

UCUENCA

Facultad de Ingeniería

Maestría en Electricidad Mención Redes Eléctricas Inteligentes

Restauración del servicio en sistemas de distribución considerando clientes prioritarios y microrredes

Trabajo de titulación previo a la obtención del título de Magíster en Electricidad Mención Redes Eléctricas Inteligentes

Autor:

Eduardo Luis Campoverde Encalada

CI: 1105161739

Correo electrónico: eduardce_925@hotmail.com

Director:

Santiago Patricio Torres Contreras

CI: 0102448958

Cuenca, Ecuador

26-octubre-2022

Resumen:

La instalación de equipamiento de automatización de distribución (DA) permite la implementación de la aplicación avanzada FLISR (Fault Location, Isolation and Service Restoration). Esta implementación proporciona a la red la capacidad de autorreparación para mejorar su resiliencia.

La restauración del servicio (SR) es un componente esencial de la aplicación FLISR. Permite que luego de detectada y aislada una falla permanente en un área del sistema, se genere y ejecute automáticamente un plan para restaurar el servicio en las áreas no afectadas, empleando el menor número de maniobras. Existen algunas investigaciones que se enfocan en una solución de SR con diferentes arquitecturas de automatización y técnicas de optimización. Sin embargo, los algoritmos son limitados y requieren de más estudios para tener en cuenta escenarios como clientes prioritarios, integración de reconfiguración de red y operación de microrredes, incertidumbres de recursos energéticos distribuidos (DER) y cargas; con el propósito de lograr una red inteligente capaz de autorrepararse de eventos extremos.

El presente trabajo de titulación propone un algoritmo de optimización centralizado basado en la combinación de técnicas metaheurísticas de Evolución Diferencial (DE) y Aprendizaje Incremental Basado en Población Continúa (PBILc) para resolver el problema de la restauración del servicio en Sistemas de Distribución. Este algoritmo considera los clientes prioritarios y el despacho de generación distribuida (DG) para soporte de la red principal, o alimentación local de las cargas, formándose microrredes. De esta forma, se provee al sistema de capacidad de autorreparación para atender a más escenarios de restauración.

Palabras clave: Clientes prioritarios. Generación distribuida. Metaheurística híbrida DE-PBILc. Reconfiguración de red. Restauración del servicio.

Abstract:

The distribution automation (DA) allows the implementation of the advanced application FLISR (Fault Location, Insolation, and Service Restoration). This implementation provides the network the self-healing capability to improve its resilience.

Service Restoration (SR) is an essential component of the FLISR application. It allows generating and executing automatically a plan to restore the service in healthy areas using the least number of maneuvers after detecting and isolating a permanent fault in the system area. There is some research that focuses on a solution based on SR with different architectures of automatization and optimization techniques. However, the algorithms are limited and require more work to consider scenarios such as priority clients, network reconfiguration and operation of microgrids, uncertainties of distributed energy resources (DER) and loads, to achieve a smart grid capable of self-healing itself from extreme events.

The present research proposes an optimization centralized algorithm based on the combination of metaheuristic techniques of Differential Evolution (DE) and Continuous Population-Based Incremental Learning (PBILc) to solve the service restoration problem in Distribution Systems. This algorithm considers priority clients and distributed generation (DG) dispatch to support the main network, or local power supply of the loads by micro-grids. In this way, the system is provided with a self-healing capacity to meet more restoration scenarios.

Keywords: Priority customers. Distributed generation. Hybrid metaheuristic DE-PBILc. Network reconfiguration. Service restoration.

