IEEE Std 1159, 2019].

- **Sonido de muescas de voltaje:** El sonido de muesca de voltaje es el fenómeno producido por una variación de la muesca de voltaje, cuando una resonancia del sistema da como resultado una respuesta de sonido en cada una de las muescas de conmutación [IEEE Std 1159, 2019].
- **Ruido eléctrico:** El ruido es el fenómeno que ocurre en un sistema eléctrico cuando la onda distorsionada contiene señales eléctricas no deseadas con contenido espectral de banda ancha generalmente menor a 200 kHz que se superponen al voltaje o corriente fundamental en los conductores de fase o señales eléctricas no deseadas que se encuentran en conductores neutros o líneas de señal [IEEE Std 1159, 2019].

### **2.2.5. Fluctuaciones de voltaje**

Las fluctuaciones de voltaje son fenómenos que producen variaciones sistemáticas en la envolvente de voltaje o varias variaciones de voltaje aleatorias, cuya amplitud se encuentra entre un intervalo de 0.95 *p.u.* a 1.05 *p.u.* [IEEE Std 1159, 2019].

### **2.2.6. Variaciones de frecuencia de potencia**

Las variaciones de frecuencia de energía son fenómenos que ocurren cuando la frecuencia fundamental del sistema de energía desvía de su valor nominal especificado debido al cambio del equilibrio entre carga y potencia de generación [IEEE Std 1159, 2019].

## **2.3. Causas y efectos de los armónicos en los sistemas de potencia**

Los armónicos en el sistema de potencia no son un fenómeno nuevo. La preocupación por la distorsión armónica se ha introducido a lo largo de la historia de los sistemas de energía eléctrica. Steinmetz publicó un libro en 1916 que dedicó considerable atención al estudio de los armónicos en los sistemas eléctricos trifásicos. Su principal preocupación eran

las corrientes de tercer armónico provocadas por el hierro saturado en transformadores y máquinas, y fue el primero en proponer conexiones delta para bloquear las corrientes de tercer armónico. Más tarde, con la innovación de la electrificación rural y el servicio telefónico, los circuitos eléctricos y telefónicos se colocaron a menudo en derechos de paso comunes; en consecuencia, las corrientes armónicas producidas por las corrientes de magnetización del transformador causaron interferencia inductiva con los sistemas telefónicos de cable abierto [Abdelaziz et al., 2012]. A continuación vemos causas y efectos de los armónicos

### 2.3.1. Causas

De acuerdo con [Grady and Santoso, 2001], las causas de los armónicos se pueden separar en dos categorías de fuentes a saber; saturación de los dispositivos y dispositivos basados en electrónica de potencia. En la Tabla 2.1 se presenta las diferentes tipos de cargas que generan armónicos en los SEP.

Tabla 2.1: Indicaciones sobre el espectro armónico inyectado por diferentes cargas

| Tipos de carga                      | Armónicos generados                        | Comentarios                            |
|-------------------------------------|--|--|
| Transformador                       | Orden par e impar                          | Componente en CD                       |
| Motor asíncrono                     | Orden impar                                | Inter y subarmónicos                   |
| Lámpara descarga                    | 3 más impares                              | Puede llegar al 30 por ciento de $I_1$ |
| Soldadura arco                      | 3  |  |
| Horno arco CA                       | Espectro variable inestable                | No lineal-asimétrico                   |
| Rectificadores con filtro inductivo | $h = K * P \pm 1$<br>$I_h = \frac{I_1}{h}$ | SAI-Variadores V                       |
| Rectificadores con filtro           | $h = K * P \pm 1$                          | alimentación                           |
| Capacitivo                          | $I_h = \frac{I_1}{h}$                      | equipos electrónicos                   |
| Ciclo-convertidores                 | variables                                  | variadores V                           |
| Reguladores PWM                     | variables                                  | SAI-convertidor CD-CA                  |

Donde

- $h$  es el orden de armónico

- $K = 1, 2, 3, 4, \dots$
- $P$  es el número de pulso del convertidor

### 2.3.2. Efectos de armónicos

Los problemas que surgen del flujo de corriente armónica son sencillos y directos. Los dos problemas principales que veremos son el calentamiento anormal de los componentes y la interrupción del funcionamiento. Los armónicos producen los efectos en equipos eléctricos, en esta parte vamos a ver los diferentes equipos sujetos en los armónicos.

#### 2.3.2.1. Efectos en bancos de capacitores

Una razón por la que los armónicos son un problema para los capacitores es porque la impedancia capacitiva disminuye a medida que aumenta la frecuencia. Por lo tanto, a medida que nuestras corrientes aumentan en orden armónico, los condensadores actuarán cada vez más como un corto. Esto básicamente significa que los condensadores atraen los armónicos de alta frecuencia. El resultado es que los condensadores se sobrecargarán debido al aumento del flujo de corriente [Smith, 2002].

Otro problema entre condensadores y armónicos es la posibilidad de resonancia [Wagner et al., 1993]. Esta condición ocurre cuando las reactancias inductiva y capacitiva de un sistema son iguales y se cancelan entre sí. Cuando ocurre la resonancia, la única impedancia efectiva es la resistencia del sistema. Esto implica que el sistema ve un valor de impedancia menor al esperado. Por lo tanto, las corrientes están menos impedidas y puede producirse una amplificación eficaz. La resonancia causará daños severos y destructivos a los capacitores y también a otras partes del sistema [Smith, 2002].

En [ANSI/IEEE Std 18, 1980] establece limitaciones en el voltaje, la corriente y la potencia reactiva para los bancos de capacitores que se pueden usar para determinar los niveles máximos de armónicos permitidos. Esta norma indica que el condensador se

puede operar continuamente dentro de las siguientes limitaciones, incluidos los componentes armónicos: 110 por ciento del voltaje rms nominal, 120 por ciento del voltaje pico nominal, 180 por ciento de la corriente rms nominal y 135 por ciento de la potencia reactiva nominal.

#### **2.3.2.2. Equipos electrónicos y cómputo**

Los equipos electrónicos (convertidores) dependen de los cruces por cero de la forma de onda, pero con la presencia de las componentes de armónicos estos pueden estar afectar por culpa de la distorsión armónica. En los sistemas computacionales, las componentes armónicas afectan el funcionamiento de la tarjeta madre de computadoras.

#### **2.3.2.3. Medidores de kW-hr y relevadores de sobre corriente**

Los armónicos generan un par adicional en el disco de inducción, que puede causar un funcionamiento incorrecto, ya que estos dispositivos están calibrados para un funcionamiento preciso solo en la frecuencia fundamental. Con el mal funcionamiento de los aparatos esto va a provocar una medición incorrecta en el momento de la lectura de los valores.

#### **2.3.2.4. Transformadores**

El efecto principal de los armónicos del sistema de potencia en los transformadores es el calor adicional generado por las pérdidas causadas por el contenido armónico de la corriente de carga. Otros problemas incluyen la posible resonancia entre la inductancia del transformador y la capacitancia del sistema, tensiones mecánicas de aislamiento (devanado y laminación) debido a los ciclos de temperatura y la posible vibración del núcleo pequeño [Wagner et al., 1993].

El calentamiento adicional causado por los armónicos del sistema requiere una reducción de la capacidad de carga para permanecer o el uso de transformadores especiales diseñados para corrientes de carga no sinusoidales. La vida útil del transformador se reducirá

como resultado de operar por encima de las temperaturas nominales [Wagner et al., 1993].

#### **2.3.2.5. Motores**

La presencia de componentes armónicos tanto de voltaje como de corriente puede ocasionar pérdidas de potencia por calentamiento tanto en los devanados como en el núcleo estatórico y rotórico. Además la naturaleza secuencial (corrientes armónicas de secuencia negativa) de algunas corrientes armónicas en los sistemas trifásicos produce pares antagónicos en el rotor. Los voltajes no sinusoidales aplicados a máquinas eléctricas pueden causar sobrecalentamiento, pares pulsantes o ruido. Además de las aplicaciones a través de la línea, los motores de accionamiento de velocidad ajustable se alimentan desde inversores que pueden producir una distorsión de voltaje significativa [Wagner et al., 1993].

El sobrecalentamiento del rotor ha sido el principal problema asociado con la distorsión de voltaje. Las pérdidas en las máquinas eléctricas dependen del espectro de frecuencia del voltaje aplicado. Las pérdidas de núcleo y parásitas pueden llegar a ser significativas en un motor de inducción con un rotor sesgado alimentado por un inversor que produce altas frecuencias armónicas. Un aumento en la vida útil del motor. Los motores monofásicos son los más afectados. El aumento de temperatura no es irregular en todo el motor; Aparecen puntos calientes cerca de los conductores dentro de las porciones del núcleo de hierro. Si los armónicos varían en el tiempo, el motor puede tolerar niveles de distorsión pico más altos sin aumentar la temperatura del punto caliente. Esto es posible porque la constante de tiempo térmica del motor es mucho más larga que el período de variación armónica [Wagner et al., 1993].

#### **2.3.2.6. Relevadores de disco de inducción electromecánicos**

En estos equipos se producen componentes de par adicionales que pueden alterar las características de retardo de tiempo de los relevadores. Así los componentes armónicos afectan la precisión y la fiabilidad de estos equipos. Estos da como resultandos mediciones incorrectos

y disparo incorrecto de relés [Kennedy, 2000].

### 2.3.2.7. Interruptores de circuito

En estos equipos es posible que las bobinas de escape no funcionen correctamente en presencia de corrientes armónicas que podría crear un disparo falso en los circuitos de protección [Kennedy, 2000].

A continuación, podemos resumir las causas y efectos de los armónicos en la Tabla 2.2.

Tabla 2.2: Resumen de causas y efectos de los armónicos

| Efectos de los armónicos        | Causas   | Consecuencias  |
|---------------------------------|--|--|
| Sobre el conductor              | <ul style="list-style-type: none"> <li>* Las intensidades armónicas provocan el aumento de la <math>I_{RMS}</math></li> <li>* El efecto piel reduce la sección efectiva de los conductores a media que aumenta la frecuencia</li> </ul>        | <ul style="list-style-type: none"> <li>* Disparos intempestivos de las protecciones,</li> <li>* Sobrecalentamiento de los conductores.</li> </ul>                    |
| Sobre el conductor de neutro    | <ul style="list-style-type: none"> <li>* Cuando existe una carga trifásica + neutro equilibrada que genera armónicos. impares múltiplos de 3</li> </ul>  | Cierre de los armónicos <ul style="list-style-type: none"> <li>* homopolares sobre el neutro que provoca calentamientos y sobreintensidades.</li> </ul>              |
| Sobre los transformadores       | <ul style="list-style-type: none"> <li>* Aumento de la <math>I_{RMS}</math></li> <li>* Las pérdidas por Foucault son proporcionales al cuadrado de la frecuencia, las perdidas por histéresis con proporcionales a las frecuencias.</li> </ul> | <ul style="list-style-type: none"> <li>* Aumento de los calentamientos por efecto Joule en los devanados,</li> <li>* Aumento de las pérdidas en el hierro</li> </ul> |
| Sobre los motores               | <ul style="list-style-type: none"> <li>* Análogas a las de los transformadores y generación de un campo adicional al principal</li> </ul>  | <ul style="list-style-type: none"> <li>* Análogas a las de los transformadores más pérdidas de rendimiento.</li> </ul>   |
| Sobre los bancos de capacitores | <ul style="list-style-type: none"> <li>* Disminución de la impedancia del condensador con el aumento de la frecuencia.</li> </ul>  | <ul style="list-style-type: none"> <li>* Envejecimiento prematuro, amplificación de los armónicos existentes.</li> </ul>   |

## 2.4. Recomendaciones de los límites de armónicos en los sistemas de potencia

En [IEEE Std 519, 2014] se presentan los límites permitidos de distorsión de voltaje y corriente en los SEP. La Tabla 2.3 presenta los límites de distorsión de voltaje en el Punto de acoplamiento común (PCC). Las Tablas 2.4, 2.5 y 2.6 presentan la Distorsión total de demanda (TDD) en función de las componentes armónicas de corriente. En donde  $I_{SC}$  es la corriente máxima de cortocircuito en el PCC y  $I_L$  es la corriente de carga de demanda máxima (componente de frecuencia fundamental) en el PCC en condiciones normales de funcionamiento de carga.

Tabla 2.3: Límites de distorsión de voltaje

| Voltaje de bus $V$ en PCC     | Armónica individual [%] | THD [%] |
|-------------------------------|-------------------------|---------|
| $V \leq 1.0$ kV               | 5.0                     | 8.0     |
| $1\text{kV} < V \leq 69$ kV   | 3.0                     | 5.0     |
| $69\text{kV} < V \leq 161$ kV | 1.5                     | 2.5     |
| $161\text{kV} < V$            | 1.0                     | 1.5     |

Tabla 2.4: Límites de distorsión de corriente para sistemas clasificados de 120 V a 69 kV

| Distorsión máxima de corriente armónica en porcentaje de $I_L$ |                 |                  |                  |                  |                  |         |
|--|-----------------|------------------|------------------|------------------|------------------|---------|
| Orden armónico individual (armónicos impares)                  |                 |                  |                  |                  |                  |         |
| $I_{SC}/I_L$   | $3 \leq h < 11$ | $11 \leq h < 17$ | $17 \leq h < 23$ | $23 \leq h < 35$ | $35 \leq h < 50$ | TDD [%] |
| $< 20$   | 4.0             | 2.0              | 1.5              | 0.6              | 0.3              | 5.0     |
| $20 < 50$  | 7.0             | 3.5              | 2.5              | 1.0              | 0.5              | 8.0     |
| $50 < 100$   | 10.0            | 4.5              | 4.0              | 1.5              | 0.7              | 12.0    |
| $100 < 1000$   | 12.0            | 5.5              | 5.0              | 2.0              | 1.0              | 15.0    |
| $> 1000$   | 15.0            | 7.0              | 6.0              | 2.5              | 1.4              | 20.0    |

Tabla 2.5: Límites de distorsión de corriente para sistemas clasificados por encima de 69 kV hasta 161 kV

| Distorsión máxima de corriente armónica en porcentaje de $I_L$ |                 |                  |                  |                  |                  |         |
|--|-----------------|------------------|------------------|------------------|------------------|---------|
| Orden armónico individual (armónicos impares)                  |                 |                  |                  |                  |                  |         |
| $I_{SC}/I_L$   | $3 \leq h < 11$ | $11 \leq h < 17$ | $17 \leq h < 23$ | $23 \leq h < 35$ | $35 \leq h < 50$ | TDD [%] |
| < 20   | 2.0             | 1.0              | 0.75             | 0.3              | 0.15             | 2.5     |
| 20 < 50  | 3.5             | 1.75             | 1.25             | 0.5              | 0.25             | 4.0     |
| 50 < 100   | 5.0             | 2.25             | 2.0              | 0.75             | 0.35             | 6.0     |
| 100 < 1000   | 6.0             | 2.75             | 2.5              | 1.0              | 0.5              | 7.5     |
| > 1000   | 7.5             | 3.5              | 3.0              | 1.25             | 0.7              | 10.0    |

Tabla 2.6: Límites de distorsión de corriente para sistemas clasificados > 161 kV

| Distorsión máxima de corriente armónica en porcentaje de $I_L$ |                 |                  |                  |                  |                  |         |
|--|-----------------|------------------|------------------|------------------|------------------|---------|
| Orden armónico individual (armónicos impares)                  |                 |                  |                  |                  |                  |         |
| $I_{SC}/I_L$   | $3 \leq h < 11$ | $11 \leq h < 17$ | $17 \leq h < 23$ | $23 \leq h < 35$ | $35 \leq h < 50$ | TDD [%] |
| < 25   | 1.0             | 0.5              | 0.38             | 0.15             | 0.1              | 1.5     |
| 25 < 50  | 2.0             | 1.0              | 0.75             | 0.3              | 0.15             | 2.5     |
| $\geq 50$  | 3.0             | 1.5              | 1.15             | 0.45             | 0.22             | 3.75    |

## 2.5. Serie de Fourier

En general, una forma de onda periódica no sinusoidal  $f(t)$  con frecuencia angular  $\omega$  se puede expresar [Mohan et al., 2003] y [González, 2003] como:

$$f(t) = a_0 + \sum_{h=1}^{\infty} a_h \cos(h\omega t) + b_h \sin(h\omega t) \quad (2.1)$$

Sustituyendo  $\omega = \frac{2\pi}{T}$  en la ecuación (2.1) nos da

$$f(t) = a_0 + \sum_{h=1}^{\infty} a_h \cos\left(\frac{2h\pi t}{T}\right) + b_h \sin\left(\frac{2h\pi t}{T}\right) \quad (2.2)$$

Donde :

$$a_0 = \frac{1}{T} \int_0^T f(t) dt \quad (2.3)$$

$$a_h = \frac{1}{T} \int_0^T f(t) \cos\left(\frac{2h\pi t}{T}\right) dt, h = 0, \dots, \infty \quad (2.4)$$

$$b_h = \frac{1}{T} \int_0^T f(t) \sin\left(\frac{2h\pi t}{T}\right) dt, h = 1, \dots, \infty \quad (2.5)$$

La serie de Fourier nos permite escribir cualquier señal periódica como una suma infinita de señales pequeñas, por lo que los parámetros  $a_0$ ,  $a_h$  y  $b_h$  comúnmente se denominan coeficiente de Fourier que debe calcularse.

## 2.6. Cantidades para la cuantificación de los armónicos

### 2.6.1. Transformada rápida de Fourier

La evolución de los esquemas de control para los convertidores electrónicos de potencia exige capacidades de procesamiento cada vez más altas, con altos consumos de tiempo y de recursos. En cuanto a compatibilidad, uno de los retos más importantes es el relacionado con distorsión armónica y el factor de potencia. En el diseño de sistemas electrónicos para medición y corrección, uno de los algoritmos más importantes es el de la Transformada Rápida de Fourier (FFT) [José I. et al., 2012]. La FFT es una herramienta conocida y ampliamente utilizada en muchos campos científicos y de ingeniería [Zhang et al., 2013], [Zhao and Zhao, 2018], [Mohapatra and Mohapatra, 2017].

En [Cooley and Tukey, 1965] se dice: la FFT es un algoritmo que reduce el tiempo de cálculo de  $n^2$  pasos a  $n \cdot \log_2(n)$ . El único requisito es que el número de puntos en la serie tiene que ser una potencia de 2 ( $2^n$  puntos), por ejemplo 16, 32, 64, 128, 252, 512 etc. La FFT se expresa de la siguiente manera:

$$X(k) = \sum_{j=1}^N x(j)\omega_N^{(j-1)(k-1)} \quad (2.6)$$

donde:

- $\omega_N = \exp(-\frac{2\pi i}{N})$
- $x(j)$ : señal que queremos encontrar su transformada de Fourier

### 2.6.2. Distorsión armónica total

En [IEEE Std 519, 1993], este término ha entrado en el uso común para definir el “factor de distorsión” de tensión o corriente. El factor de distorsión (Factor Armónico) es la relación de la raíz cuadrada de los armónicos contenidos al valor efectivo de la cantidad fundamental, expresado en porcentaje de la fundamental. La distorsión armónica total del voltaje ( $THD_V$ ) en [IEEE Std 1459-2010, 2010] se define para ecuación (2.7).

$$THD_V = \frac{\sqrt{V_0^2 + \sum_{h \neq 1} V_h^2}}{V_1} \quad (2.7)$$

Donde

- $h$  es el orden de armónico
- $V_0$  es la componente de CD del voltaje
- $V_1$  es la componente fundamental del voltaje

La distorsión armónica total de la corriente ( $THD_I$ ) en [IEEE Std 1459-2010, 2010] se define para ecuación (2.8).

$$THD_I = \frac{\sqrt{I_0^2 + \sum_{h \neq 1} I_h^2}}{I_1} \quad (2.8)$$

Donde

- $h$  es el orden de armónico
- $I_0$  es la componente de CD de la corriente
- $I_1$  es la componente fundamental de la corriente

## 2.7. Conclusiones

La electrónica de potencia ofrece muchas ventajas en el campo de la ingeniería eléctrica, como en la transmisión de energía eléctrica en cd con HVDC y AC con FACTS, como a los aparatos de los usuarios finales también ofrece ventajas. Pero los equipos electrónicos son parte de una carga no lineal. Las cargas no lineales son más frecuentes en los equipos utilizados por la empresa proveedora y los usuarios finales. Las cargas no lineales consumen corrientes armónicas que pueden causar problemas tanto a la empresa proveedora como al equipo del usuario final. Los armónicos son la fuente de muchas perturbaciones en las redes eléctricas, sin embargo, no se puede evitar el uso de equipos electrónicos, por lo cual es importante buscar técnicas de atenuación para remediarlo.

Los armónicos son problemas reales que existen en los sistemas de alimentación y deben tomarse en serio porque son capaces de generar pérdidas económicas y materiales. Es absolutamente importante que las instalaciones que no pueden prescindir de cargas no lineales puedan al menos producir menos de acuerdo con el estándar IEEE 519-2010.



## Capítulo 3

# Cómputo en paralelo basado en threads

### 3.1. Introducción

La humanidad ha vivido grandes conflictos, el más conocido es la segunda Guerra Mundial, durante a cual se dio el nacimiento de herramientas de cálculo excepcionales, las cuales fueron utilizadas en la resolución de problemas tales como logística de transporte, modelado matemático, simulación etc.

La mayoría de equipos, dispositivos y herramientas son fruto de la investigación de las fuerzas armadas para poder obtener la victoria frente a sus oponentes. La creación de máquinas de calcular como la *Mathematical Analyzer, Numerical Integrator, and Computer (MANIAC)*, antecesoras de las computadoras actuales, permitió resolver problemas complejos utilizando algoritmos cada vez más sofisticados.

Sin embargo, sea cual sea el desarrollo de un algoritmo, siempre hay un tiempo de ejecución de este programa. Cuando el programa contiene más secuencias, el tiempo de procesamiento aumenta. Por ello es recomendable realizar una paralelización de determinados

pasos para poder reducir el tiempo de ejecución, de ahí la necesidad de utilizar el cómputo en paralelo [Liu et al., 2012].

Este capítulo, se abordan los conceptos básicos en torno al cómputo en paralelo, las arquitecturas computacionales y a la plataforma para el cómputo en paralelo.

## **3.2. Definiciones**

### **3.2.1. Cómputo en paralelo**

El cómputo en paralelo es el uso de una computadora paralela para reducir el tiempo necesario de solución a un problema computacional. El cómputo en paralelo ahora se considera un estándar para que los científicos e ingenieros resuelvan problemas en áreas tan diversas como la evolución galáctica, el modelado climático, el diseño de aviones, la dinámica molecular así como el análisis de los sistemas eléctricos [Quin, 2003], [Giloi, 1993].

### **3.2.2. Computadora paralela**

Una computadora en paralelo es un sistema computacional con múltiples procesadores que admite la programación en paralelo. Existen dos categorías importantes de computadoras paralelas son: las multicomputadoras y los multiprocesadores.

#### **3.2.2.1. Multicomputadora**

Una multicomputadora es una computadora paralela construida a partir de varias computadoras y conectadas a través de una red de cómputo. Los procesadores en diferentes computadoras interactúan pasándose mensajes entre sí [Quin, 2003]. En la Figura 3.1 se observa un ejemplo de una multicomputadora con 4 computadoras conectada en la red.

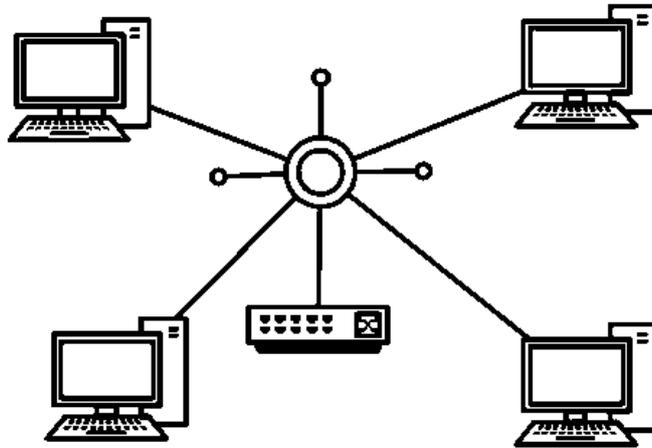


Figura 3.1: Red de cómputo en paralelo

#### 3.2.2.2. Multiprocessor

Una computadora multiprocesador es un sistema integrado para varios procesadores ubicadas en una sola tarjeta. En esta computadora todos los procesadores comparten una memoria. Esta memoria compartida admite la comunicación y sincronización entre procesadores [Quin, 2003]. En la Figura 3.2 se ilustra la representación un multiprocesador con  $n$  procesadores.

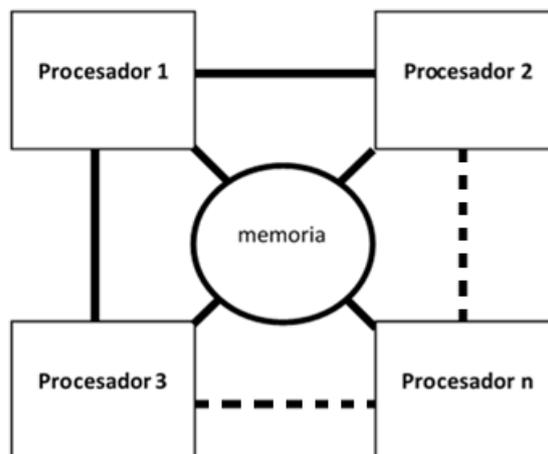


Figura 3.2: Esquema de un sistema multiprocesador

### 3.2.3. Programación en paralelo

La programación en paralelo es la programación en un lenguaje que permite indicar explícitamente cómo diferentes partes del cálculo pueden ser ejecutadas simultáneamente por diferentes procesadores [Quin, 2003]. En la Figura 3.3 se ilustra el esquema de una paralelización de un programa con 3 procesadores.

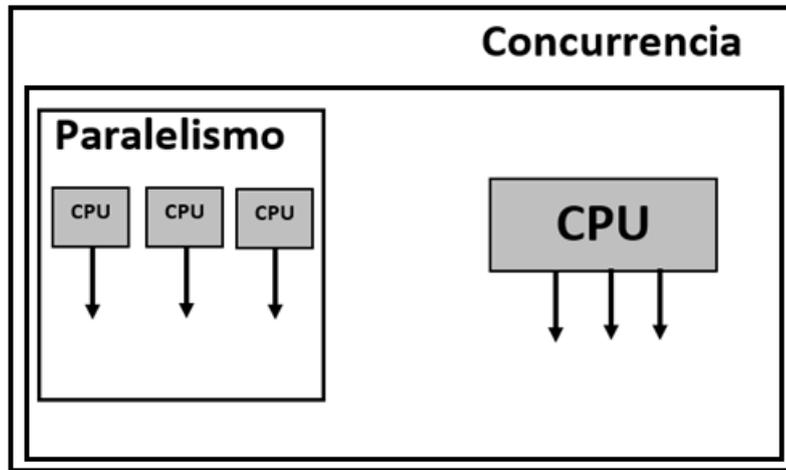


Figura 3.3: Programación en paralelo

## 3.3. Plataforma Threads

### 3.3.1. Concepto de Thread

Los threads representan una plataforma que utiliza recursos de la CPU para resolver problemas. Esta plataforma puede usar un sólo thread para resolver un problema paso a paso con un solo procesador cuando el proceso no tiene secciones que se puedan paralelizar. Este proceso se denomina secuencial. Si el problema es grande y tiene secciones que podrían paralelizarse, estas secciones se asignan a varios threads que se distribuyen a los subprocesos asignados a los procesadores. Los procesadores de subprocesos múltiples simultáneos pueden mejorar el rendimiento al permitir que varios subprocesos independientes emitan

múltiples instrucciones en cada ciclo. Sin embargo, todas las estrategias de asignación de recursos de subprocesos múltiples simultáneos existentes asignan los recursos de acuerdo con el estado de las instrucciones de los subprocesos y sin considerar las prioridades de subprocesos que otorga el sistema operativo [Cheng and Yang, 2009], [Miao et al., 2009]. Este último proceso se conoce como cómputo en paralelo.

En la Figura 3.4 se muestra los dos tipos del modelo de threads. En la Figura 3.4 (a) podemos ver el concepto de uso un thread conocido en inglés *“Single-thread proces”* y de la Figura 3.4 (b) se muestra el uso de varios hilos conocido en inglés *“Multithreading”*, donde cada proceso se hace por el medio de un thread; los threads comparten los mismos recursos de código, datos, y archivos.

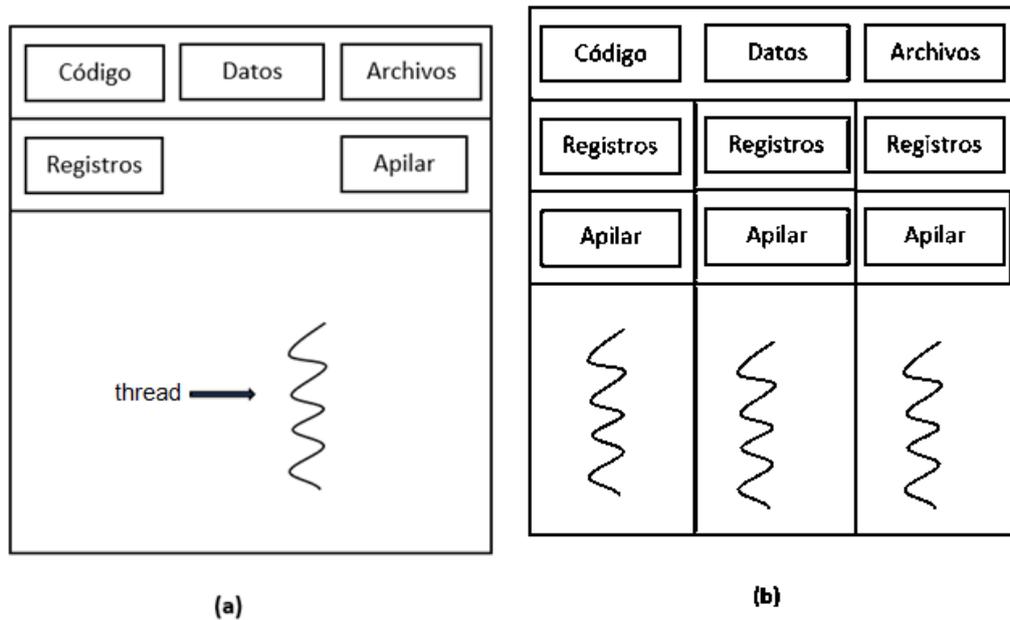


Figura 3.4: Concepto de simple thread y varias threads

La Figura 3.5 (a) muestra el tipo de esquema de programación con un sólo thread que hace el tratamiento de todas las tareas y en la Figura 3.5 (b) se muestra el uso de varios threads (*“Multithreading”*), donde podemos observar que cada thread trata un sólo tarea.

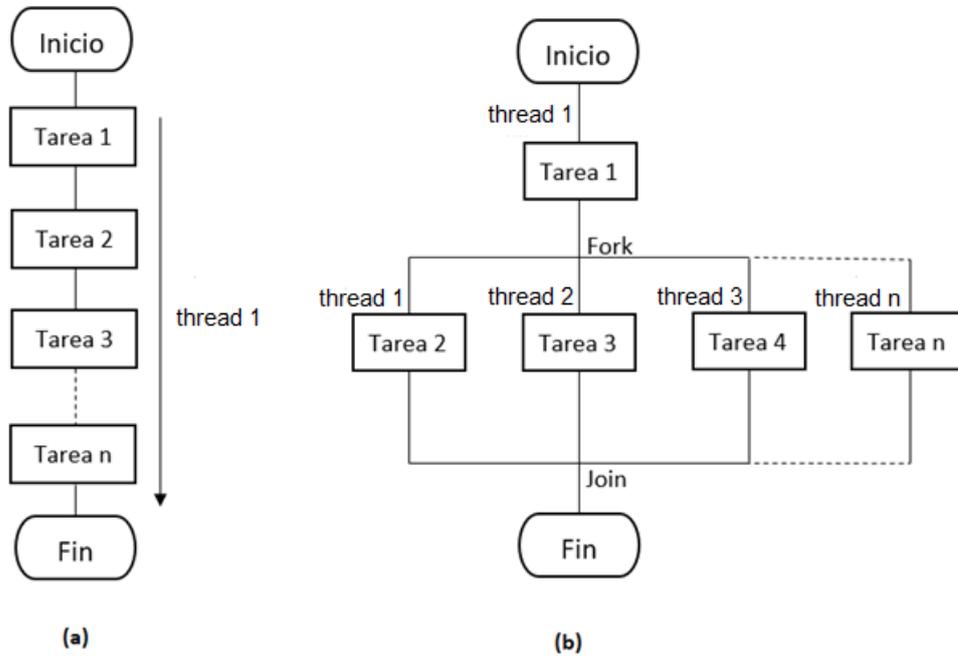


Figura 3.5: Esquema de threads

### 3.3.2. Comandos principales de la plataforma Thread en Lenguaje C

#### 3.3.2.1. pthread\_create

La función `pthread_create()` se usa para crear un nuevo hilo, con atributos especificados por `attr`, dentro de un proceso. Si `attr` es `NULL`, se utilizan los atributos predeterminados. Si los atributos especificados por `attr` se modifican más tarde, los atributos del hilo se ven afectados. Una vez completado con éxito, `pthread_create()` almacena el ID del hilo creado en la ubicación a la que hace referencia el hilo [Michael, 2010]. La sintaxis de la función es:

```
int pthread_create(pthread_t*restrict thread,
const pthread_attr_t*restrict attr,
void*(*start_routine)(void*),
void*restrict arg);
```

### 3.3.2.2. pthread\_attr\_init

La función `pthread_attr_init ()` inicializa un objeto de atributos de hilo `attr` con el valor predeterminado para todos los atributos individuales usados por una implementación dada [Michael, 2010]. La sintaxis de la función es:

```
int pthread_attr_init(pthread_attr_t*attr);
```

### 3.3.2.3. pthread\_join

La función `pthread_join ()` espera a que termine el hilo especificado por `hilo`. Si ese hilo ya ha terminado, entonces `pthread_join ()` regresa inmediatamente. El hilo especificado por `hilo` debe poder unirse [Michael, 2010]. La sintaxis de la función es:

```
int pthread_join(pthread_t thread, void **retval);
```

### 3.3.2.4. pthread\_exit

La función `pthread_exit ()` termina el hilo de llamada y hace que el valor `value_ptr` esté disponible para cualquier unión exitosa con el hilo de terminación. Todos los controladores de limpieza de cancelación que se han insertado y aún no se muestran en el orden inverso al que se enviaron y luego se ejecutaron. Después de que se hayan ejecutado todos los controladores de limpieza de cancelación, si el subproceso tiene datos específicos del subproceso, se llamarán las funciones de destructor apropiadas en un orden no especificado [Michael, 2010]. La sintaxis de la función es:

```
noreturn void pthread_exit(void *retval);
```

### 3.3.2.5. pthread\_mutex\_init

La función `pthread_mutex_init ()` inicializará el mutex referenciado por `mutex` con atributos especificados por `attr`. Si `attr` es `NULL`, se utilizan los atributos de exclusión mutua

predeterminados; el efecto será el mismo que pasar la dirección de un objeto de atributos de exclusión mutua por defecto. Tras la inicialización satisfactoria, el estado del mutex se inicializa y desbloquea [Michael, 2010]. La sintaxis de la función es:

```
int pthread_mutex_init(pthread_mutex_t*restrict mutex,  
const pthread_mutexattr_t*restrict attr);
```

### 3.3.2.6. `pthread_mutex_destroy`

La función `pthread_mutex_destroy ()` destruirá el objeto mutex al que hace referencia `mutex`; el objeto `mutex` se vuelve, en efecto, no inicializado. Una implementación puede hacer que `pthread_mutex_destroy ()` establezca el objeto referenciado por `mutex` en un valor no válido [Michael, 2010]. La sintaxis de la función es:

```
int pthread_mutex_destroy(pthread_t*mutex);
```

## 3.4. **Diseño de algoritmos en paralelo por la Metodología de Ian Foster**

Ian Foster ha propuesto una metodología [Foster, 1995] de cuatro pasos para diseñar algoritmos paralelos. Fomenta el desarrollo de algoritmos paralelos escalables al retrasar las consideraciones dependientes de la máquina a los pasos posteriores. Los cuatro pasos del diseño se denominan partición, comunicación, aglomeración y mapeo. En esta sección explicamos cada uno de estos pasos [Quin, 2003]. En la Figura 3.6 se muestra el esquema de los pasos de la metodología de Ian Foster por un proceso de paralelización de un problema.

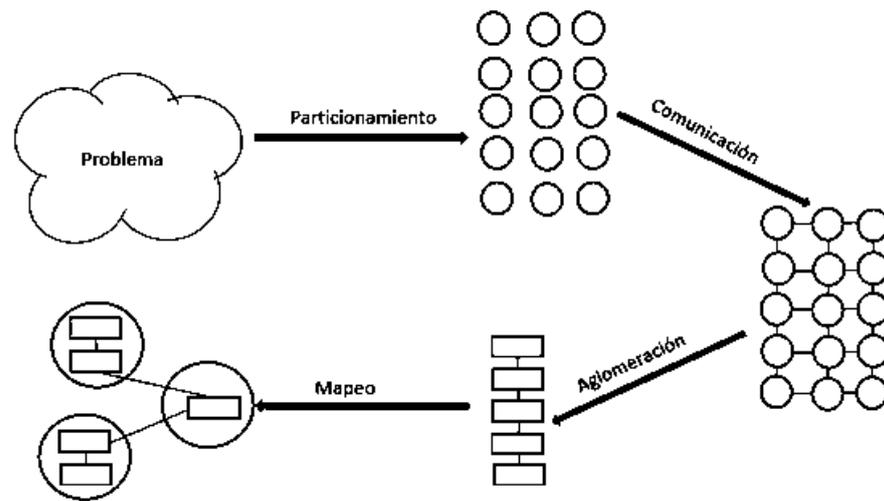


Figura 3.6: Metodología de diseño de algoritmos paralelos de Ian Foster

### 3.4.1. Particionamiento

El particionamiento es el primer paso en la metodología de Ian Foster que consiste en descomponer un problema en subproblemas de tal manera que se reduzca la dificultad del problema. El objetivo de esta tarea es reducir la complejidad del problema. La estrategia se puede llevar a cabo sobre la base de datos o cálculos. Una buena partición divide tanto el cálculo como los datos en muchas piezas pequeñas. [Foster, 1995].

### 3.4.2. Comunicación

La comunicación es el segundo paso, por lo cual se identifica las tareas primitivas y se determina el patrón de comunicación entre ellas. Este dos principales tipos de comunicaciones; la comunicación local se hace cuando una operación necesita datos de un pequeño número de otras tareas. Una operación de comunicación global es aquella en la que deben participar muchas tareas [Foster, 1995], [Quin, 2003].

### 3.4.3. Aglomeración

La aglomeración es el tercer paso que consiste en revisar las decisiones tomadas en la participación y comunicación con el fin de obtener un algoritmo que se ejecutará de manera eficiente [Foster, 1995].

### 3.4.4. Mapeo

El mapeo es el proceso de asignar tareas a los procesadores. Si estamos ejecutando nuestro programa en un multiprocesador centralizado. El sistema operativo asigna automáticamente los procesos a los procesadores. Por lo tanto, nuestra discusión asume que el sistema de destino es una computadora paralela de memoria distribuida [Quin, 2003]. Los objetivos del mapeo son maximizar la utilización del procesador y minimizar la comunicación entre procesadores. La utilización del procesador es el porcentaje promedio de tiempo que los procesadores del sistema están ejecutando activamente las tareas necesarias para la solución del problema. La utilización del procesador se maximiza cuando el cálculo se equilibra de manera uniforme, lo que permite que todos los procesadores comiencen y finalicen la ejecución al mismo tiempo [Quin, 2003].

## 3.5. Métricas para el procesamiento en paralelo

### 3.5.1. *Speed-up*

Diseñamos e implementamos programas paralelos con la esperanza de que se ejecuten más rápido que sus contrapartes secuenciales [Quin, 2003]. *Speed-up* es la relación entre el tiempo de ejecución secuencial y el tiempo de ejecución en paralelo:

$$speed-up = \frac{\text{Tiempo de ejecución secuencial}}{\text{Tiempo de ejecución paralelo}} \quad (3.1)$$

### 3.5.2. Eficiencia

La eficiencia de un programa paralelo es una medida de la utilización del procesador. Definimos la eficiencia como la aceleración dividida por el número de procesadores utilizados:

$$\text{Eficiencia} = \frac{\text{Tiempo de ejecución secuencial}}{\text{procesadores utilizadas} \times \text{Tiempo de ejecución en paralelo}} \quad (3.2)$$

### 3.5.3. Ley de Amdahl

La ley de Amdahl se puede escribir de la siguiente manera: Sea  $f$  la fracción de operaciones en un cálculo que debe realizarse secuencialmente, donde  $0 \leq f \leq 1$ . El  $\psi$  de aceleración máxima que puede alcanzar una computadora en paralelo con procesadores  $p$  que realizan el cálculo es

$$\psi \leq \frac{1}{f + \frac{(1-f)}{p}} \quad (3.3)$$

La ley de Amdahl se basa en el supuesto de que estamos tratando de resolver un problema de tamaño fijo lo más rápido posible. Proporciona un límite superior en la aceleración que se puede lograr aplicando un cierto número de procesadores para resolver el problema en paralelo. También se puede utilizar para determinar la aceleración asintótica que se puede lograr a medida que aumenta el número de procesadores [Quin, 2003].

## 3.6. Conclusiones

En este presente Capítulo hemos definido; el cómputo en paralelo, computadora paralela, programación en paralelo. Después pudimos hablar sobre la metodología de Ian Foster que se divide en cuatro etapas; particionamiento, comunicación, mapeo y aglomeración que

permiten realizar el diseño de paralelización. Además hemos visto la plataforma de threads donde vimos dos conceptos muy importantes a saber; el cómputo secuencial usando el concepto de un solo thread (*Single-thread*) y el cómputo en paralelo usando varias threads (*Multithreading*). Al final, se presentó métricas para el procesamiento en paralelo que vos permite evaluar el rendimiento del cómputo en paralelo donde el más importante conocido por el *speed-up*. Debido al desarrollo de este Capítulo y en la actualidad de la importancia del cómputo en paralelo en nuestra área de ingeniería eléctrica podemos afirmar que el cómputo en paralelo es una técnica muy importante para nuestro trabajo dado que nos va a ayudar a alcanzar nuestro objetivo en la investigación de esta tesis.

## Capítulo 4

# Propuesta de paralelización de algoritmos usados para el cálculo de métricas del análisis armónico

### 4.1. Introducción

En este capítulo se presentan las propuestas para la paralelización de los algoritmos usados en el cálculo de la transformada rápida de Fourier, la distorsión armónica total y la respuesta a la frecuencia de una red eléctrica. Estos algoritmos son paralelizados usando la plataforma de procesamiento en paralelo basada en threads.

### 4.2. Propuesta de paralelización del cálculo de la FFT

Considérese la expresión matemática usada para el cálculo de la FFT, dada por (4.1):

$$X(k) = \sum_{j=1}^N x(j)\omega_N^{(j-1)(k-1)} \quad (4.1)$$

Donde

- $N$  es el número de muestras de la señal
- $k$  es el orden armónico

De la expresión (4.1) se aprecia que cada uno de los armónicos pueden ser calculados de manera independiente, por lo que, si se dispusiera de un número  $P$  de elementos de proceso, el cual fuera igual al número de armónicos que se deseara calcular de la señal, entonces, se podría hacer un mapeo, para que cada elemento de proceso calculara un armónico, tal y como se muestra en la Figura 4.1,

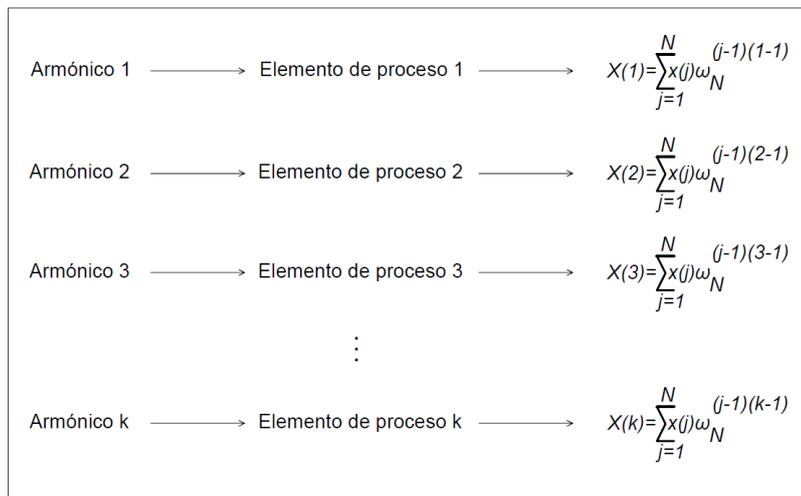


Figura 4.1: Distribución de los armónicos en los procesadores

De la Figura 4.1 se aprecia que cada elemento de proceso calcula un armónico, de los  $K$  armónicos que se desea determinar de la forma de estudio de estudio. Desafortunadamente, no es muy común contar con un número muy grande de elementos de proceso, por lo que, si se dispone de un número  $P$  de elementos de proceso, el cual es menor al número  $K$ , de armónicos que se desean calcular, entonces, se deberá de asignar a cada elemento de proceso el cálculo de más de un armónico. Por ejemplo, si se dispone de 8 elementos de proceso, y se desean calcular los primeros 40 armónicos de una señal, entonces la división de los 40 armónicos entre los 8 elementos de proceso, da como resultado 5, esto significa que cada elemento de proceso calculará 5 armónicos. En la Figura 4.2 se muestra la forma del mapeo anteriormente descrito.