Índice de Contenidos

INTRODUCCIÓN.....	14
1 APLICACIÓN AVANZADA DE AUTOMATIZACIÓN DE DISTRIBUCIÓN FLISR	16
1.1 Introducción.....	16
1.2 Aplicación avanzada de la automatización de distribución FLISR.....	16
1.3 Desafíos para la implementación de la aplicación FLISR.....	18
1.4 Beneficios de la aplicación FLISR.....	19
1.5 Restricciones a satisfacer por la aplicación FLISR.....	20
1.6 Pasos de la aplicación FLISR	20
1.6.1 Detección de la falla	21
1.6.2 Localización de la falla.....	21
1.6.3 Aislamiento de la falla.....	22
1.6.4 Restauración del servicio.....	23
1.6.5 Secuencia de reversa	23
2 RESTAURACIÓN DEL SERVICIO EN LOS SISTEMAS DE DISTRIBUCIÓN.....	24
2.1 Introducción.....	24
2.2 Arquitecturas de Automatización para la restauración del servicio.....	26
2.3 Arquitectura de automatización sin sistemas de comunicación.....	26
2.4 Arquitecturas de automatización con sistemas de comunicación.....	28
2.4.1 Arquitectura centralizada	28
2.4.1.1 <i>Sistema Experto</i>	30
2.4.1.2 <i>Algoritmos Heurísticos</i>	31
2.4.1.3 <i>Teoría de grafos</i>	32
2.4.1.4 <i>Programación Matemática</i>	33
2.4.1.5 <i>Algoritmos Metaheurísticos</i>	34
2.4.2 Arquitecturas distribuida y jerárquica	35
2.4.2.1 <i>Sistema Multi-agente (MAS)</i>	38
2.4.2.2 <i>Programación matemática</i>	39
2.5 Comparativa entre las arquitecturas de automatización.....	39
2.6 Aplicaciones prácticas de restauración del servicio	41
3 METODOLOGÍA DEL ALGORITMO DE RESTAURACIÓN DEL SERVICIO	43
3.1 Introducción.....	43
3.2 Formulación matemática.....	44
3.2.1 Reconfiguración del sistema para restauración del servicio.....	44

3.2.2	Unidades de generación distribuida para restauración del servicio.....	46
3.2.3	Clientes prioritarios para restauración del servicio.....	47
3.3	Técnicas metaheurísticas.....	48
3.3.1	Evolución diferencial (DE).....	49
3.3.2	Aprendizaje incremental basado en población continua (PBILc).....	51
3.3.3	Algoritmo DE-PBILc.....	52
3.4	Implementación del algoritmo de Restauración del Servicio.....	54
4	RESULTADOS DEL ALGORITMO DE RESTAURACIÓN DEL SERVICIO.....	69
4.1	Introducción.....	69
4.2	Sistema de prueba estándar.....	69
4.3	Casos de estudio.....	70
4.3.1	Caso 1: Reconfiguración del sistema para la restauración.....	71
4.3.1.1	<i>Localización de falla en líneas 7, 9, 14, 28 y 32.....</i>	<i>71</i>
4.3.1.2	<i>Localización de falla en líneas 3, 4 y disparo en línea 9.....</i>	<i>71</i>
4.3.2	Caso 2: Integración de unidades DG para la restauración.....	72
4.3.2.1	<i>Localización de falla en líneas 4, 28 y 29 con integración DG.....</i>	<i>73</i>
4.3.3	Caso 3: Consideración de clientes prioritarios para la restauración.....	73
4.3.3.1	<i>Localización de falla en líneas 4, 28 y 29.....</i>	<i>74</i>
4.3.3.2	<i>Localización falla en líneas 4, 28 y 29 con adición DG parcial.....</i>	<i>75</i>
4.4	Resultados de la simulación.....	76
4.4.1	Resultados para Caso 1.....	77
4.4.1.1	<i>Plan SR - Falla en líneas 7, 9, 14, 28 y 32.....</i>	<i>77</i>
4.4.1.2	<i>Plan SR - Falla en líneas 3, 4 y disparo en línea 9.....</i>	<i>80</i>
4.4.2	Resultados para Caso 2.....	83
4.4.2.1	<i>Plan SR - Falla en líneas 4, 28 y 29 con integración DG.....</i>	<i>83</i>
4.4.3	Resultados para Caso 3.....	85
4.4.3.1	<i>Plan SR - Falla en líneas 4, 28 y 29.....</i>	<i>85</i>
4.4.3.2	<i>Plan SR - Falla en líneas 4, 28 y 29 con adición DG parcial.....</i>	<i>88</i>
4.5	Comparación de desempeño del algoritmo de SR.....	91
4.5.1	Comparación de resultados para Caso 1.....	91
4.5.1.1	<i>Plan SR - Falla en líneas 7, 9, 14, 28 y 32.....</i>	<i>91</i>
4.5.1.2	<i>Plan SR - Falla en líneas 3, 4, y disparo en línea 9.....</i>	<i>93</i>
4.5.2	Comparación de resultados para Caso 2.....	96
4.5.2.1	<i>Plan SR - Falla en líneas 4, 28 y 29 con integración DG.....</i>	<i>96</i>
4.5.3	Comparación de resultados para Caso 3.....	98
4.5.3.1	<i>Plan SR - Falla en líneas 4, 28 y 29.....</i>	<i>98</i>
4.5.3.2	<i>Plan SR - Falla en líneas 4, 28 y 29 con adición DG parcial.....</i>	<i>100</i>
4.6	Extensión de análisis de desempeño del algoritmo de SR.....	102
4.6.1	Influencia del procesamiento en paralelo.....	102
4.6.2	Impacto del tamaño de la población.....	105

5	CONCLUSIONES Y RECOMENDACIONES	109
5.1	Conclusiones	109
5.2	Recomendaciones	111
	ANEXO A DATOS DE RED DEL SISTEMA DE PRUEBA ESTÁNDAR.....	112
A.1	Datos de nodos	112
A.2	Datos de líneas.....	113
A.2.1.	Líneas operativas	113
A.2.2.	Líneas de enlace	114
	REFERENCIAS	115

Índice de Figuras

Figura 1.1 Restauración del servicio en secciones no afectadas del alimentador sin la inteligencia FLISR [6].	17
Figura 1.2 Restauración del servicio en secciones no afectadas del alimentador red con la inteligencia FLISR [6].	18
Figura 1.3 Secuencia de pasos y tecnologías usadas para la aplicación FLISR [10].	20
Figura 1.4 Detección de la falla, paso 1 de la aplicación FLISR [6].	21
Figura 1.5 Localización de la falla, paso 2 de la aplicación FLISR [6].	22
Figura 1.6 Aislamiento de la falla, paso 3 de la aplicación FLISR [6].	22
Figura 1.7 Restauración del servicio, paso 4 de la aplicación FLISR [6].	23
Figura 2.1 Topología inicial del sistema de prueba [4].	25
Figura 2.2 Procedimientos de restauración de servicio en el sistema de prueba [4].	25
Figura 2.3 Interruptor seccionador para análisis de estado abierto/cerrado de red [12].	27
Figura 2.4 Arquitecturas de automatización con uso de sistemas de comunicación [11].	28
Figura 2.5 Componentes de la arquitectura de control centralizada [6].	29
Figura 2.6 Componentes del sistema experto [4].	30
Figura 2.7 Gráfico abstracto del sistema de prueba tras el aislamiento de la falla en la Zona 1 [4].	32
Figura 2.8 Escenario de aplicación de inteligencia distribuida para la restauración del servicio [14].	36
Figura 2.9 Comparativa entre el Tiempo de Respuesta y la Complejidad de las arquitecturas de automatización [6].	39
Figura 3.1 Generación del vector mutante v_i en DE [41].	49
Figura 3.2 Ilustración de evolución de PDF en PBILc [42].	51
Figura 3.3 Diagrama de flujo del algoritmo para la resolución del problema de SR.	55
Figura 3.4 Diagrama de flujo del análisis de estado de falla para la obtención del camino de restauración.	58
Figura 3.5 Diagrama de flujo de metodología de deslastre de carga según la prioridad de clientes.	60
Figura 3.6 Diagrama de flujo de metodología de interruptores de línea a conmutar según el deslastre de carga.	61
Figura 3.7 Diagrama de flujo de generación de la población inicial considerando la conmutación de interruptores de línea del sistema.	63
Figura 3.8 Diagrama de flujo de generación de la población inicial considerando la conmutación de interruptores de línea y las unidades DG del sistema.	64
Figura 3.9 Diagrama de flujo de generación de la población inicial considerando la conmutación de interruptores de línea, las unidades DG y el deslastre de clientes.	65
Figura 4.1 Diagrama unifilar del sistema de distribución de 33 nodos de IEEE [43].	70
Figura 4.2 Localización de falla en líneas 7, 9, 14, 28 y 32 en el sistema de prueba estándar.	71