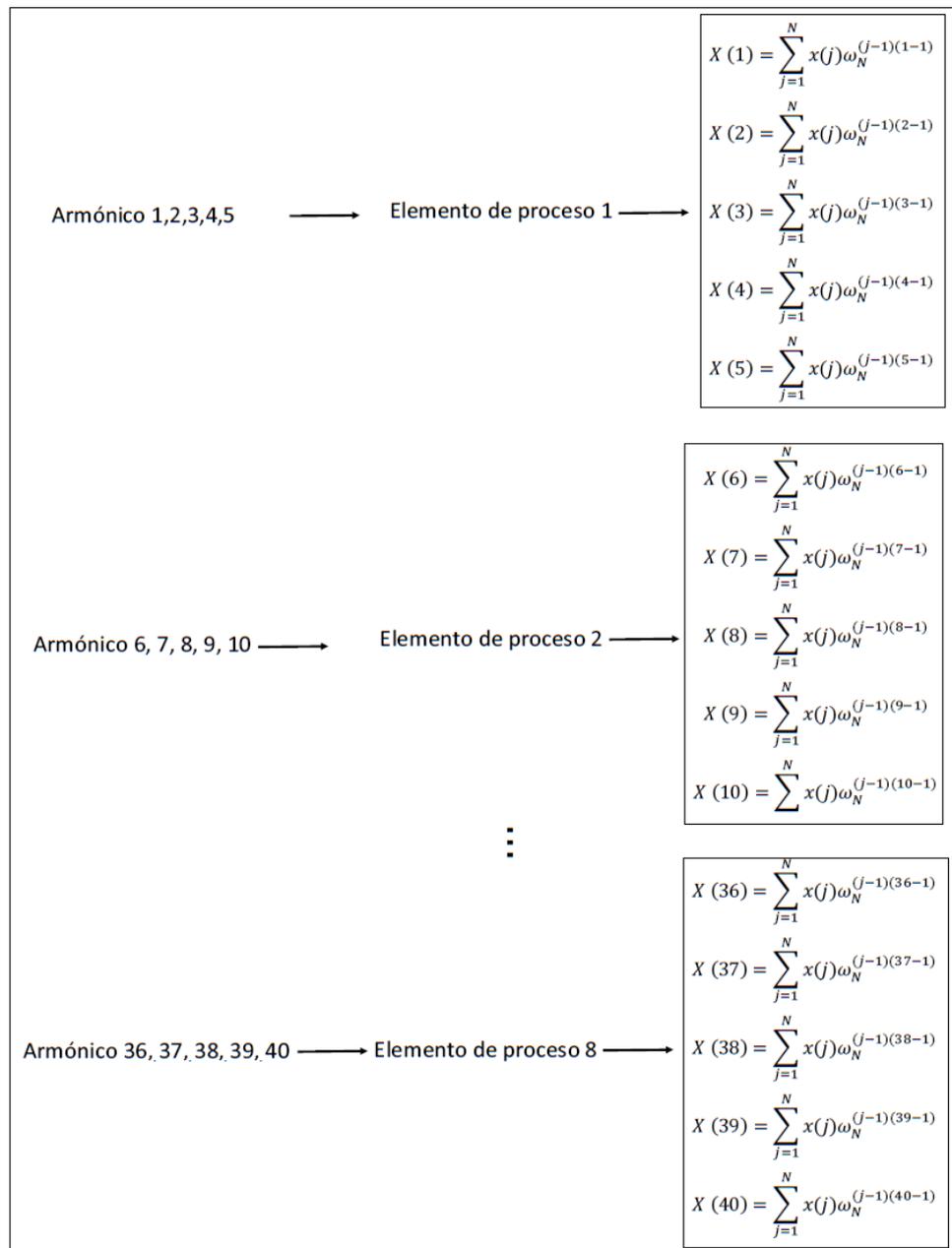


Figura 4.2: Distribución de los armónicos en 8 procesadores

No siempre la división entre el número de tareas a realizar y el número de elementos de proceso resulta en una cantidad entera. Bajo esta situación, se tomará la decisión de repartir el remanente de la división entre los primeros elementos de proceso, con el objeto

de tratar de equilibrar los cálculos entre los elementos de proceso. El algoritmo paralelo para el cálculo de la FFT se muestra en la Tabla 4.1. y en la Figura 4.3 se presenta su diagrama de flujo.

Tabla 4.1: Algoritmo de cálculo en paralelo de la FFT

1. Inicio
2. Establecer el número  $P$  de elementos de cómputo a utilizar
3. Establecer el número  $K$  de armónicos a calcular
4. Calcular el número  $m$  de armónicos a ser calculados por cada elemento de cómputo
5. Realizar el mapeo a cada elemento de proceso
6. Esperar los resultados por parte de cada elemento de proceso
7. Poner a disposición los resultados obtenidos
8. Fin

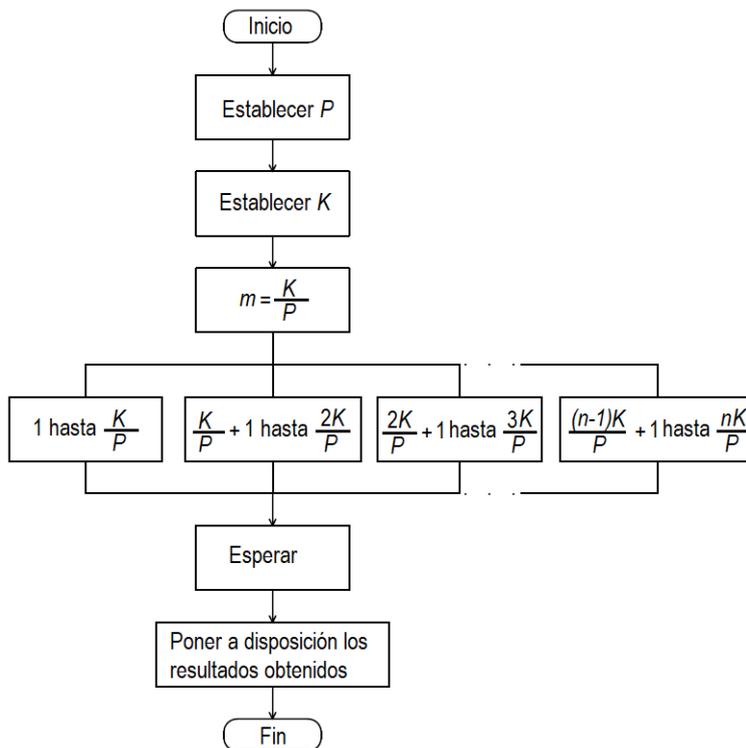


Figura 4.3: Diagrama de flujo de la FFT

Los resultados son colocados en un vector de tamaño  $K$ , el cual se encuentra en la memoria compartida. Con el objeto de que, en un mismo instante de tiempo, más de

un hilo no pueda acceder al vector en el cual se almacenan los  $k$  armónicos, en los que se descompone la señal, se ha utilizado un esquema de variables de exclusión (`mutex_lock`). La Figura 4.4 muestra la función que realiza el cálculo de un rango de armónicos. En el código se aprecia que la función recibe como parámetros el armónico inicial del rango, así como el armónico final del rango de armónicos a ser calculados. Los ciclos anidados contenidos en la función tienen como objeto, primero, calcular el  $k$  –ésimo armónico del rango, en tanto que el ciclo interno calcula el armónico  $k$ . Una vez que es calculado el armónico  $k$ , este se almacena en el vector `res`. Para almacenar el armónico en el vector `res` se utiliza un esquema de variables de exclusión para evitar que más de un elemento de proceso pueda acceder al mismo. Debe de recordarse que la función presentada en la figura anterior se ejecuta de manera simultánea en los  $P$  elementos de proceso disponibles. Cuando dos elementos de proceso han realizado el cálculo de un armónico, y los dos quieren modificar el vector `res`, se establece que el primero que termina bloquea al vector `res` para su modificación, por lo tanto, el segundo elemento de proceso deberá esperar.

```

void *fft(void *args)
{
    int ini,fin;
    int i,j;
    complex double suma, valor;
    datos *mis_datos=(datos*)args;
    ini = mis_datos->inicio;
    fin = mis_datos->final;
    for (i=ini; i<fin; i++)
    {
        real(suma) = 0.0;
        imag(suma) = 0.0;
        for (j=0; j<puntos; j++)
        {
            real(valor) = 0.0;
            imag(valor) = -2.0*pi*i*j/puntos;
            suma = suma + vector[j]*exponente(valor);
        }
        pthread_mutex_lock(&mutex);
        res[i] = suma;
        pthread_mutex_unlock(&mutex);
    }
    return NULL;
}

```

Figura 4.4: Código de cálculo de un rango de armónico

La Figura 4.5 muestra la sección del código encargada de hacer el reparto de los  $K$  armónicos a calcular en los  $P$  elementos de proceso disponibles.

```

void reparte_tareas(void)
{
    div_t x;
    int *cuantos;
    int i;
    datos *datos_envio;
    datos_envio = (datos*)malloc(procesadores*sizeof(datos));
    cuantos = (int*)malloc(procesadores*sizeof(int));
    pthread_t *thread_id;
    thread_id = (pthread_t*)malloc(procesadores*sizeof(pthread_t));
    x = div(armonicos,procesadores);
    for (i=0; i<procesadores; i++)
        cuantos[i] = x.quot;
    for (i=0; i<x.rem; i++)
        cuantos[i] = cuantos[i] + 1;
    datos_envio[0].inicio = 0;
    datos_envio[0].final = datos_envio[0].inicio + cuantos[0];
    for (i=1; i<procesadores; i++)
    {
        datos_envio[i].inicio = datos_envio[i-1].final;
        datos_envio[i].final = datos_envio[i].inicio + cuantos[i];
    }
}

```

Figura 4.5: Código de repartición de los armónicos por cada elemento de proceso

De la Figura 4.5 se aprecia que el número de armónicos a ser calculados por cada elemento de proceso se almacena en el vector `cuantos`. La variable `x` almacena el cociente de la división entre  $K$  y  $P$ . El residuo se distribuye entre los primeros elementos de proceso.

La Figura 4.6 muestra la sección del programa en que se hace el mapeo de las actividades a los  $P$  elementos de proceso. La variable `procesadores` contiene almacena el número de elementos de proceso a ser utilizados.

```

for (i=0; i<procesadores; i++)
    pthread_create(&thread_id[i],NULL,fft,&datos_envio[i]);

```

Figura 4.6: Sección de actividades de procesadores

La Figura 4.7 muestra el proceso en el cual se verifica que todos los elementos de proceso hayan finalizado las tareas que le fueron asignadas.

```

for (i=0; i<procesadores; i++)
    pthread_join(thread_id[i],NULL);

```

Figura 4.7: Código de verificación de la finalización de las tareas

La señal a la cual se le calculará su espectro armónico se obtiene a través de la solución en estado estable de la red eléctrica a ser analizada. Una vez alcanzado el estado estable, el último ciclo de la variable de interés se almacena en un archivo de datos, el cual es el insumo de entrada para el programa desarrollado.

El algoritmo desarrollado e implementado tiene la versatilidad de poder funcionar con señales con un número  $N$  de muestras. El algoritmo permite también calcular un número variable de armónicos. Esta versatilidad es utilizada en los casos que serán presentados en el Capítulo No. 5 de esta tesis, en dónde se probará el algoritmo desarrollado usando diversos valores de muestras de la señal, así como calculado diversas cantidades de armónicos.

### 4.3. Propuesta de paralelización del cálculo del THD

Considérese la expresión matemática en [IEEE Std 1459-2010, 2010] usada para el cálculo del THD, dada por la siguiente ecuación:

$$THD_I = \frac{\sqrt{I_0^2 + \sum_{h \neq 1}^n I_h^2}}{I_1} \quad (4.2)$$

Donde

- $h$  es el orden de armónico
- $I_0$  es la componente de corriente directa
- $I_1$  es la componente fundamental de la corriente
- $I_n$  es la  $n$ -ésima componente de la corriente

De la expresión (4.2) muestra cómo se calcula la THD de una señal con respecto a la fundamental. Una red se compone de varios nodos ligados entre sí para líneas de transmisión en las cuales se transmite flujos de potencia a través los enlaces de corrientes y voltaje. La red siempre tiene más de una señal que además aumenta con el tamaño de la red, de tal modo que se necesita el uso del cómputo en paralelo para el cálculo de la THD de un conjunto de

señales. En la Figura 4.8 se muestra un ejemplo de una red que tiene 40 señales y disponemos de 8 elementos de procesadores. Las señales son variables de estado encontrados en estado estable, de tal modo que se calcula de la THD de señales. Los 40 Variable de estado (VE) se distribuyen en los 8 elementos de proceso por lo cual cada elemento de proceso calcula 5 VE.

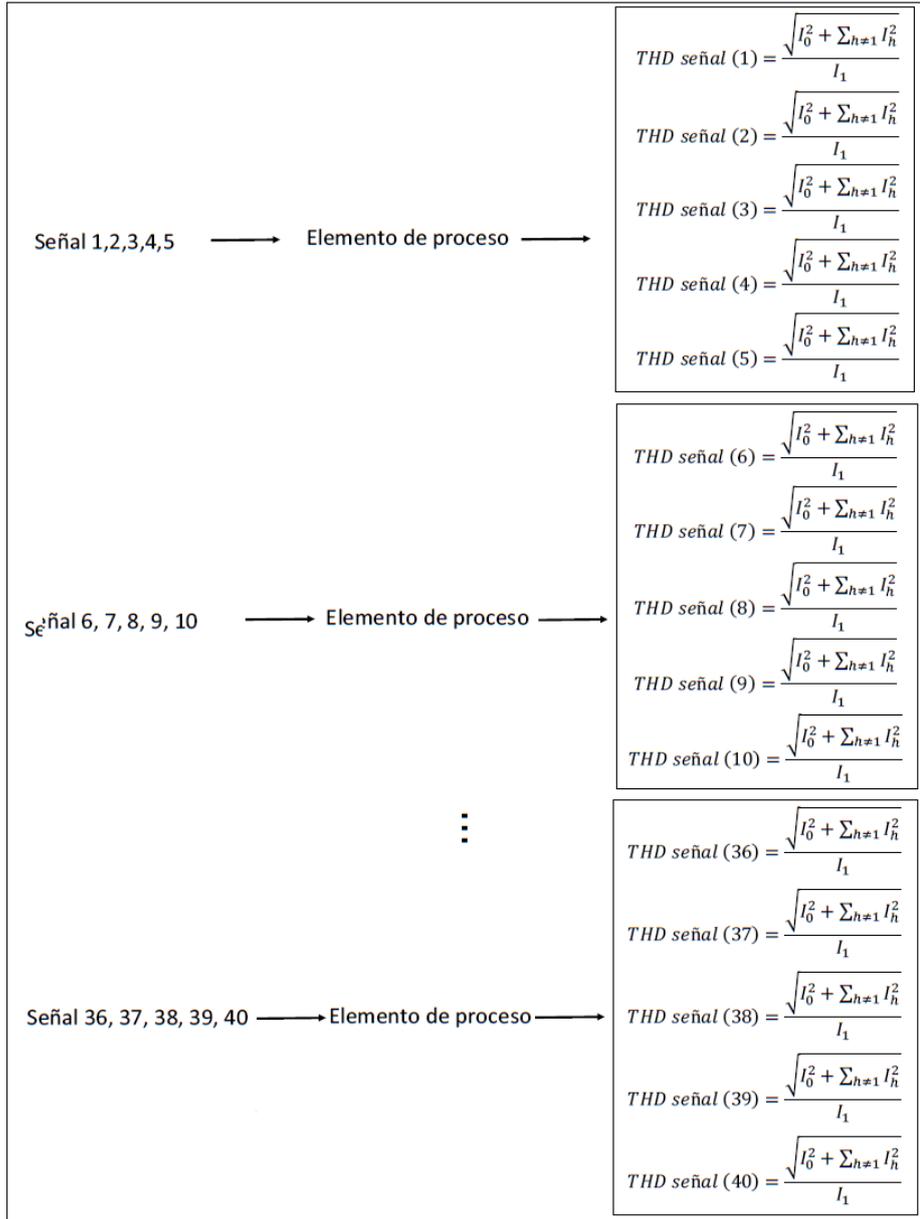


Figura 4.8: Distribución de los armónicos en los 8 procesadores

De la Figura 4.8 se aprecia que como cada elemento de proceso calcula el THD de cada señal de un conjunto de señales. El algoritmo paralelo para el cálculo de la FFT se muestra en la Tabla 4.2. y en la Figura 4.9 se presenta su diagrama de flujo.

Tabla 4.2: Algoritmo de cálculo en paralelo de la THD

1. Inicio
2. Lectura del archivo
3. Establecer el número  $P$  de elementos de cómputo a utilizar
4. Establecer el número  $N$  de muestreo
5. Establecer el número  $n$  de armónico que desea calcular
6. Calcular el número  $m$  de conjunto de señal a ser calculados por cada elemento de cómputo
7. Realizar el mapeo a cada elemento de proceso
8. Esperar los resultados por parte de cada elemento de proceso
9. Poner a disposición los resultados obtenidos
10. Fin

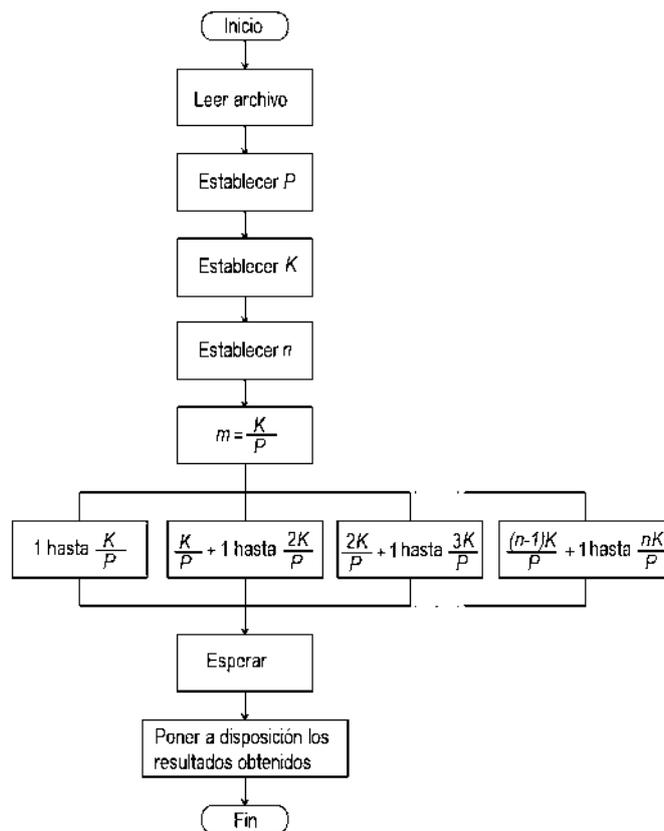


Figura 4.9: Diagrama de flujo del THD

Los resultados son colocados en un vector de tamaño  $K$ , el cual se encuentra en la memoria compartida. Con el objeto de que, en un mismo instante de tiempo, más de un hilo no pueda acceder al vector en el cual se almacenan los  $K$  armónicos, de los que se descompone la señal, se ha utilizado un esquema de variables de exclusión (`mutex_lock`). La Figura 4.10 muestra la función que realiza el cálculo de THD con base a un rango de armónicos.

```

void *fft(void *args)
{
    int ini,fin;
    int i,j,k,ii;
    float fund;
    int tecla;
    complex double suma,valor;
    float sumatoria;
    float thd;
    complex double *res;
    datos *mis_datos=(datos*)args;
    ini = mis_datos->inicio;
    fin = mis_datos->final;
    res = new complex double[40];
    for (i=ini; i<fin; i++)
    {
        for (k=0; k<40; k++)
        {
            real(suma) = 0.0;
            imag(suma) = 0.0;
            for (j=0; j<512; j++)
            {
                real(valor) = 0.0;
                imag(valor) = -2.0*pi*k*j/512;
                suma = suma + matriz[j][i+1]*exponente(valor);
            }
            res[k] = suma;
        }
        fund = absoluto(res[1]);
        sumatoria = 0.0;
        for (ii=2; ii<40; ii++)
            sumatoria = sumatoria + pow(absoluto(res[ii]),2.0);
        thd = sqrt(sumatoria)/fund;
        pthread_mutex_lock(&mutex);
        vector_thd[i] = thd;
        pthread_mutex_unlock(&mutex);
    }
    return NULL;
}

```

Figura 4.10: Código de cálculo de la THD a un rango de armónico

En el código se aprecia que la función recibe como parámetros el armónico inicial del rango así como el armónico final del rango de armónicos a ser calculados. Los ciclos anidados contenidos en la función tienen como objeto, el primero, calcular el  $k$  –ésimo armónico del rango, en tanto que el ciclo interno, calcula el armónico  $k$ . Una vez que es calculado el armónico  $k$ , este se almacena en el vector `res`. Para almacenar el armónico en el vector `res` se utiliza un esquema de variables de exclusión para evitar que más de un elemento de proceso pueda acceder al mismo. Debe de recordarse que la función presentada en la tabla anterior se ejecuta de manera simultánea en los  $P$  elementos de proceso disponibles.

Cuando dos elementos de proceso han realizado el cálculo de un armónico, y los dos quieren modificar el vector `res`, se establece que el primero que termina bloquea al vector `res` para su modificación, por lo tanto, el segundo elemento de proceso deberá esperar.

La Figura 4.11 muestra la sección del código encargada de hacer el reparto de los  $K$  armónicos a calcular en los  $P$  elementos de proceso disponibles.

```
void reparte_tareas(void)
{
    div_t x;
    int *cuantos;
    int i;
    int tecla;
    datos *datos_envio;
    datos_envio = (datos*)malloc(procesadores*sizeof(datos));
    cuantos = (int*)malloc(procesadores*sizeof(int));
    pthread_t *thread_id;
    thread_id = (pthread_t*)malloc(procesadores*sizeof(pthread_t));
    x = div(ve,procesadores);
    for (i=0; i<procesadores; i++)
        cuantos[i] = x.quot;
    for (i=0; i<x.rem; i++)
        cuantos[i] = cuantos[i] + 1;
    datos_envio[0].inicio = 0;
    datos_envio[0].final = datos_envio[0].inicio + cuantos[0];
    for (i=1; i<procesadores; i++)
    {
        datos_envio[i].inicio = datos_envio[i-1].final;
        datos_envio[i].final = datos_envio[i].inicio + cuantos[i];
    }
}
```

Figura 4.11: Código de cálculo de un rango de armónico

De la Figura 4.11 se aprecia que el número de armónicos a ser calculados por cada elemento de proceso se almacena en el vector `cuantos`. La variable  $x$  almacena el cociente de la división entre  $K$  y  $P$ . El residuo se distribuye entre los primeros elementos de proceso.

La Figura 4.12 muestra la sección del programa en que se hace el mapeo de las actividades a los  $P$  elementos de proceso. La variable `procesadores` contiene almacena el número de elementos de proceso a ser utilizados.

```
for (i=0; i<procesadores; i++)
    pthread_create(&thread_id[i],NULL,fft,&datos_envio[i]);
```

Figura 4.12: Sección de actividades de procesadores

La Figura 4.13 muestra el proceso en el cual se verifica que todos los elementos de proceso hayan finalizado las tareas que le fueron asignadas.

```
for (i=0; i<procesadores; i++)
    pthread_join(thread_id[i],NULL);
```

Figura 4.13: Código de verificación de la finalización de las tareas

El cálculo de la THD se hace por el medio de los archivos de datos obtenido en el estado estable lo cual se utiliza para el cómputo en paralelo en donde cada elemento de proceso calcula un conjunto de señal (VE). El algoritmo desarrollado e implementado se va a usar en el Capítulo No. 5 de esta tesis, en dónde se probará el algoritmo desarrollado por ciertos tipos de redes donde tenemos los archivos indicados el número de VE.