Figura 4.3 Localización de falla en líneas 3, 4 y disparo en línea 9 en el sistema de prueba estándar.....	72
Figura 4.4 Localización de falla en líneas 4, 28 y 29 en el sistema de prueba estándar con la integración de unidades DG.	73
Figura 4.5 Localización de falla en líneas 4, 28 y 29 en el sistema de prueba estándar... 75	75
Figura 4.6 Localización de falla en líneas 4, 28 y 29 en el sistema de prueba estándar con la integración de unidades DG 2 a 4.	76
Figura 4.7 Proceso de convergencia del algoritmo de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.....	77
Figura 4.8 Topología de red resultante del algoritmo de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.....	79
Figura 4.9 Proceso de convergencia del algoritmo de SR para Caso 1 – Falla en líneas 3, 4 y disparo en 9. (a) Gráfica en vista original. (b) Gráfica aumentada.....	81
Figura 4.10 Topología de red resultante del algoritmo de SR para Caso 1 – Falla en líneas 3, 4 y disparo en línea 9.	82
Figura 4.11 Proceso de convergencia del algoritmo de SR para Caso 2 – Falla en líneas 4, 28 y 29 con integración DG.....	84
Figura 4.12 Topología de red resultante del algoritmo de SR para Caso 2 - Falla en líneas 4, 28 y 29 con integración DG.....	85
Figura 4.13 Proceso de convergencia del algoritmo de SR para Caso 3 – Falla en líneas 4, 28 y 29.....	86
Figura 4.14 Topología de red resultante del algoritmo de SR para Caso 3 - Falla en líneas 4, 28 y 29.....	88
Figura 4.15 Proceso de convergencia del algoritmo de SR para Caso 3 – Falla en líneas 4, 28 y 29 con integración DG parcial.	89
Figura 4.16 Topología de red resultante del algoritmo de SR para Caso 3 - Falla en líneas 4, 28 y 29 con integración DG parcial.	91
Figura 4.17 Proceso de convergencia de los algoritmos de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32. (a) Técnica DE. (b) Técnica PBILc.	92
Figura 4.18 Proceso de convergencia del algoritmo con DE para Caso 1 – Falla en líneas 3, 4 y disparo en 9. (a) Gráfica en vista original. (b) Gráfica aumentada.....	94
Figura 4.19 Proceso de convergencia del algoritmo con PBILc para Caso 1 – Falla en líneas 3, 4 y disparo en 9. (a) Gráfica en vista original. (b) Gráfica aumentada.....	95
Figura 4.20 Proceso de convergencia de los algoritmos de SR para Caso 2 – Falla en líneas 4, 28 y 29 con integración DG. (a) Técnica DE. (b) Técnica PBILc.....	97
Figura 4.21 Proceso de convergencia de los algoritmos de SR para Caso 3 – Falla en líneas 4, 28 y 29. (a) Técnica DE. (b) Técnica PBILc.....	99
Figura 4.22 Proceso de convergencia de los algoritmos de SR para Caso 3 – Falla en líneas 4, 28 y 29 con integración DG parcial. (a) Técnica DE. (b) Técnica PBILc.	101
Figura 4.23 Proceso de convergencia del algoritmo de SR para Caso 2 al variar los núcleos del procesamiento en paralelo. (a) 2 núcleos. (b) 4 núcleos. (c) 6 núcleos.	104
Figura 4.24 Proceso de convergencia del algoritmo de SR para Caso 2 al variar el tamaño de la población. (a) 20 individuos. (b) 40 individuos . (c) 80 individuos.	107