#### 4.4. Propuesta de paralelización del cálculo de la DPI de una red eléctrica

La respuesta a la frecuencia de una red eléctrica de potencia en CA se puede encontrar a través de la construcción, a una frecuencia específica, de la matriz de admitancias. De punto de vista de análisis la función de transferencia de una red eléctrica es su matriz de admitancias relacionada con las corrientes nodales y los voltajes nodales que aparecen a niveles de los nodos de la red, se representa, la ecuación matricial a cualquier frecuencia  $f$  con base a la ecuación (4.3):

$$\tilde{I}_f = Y_f \tilde{V}_f \quad (4.3)$$

A partir de la ecuación (4.3) se encuentra la matriz de admitancias  $Y_f$  donde su inversa se conoce como la matriz de impedancias  $Z_f$  que se puede expresar a detalle a través de este medio:

$$Z_f = \begin{pmatrix} Z_{1,1} & Z_{1,2} & \cdots & Z_{1,n} \\ Z_{2,1} & Z_{2,2} & \cdots & Z_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{n,1} & Z_{n,2} & \cdots & Z_{n,n} \end{pmatrix} \quad (4.4)$$

En donde cada elemento de la diagonal  $Z_{j,j}$  representa la impedancia del nodo  $j$  de la red. El conjunto de puntos  $Z_{j,j}$  para diferentes frecuencias representa la respuesta de la impedancia del nodo  $j$  de la red a la frecuencia. Este conjunto de puntos se conoce como Respuesta a la frecuencia de la impedancia (DPI) del nodo  $j$  [Ramos-Paz, 2007].

La combinación de elementos capacitivos e inductivos vistos, desde un nodo en particular del sistema, puede resultar en resonancias en serie o en paralelo. Las resonancias serie están asociadas con un elevado flujo de corrientes armónicas a través de ciertos elementos y nodos en la red, en tanto que el resultado de las resonancias en paralelo puede ser la presencia de voltajes armónicos excesivos a través de los elementos y nodos de la red [Ramos-Paz, 2007].

Los elementos en la diagonal ( $Z_{j,j}$ ) de la matriz de impedancia  $Z_f$  son ellos que define la DPI al nodo  $j$ . Se puede calcular la DPI con un solo procesador cada matriz  $Z_f$  para cada frecuencia una después de otra (Cálculo secuencial). Debido a que los cálculos de matrices  $Z_f$  para frecuencias  $f_0, f_1, f_2, \dots, f_n$  son independientes y por razón de cálculo más preciso de  $Z_{j,j}$ , se necesita un paso de frecuencia muy pequeño con la finalidad de obtener una reproducción precisa de las resonancias en serie y paralelo del sistema. Por lo tanto, el paso de frecuencia más pequeña es y más requiera esfuerzo computacional para obtener  $Z_{j,j}$  donde la necesidad del cómputo en paralelo sería la mejor alternativa [Ramos-Paz, 2007]. En Figura 4.14 se muestra como cada fragmento de frecuencias está distribuyendo a cada elemento de proceso.

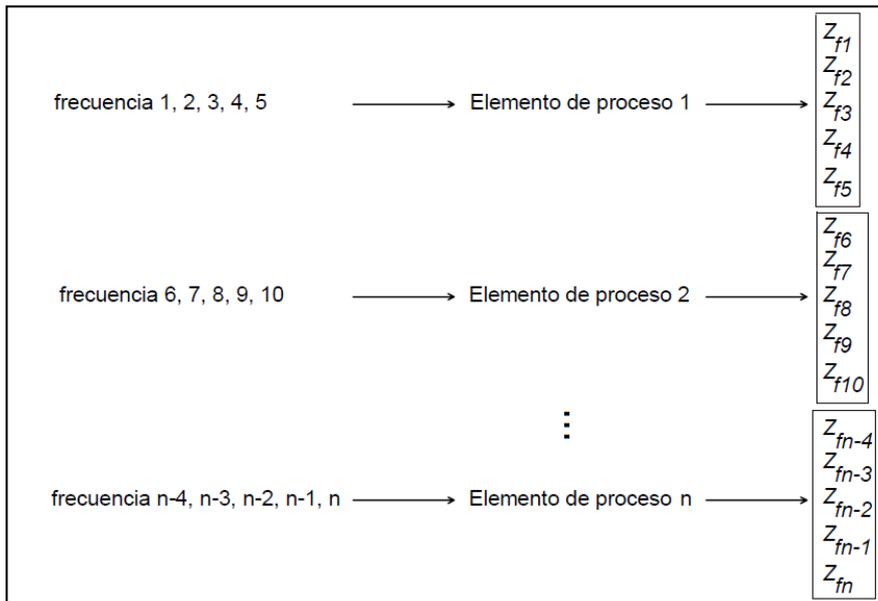


Figura 4.14: Distribución de las frecuencia en  $P$  procesadores

En la Figura 4.14 se muestra como cada conjunto de frecuencia está repartida a cada elemento de proceso para calcular la matriz  $Z_f$  donde los elementos en la diagonal de esta matriz son los puntos de la respuesta a frecuencia de la red a esta frecuencia específica. El algoritmo de cálculo de la DPI está en la Tabla 4.3 y en la Figura 4.15 se presenta diagrama de flujo.

Tabla 4.3: Algoritmo de cálculo en paralelo de la DPI

1. Inicio
2. Lectura del archivo
3. Establecer el número  $P$  de elementos de cómputo a utilizar
4. Establecer el paso de frecuencia
5. Calcular el fragmento del vector de frecuencia por cada elemento de cómputo
6. Realizar el mapeo a cada elemento de proceso
7. Calcular la matriz de impedancia
8. Esperar los resultados por parte de cada elemento de proceso
9. Poner a disposición los resultados obtenidos
10. Fin

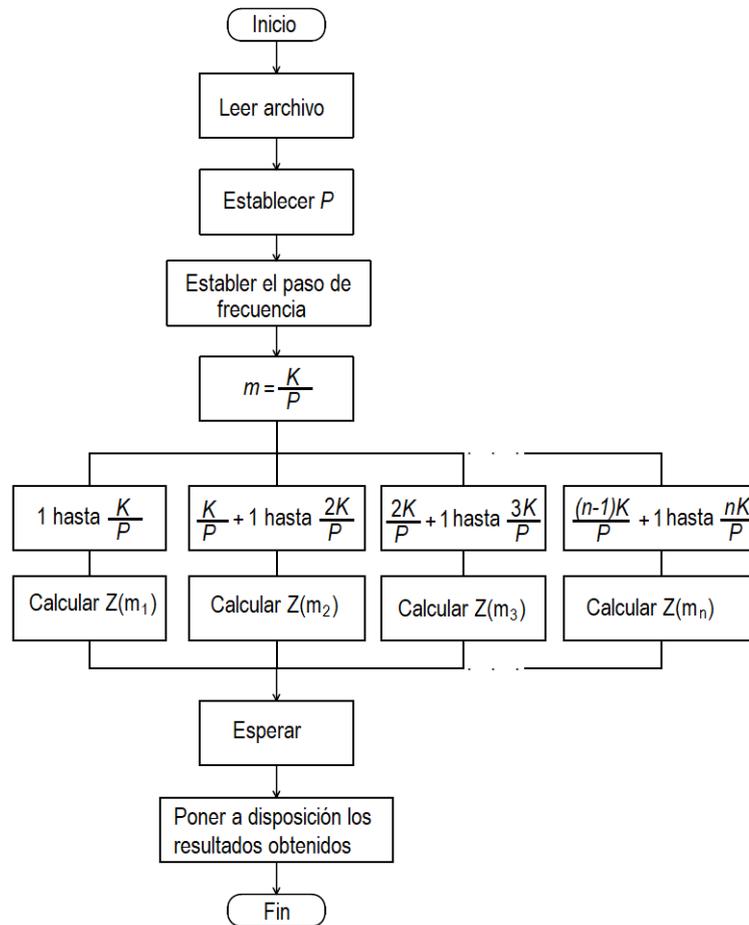


Figura 4.15: Diagrama de flujo del DPI

Los resultados son colocados en un vector *resultados*, el cual se encuentra en la memoria compartida. Con el objeto de que, en un mismo instante de tiempo, más de un hilo no pueda acceder al vector en el cual se almacenan las matrices de impedancias, donde en los diagonales se ubican los puntos de DPI, se ha utilizado un esquema de variables de exclusión (*mutex\_lock*). La Figura 4.16 muestra la función que realiza el cálculo de la DPI con proceso de inversión matricial.

En el código se aprecia que la función recibe como parámetros  $r$ ,  $l$ ,  $c$ , de la red a analizar que permite construir la matriz de admitancia. Después de tener la matriz de admitancia se calcula la matriz inversa del *ybus* llamado matriz de impedancia. Una vez

que se calcula la matriz de impedancias  $k$ , esta se almacena en el vector *resultados*. Para almacenar las matrices de impedancia en el vector *resultados* se utiliza un esquema de variables de exclusión para evitar que más de un elemento de proceso pueda acceder al mismo. Debe de recordarse que la función presentada en la Figura 4.16 se ejecuta de manera simultánea en los  $P$  elementos de proceso disponibles. Cuando dos elementos de proceso han realizado el cálculo de una matriz de impedancias, y los dos quieren modificar el vector *resultados*, se establece que el primero que termina bloquea al vector *resultados* para su modificación, por lo tanto el segundo elemento de proceso deberá esperar.

```
// proceso de inversión matricial
for {k=0; k<orden; k++}
{
    pivote = ybus[k*orden+k];
    for (i=0; i<orden; i++)
        for (j=0; j<orden; j++)
            if ((j!=k) && (i!=k))
                ybus[i*orden+j]=ybus[i*orden+j]
                    -ybus[i*orden+k]*ybus[k*orden+j]/ybus[k*orden+k];
    for (i=0; i<orden; i++)
    {
        ybus[i*orden+k] = (-1*ybus[i*orden+k])/pivote;
        ybus[k*orden+i] = (-1*ybus[k*orden+i])/pivote;
    }
    ybus[k*orden+k] = -1/pivote;
}
for (i=0; i<orden; i++)
    for (j=0; j<orden; j++)
        ybus[i*orden+j] = -1*ybus[i*orden+j];

// proceso de almacenamiento de resultados
pthread_mutex_lock(&sum_mutex);
inicio = int((referencial/h-1)*orden*orden);
for (i=0; i<orden; i++)
    for (j=0; j<orden; j++)
        resultados[inicio*orden+i+j]=sqrt((__real__ ybus[i*orden+j])*(__real__ ybus[i*orden+j])
            +(__imag__ ybus[i*orden+j])*(__imag__ ybus[i*orden+j]));
pthread_mutex_unlock(&sum_mutex);
contador++;
referencial = referencial + h;
```

Figura 4.16: Código de cálculo de la DPI en rango de frecuencia

La Figura 4.17 muestra la sección del código encargada de hacer el reparto de los  $K$  frecuencia a calcular en los  $P$  elementos de proceso disponibles.

```

void calcula_envios()
{
    div_t x;
    int num_tareas_hilo, i;
    double inicio;
    pthread_attr_t attr;
    pthread_mutex_init(&sum_mutex, NULL);
    pthread_attr_init(&attr);
    pthread_t *thread_id;
    thread_arg_t *thread_arg;
    thread_id = new pthread_t[hilos];
    thread_arg = new thread_arg_t[hilos];
    tareas = new double [hilos];
    x = div(f_max, hilos);
    num_tareas_hilo = x.quot;
    for (i=0; i<hilos; i++)
        tareas[i] = x.quot;
    for (i=0; i<x.rem; i++)
        tareas[i] = tareas[i]+1;
    inicio = 0;
    for (i=0; i<hilos; i++)
    {
        thread_arg[i].principio = inicio;
        thread_arg[i].frecuencias = tareas[i];
        thread_arg[i].h = h;
        inicio = inicio + tareas[i];
    }
}

```

Figura 4.17: Código de repartición de rango de frecuencia a cada procesadores

De la Figura 4.17 se aprecia que el número de matrices a ser calculados por cada elemento de proceso se almacena en el vector cuantos. La variable  $x$  almacena el cociente de la división entre  $K$  y  $P$ . El residuo se distribuye entre los primeros elementos de proceso.

La Figura 4.18 muestra la sección del programa en que se hace el mapeo de las actividades a los  $P$  elementos de proceso. La variable procesadores contiene el número de elementos de proceso a ser utilizados.

```

for (i=0; i<hilos; i++)
    pthread_create(&thread_id[i], NULL, forma_e_inversa, &thread_arg[i]);

```

Figura 4.18: Sección de actividades de procesadores

La Figura 4.19 muestra el proceso en el cual se verifica que todos los elementos de proceso hayan finalizado las tareas que le fueron asignadas.

```
pthread_join(thread_id[i],NULL);  
pthread_mutex_destroy(&sum_mutex);
```

Figura 4.19: Código de verificación de la finalización de las tareas

El cálculo de la DPI se hace por el medio de los archivos de datos que presenta las características de redes lo cual se utiliza para el cómputo en paralelo en donde cada elemento de proceso calcula un fragmento de frecuencia para la determinación de la DPI. El algoritmo desarrollado e implementado se va a usar en el Capítulo No. 5, en dónde se verifica el algoritmo desarrollado.

## 4.5. Conclusiones

En el presente Capítulo se han presentado las tres propuestas de paralelización de algoritmos de tres métricas usadas en el análisis armónico para el cálculo de la FFT, el THD y la DPI.

La propuesta de cálculo de FFT en paralelo presentada en este capítulo se realiza de manera que el número de armónicos se distribuye entre los elementos de proceso disponibles. El proceso se realiza de tal manera que cada elemento del proceso calcula un conjunto de armónicos asignados a él. Cada conjunto de armónicos se distribuye según una división entre el número de armónicos y los elementos del proceso mediante una división exacta o, por el contrario, el resto de esta división se distribuye entre los primeros elementos del proceso.

Para el caso de la THD, se propone un algoritmo que permite calcular el THD de diferentes Variable de estado (VE) que están almacenadas en un archivo donde la solución fue encontrada en estado estable. La propuesta es como tal cada elemento de proceso calcula un conjunto de VE repartida de manera uniforme a cada elemento de proceso disponible por el cálculo.

En el caso de la DPI, se presentó una propuesta de cálculo en paralelo lo cual se trata de la fragmentación de un vector de frecuencia que permite calcular la matriz de impedancia de la red a estudiar por la técnica de inversión de la matriz de admitancia en donde cada matriz de impedancia tiene una frecuencia propia donde en la diagonal se ubican los DPI. En razón de un paso de frecuencia muy pequeña que ocasiona un esfuerzo computacional elevado, se implementó un algoritmo en que cada elemento de proceso calcula un rango de frecuencias asignadas a él.



## Capítulo 5

# Casos de Estudio

### 5.1. Introducción

En este capítulo se presentan diversos casos de estudio, en los cuales se aplican los algoritmos paralelizados, propuestos en esta tesis, para el cálculo de la transformada rápida de Fourier, la distorsión armónica total y la respuesta a la frecuencia de un sistema eléctrico. Los casos de estudio nos demuestran la ventaja del uso del cómputo en paralelo.

### 5.2. Características de la computadora utilizada para el cómputo en paralelo

La computadora utilizada en ejecución de los casos de estudio presentados en esta tesis es un servidor HP, el cual cuenta con 2 procesadores Intel Quad core con una velocidad de 2004.174 MHz. En las Figuras 5.1 se muestra una vista frontal del equipo de cómputo utilizado para la ejecución de los diversos casos de estudio presentados en esta tesis, en tanto que en la Figura 5.2 se muestra el interior del servidor.



Figura 5.1: Presentación de la computadora

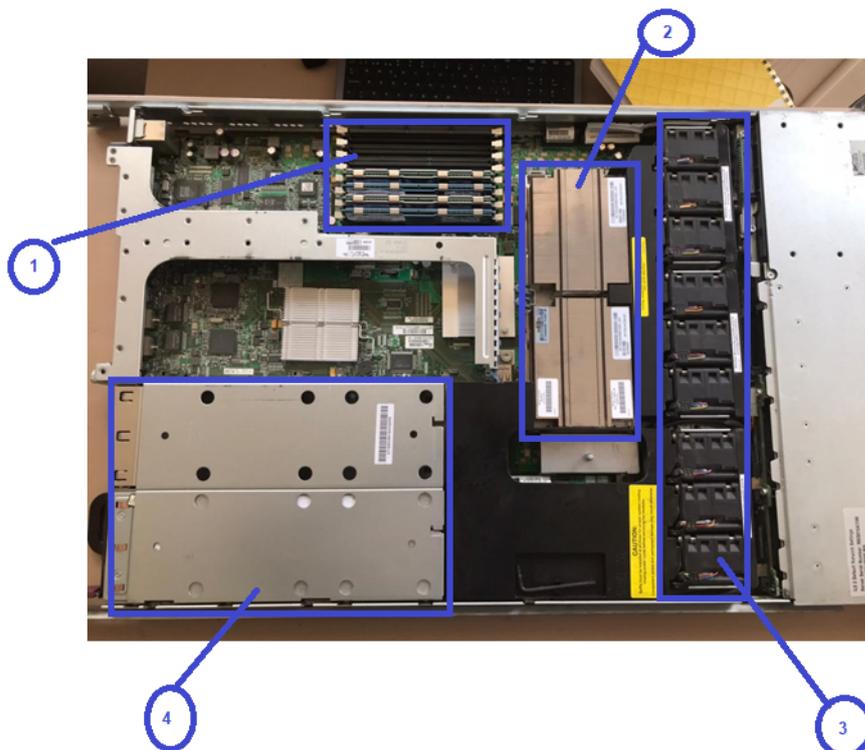


Figura 5.2: Presentación interna de la Computadora

En la Figura 5.2 se observan los siguientes elementos:

- 1 Memoria del servidor.
- 2 Procesadores del servidor
- 3 Ventiladores.
- 4 Sistema de refrigeración.

Todos los tiempos de cómputo presentados en este capítulo se obtienen a partir de 10 mediciones, en las cuales se eliminan los tiempos más pequeños y más altos y se promedian los 8 tiempos restantes. Este tratamiento estadístico de las mediciones se realiza con el objeto de no tomar en cuenta los tiempos en los cuales el sistema pudo haber generado una interrupción a la ejecución de los casos de estudio.

### **5.3. Caso de Estudio No. 1: Cálculo en paralelo de la FFT**

En este estudio se realiza el cálculo en paralelo de la FFT de una VE tomada de la red de IEEE de 14 nodos que ha sido modificada con la incorporación de cargas no lineales, en donde se incluyó una rama magnetizante en los nodos 10, 11, 13 y 14. El algoritmo propuesto será ejecutado usando 1, 2, hasta 8 elementos de proceso, con el objeto de demostrar el efecto de disminución de tiempo con el uso de un número mayor de elementos de proceso. En la Figura 5.3 se muestra la señal de voltaje en el nodo 10, la cual está distorsionada debido a la presencia de armónicos. Se observa que el tercer armónico tiene una magnitud de 14.5 % con respecto al voltaje fundamental, lo que no es aceptable con respecto a lo establecido por el estándar IEEE 519 de 2014 ver Tabla la 2.3.

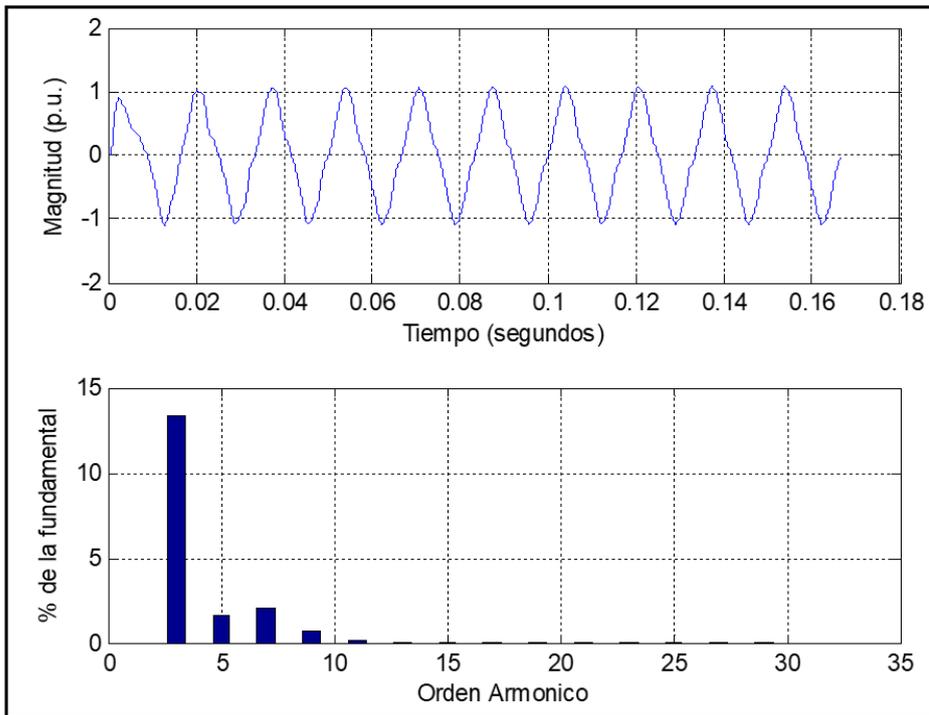


Figura 5.3: Voltaje en el nodo 10

A través del sistema desarrollado por [Ramos-Paz, 2007], se encuentra el modelo matemático del sistema de prueba del IEEE de 14 nodos. Una vez encontrado el modelo matemático se determina el estado estacionario periódico del mismo. A partir de la forma de onda, de cada variable de estado, tomada en el estado estable, se calcula la FFT de manera paralela. En este caso de estudio se considera el cálculo de los primeros 40 y 80 armónicos, además se hace un muestreo de la señal de la VE con; 512, 1024, 2048, 4096, 8192 y 16384 puntos por periodo, lo cual implica un aumento en el esfuerzo computacional en el proceso del cálculo de la FFT.

### 5.3.1. Medición de 40 armónicos

En la Tabla 5.1 se muestra el tiempo de cómputo requerido para el cálculo de la FFT de la VE de estado seleccionada. En esta tabla se presentan los tiempos de cómputo cuando se hace un muestreo con 512, 1024, 2048, 4096, 8192 y 16384 puntos por periodo. En la primera

columna de esta tabla se muestra el número de elementos de proceso utilizados. Se observa que el uso de un solo elemento de proceso produce un cálculo eminentemente secuencial. Con el uso de 2 elementos de proceso se observa que, para 512 puntos, se disminuye el tiempo de cómputo de 8 ms a 6 ms, lo cual representa una disminución del 25 %. Se observa que con el uso de 8 elementos de proceso se alcanza un tiempo de 4 ms, lo cual representa una disminución del tiempo de cómputo de un 50 %.

Tabla 5.1: Resultados de Mediciones de Tiempo [s] de 40 armónicos

| No. de Procesadores | Puntos de muestra por periodo |       |       |       |       |       |
|---------------------|-------------------------------|-------|-------|-------|-------|-------|
|                     | 512                           | 1024  | 2048  | 4096  | 8192  | 16384 |
| 1                   | 0.008                         | 0.013 | 0.022 | 0.040 | 0.077 | 0.150 |
| 2                   | 0.006                         | 0.008 | 0.013 | 0.022 | 0.042 | 0.080 |
| 3                   | 0.005                         | 0.006 | 0.009 | 0.016 | 0.029 | 0.055 |
| 4                   | 0.004                         | 0.005 | 0.008 | 0.014 | 0.024 | 0.047 |
| 5                   | 0.004                         | 0.005 | 0.007 | 0.012 | 0.020 | 0.039 |
| 6                   | 0.004                         | 0.005 | 0.006 | 0.011 | 0.018 | 0.034 |
| 7                   | 0.004                         | 0.005 | 0.006 | 0.010 | 0.017 | 0.030 |
| 8                   | 0.004                         | 0.005 | 0.006 | 0.010 | 0.015 | 0.028 |

En los datos presentados en la Tabla 5.1 se aprecia que conforme se aumenta el número de puntos de muestreo por periodo, se va incrementando el tiempo de cómputo con el uso de un solo elemento de proceso. La aplicación de un mayor número de elementos de proceso va generando una disminución en el tiempo de cómputo. Se aprecia que para el uso de 16384 puntos por periodo y 8 elementos de proceso se reduce el tiempo de cómputo en un 81.33 %.