Índice de Tablas

Tabla 2.1 Criterios de selección de las arquitecturas de automatización [6], [11].	40
Tabla 2.2 Aplicaciones DA en arquitecturas de automatización [6], [11].	40
Tabla 3.1 Niveles de criticidad de carga [3].	48
Tabla 3.2 Parámetros de metaheurísticas DE-PBILc.	56
Tabla 3.3 Costos de operaciones de restauración en la función objetivo de SR.	67
Tabla 4.1 Localización y capacidad de unidades DG añadidas al sistema estándar [47].	72
Tabla 4.2 Niveles de criticidad de carga asignados en el sistema de prueba estándar.	74
Tabla 4.3 Plan de restauración del algoritmo de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.	78
Tabla 4.4 Desempeño del algoritmo de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.	78
Tabla 4.5 Plan de restauración del algoritmo de SR para Caso 1 – Falla en líneas 3, 4 y disparo en línea 9.	80
Tabla 4.6 Desempeño del algoritmo de SR para Caso 1 – Falla en líneas 3, 4 y disparo en línea 9.	82
Tabla 4.7 Plan de restauración del algoritmo de SR para Caso 2 – Falla en líneas 4, 28 y 29 con integración DG.	83
Tabla 4.8 Desempeño del algoritmo de SR para Caso 2 - Falla en líneas 4, 28 y 29 con integración DG.	84
Tabla 4.9 Plan de restauración del algoritmo de SR para Caso 3 – Falla en líneas 4, 28 y 29.	87
Tabla 4.10 Desempeño del algoritmo para Caso 3 - Falla en líneas 4, 28 y 29.	87
Tabla 4.11 Plan de restauración del algoritmo para Caso 3 – Falla en líneas 4, 28 y 29 con integración DG parcial.	89
Tabla 4.12 Desempeño del algoritmo de SR para Caso 3 - Falla en líneas 4, 28 y 29 con integración DG parcial.	90
Tabla 4.13 Desempeño de los algoritmos de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.	93
Tabla 4.14 Desempeño de los algoritmos de SR para Caso 1 – Falla en líneas 3, 4 y disparo en línea 9.	96
Tabla 4.15 Desempeño de los algoritmos de SR para Caso 2 – Falla en líneas 4, 28 y 29 con integración DG.	98
Tabla 4.16 Desempeño de los algoritmos de SR para Caso 3 – Falla en líneas 4, 28 y 29.	100
Tabla 4.17 Desempeño de los algoritmos de SR para Caso 3 – Falla en líneas 4, 28 y 29 con integración DG parcial.	102
Tabla 4.18 Desempeño del algoritmo de SR para Caso 2 al variar los núcleos del procesamiento en paralelo.	105
Tabla 4.19 Desempeño del algoritmo de SR para Caso 2 al variar el tamaño de la población.	107

Cláusula de licencia y autorización para publicación en el Repositorio Institucional

Eduardo Luis Campoverde Encalada en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de titulación "RESTAURACIÓN DEL SERVICIO EN SISTEMAS DE DISTRIBUCIÓN CONSIDERANDO CLIENTES PRIORITARIOS Y MICRORREDES", de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN reconozco a favor de la Universidad de Cuenca una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos.

Asimismo, autorizo a la Universidad de Cuenca para que realice la publicación de este trabajo de titulación en el repositorio institucional, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Cuenca, 26 de octubre del 2022



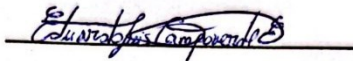
Eduardo Luis Campoverde Encalada

C.I: 1105161739

Cláusula de Propiedad Intelectual

Eduardo Luis Campoverde Encalada, autor del trabajo de titulación "RESTAURACIÓN DEL SERVICIO EN SISTEMAS DE DISTRIBUCIÓN CONSIDERANDO CLIENTES PRIORITARIOS Y MICRORREDES", certifico que todas las ideas, opiniones y contenidos expuestos en la presente investigación son de exclusiva responsabilidad de su autor.

Cuenca, 26 de octubre del 2022



Eduardo Luis Campoverde Encalada

C.I: 1105161739

Agradecimientos

A Dios, por permitirme estar con vida y culminar de manera satisfactoria esta etapa de formación profesional.

A mi mamá, por su amor incondicional y su apoyo en las situaciones más difíciles, y a mis familiares, por estar presentes en cada momento y haber contribuido en mi formación.

A la Universidad de Cuenca, por su ardua gestión para ofertar este vanguardista Programa de Maestría que fortalece el desarrollo tecnológico-científico de los profesionales, proporcionándonos los conocimientos y herramientas necesarias para afrontar los retos de los futuros sistemas de energía eléctrica.

Finalmente, a cada uno de los docentes de este programa, por compartir sus conocimientos y experiencias que enriquecieron mi formación profesional. De manera especial a mi director de tesis, por su colaboración y guía en la realización de este proyecto de titulación, gracias.

Eduardo Luis

Dedicatoria

A mi mamá, Jenny, por ser el pilar fundamental en mi vida y fuente de motivación para afrontar este y otros desafíos. Para mi abuelita Regina, por el amor y cuidado que me ha brindado en todos estos años. A mi hermano Cristian y a mis tíos Tobar y Colón, que han estado presentes en cada etapa de mi vida. Para mis abuelitos Pancho (+), Amadita (+) y Licho (+), que me protegen desde arriba.

Eduardo Luis

INTRODUCCIÓN

Los sistemas de distribución operan en la práctica en configuración radial. Esta topología de red ofrece una baja confiabilidad; esto es, una falla en cualquier componente producirá una interrupción para todas las cargas aguas abajo de la localización de falla.

En la actualidad, las empresas distribuidoras efectúan inversiones en sistemas tecnológicos para mejorar la confiabilidad, robustez, reducción de pérdidas, reducción de fallas y extensión de la vida útil de los equipos de distribución. La instalación de equipamiento de automatización de distribución (Distribution automation - DA) permite implementar la aplicación avanzada FLISR (Fault Location, Insolation and Service Restoration), que proporciona a la red la capacidad de autorreparación (self-healing) para mejorar su resiliencia.

La aplicación FLISR se compone de dos etapas: *Localización de la falla y aislamiento*, que detecta la ubicación de la falla permanente y aísla las zonas defectuosas lo más rápido posible, y la *restauración del servicio (SR)*, que ejecuta un plan de maniobras para minimizar el número de clientes interrumpidos y operaciones de conmutación, sin violar las restricciones operativas y eléctricas del sistema como son: balance de potencia, límites de voltaje, capacidad de corriente de los conductores y configuración radial.

Existen algunos trabajos investigativos que se enfocan en la solución de SR con diferentes arquitecturas de automatización y técnicas de optimización. Sin embargo, los algoritmos de optimización son limitados y se requieren más estudios que tengan en cuenta escenarios como clientes prioritarios, integración de reconfiguración de red y operación de microrredes, incertidumbres de recursos energéticos distribuidos (DER) y cargas; con el propósito de lograr una red inteligente capaz de autorrepararse de eventos extremos.

Para implementar la inteligencia de SR, los algoritmos de optimización distribuidos están en auge, en especial para redes extensas y modernas. Sin embargo, con sistemas de automatización y comunicación heredados de las empresas distribuidoras este enfoque puede no resultar viable. En aplicaciones prácticas, los enfoques centralizados que usan algoritmos basados en reglas lógicas se prefieren dado que pueden brindar soluciones de buena calidad en tiempos de cómputo razonables, y su implementación resulta menos compleja desde el punto de vista técnico, en especial para sistemas de pequeña escala.

El presente trabajo de titulación propone un algoritmo de optimización centralizado basado en una metaheurística híbrida que combina las técnicas de Evolución Diferencial (DE) y Aprendizaje Incremental Basado en Población Continua (PBILc) para resolver el problema de la restauración del servicio en sistemas de distribución. Este algoritmo considera los clientes prioritarios y el despacho de generación distribuida (DG) para soporte de la red principal o alimentación local de las cargas. De esta forma, se dota al sistema de capacidad de autorreparación para atender a más escenarios de restauración, teniendo en cuenta que pueden existir cargas donde no es admisible la pérdida del servicio eléctrico (ej. Hospitales, bancos, grandes fábricas) y unidades DG que pueden despacharse para alimentar a las cargas localmente, operando en modo microrred.

Las contribuciones que intenta brindar este trabajo investigativo son las siguientes: i) La implementación de un algoritmo de optimización metaheurístico para resolver el problema de la restauración del servicio en sistemas de distribución y ii) la extensión de escenarios de restauración en la obtención de soluciones.

El documento se encuentra estructurado de la siguiente manera. El Capítulo I presenta una descripción de la aplicación avanzada de automatización FLISR en los sistemas de distribución. El Capítulo II contiene la revisión del estado del arte de las diferentes arquitecturas de automatización y técnicas de optimización para resolver el problema de la restauración del servicio en redes de distribución. El Capítulo III presenta la metodología del algoritmo de optimización centralizado basado en las técnicas metaheurísticas DE-PBILc para la solución, que incluye la extensión de los escenarios de restauración. El Capítulo IV proporciona los resultados computacionales de la solución de restauración del servicio en el sistema de prueba estándar usado como red de distribución. Finalmente, el Capítulo V contiene las conclusiones y recomendaciones del trabajo.