En la Tabla 5.2 se muestra el *speed-up* obtenido a través del uso del cómputo en paralelo. Se aprecia que conforme aumenta el tamaño del problema y aumenta el número de elementos de proceso, se incrementa el *speed-up*, alcanzándose un *speed-up* máximo de 5.399 para el caso de una muestra de 16384 puntos por periodo y utilizando 8 elementos de proceso es 5.399 veces más rápido que el programa secuencial, es decir, el programa que usa un solo elemento de proceso.

En la Figura 5.4 se muestra el comportamiento del tiempo de cómputo conforme

Tabla 5.2: Resultados de Cálculo del *speed-up* de 40 armónicos

| No. de Procesadores | Puntos de muestra por periodo |       |       |       |       |       |
|---------------------|-------------------------------|-------|-------|-------|-------|-------|
|                     | 512                           | 1024  | 2048  | 4096  | 8192  | 16384 |
| 1                   | 1.000                         | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| 2                   | 1.333                         | 1.625 | 1.692 | 1.818 | 1.833 | 1.889 |
| 3                   | 1.600                         | 2.167 | 2.444 | 2.500 | 2.655 | 2.720 |
| 4                   | 2.000                         | 2.600 | 2.750 | 2.857 | 3.208 | 3.200 |
| 5                   | 2.000                         | 2.600 | 3.143 | 3.333 | 3.850 | 3.842 |
| 6                   | 2.000                         | 2.600 | 3.667 | 3.636 | 4.278 | 4.450 |
| 7                   | 2.000                         | 2.600 | 3.667 | 4.000 | 4.529 | 5.077 |
| 8                   | 2.000                         | 2.600 | 3.667 | 4.000 | 5.133 | 5.399 |

se incrementa el número de elementos de proceso. Se observa que conforme aumenta el número de elementos de proceso se disminuye el tiempo de cómputo, siendo notorio que el mayor decremento en el tiempo de cómputo se da cuando se pasa del uso de un elemento de cómputo a dos elementos de cómputo. De esta misma gráfica se aprecia que para los casos de estudio de 512, 1024, 2048 y 4096 puntos por periodo, no existe un decremento en la velocidad con el uso de 6 o más elementos de proceso. Esto indica que se alcanzado el tiempo secuencial del algoritmo, es decir, la parte del mismo que no puede ser paraleliza. El uso adicional de elementos de proceso, en el mejor de los casos no reducirá el tiempo de cómputo, en el escenario más adverso incrementará el tiempo de cómputo.

En la Figura 5.5 se muestra una gráfica en que relaciona el número de elementos de cómputo con el *speed-up* obtenido en los diversos casos de estudio. Se aprecia que conforme aumenta el tamaño del problema y el número de elemento de proceso se alcanza un valor máximo del *speed-up* para el caso de 512 puntos por periodo, el *speed-up* máximo alcanzado es de 2.0 en tanto que para el caso de 16384 puntos por periodo el *speed-up* máximo alcanzado es cercano a 5.4. De aquí se puede concluir que el algoritmo propuesto genera mejores resultados para problemas de mayores dimensiones. Sin embargo, para el caso de problemas de menor dimensión, genera una disminución considerable en el tiempo de cómputo.

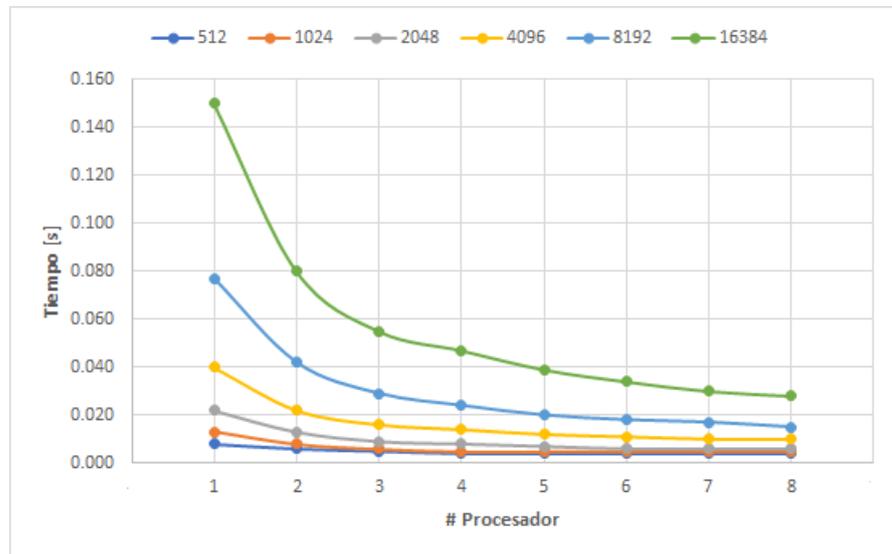
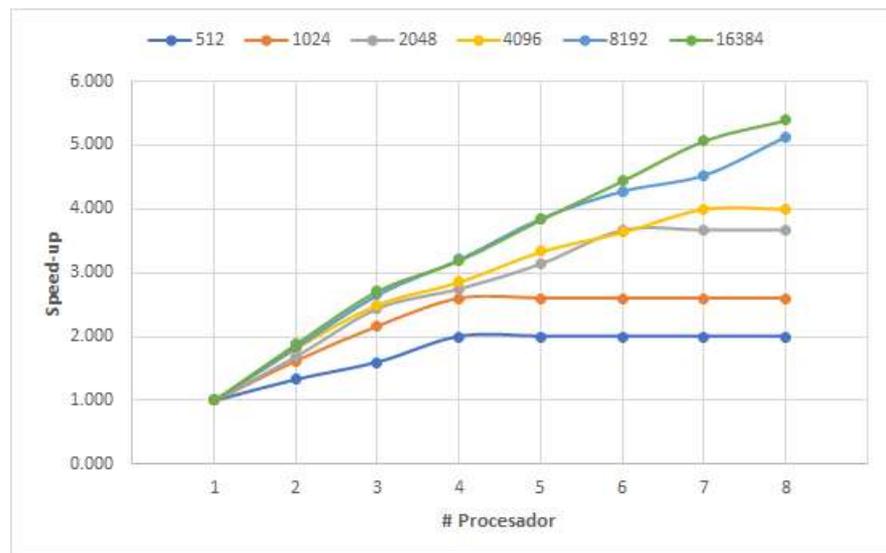


Figura 5.4: Gráfica de Tiempo de la Paralelización de la FFT con 40 armónicos

Figura 5.5: Gráfica del *Speed-up* de la Paralelización de la FFT con 40 armónicos

### 5.3.2. Medición de 80 armónicos

En la Tabla 5.3 se muestra el tiempo de cómputo requerido para el cálculo de la FFT de la VE de estado seleccionada. En esta tabla se presentan los tiempos de cómputo cuando se

hace un muestreo con 512, 1024, 2048, 4096, 8192 y 16384 puntos por periodo. En la primera columna de esta tabla se muestra el número de elementos de proceso utilizados. Se observa que el uso de un solo elemento de proceso produce un cálculo eminentemente secuencial. Con el uso de 2 elementos de proceso se observa que, para 512 puntos, se disminuye el tiempo de cómputo de 13 ms a 8 ms, lo cual representa una disminución del 38%. Se observa que con el uso de 8 elementos de proceso se alcanza un tiempo de 5 ms, lo cual representa una disminución del tiempo de cómputo de un 61.5%.

Tabla 5.3: Resultados de Mediciones de Tiempo [s] de 80 armónicos

| No. de Procesadores | Puntos de muestra por periodo |       |       |       |       |       |
|---------------------|-------------------------------|-------|-------|-------|-------|-------|
|                     | 512                           | 1024  | 2048  | 4096  | 8192  | 16384 |
| 1                   | 0.013                         | 0.022 | 0.042 | 0.078 | 0.152 | 0.299 |
| 2                   | 0.008                         | 0.013 | 0.023 | 0.041 | 0.079 | 0.154 |
| 3                   | 0.006                         | 0.010 | 0.016 | 0.029 | 0.055 | 0.106 |
| 4                   | 0.005                         | 0.008 | 0.014 | 0.023 | 0.044 | 0.083 |
| 5                   | 0.005                         | 0.007 | 0.012 | 0.020 | 0.038 | 0.071 |
| 6                   | 0.005                         | 0.007 | 0.010 | 0.018 | 0.031 | 0.059 |
| 7                   | 0.005                         | 0.007 | 0.010 | 0.018 | 0.028 | 0.055 |
| 8                   | 0.005                         | 0.007 | 0.010 | 0.016 | 0.024 | 0.053 |

En los datos presentados en la Tabla 5.3 se aprecia que conforme se aumenta el número de puntos de muestreo por periodo, se va incrementando el tiempo de cómputo con el uso de un solo elemento de proceso. La aplicación de un mayor número de elementos de proceso va generando una disminución en el tiempo de cómputo. Se aprecia que para el uso de 16384 puntos por periodo y 8 elementos de proceso se reduce el tiempo de cómputo en un 82.3%.

En la Tabla 5.4 se muestra el *speed-up* obtenido a través del uso del cómputo en paralelo. Se aprecia que conforme aumenta el tamaño del problema y aumenta el número de elementos de proceso, se incrementa el *speed-up*, alcanzándose un *speed-up* máximo de 6.333 para el caso de una muestra de 8192 puntos por periodo y utilizando 8 elementos de proceso es 6.333 veces más rápido que el programa secuencial, es decir, el programa que usa un solo elemento de proceso.

Tabla 5.4: Resultados de Cálculo del *speed-up* de 80 armónicos

| No. de Procesadores | Puntos de muestra por periodo |       |       |       |       |       |
|---------------------|-------------------------------|-------|-------|-------|-------|-------|
|                     | 512                           | 1024  | 2048  | 4096  | 8192  | 16384 |
| 1                   | 1.000                         | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| 2                   | 1.625                         | 1.692 | 1.826 | 1.886 | 1.924 | 1.942 |
| 3                   | 2.167                         | 2.200 | 2.625 | 2.693 | 2.764 | 2.821 |
| 4                   | 2.600                         | 2.750 | 3.000 | 3.352 | 3.455 | 3.602 |
| 5                   | 2.600                         | 3.143 | 3.500 | 3.905 | 4.000 | 4.211 |
| 6                   | 2.600                         | 3.143 | 4.200 | 4.399 | 4.903 | 5.068 |
| 7                   | 2.600                         | 3.143 | 4.200 | 4.400 | 5.429 | 5.436 |
| 8                   | 2.600                         | 3.143 | 4.200 | 5.016 | 6.333 | 5.642 |

En la Figura 5.6 se muestra el comportamiento del tiempo de cómputo conforme se incrementa el número de elementos de proceso. Se observa que conforme aumenta el número de elementos de proceso se disminuye el tiempo de cómputo, siendo notorio que el mayor decremento en el tiempo de cómputo se da cuando se pasa del uso de un elemento de cómputo a dos elementos de cómputo. De esta misma gráfica se aprecia que para los casos de estudio de 512, 1024, 2048 y 4096 puntos por periodo, no existe un decremento en la velocidad con el uso de 6 o más elementos de proceso. Esto indica que se alcanzado el tiempo secuencial del algoritmo, es decir, la parte del mismo que no puede ser paraleliza. El uso adicional de elementos de proceso, en el mejor de los casos no reducirá el tiempo de cómputo, en el escenario más adverso incrementará el tiempo de cómputo.

En la Figura 5.7 se muestra una gráfica en que relaciona el número de elementos de cómputo con el *speed-up* obtenido en los diversos casos de estudio. Se aprecia que conforme aumenta el tamaño del problema y el número de elemento de proceso se alcanza un valor máximo del *speed-up* para el caso de 512 puntos por periodo, el *speed-up* máximo alcanzado es de 2.6 en tanto que para el caso de 8192 puntos por periodo el *speed-up* máximo alcanzado es de 6.333. De aquí se puede concluir que el algoritmo propuesto genera mejores resultados para problemas de mayores dimensiones. Sin embargo, para el caso de problemas de menor dimensión, genera una disminución considerable en el tiempo de cómputo. Por el caso de 16384 puntos por periodo es posible obtener un *speed-up* mayor que para los 8192 puntos por periodo si disponemos de más de 8 elementos de proceso.

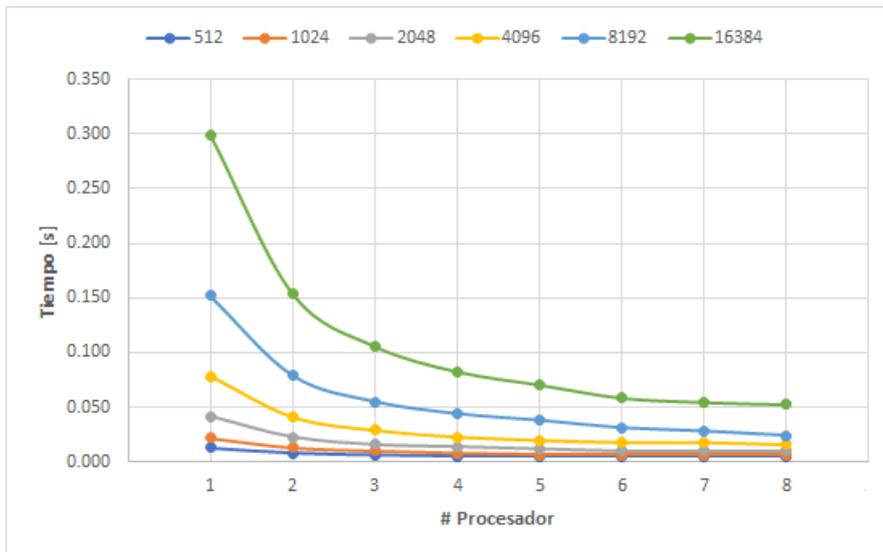


Figura 5.6: Gráfica de Tiempo de la Paralelización de la FFT con 80 armónicos

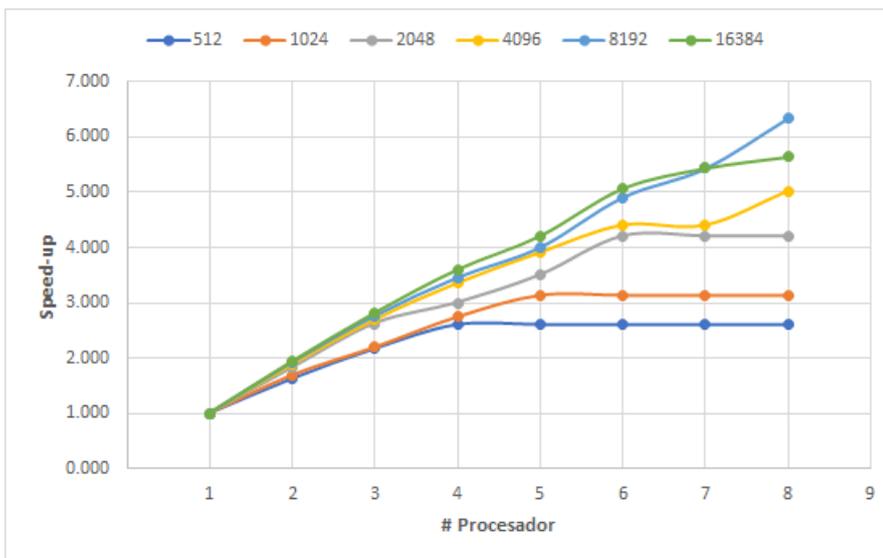


Figura 5.7: Gráfica del *Speed-up* de la Paralelización de la FFT con 80 armónicos

## 5.4. Caso de Estudio No. 2: Cálculo en paralelo de la THD

En este caso de estudio se presenta el cálculo del THD de las variables de estado de los sistemas modificados de IEEE de 14, 30, 57 y 118 nodos. El proceso consiste en determinar el estado estacionario periódico del sistema a ser estudiado, una vez encontrada la respuesta en el estado permanente, se toma el último ciclo de cada una de las VE, con que fue modelado matemáticamente y se procede a calcular de manera simultánea el THD. Para este caso de estudio se consideran para 512 puntos por periodo. El proceso se realiza de tal manera que para cada uno de estos cuatro sistemas se encuentra la solución en estado estable, que nos da un vector de VE conformemente al sistema. De esta solución tenemos un vector de tamaño  $n$  que nos permite distribuir el número de VE en 1, 2, hasta 8 elementos de proceso. Así de este proceso de cómputo medimos el tiempo de ejecución del cómputo.

En la Tabla 5.5 se muestra el tiempo de cómputo requerido para el cálculo del THD de las VE de los 4 sistemas prueba de IEEE de 14, 30, 57 y 118 nodos. En la primera columna de esta tabla se muestra el número de elementos de proceso utilizados. Se observa que el uso de un solo elemento de proceso produce un cálculo eminentemente secuencial. Con el uso de 2 elementos de proceso se observa que, para 14 nodos, se disminuye el tiempo de cómputo de 245 ms a 132 ms, lo cual representa una disminución del 46.1 %. Se observa que con el uso de 8 elementos de proceso se alcanza un tiempo de 55 ms, lo cual representa una disminución del tiempo de cómputo de un 77.6 %.

Tabla 5.5: Resultados de Mediciones de Tiempo [s] cálculo del THD

| No. de Procesadores | Sistemas de prueba de IEEE |          |          |           |
|---------------------|----------------------------|----------|----------|-----------|
|                     | 14 nodos                   | 30 nodos | 57 nodos | 118 nodos |
| 1                   | 0.245                      | 0.399    | 0.740    | 1.833     |
| 2                   | 0.132                      | 0.214    | 0.393    | 0.969     |
| 3                   | 0.094                      | 0.151    | 0.276    | 0.680     |
| 4                   | 0.076                      | 0.122    | 0.218    | 0.535     |
| 5                   | 0.065                      | 0.102    | 0.184    | 0.447     |
| 6                   | 0.059                      | 0.089    | 0.160    | 0.392     |
| 7                   | 0.058                      | 0.083    | 0.145    | 0.349     |
| 8                   | 0.055                      | 0.077    | 0.139    | 0.325     |

En los datos presentados en la Tabla 5.5 se aprecia que conforme se aumenta el número de VE, se va incrementando el tiempo de cómputo con el uso de un solo elemento de proceso. La aplicación de un mayor número de elementos de proceso va generando una disminución en el tiempo de cómputo. Se aprecia que para el caso de 118 nodos y 8 elementos de proceso se reduce el tiempo de cómputo en un 82.3 %.

En la Tabla 5.6 se muestra el *speed-up* obtenido a través del uso del cómputo en paralelo. Se aprecia que conforme aumenta el tamaño del problema y aumenta el número de elementos de proceso, se incrementa el *speed-up*, alcanzándose un *speed-up* máximo de 5.634 para el caso de un sistema de 118 nodos y utilizando 8 elementos de proceso es 5.634 veces más rápido que el programa secuencial, es decir, el programa que usa un solo elemento de proceso.

Tabla 5.6: Resultados de Cálculo del *speed-up* del THD

| No. de Procesadores | Sistemas de prueba de IEEE |          |          |           |
|---------------------|----------------------------|----------|----------|-----------|
|                     | 14 nodos                   | 30 nodos | 57 nodos | 118 nodos |
| 1                   | 1.000                      | 1.000    | 1.000    | 1.000     |
| 2                   | 1.855                      | 1.866    | 1.883    | 1.892     |
| 3                   | 2.604                      | 2.642    | 2.678    | 2.694     |
| 4                   | 3.236                      | 3.273    | 3.391    | 3.429     |
| 5                   | 3.769                      | 3.915    | 4.014    | 4.100     |
| 6                   | 4.135                      | 4.486    | 4.637    | 4.673     |
| 7                   | 4.193                      | 4.811    | 5.093    | 5.252     |
| 8                   | 4.432                      | 5.182    | 3.325    | 5.634     |

En la Figura 5.8 se muestra el comportamiento del tiempo de cómputo conforme se incrementa el número de elementos de proceso. Se observa que conforme aumenta el número de elementos de proceso se disminuye el tiempo de cómputo, siendo notorio que el mayor decremento en el tiempo de cómputo se da cuando se pasa del uso de un elemento de cómputo a dos elementos de cómputo.

En la Figura 5.9 se muestra una gráfica que relaciona el número de elementos de cómputo con el *speed-up* obtenido en los diversos casos de estudio. Se aprecia que conforme aumenta el tamaño del problema y el número de elemento de proceso se alcanza un valor

máximo del *speed-up* para el caso de 14 nodos, el *speed-up* máximo alcanzado es de 4.432 en tanto que para el caso de 118 nodos el *speed-up* máximo alcanzado es de 5.634. De aquí se puede concluir que el algoritmo propuesto genera mejores resultados para problemas de mayores dimensiones. Sin embargo, para el caso de problemas de menor dimensión, genera una disminución considerable en el tiempo de cómputo.