Capítulo 1

1 APLICACIÓN AVANZADA DE AUTOMATIZACIÓN DE DISTRIBUCIÓN FLISR

1.1 Introducción

El sistema de distribución es la parte del sistema eléctrico de potencia (SEP) que se encarga del suministro de energía desde la subestación de distribución hasta los usuarios finales [1]. Tradicionalmente, opera en configuración radial para un diseño simple, de bajo costo y una coordinación de protecciones sencilla [2]. Esta topología de red ofrece una baja confiabilidad, una falla en cualquier componente producirá una interrupción para todas las zonas aguas abajo de la localización de falla.

Las estadísticas señalan que las principales fallas en las redes de distribución ocurren debido al clima, al desgaste de componentes y a los accidentes [2]. Se estima que el 80 % de todas las interrupciones de la red se encuentran en los sistemas de distribución o son causadas por ellos [3], siendo los principales contribuyentes de la indisponibilidad de energía eléctrica a los usuarios [2], [4].

En la actualidad, las empresas distribuidoras efectúan inversiones en sistemas tecnológicos como son los equipos de seccionamiento inteligente de alimentador, sistemas de comunicación e inteligencia de control para mejorar la confiabilidad, robustez, reducción de pérdidas, reducción de fallas y extensión de la vida útil de los equipos de distribución [5]. La instalación de equipamiento de automatización de distribución (DA) facilita implementar la aplicación avanzada FLISR, que proporciona a la red la capacidad de autorreparación [2], [4], con el propósito de mejorar la confiabilidad y eficiencia de operación.

1.2 Aplicación avanzada de la automatización de distribución FLISR

La aplicación avanzada de automatización de distribución FLISR (Fault Locación, Isolation and Service Restoration) proporciona a la red la capacidad de autorreparación (self-healing), por lo que, en situaciones de emergencia, las redes restablecerán el servicio automáticamente y a la máxima brevedad, a la mayor cantidad de clientes posibles, mientras los grupos operativos efectúan las reparaciones necesarias [6].

La aplicación FLISR permite que el sistema de distribución se restaure de forma automática e inteligente al mejor estado posible [2], ejecutándose un plan de maniobras para re-energizar las secciones no afectadas que se quedaron sin servicio tras una falla permanente en la red. Para su implementación se requiere del uso de equipos de seccionamiento inteligente de alimentador (reconectores, seccionadores, seccionalizadores, indicadores de paso de falla, etc.), sistemas de comunicación y algoritmos de control (lógicas) [2], [6].

Sin la implementación de la aplicación FLISR, las maniobras de operación y búsqueda de fallas son realizadas de forma manual por los grupos operativos, que necesitan recorrer físicamente la ruta del alimentador [6], situación que extiende significativamente el tiempo de restauración del servicio a las secciones no afectadas. La Figura 1.1 ilustra la línea de tiempo con la secuencia de actividades que se ejecutan sin la inteligencia FLISR. Tras la ocurrencia de la falla, los clientes reportan la interrupción del servicio y el grupo operativo se desplaza hasta la ubicación aproximada de la falla identificada por los equipos de seccionamiento de alimentador o estimada mediante OMS (Outage Management System) [6]. Localizado el daño, se aísla la sección con falla y se ejecutan maniobras manuales para restablecer el servicio a las zonas no afectadas. Luego, se procede a efectuar las reparaciones necesarias. Bajo este escenario, los clientes conectados a las secciones no afectadas del alimentador experimentan una interrupción del servicio por varios minutos o incluso horas.