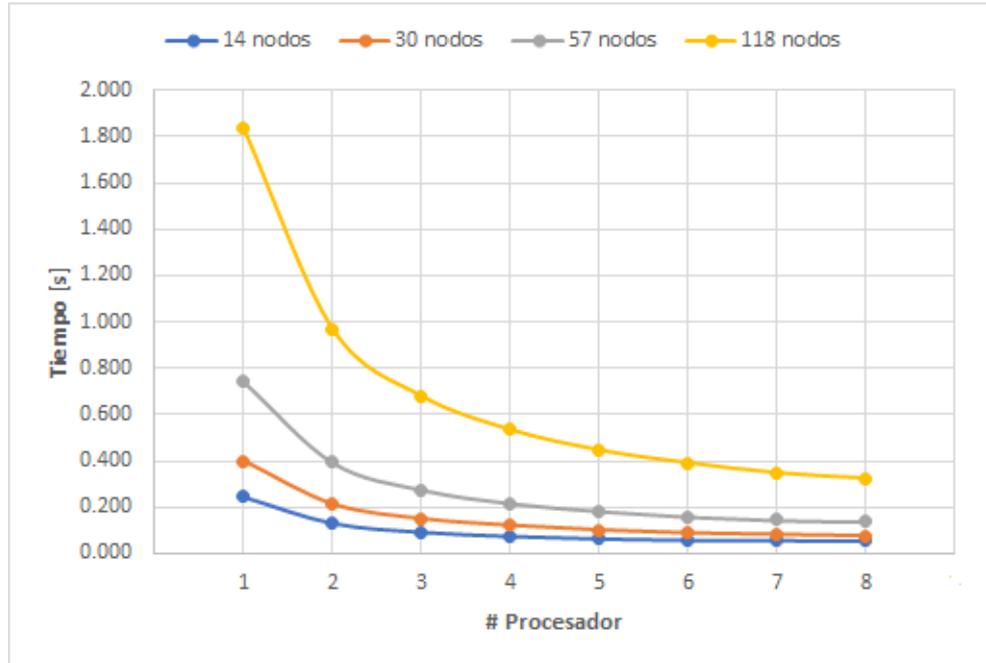


Figura 5.8: Gráfica de Tiempo de la Paralelización del THD

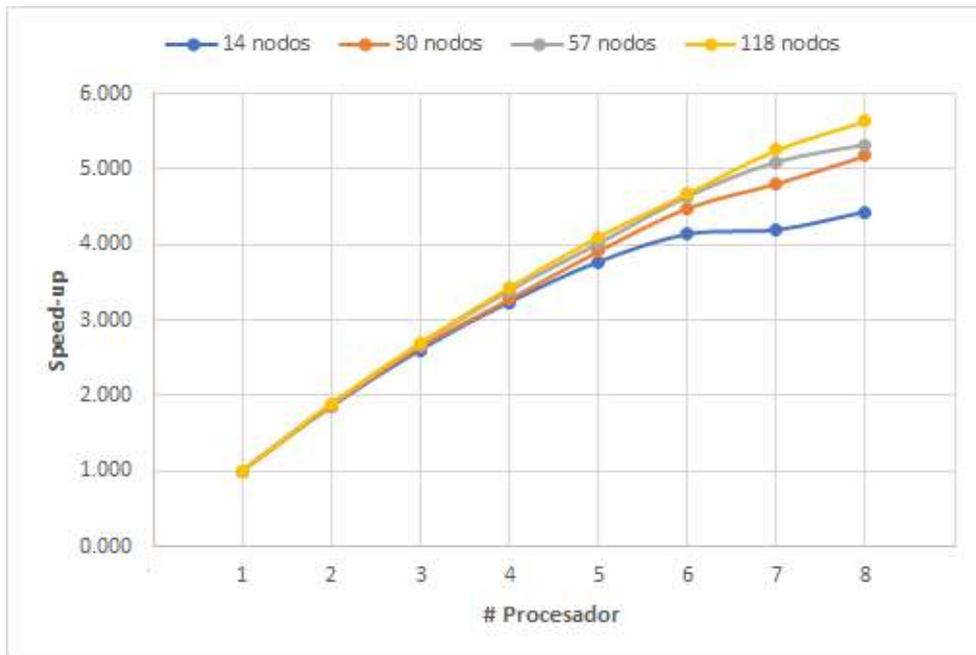


Figura 5.9: Gráfica del *speed-up* de la Paralelización del THD

A continuación, se presenta la información sobre la modificación del sistema de Prueba de IEEE de 14 nodos donde se incluyeron ramas magnetizante en los nodos 10, 11, 13 y 14. El sistema de prueba del IEEE de 14 nodos se modeló matemáticamente por medio del uso de 43 variables de estado. La Figura 4.10 muestra el comportamiento del voltaje distorsionado en el nodo 11, debido a la presencia del tercer armónico de magnitud cuyo es del 14.5 % por encima del límite establecida en la norma [IEEE Std 519, 2014]. En la Figura 5.11 se muestra la corriente del nodo 6 al nodo 12, se observa que la corriente está distorsionada por una tercera armónica de magnitud de 65% de la fundamental. La Figura 5.12 muestra la corriente entregada por el generador conectado al nodo 1, la cual está distorsionada por las armónicas y se observa una tercera armónica de magnitud del 21 % lo cual contribuye de manera significativa a la distorsión de la corriente.

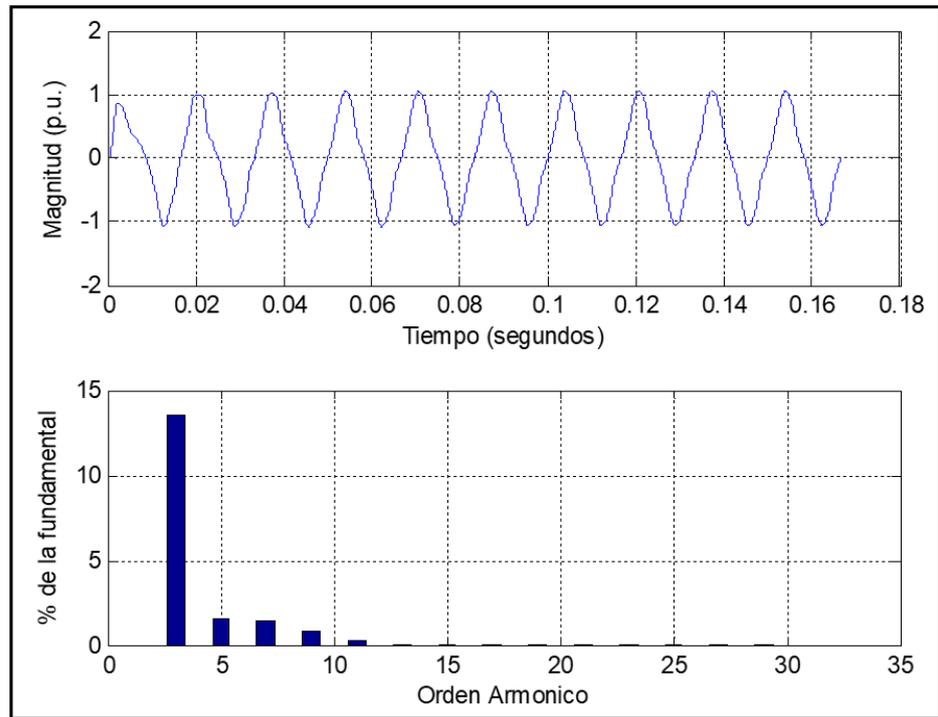


Figura 5.10: Voltaje en el nodo 11

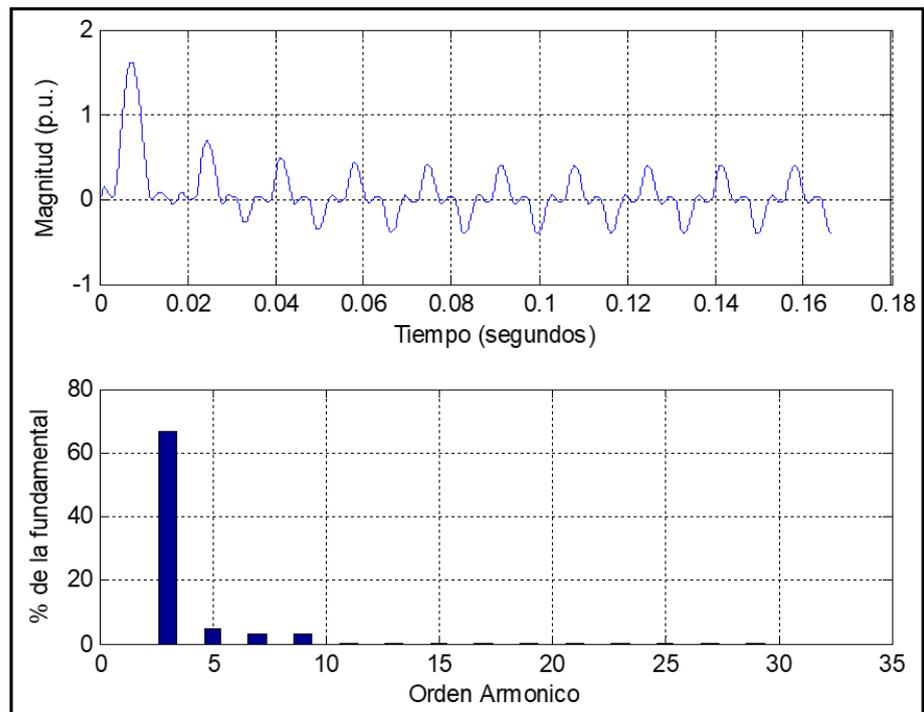


Figura 5.11: Corriente del nodo 6 al nodo 12

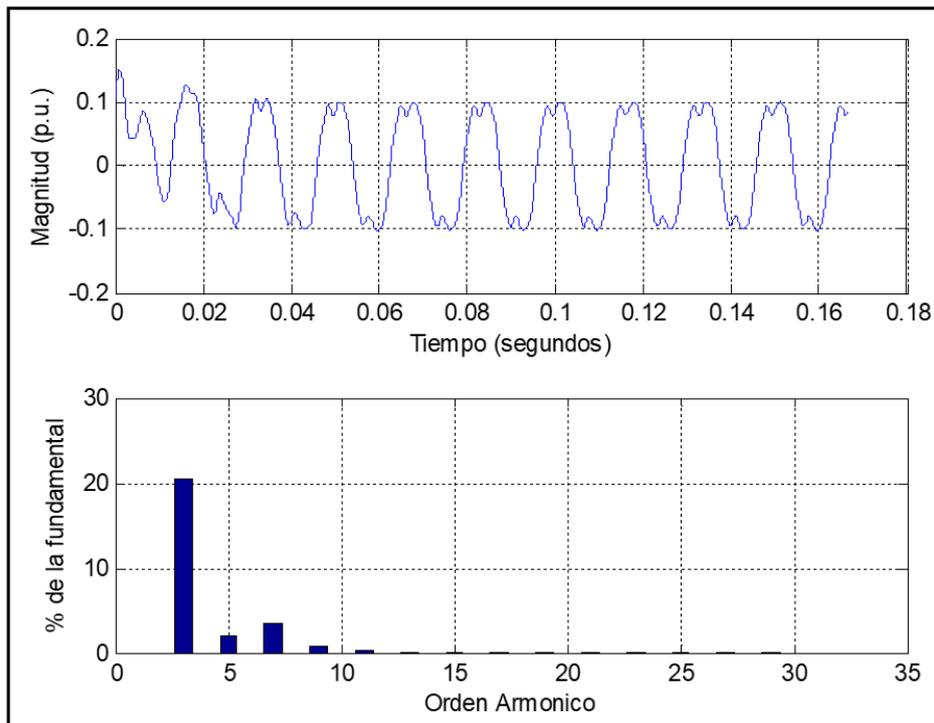


Figura 5.12: Corriente entregada por el generador conectado al nodo 1

En la Tabla 5.7 se muestra el valor del THD en las variables de estado del sistema de prueba del IEEE de 14 nodos que se modeló matemáticamente. En la Tabla 5.7 se presenta las variables de estado que no cumplen con las recomendaciones de [IEEE Std 519, 2014] que están en el Capítulo 2 precisamente en la sección 2.4.

Tabla 5.7: Variables de estado del sistema de 14 nodos con sus THD

| <b>VE</b> | <b>Significado</b>                                       | <b>THD[%]</b> |
|-----------|--|---------------|
| 11        | Corriente del nodo 5 al nodo 11                          | 75.296        |
| 12        | Corriente del nodo 6 al nodo 12                          | 67.273        |
| 13        | Corriente del nodo 6 al nodo 13                          | 77.258        |
| 15        | Corriente del nodo 7 al nodo 9                           | 91.000        |
| 18        | Corriente del nodo 10 al nodo 11                         | 16.735        |
| 19        | Corriente del nodo 12 al nodo 13                         | 57.628        |
| 20        | Corriente del nodo 13 al nodo 14                         | 23.832        |
| 30        | Voltaje en el nodo 10                                    | 13.516        |
| 31        | Voltaje en el nodo 11                                    | 13.653        |
| 32        | Voltaje en el nodo 12                                    | 14.033        |
| 33        | Voltaje en el nodo 13                                    | 11.141        |
| 34        | Voltaje en el nodo 14                                    | 11.998        |
| 39        | Corriente entregada por el genera<br>Conectada al nodo 1 | 20.918        |
| 40        | Corriente entregada por el genera<br>Conectada al nodo 2 | 69.418        |
| 41        | Corriente entregada por el genera<br>Conectada al nodo 3 | 50.855        |
| 42        | Corriente entregada por el genera<br>Conectada al nodo 6 | 92.304        |

## 5.5. Caso de Estudio No. 3: Cálculo en paralelo de la DPI

A continuación, se presentan dos casos de estudio, en los cuales se determina de manera paralela la respuesta a la frecuencia en una red eléctrica. En el Caso de Estudio 3.1 se calcula el DPI de una red eléctrica de 3 nodos, en tanto que el Caso de Estudio 3.2 se muestra el cálculo la DPI de una red eléctrica de 5 nodos. En ambos casos se modifica el paso de frecuencia con valores de 1, 0.1, 0.01, 0.001 y 0.0001, además de que se calcula la respuesta desde la frecuencia fundamental hasta el 40vo armónico y el 80vo armónico.

### 5.5.1. Caso de Estudio 3.1. Red de 3 nodos con 2 líneas de transmisión

En este caso de estudio se calcula la respuesta a la frecuencia de la red eléctrica mostrada en la Figura 5.13, la cual consta de 3 nodos. Se presenta el cálculo de la respuesta a la frecuencia en base a la variación del paso de frecuencia, usando valores de 1, 0.1, 0.01, 0.001, 0.0001. En este estudio se utiliza desde 1 hasta 8 elementos de proceso.

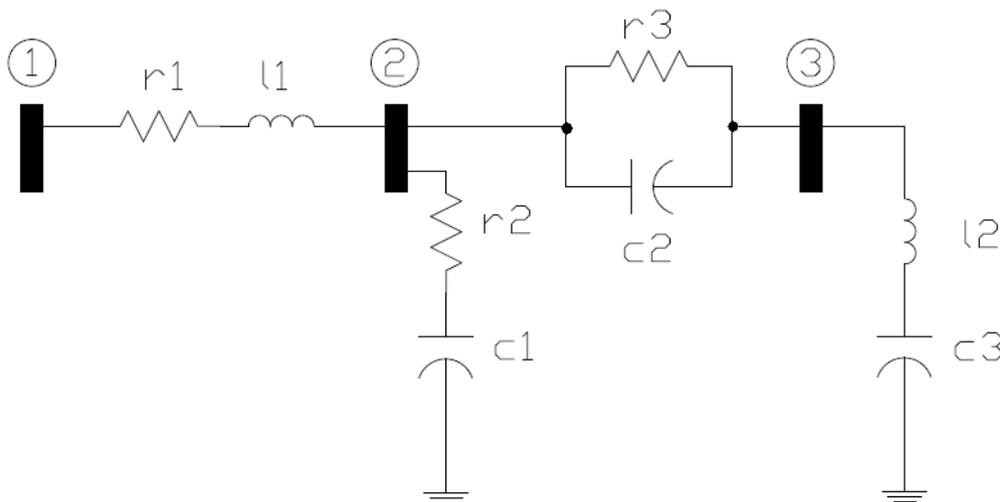


Figura 5.13: Esquema del sistema de 3 nodos

### 5.5.1.1. Medición de 40 armónicos

En la Tabla 5.8 se muestra el tiempo de cómputo requerido para el cálculo de la DPI del sistema eléctrico. En esta tabla se presentan los tiempos de cómputo cuando se usa un paso de frecuencia de 1, 0.1, 0.01, 0.001 y 0.0001. En la primera columna de esta tabla se muestra el número de elementos de proceso utilizados. Se observa que un paso de 1, no hay ventaja de usar el cómputo en paralelo debido a que el esfuerzo computacional es mínimo. Se observa que con el paso de 0.1, la ventaja del uso del cómputo en paralelo se ve a partir de 3 elementos de proceso, se disminuye el tiempo de 4 ms a 3 ms, lo cual representa una disminución del 25% y se observa con 3 elementos de proceso se alcanza el límite de la paralelización del programa.

Se observa que el uso de un solo elemento de proceso produce un cálculo eminentemente secuencial. Con el uso de 2 elementos de proceso se observa que para un paso de frecuencia de 0.0001, se disminuye el tiempo de cómputo de 1057 ms a 644 ms, lo cual representa una disminución del 39.1%. Se observa que con el uso de 8 elementos de proceso se alcanza un tiempo de 504 ms, lo cual representa una disminución del tiempo de cómputo de un 52.3%.

Tabla 5.8: Resultados de Mediciones de Tiempo [s] de 40 armónicos

| No. de Procesadores | Paso de frecuencia |          |           |            |             |
|---------------------|--------------------|----------|-----------|------------|-------------|
|                     | Paso=1             | Paso=0.1 | Paso=0.01 | Paso=0.001 | Paso=0.0001 |
| 1                   | 0.003              | 0.004    | 0.013     | 0.108      | 1.057       |
| 2                   | 0.003              | 0.004    | 0.009     | 0.069      | 0.644       |
| 3                   | 0.003              | 0.003    | 0.008     | 0.058      | 0.530       |
| 4                   | 0.003              | 0.003    | 0.008     | 0.056      | 0.512       |
| 5                   | 0.003              | 0.003    | 0.008     | 0.056      | 0.511       |
| 6                   | 0.003              | 0.003    | 0.008     | 0.056      | 0.510       |
| 7                   | 0.003              | 0.003    | 0.008     | 0.056      | 0.507       |
| 8                   | 0.003              | 0.003    | 0.008     | 0.056      | 0.504       |

En los datos presentados en la Tabla 5.8 se aprecia que conforme se disminuye el paso de frecuencia, se va incrementando el tiempo de cómputo con el uso de un solo elemento de proceso. La aplicación de un mayor número de elementos de proceso va generando una

disminución en el tiempo de cómputo. Se aprecia que para el uso de un paso de frecuencia de 0.0001 y 8 elementos de proceso se reduce el tiempo de cómputo en un 52.3 %.

Tabla 5.9: Resultados del cálculo del *speed-up* de 40 armónicos

| No. de Procesadores | Paso de frecuencia |          |           |            |             |
|---------------------|--------------------|----------|-----------|------------|-------------|
|                     | Paso=1             | Paso=0.1 | Paso=0.01 | Paso=0.001 | Paso=0.0001 |
| 1                   | 1.000              | 1.000    | 1.000     | 1.000      | 1.000       |
| 2                   | 1.000              | 1.000    | 1.444     | 1.565      | 1.641       |
| 3                   | 1.000              | 1.333    | 1.625     | 1.862      | 1.994       |
| 4                   | 1.000              | 1.333    | 1.625     | 1.929      | 2.064       |
| 5                   | 1.000              | 1.333    | 1.625     | 1.929      | 2.068       |
| 6                   | 1.000              | 1.333    | 1.625     | 1.929      | 2.073       |
| 7                   | 1.000              | 1.333    | 1.625     | 1.929      | 2.085       |
| 8                   | 1.000              | 1.333    | 1.625     | 1.929      | 2.097       |

En la Tabla 5.9 se muestra el *speed-up* obtenido a través del uso del cómputo en paralelo. Se aprecia que conforme aumenta el tamaño del problema y aumenta el número de elementos de proceso, se incrementa el *speed-up*, alcanzándose un *speed-up* máximo de 2.097 para el caso de un paso de frecuencia de 0.0001 y utilizando 8 elementos de proceso es 2.097 veces más rápido que el programa secuencial, es decir, el programa que usa un solo elemento de proceso.

En la Figura 5.14 se muestra el comportamiento del tiempo de cómputo conforme se incrementa el número de elementos de proceso. Se observa que conforme aumenta el número de elementos de proceso se disminuye el tiempo de cómputo, siendo notorio que el mayor decremento en el tiempo de cómputo se da cuando se pasa del uso de un elemento de cómputo a dos elementos de cómputo por los pasos de frecuencia de 0.01, 0.001 y 0.0001. De esta misma gráfica se aprecia que para los casos de estudio de 1, 0.1, 0.01 y 0.001 pasos de frecuencia, no existe un decremento en la velocidad con el uso de 3 o más elementos de proceso. Esto indica que se alcanzado el tiempo secuencial del algoritmo, es decir, la parte del mismo que no puede ser paraleliza. El uso adicional de elementos de proceso, en el mejor de los casos no reducirá el tiempo de cómputo, en el escenario más adverso incrementará el tiempo de cómputo.

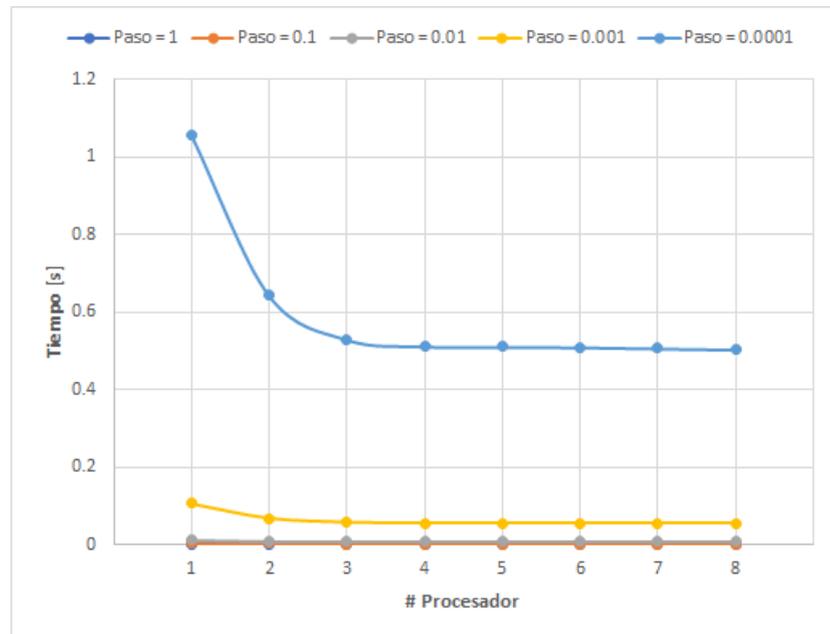


Figura 5.14: Gráfica de Tiempo de la Paralelización del DPI con 40 armónicos

En la Figura 5.15 se muestra una gráfica en que relaciona el número de elementos de cómputo con el *speed-up* obtenido en los diversos casos de estudio. Se aprecia que conforme aumenta el tamaño del problema y el número de elemento de proceso se alcanza un valor máximo del *speed-up* para el caso de 0.1 paso de frecuencia, el *speed-up* máximo alcanzado es de 1.333 en tanto que para el caso de 0.0001 paso de frecuencia el *speed-up* máximo alcanzado es de 2.097. De aquí se puede concluir que el algoritmo propuesto genera mejores resultados para problemas de mayores dimensiones. Sin embargo, para el caso de problemas de menor dimensión, genera una disminución considerable en el tiempo de cómputo.

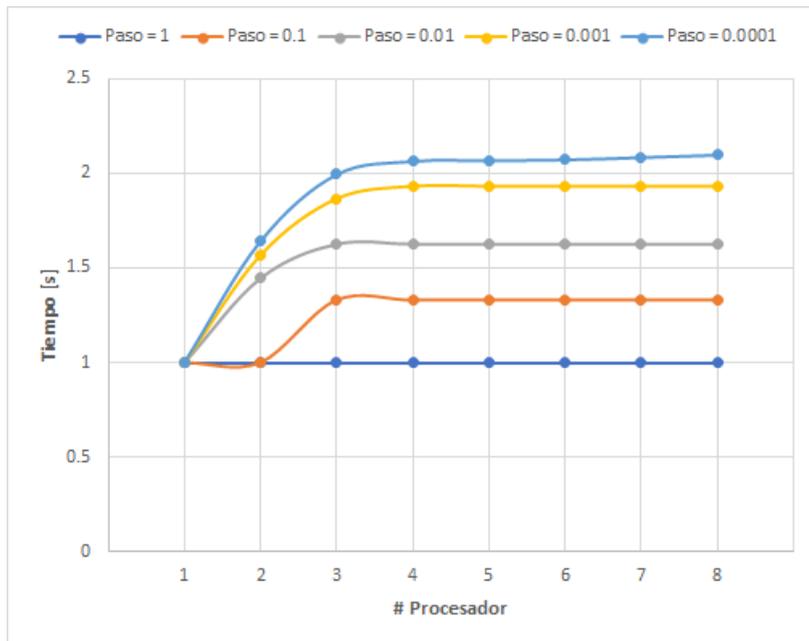


Figura 5.15: Gráfica del *speed-up* de la Paralelización del DPI con 40 armónicos

### 5.5.1.2. Medición de 80 armónicos

En la Tabla 5.10 se muestra el tiempo de cómputo requerido para el cálculo de la DPI del sistema eléctrico. En esta tabla se presentan los tiempos de cómputo cuando se usa un paso de frecuencia de 1, 0.1, 0.01, 0.001 y 0.0001. En la primera columna de esta tabla se muestra el número de elementos de proceso utilizados. Se observa que un paso de 1, no hay ventaja de usar el cómputo en paralelo debido a que el esfuerzo computacional es mínimo. Se observa que con el paso de 0.1, la ventaja del uso del cómputo en paralelo se ve a partir de 2 elementos de proceso, se disminuye el tiempo de 5ms a 4ms, lo cual representa una disminución del 25% y se observa con 3 elementos de proceso se alcanza el límite de la paralelización del programa.

Se observa que el uso de un solo elemento de proceso produce un cálculo eminentemente secuencial. Con el uso de 2 elementos de proceso se observa que para un paso de frecuencia de 0.0001, se disminuye el tiempo de cómputo de 2109 ms a 1300 ms, lo cual representa una disminución del 38.6%. Se observa que con el uso de 8 elementos de proceso

se alcanza un tiempo de 1004 ms, lo cual representa una disminución del tiempo de cómputo de un 52.4%.

Tabla 5.10: Resultados de Mediciones de Tiempo [s] de 80 armónicos

| No. de Procesadores | Paso de frecuencia |          |           |            |             |
|---------------------|--------------------|----------|-----------|------------|-------------|
|                     | Paso=1             | Paso=0.1 | Paso=0.01 | Paso=0.001 | Paso=0.0001 |
| 1                   | 0.003              | 0.005    | 0.024     | 0.214      | 2.109       |
| 2                   | 0.003              | 0.004    | 0.016     | 0.133      | 1.300       |
| 3                   | 0.003              | 0.004    | 0.013     | 0.108      | 1.035       |
| 4                   | 0.003              | 0.004    | 0.013     | 0.106      | 1.024       |
| 5                   | 0.003              | 0.004    | 0.013     | 0.106      | 1.023       |
| 6                   | 0.003              | 0.004    | 0.013     | 0.106      | 1.015       |
| 7                   | 0.003              | 0.004    | 0.013     | 0.106      | 1.005       |
| 8                   | 0.003              | 0.004    | 0.013     | 0.106      | 1.004       |

En los datos presentados en la Tabla 5.10 se aprecia que conforme se disminuye el paso de frecuencia, se va incrementando el tiempo de cómputo con el uso de un solo elemento de proceso. La aplicación de un mayor número de elementos de proceso va generando una disminución en el tiempo de cómputo. Se aprecia que para el uso de un paso de frecuencia de 0.0001 y 8 elementos de proceso se reduce el tiempo de cómputo en un 52.4%.

Tabla 5.11: Resultados del cálculo del *speed-up* de 80 armónicos

| No. de Procesadores | Paso de frecuencia |          |           |            |             |
|---------------------|--------------------|----------|-----------|------------|-------------|
|                     | Paso=1             | Paso=0.1 | Paso=0.01 | Paso=0.001 | Paso=0.0001 |
| 1                   | 1.000              | 1.000    | 1.000     | 1.000      | 1.000       |
| 2                   | 1.000              | 1.250    | 1.500     | 1.609      | 1.622       |
| 3                   | 1.000              | 1.250    | 1.846     | 1.981      | 2.038       |
| 4                   | 1.000              | 1.250    | 1.846     | 2.019      | 2.060       |
| 5                   | 1.000              | 1.250    | 1.846     | 2.019      | 2.062       |
| 6                   | 1.000              | 1.250    | 1.846     | 2.019      | 2.078       |
| 7                   | 1.000              | 1.250    | 1.846     | 2.019      | 2.099       |
| 8                   | 1.000              | 1.250    | 1.846     | 2.019      | 2.101       |

En la Tabla 5.11 se muestra el *speed-up* obtenido a través del uso del cómputo en paralelo. Se aprecia que conforme aumenta el tamaño del problema y aumenta el número de elementos de proceso, se incrementa el *speed-up*, alcanzándose un *speed-up* máximo de 2.101 para el caso de un paso de frecuencia de 0.0001 y 8 elementos de proceso es 2.101

veces más rápido que el programa secuencial, es decir, el programa que usa un solo elemento de proceso.

En la Figura 5.16 se muestra el comportamiento del tiempo de cómputo conforme se incrementa el número de elementos de proceso. Se observa que conforme aumenta el número de elementos de proceso se disminuye el tiempo de cómputo, siendo notorio que el mayor decremento en el tiempo de cómputo se da cuando se pasa del uso de un elemento de cómputo a dos elementos de cómputo por los pasos de frecuencia de 0.01, 0.001 y 0.0001. De esta misma gráfica se aprecia que para los casos de estudio de 1, 0.1, 0.01 y 0.001 pasos de frecuencia, no existe un decremento en la velocidad con el uso de 3 o más elementos de proceso. Esto indica que se alcanzado el tiempo secuencial del algoritmo, es decir, la parte del mismo que no puede ser paraleliza. El uso adicional de elementos de proceso, en el mejor de los casos no reducirá el tiempo de cómputo, en el escenario más adverso incrementará el tiempo de cómputo.

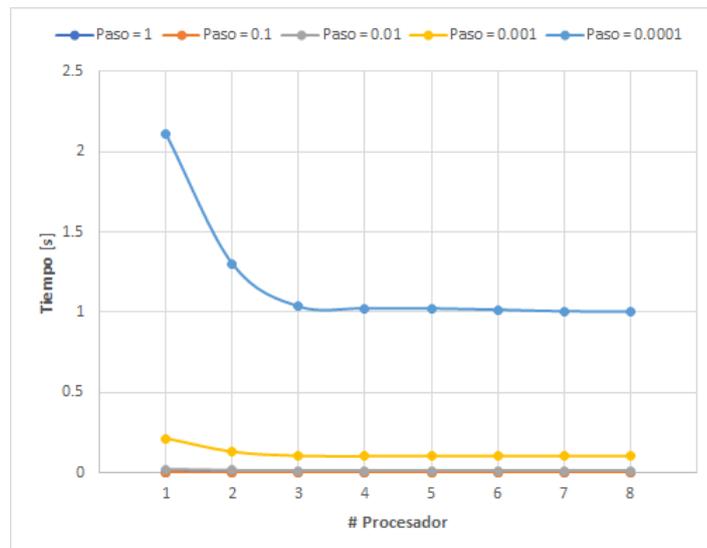


Figura 5.16: Gráfica de Tiempo de la Paralelización del DPI con 80 armónicos

En la Figura 5.17 se muestra una gráfica en que relaciona el número de elementos de cómputo con el *speed-up* obtenido en los diversos casos de estudio. Se aprecia que conforme aumenta el tamaño del problema y el número de elemento de proceso se alcanza un valor máximo del *speed-up* para el caso de 0.1 paso de frecuencia, el *speed-up* máximo alcanzado es de 1.25 en tanto que para el caso de 0.0001 paso de frecuencia el *speed-up* máximo alcanzado es de 2.101. De aquí se puede concluir que el algoritmo propuesto genera mejores resultados para problemas de mayores dimensiones. Sin embargo, para el caso de problemas de menor dimensión, genera una disminución considerable en el tiempo de cómputo.

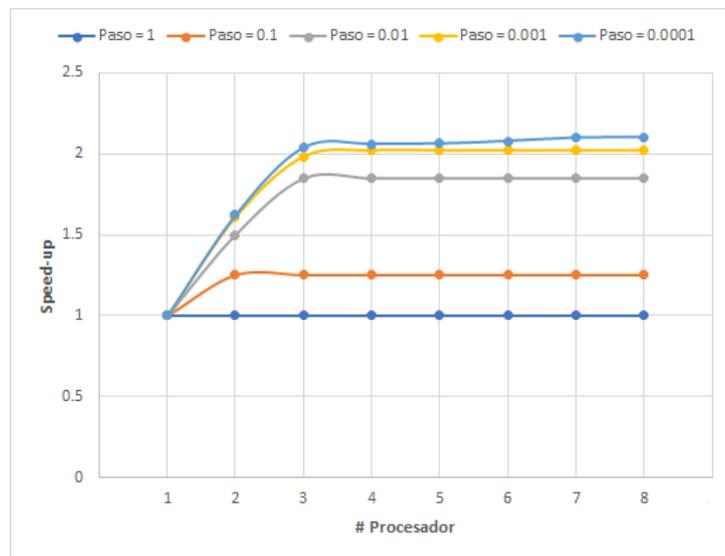


Figura 5.17: Gráfica del *speed-up* de la Paralelización del DPI con 80 armónicos

### 5.5.2. Caso de Estudio 3.2. Red de 5 nodos con 7 líneas de transmisión

En este caso de estudio se calcula la respuesta a la frecuencia de la red eléctrica mostrada en la Figura 5.18, la cual consta de 5 nodos. Se presenta el cálculo de la respuesta a la frecuencia en base a la variación del paso de frecuencia, usando valores de 1, 0.1, 0.01, 0.001, 0.0001. En este estudio se utiliza desde 1 hasta 8 elementos de proceso.

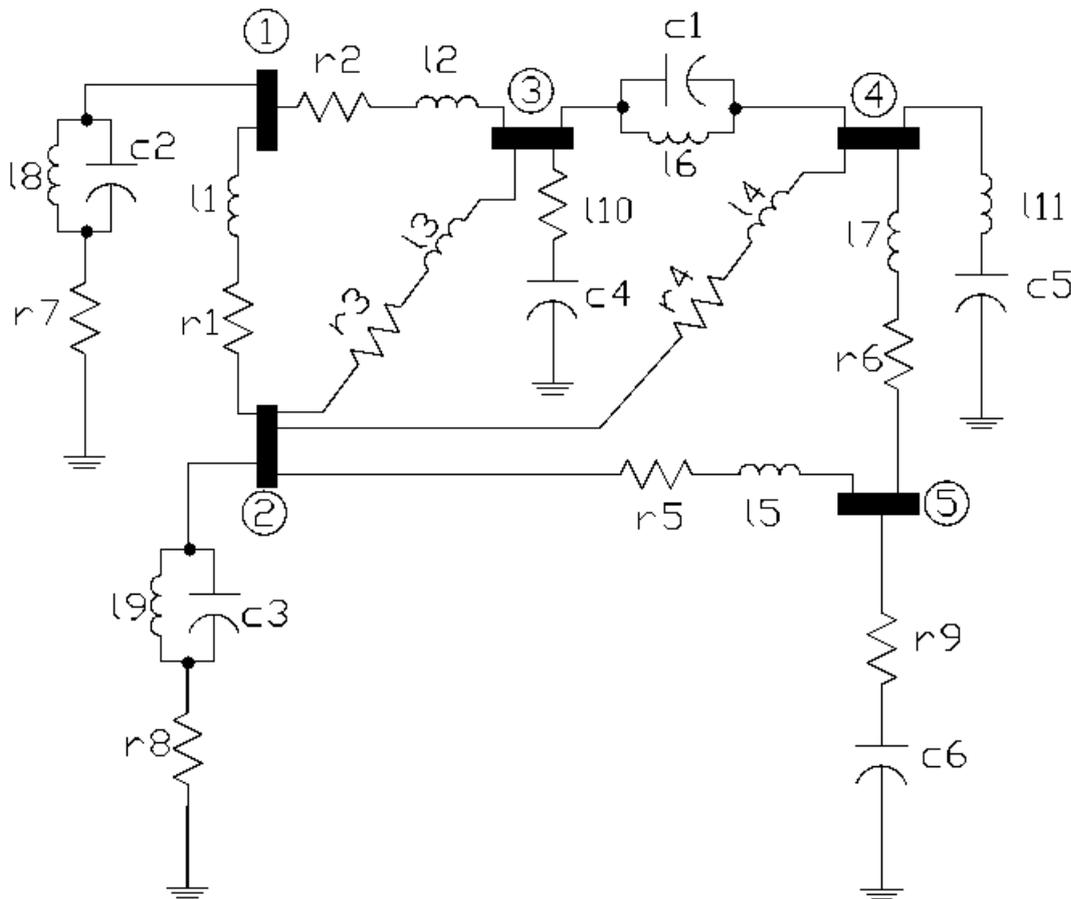


Figura 5.18: Esquema del sistema de 5 nodos

#### 5.5.2.1. Medición de 40 armónicos

En la Tabla 5.12 se muestra el tiempo de cómputo requerido para el cálculo de la DPI del sistema eléctrico. En esta tabla se presentan los tiempos de cómputo cuando se usa un

paso de frecuencia de 1, 0.1, 0.01, 0.001 y 0.0001. En la primera columna de esta tabla se muestra el número de elementos de proceso utilizados. Se observa que un paso de 1, no hay ventaja de usar el cómputo en paralelo debido a que el esfuerzo computacional es mínimo. Se observa que con el paso de 0.1, la ventaja de uso del cómputo en paralelo se ve a partir de 2 elementos de proceso, se disminuye el tiempo de 7ms a 5ms, lo cual representa una disminución del 28.6 % y se observa con 3 elementos de proceso se alcanza el límite de la paralelización del programa.

Se observa que el uso de un solo elemento de proceso produce un cálculo eminentemente secuencial. Con el uso de 2 elementos de proceso se observa que para un paso de frecuencia de 0.0001, se disminuye el tiempo de cómputo de 4110 ms a 2132 ms, lo cual representa una disminución del 48.1 %. Se observa que con el uso de 8 elementos de proceso se alcanza un tiempo de 813 ms, lo cual representa una disminución del tiempo de cómputo de un 80.2 %.

Tabla 5.12: Resultados de Mediciones de Tiempo [s] de 40 armónicos

| No. de Procesadores | Paso de frecuencia |          |           |            |             |
|---------------------|--------------------|----------|-----------|------------|-------------|
|                     | Paso=1             | Paso=0.1 | Paso=0.01 | Paso=0.001 | Paso=0.0001 |
| 1                   | 0.003              | 0.007    | 0.044     | 0.414      | 4.110       |
| 2                   | 0.003              | 0.005    | 0.024     | 0.216      | 2.132       |
| 3                   | 0.003              | 0.004    | 0.018     | 0.158      | 1.534       |
| 4                   | 0.003              | 0.004    | 0.015     | 0.122      | 1.185       |
| 5                   | 0.003              | 0.004    | 0.013     | 0.106      | 1.020       |
| 6                   | 0.003              | 0.004    | 0.013     | 0.100      | 0.941       |
| 7                   | 0.003              | 0.004    | 0.013     | 0.096      | 0.870       |
| 8                   | 0.003              | 0.004    | 0.013     | 0.096      | 0.813       |

En los datos presentados en la Tabla 5.12 se aprecia que conforme se disminuye el paso de frecuencia, se va incrementando el tiempo de cómputo con el uso de un solo elemento de proceso. La aplicación de un mayor número de elementos de proceso va generando una disminución en el tiempo de cómputo. Se aprecia que para el uso de un paso de frecuencia de 0.0001 y 8 elementos de proceso se reduce el tiempo de cómputo en un 80.2 %.

Tabla 5.13: Resultados del cálculo del *speed-up* de 40 armónicos

| No. de Procesadores | Paso de frecuencia |          |           |            |             |
|---------------------|--------------------|----------|-----------|------------|-------------|
|                     | Paso=1             | Paso=0.1 | Paso=0.01 | Paso=0.001 | Paso=0.0001 |
| 1                   | 1.000              | 1.000    | 1.000     | 1.000      | 1.000       |
| 2                   | 1.000              | 1.400    | 1.833     | 1.917      | 1.928       |
| 3                   | 1.000              | 1.750    | 2.444     | 2.620      | 2.679       |
| 4                   | 1.000              | 1.750    | 2.933     | 3.393      | 3.468       |
| 5                   | 1.000              | 1.750    | 3.385     | 3.906      | 4.029       |
| 6                   | 1.000              | 1.750    | 3.385     | 4.140      | 4.368       |
| 7                   | 1.000              | 1.750    | 3.385     | 4.313      | 4.724       |
| 8                   | 1.000              | 1.750    | 3.385     | 4.313      | 5.055       |

En la Tabla 5.13 se muestra el *speed-up* obtenido a través del uso del cómputo en paralelo. Se aprecia que conforme aumenta el tamaño del problema y aumenta el número de elementos de proceso, se incrementa el *speed-up*, alcanzándose un *speed-up* máximo de 5.055 para el caso de un paso de frecuencia de 0.0001 y 8 elementos de proceso es 5.055 veces más rápido que el programa secuencial, es decir, el programa que usa un solo elemento de proceso.

En la Figura 5.19 se muestra el comportamiento del tiempo de cómputo conforme se incrementa el número de elementos de proceso. Se observa que conforme aumenta el número de elementos de proceso se disminuye el tiempo de cómputo, siendo notorio que el mayor decremento en el tiempo de cómputo se da cuando se pasa del uso de un elemento de cómputo a dos elementos de cómputo por los pasos de frecuencia de 0.01, 0.001 y 0.0001. De esta misma gráfica se aprecia que para los casos de estudio de 1, 0.1, 0.01 y 0.001 pasos de frecuencia, no existe un decremento en la velocidad con el uso de 6 o más elementos de proceso. Esto indica que se alcanzado el tiempo secuencial del algoritmo, es decir, la parte del mismo que no puede ser paraleliza. El uso adicional de elementos de proceso, en el mejor de los casos no reducirá el tiempo de cómputo, en el escenario más adverso incrementará el tiempo de cómputo.

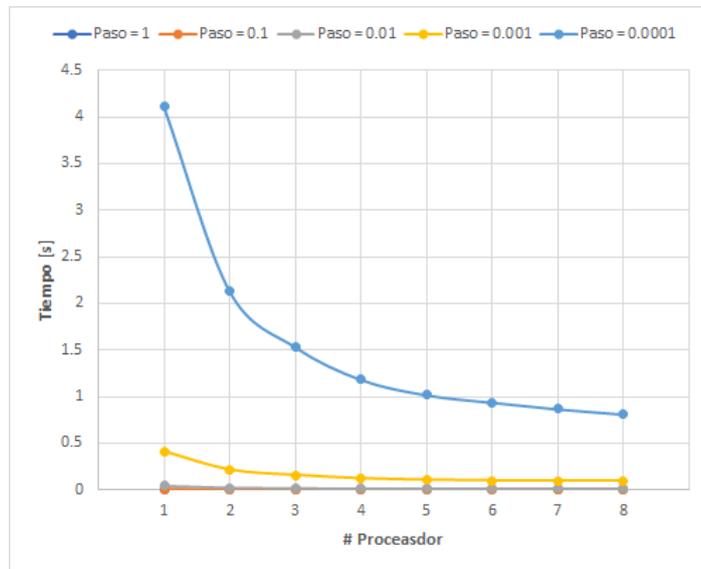


Figura 5.19: Gráfica de Tiempo de la Paralelización de la DPI con 40 armónicos

En la Figura 5.20 se muestra una gráfica en que relaciona el número de elementos de cómputo con el *speed-up* obtenido en los diversos casos de estudio. Se aprecia que conforme aumenta el tamaño del problema y el número de elementos de proceso se alcanza un valor máximo del *speed-up* para el caso de 0.1 paso de frecuencia, el *speed-up* máximo alcanzado es de 1.75 en tanto que para el caso de 0.0001 paso de frecuencia el *speed-up* máximo alcanzado es de 5.055. De aquí se puede concluir que el algoritmo propuesto genera mejores resultados para problemas de mayores dimensiones. Sin embargo, para el caso de problemas de menor dimensión, genera una disminución considerable en el tiempo de cómputo.

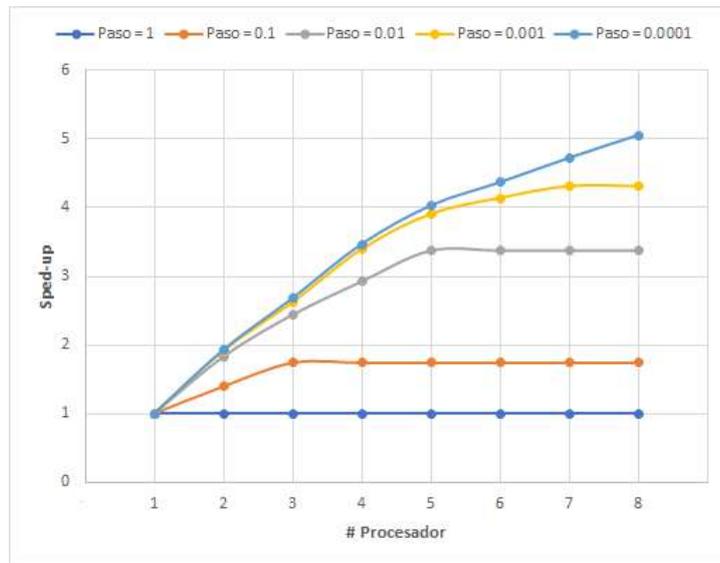


Figura 5.20: Gráfica del *speed-up* de la Paralelización del DPI con 40 armónicos

#### 5.5.2.1. Medición de 80 armónicos

En la Tabla 5.14 se muestra el tiempo de cómputo requerido para el cálculo de la DPI del sistema eléctrico. En esta tabla se presentan los tiempos de cómputo cuando se usa un paso de frecuencia de 1, 0.1, 0.01, 0.001 y 0.0001. En la primera columna de esta tabla se muestra el número de elementos de proceso utilizados. Se observa que un paso de 1, no hay ventaja de usar el cómputo en paralelo debido a que el esfuerzo computacional es mínimo. Se observa que con el paso de 0.1, la ventaja del uso del cómputo en paralelo se ve a partir de 2 elementos de proceso, se disminuye el tiempo de 11ms a 7ms, lo cual representa una disminución del 36.4% y se observa con 3 elementos de proceso se alcanza el límite de la paralelización del programa.

Se observa que el uso de un solo elemento de proceso produce un cálculo eminentemente secuencial. Con el uso de 2 elementos de proceso se observa que para un paso de frecuencia de 0.0001, se disminuye el tiempo de cómputo de 8214 ms a 4255 ms, lo cual representa una disminución del 48.2%. Se observa que con el uso de 8 elementos de proceso se alcanza un tiempo de 1613 ms, lo cual representa una disminución del tiempo de cómputo de un 80.4%.

Tabla 5.14: Resultados de Mediciones de Tiempo [s] de 80 armónicos

| No. de Procesadores | Paso de frecuencia |          |           |            |             |
|---------------------|--------------------|----------|-----------|------------|-------------|
|                     | Paso=1             | Paso=0.1 | Paso=0.01 | Paso=0.001 | Paso=0.0001 |
| 1                   | 0.003              | 0.011    | 0.085     | 0.825      | 8.214       |
| 2                   | 0.003              | 0.007    | 0.046     | 0.430      | 4.255       |
| 3                   | 0.003              | 0.005    | 0.033     | 0.302      | 2.960       |
| 4                   | 0.003              | 0.005    | 0.027     | 0.240      | 2.366       |
| 5                   | 0.003              | 0.005    | 0.024     | 0.207      | 2.036       |
| 6                   | 0.003              | 0.005    | 0.022     | 0.192      | 1.881       |
| 7                   | 0.003              | 0.005    | 0.021     | 0.179      | 1.739       |
| 8                   | 0.003              | 0.005    | 0.020     | 0.177      | 1.613       |

En los datos presentados en la Tabla 5.14 se aprecia que conforme se disminuye el paso de frecuencia, se va incrementando el tiempo de cómputo con el uso de un solo elemento de proceso. La aplicación de un mayor número de elementos de proceso va generando una disminución en el tiempo de cómputo. Se aprecia que para el uso de un paso de frecuencia de 0.0001 y 8 elementos de proceso se reduce el tiempo de cómputo en un 80.4%.

Tabla 5.15: Resultados del cálculo del *speed-up* de 80 armónicos

| No. de Procesadores | Paso de frecuencia |          |           |            |             |
|---------------------|--------------------|----------|-----------|------------|-------------|
|                     | Paso=1             | Paso=0.1 | Paso=0.01 | Paso=0.001 | Paso=0.0001 |
| 1                   | 1.000              | 1.000    | 1.000     | 1.000      | 1.000       |
| 2                   | 1.000              | 1.571    | 1.848     | 1.919      | 1.930       |
| 3                   | 1.000              | 1.833    | 2.576     | 2.732      | 2.775       |
| 4                   | 1.000              | 2.200    | 3.148     | 3.438      | 3.472       |
| 5                   | 1.000              | 2.200    | 3.542     | 3.986      | 4.034       |
| 6                   | 1.000              | 2.200    | 3.864     | 4.297      | 4.367       |
| 7                   | 1.000              | 2.200    | 4.048     | 4.609      | 4.723       |
| 8                   | 1.000              | 2.200    | 4.250     | 4.661      | 5.092       |

En la Tabla 5.15 se muestra el *speed-up* obtenido a través del uso del cómputo en paralelo. Se aprecia que conforme aumenta el tamaño del problema y aumenta el número de elementos de proceso, se incrementa el *speed-up*, alcanzándose un *speed-up* máximo de 5.092 para el caso de un paso de frecuencia de 0.0001 y 8 elementos de proceso es 5.092 veces más rápido que el programa secuencial, es decir, el programa que usa un solo elemento de proceso.

En la Figura 5.19 se muestra el comportamiento del tiempo de cómputo conforme se incrementa el número de elementos de proceso. Se observa que conforme aumenta el número de elementos de proceso se disminuye el tiempo de cómputo, siendo notorio que el mayor decremento en el tiempo de cómputo se da cuando se pasa del uso de un elemento de cómputo a dos elementos de cómputo por los pasos de frecuencia de 0.01, 0.001 y 0.0001. De esta misma gráfica se aprecia que para los casos de estudio de 1 y 0.1 pasos de frecuencia, no existe un decremento en la velocidad con el uso de 2 o más elementos de proceso. Esto indica que se alcanzado el tiempo secuencial del algoritmo, es decir, la parte del mismo que no puede ser paraleliza. El uso adicional de elementos de proceso, en el mejor de los casos no reducirá el tiempo de cómputo, en el escenario más adverso incrementará el tiempo de cómputo.

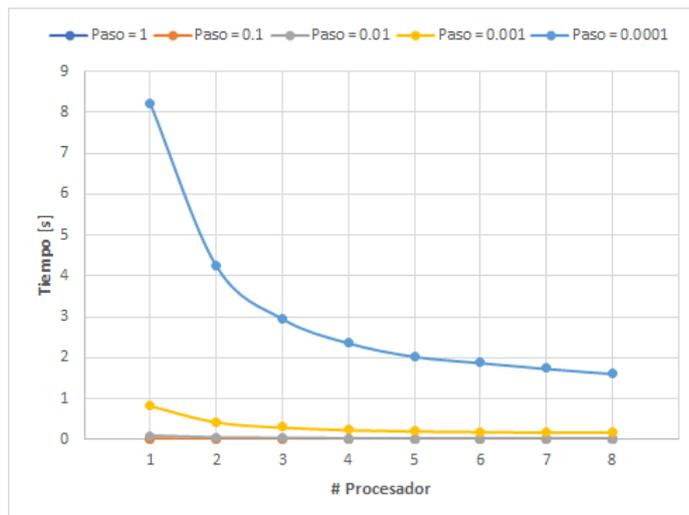


Figura 5.21: Gráfica de Tiempo de la Paralelización del DPI con 80 armónicos

En la Figura 5.21 se muestra una gráfica en que relaciona el número de elementos de cómputo con el *speed-up* obtenido en los diversos casos de estudio. Se aprecia que conforme aumenta el tamaño del problema y el número de elementos de proceso se alcanza un valor máximo del *speed-up* para el caso de 0.1 paso de frecuencia, el *speed-up* máximo alcanzado es de 2.2 en tanto que para el caso de 0.0001 paso de frecuencia el *speed-up* máximo alcanzado es de 5.092 De aquí se puede concluir que el algoritmo propuesto genera mejores resultados

para problemas de mayores dimensiones. Sin embargo, para el caso de problemas de menor dimensión, genera una disminución considerable en el tiempo de cómputo.

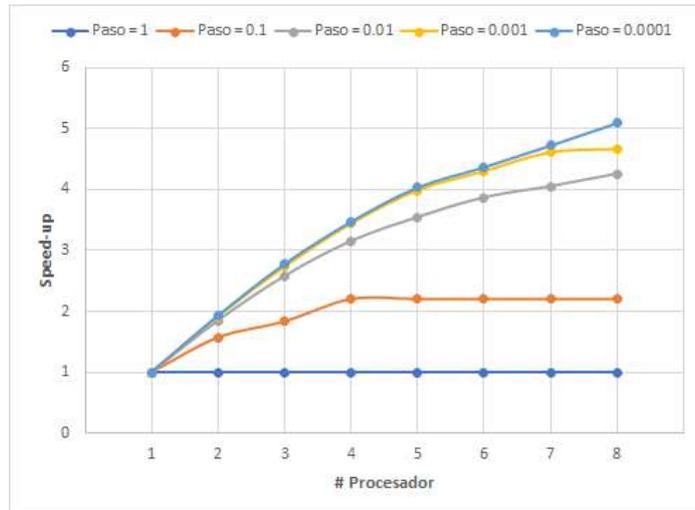


Figura 5.22: Gráfica del *speed-up* de la Paralelización del DPI con 80 armónicos

A continuación, se presenta en la Figura 5.23, la respuesta a la frecuencia en el nodo 3. En la Figura 5.24 se muestra la respuesta a la frecuencia en el nodo 4 y en la Figura 5.25 se muestra la respuesta a la frecuencia en el nodo 5. En la Figura 5.23, se observa que la red es susceptible a una resonancia paralela en las frecuencias de 540 Hz y 660 Hz es decir a dos puntos de orden  $h = 9$  y  $h = 11$ . En la Figura 5.24, se observa que la red es susceptible a una resonancia paralela en las frecuencias de 540 Hz y 660 Hz es decir a dos puntos de orden  $h = 9$  y  $h = 11$ . En la Figura 5.25, se observa que la red es susceptible a una resonancia paralela en las frecuencias de 240 Hz y 660 Hz es decir a dos puntos de orden  $h = 4$  y  $h = 11$ .

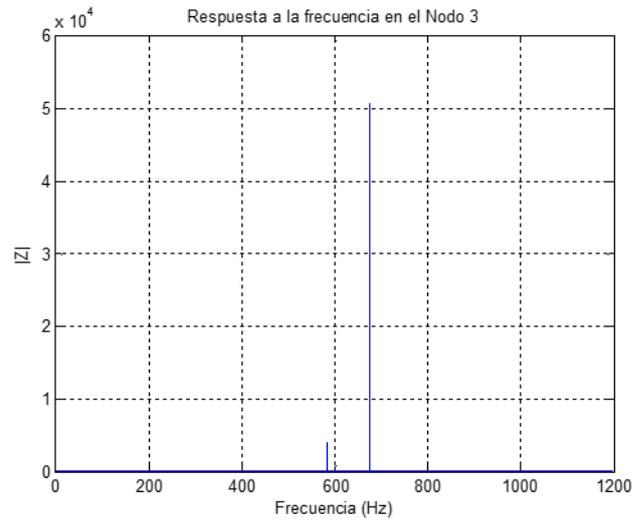


Figura 5.23: Respuesta de la impedancia a la frecuencia visto del nodo 3

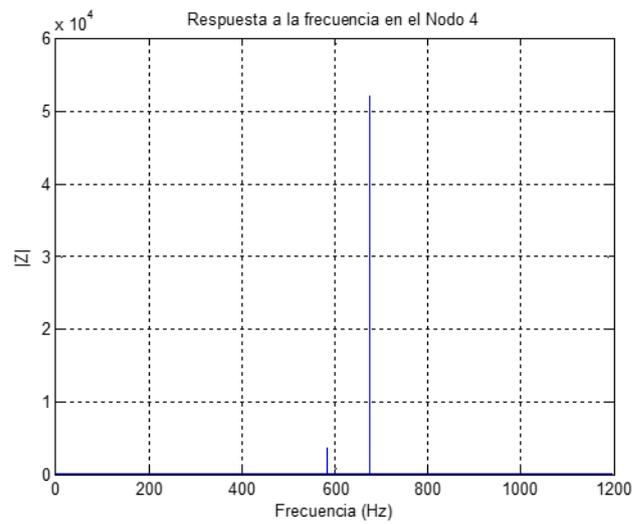


Figura 5.24: Respuesta de la impedancia a la frecuencia visto del nodo 4

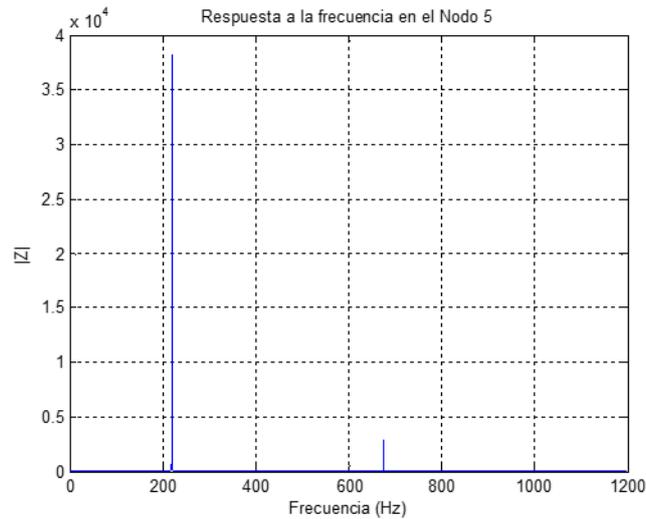


Figura 5.25: Respuesta de la impedancia a la frecuencia visto del nodo 5

## 5.6. Conclusiones

En este capítulo se ha presentado la aplicación del cómputo en paralelo, basado en threads, en el cálculo de la FFT, el THD y la DPI, en redes eléctricas.

En la propuesta presentada para el cálculo en paralelo de la FFT, se obtuvo un *speed-up* de 5.399, considerando un muestreo de 16384 puntos por periodo, así como 40 armónicos. Este *speed-up* representa una disminución del 81.33 % con respecto al tiempo de cómputo secuencial. Se muestra además que el caso de 8192 puntos por periodo, así como 80 armónicos se obtiene un *speed-up* máximo de 6.333, lo cual implica una disminución del 84.2 % con respecto al tiempo de cómputo secuencial.

En la aplicación del cómputo en paralelo, basado en threads, para el cálculo del THD en redes eléctricas, se presentaron cuatro estudios, consistentes en el cálculo del THD de las variables de estado de los sistemas de prueba modificados del IEEE de 14, 30, 57 y 118 nodos. Se observa que para el sistema modificado de prueba del IEEE de 118 nodos se alcanza un *speed-up* máximo de 5.634, lo cual implica una disminución en el tiempo de cómputo del 82.3 % con respecto al tiempo de cómputo secuencial.

Finalmente se presentó la aplicación del cómputo en paralelo, basado en threads, en el cálculo de la DPI, en redes eléctricas. Se observa que, para el sistema de 5 nodos, con el uso de un paso de frecuencia de 0.0001, se obtiene un *speed-up* máximo de 5.092, lo cual representa una disminución del 80.4% con respecto al tiempo de cómputo secuencial.

Se puede concluir que en promedio el uso de un elemento de proceso reduce aproximadamente un 10% el tiempo de cómputo secuencial, alcanzándose, reducciones cercanas la 80%, con respecto al tiempo de cómputo secuencial.

## Capítulo 6

# Conclusiones y recomendaciones para trabajos futuros

### 6.1. Conclusiones Generales

Una vez realizado el trabajo de investigación, se obtienen las siguientes conclusiones:

En esta tesis se han propuesto tres algoritmos para el cálculo en paralelo, basado en threads: la transformada rápida de Fourier, la distorsión armónica total, así como la respuesta a la frecuencia.

Para el cálculo en paralelo de la FFT, se obtuvo un *speed-up* de 5.399, considerando un muestreo de 16384 puntos por periodo así como 40 armónicos. Este *speed-up* representa una disminución del 81.33% con respecto al tiempo de cómputo secuencial. Se muestra además que para el caso de 8192 puntos por periodo así como 80 armónicos se obtiene un *speed-up* máximo de 6.333, lo cual implica una disminución del 84.2% con respecto al tiempo de cómputo secuencial.

En la aplicación del cómputo en paralelo, basado en threads, para el cálculo del THD en redes eléctricas, se presentaron cuatro estudios, consistentes en el cálculo del THD

de las variables de estado de los sistemas de prueba modificados del IEEE de 14, 30, 57 y 118 nodos. Se observa que para el sistema modificado de prueba del IEEE de 118 nodos se alcanza un *speed-up* máximo de 5.634, lo cual implica una disminución en el tiempo de cómputo del 82.3% con respecto al tiempo de cómputo secuencial.

Finalmente se presentó la aplicación del cómputo en paralelo, basado en threads, en el cálculo de la DPI, en redes eléctricas. Se observa que para el sistema de 5 nodos, con el uso de un paso de frecuencia de 0.0001, se obtiene un *speed-up* máximo de 5.092, lo cual representa una disminución del 80.4% con respecto al tiempo de cómputo secuencial.

Se puede concluir que en promedio el uso de un elemento de proceso reduce en aproximadamente en un 10% el tiempo de cómputo secuencial, alcanzándose, reducciones cercanas al 80%, con respecto al tiempo de cómputo secuencial.

Los resultados presentados en esta tesis permiten demostrar que el uso del procesamiento en paralelo, basado en threads, es una alternativa real para la disminución del tiempo de cómputo en problemas de grandes dimensiones.

Se concluye además de que es viable, aplicar el cómputo en paralelo, basado en threads, en la solución a diversos problemas del área de la ingeniería eléctrica, en específico a problemas relacionados con la calidad de la energía.

Se concluye que la programación en paralelo debe de ser el modelo de programación utilizada en el ámbito científico.

## 6.2. Trabajos Futuros

Los trabajos futuros de investigación asociados con esta tesis se propone:

- Aplicar el cómputo en paralelo, basado en thread, en la simulación del filtrado armónico dinámico, en redes eléctricas.
- Aplicar el cómputo en paralelo, basado en threads, en el diseño e implementación de

interfaces virtuales, aplicadas al monitoreo de índices de calidad de la energía.

- Diseñar algoritmos, aplicados a la solución de problemas de calidad de la energía, que puedan combinar el uso de los threads y de las unidades de procesamiento gráfico.



# Referencias

- [Abdelaziz et al., 2012] Abdelaziz, A., Mekhamer, S., and Ismael, S. M. (2012). Sources and mitigation of harmonics in industrial electrical power systems: State of the art. *Journal of Power and Energy Engineering*, pages 320–332.
- [Atanasoae and Pentiuic, 2014] Atanasoae, P. and Pentiuic, R. (2014). Indices for the power quality monitoring in the romanian power transmission system. In *2014 16th International Conference on Harmonics and Quality of Power (ICHQP)*, pages 68–71.
- [Cheng and Yang, 2009] Cheng, L. and Yang, Q. (2009). Thread priority sensitive simultaneous multi-threading fair scheduling strategy. In *2009 International Conference on Computational Intelligence and Software Engineering*, pages 1–4.
- [Cobben et al., 2007] Cobben, J., Bhattacharyya, S., Myrzik, J., and Kling, W. (2007). Integrated approach for power quality requirements at the point of connection. In *2007 9th International Conference on Electrical Power Quality and Utilisation*, pages 1–5.
- [Cooley and Tukey, 1965] Cooley, J. W. and Tukey, J. W. (1965). An algorithm for the machine calculation of complex fourier series. *Math. Comput.*, pages 297–301.
- [C.Sankaran, 2002] C.Sankaran (2002). *Power Quality*. Press CRC.
- [Dugan et al., 2002] Dugan, R., Santoso, S., McGranaghan, M., and Beaty, H. (2002). *Electrical Power Systems Quality*. McGraw-Hill professional engineering. McGraw-Hill Education.

- [Eghtedarpour et al., 2014] Eghtedarpour, N., Karimi, M., and Tavakoli, M. (2014). Harmonic resonance in power systems - a documented case. In *2014 16th International Conference on Harmonics and Quality of Power (ICHQP)*, pages 857–861.
- [Foster, 1995] Foster, I. (1995). *Designing and building parallel programs: concepts and tools for parallel software engineering*. Massachusetts, addison-wesley edition.
- [Giloi, 1993] Giloi, W. (1993). Parallel programming models and their interdependence with parallel architectures. In *Proceedings of Workshop on Programming Models for Massively Parallel Computers*, pages 2–11.
- [González, 2003] González, C. C. (2003). *Fundamentos del análisis de Fourier*. Vigo. GAMESAL.
- [Grady and Santoso, 2001] Grady, W. and Santoso, S. (2001). Understanding power system harmonics. *Power Engineering Review, IEEE*, pages 8 – 11.
- [Ise et al., 2000] Ise, T., Hayashi, Y., and Tsuji, K. (2000). Definitions of power quality levels and the simplest approach for unbundled power quality services. In *Ninth International Conference on Harmonics and Quality of Power. Proceedings (Cat. No.00EX441)*, volume 2, pages 385–390.
- [José I. et al., 2012] José I., R., César E., L., and Fredy H., M. (2012). Implementation of FFT transform on a FPGA oriented toward its application in power electronic converters. *Revista Tekhnê*, pages 21 – 32.
- [Kennedy, 2000] Kennedy, B. W. (2000). *Power Quality Primer*. McGraw-Hill Companies.
- [Kumar and Zare, 2014] Kumar, D. and Zare, F. (2014). Analysis of harmonic mitigations using hybrid passive filters. In *2014 16th International Power Electronics and Motion Control Conference and Exposition*, pages 945–951.
- [Liu et al., 2012] Liu, B., Zydek, D., Selvaraj, H., and Gewali, L. (2012). Accelerating high performance computing applications: Using CPUs, GPUs, hybrid CPU/GPU, and FPGAs. In *2012 13th International Conference on Parallel and Distributed Computing, Applications and Technologies*, pages 337–342.

- [Long et al., 2013] Long, Y., Xiao, X., Xu, Y., Yu, B., Xu, Y., and Hao, J. (2013). MMC-UPQC: Application of modular multilevel converter on unified power quality conditioner. In *2013 IEEE Power Energy Society General Meeting*, pages 1–5.
- [Miao et al., 2009] Miao, Q., Huang, M., and Wei, Q. (2009). Parallel computing of numerical manifold method with OpenMP. In *2009 IEEE Youth Conference on Information, Computing and Telecommunication*, pages 174–177.
- [Michael, 2010] Michael, K. (2010). *The Linux Programming Interface*. A Linux and UNIX System Programming Handbook. No Starch Press.
- [Mohan et al., 2003] Mohan, N., Undeland, T. M., and Robbins, W. P. (2003). *Power Electronics. Converters, Applications and Design*. John Wiley and Sons, Inc, third edition.
- [Mohapatra and Mohapatra, 2017] Mohapatra, B. N. and Mohapatra, R. K. (2017). FFT and sparse FFT techniques and applications. In *2017 Fourteenth International Conference on Wireless and Optical Communications Networks (WOCN)*, pages 1–5.
- [Mora-Barajas and Banuelos-Sanchez, 2010] Mora-Barajas, M. and Banuelos-Sanchez, P. (2010). Contaminación armónica producida por cargas no lineales de baja potencia: modelo matemático y casos prácticos. *Ingeniería, investigación y tecnología*, pages 189 – 198.
- [Motta and Faúndes, 2016] Motta, L. and Faúndes, N. (2016). Active / passive harmonic filters: Applications, challenges trends. In *2016 17th International Conference on Harmonics and Quality of Power (ICHQP)*, pages 657–662.
- [Quin, 2003] Quin, M. J. (2003). *Parallel Programming in C with MPI and OpenMP*. McGraw-Hill Companies.
- [Ramos-Paz, 2007] Ramos-Paz, A. (2007). *Técnica para la Generación Automática de Ecuaciones Diferenciales No Autónomas para Representar el Comportamiento Dinámico de Sistemas Eléctricos NoLineales Incorporando Herramientas Avanzadas de Cómputo*. Tesis de Doctorado. UMSNH-FIE-DEP.

- [Singh et al., 1999] Singh, B., Al-Haddad, K., and Chandra, A. (1999). A review of active filters for power quality improvement. *IEEE Transactions on Industrial Electronics*, 46(5):960–971.
- [Smith, 2002] Smith, C. W. (2002). Power systems and harmonic factors. *IEEE Potentials*, 20(5):10–12.
- [Sundaram et al., 1994] Sundaram, A., Mehta, H., and Tahiliani, V. (1994). Custom power - eprl’s response to power quality issues. pages 398–403.
- [Suslov et al., 2018] Suslov, K., Solonina, N., and Gerasimov, D. (2018). Assessment of an impact of power supply participants on power quality. In *2018 18th International Conference on Harmonics and Quality of Power (ICHQP)*, pages 1–5.
- [ANSI/IEEE Std 18, 1980] ANSI/IEEE Std 18 (1980). Ieee standard for shunt power capacitors. *ANSI/IEEE Std 18-1980*, pages 1–23.
- [IEEE Std 1159, 2019] IEEE Std 1159 (2019). IEEE recommended practice for monitoring electric power quality - redline. *IEEE Std 1159-2019 (Revision of IEEE Std 1159-2009) - Redline*, pages 1–180.
- [IEEE Std 1459-2010, 2010] IEEE Std 1459-2010 (2010). IEEE standard definitions for the measurement of electric power quantities under sinusoidal, nonsinusoidal, balanced, or unbalanced conditions. *IEEE Std 1459-2010 (Revision of IEEE Std 1459-2000)*, pages 1–50.
- [IEEE Std 519, 1993] IEEE Std 519 (1993). IEEE recommended practices and requirements for harmonic control in electrical power systems. *IEEE Std 519-1992*, pages 1–112.
- [IEEE Std 519, 2014] IEEE Std 519 (2014). IEEE recommended practice and requirements for harmonic control in electric power systems. *IEEE Std 519-2014 (Revision of IEEE Std 519-1992)*, pages 1–29.
- [Std EN 50160, 1995] Std EN 50160 (1995). Application guide to the european standard en 50160 on “voltage characteristics of electricity supplied by public distribution systems”. In *GuideToEN50160*, pages 1–44.

- 
- [Wagner et al., 1993] Wagner, V. E., Balda, J. C., Griffith, D. C., McEachern, A., Barnes, T. M., Hartmann, D. P., Phileggi, D. J., Emmanuel, A. E., Horton, W. F., Reid, W. E., Ferraro, R. J., and Jewell, W. T. (1993). Effects of harmonics on equipment. *IEEE Transactions on Power Delivery*, pages 672–680.
- [Zhang et al., 2013] Zhang, X., Shen, K., Xu, C., and Wang, K. (2013). Design and implementation of parallel FFT on CUDA. In *2013 IEEE 11th International Conference on Dependable, Autonomic and Secure Computing*, pages 583–589.
- [Zhao and Zhao, 2018] Zhao, Z. and Zhao, Y. (2018). The optimization of FFT algorithm based with parallel computing on GPU. In *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pages 2003–2007.