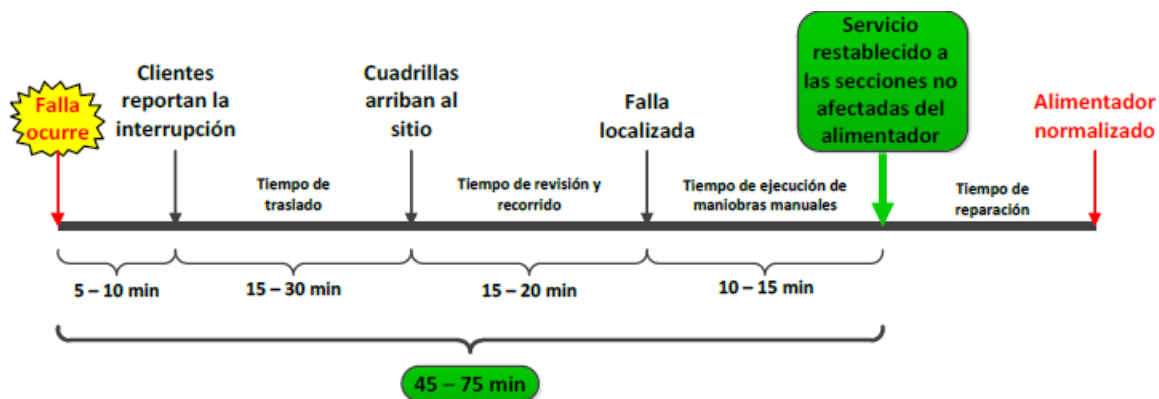


Figura 1.1 Restauración del servicio en secciones no afectadas del alimentador sin la inteligencia FLISR [6].

Con la aplicación FLISR, los clientes conectados a las secciones no afectadas del alimentador experimentan una interrupción del servicio por pocos minutos, como se ilustra en la Figura 1.2. La inteligencia FLISR detecta automáticamente la ocurrencia de la falla, determina su localización, ejecuta los comandos de telecontrol para abrir los equipos de seccionamiento que aíslan la sección con falla, y cierra otros equipos

de seccionamiento para restablecer el servicio a las secciones no afectadas. La ejecución de estas acciones se realiza mediante control automático, es decir, sin la intervención humana [2], [6].

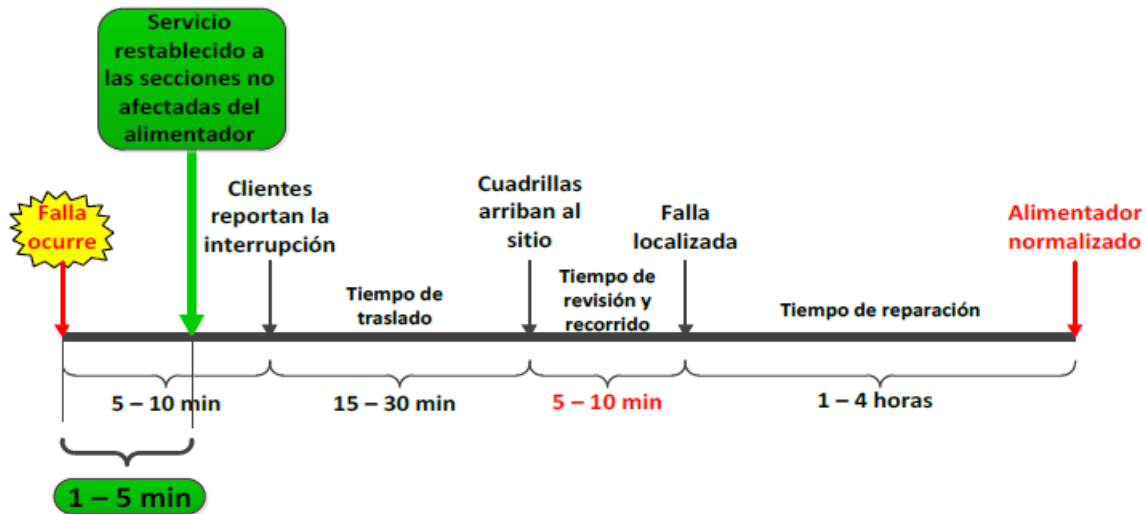


Figura 1.2 Restauración del servicio en secciones no afectadas del alimentador red con la inteligencia FLISR [6].

1.3 Desafíos para la implementación de la aplicación FLISR

La instalación de tecnología de automatización de distribución (DA) es clave para alcanzar un sistema de distribución inteligente [6], [7], que posea la capacidad de autorreparación [4]. Por consiguiente, la implementación de la aplicación FLISR representa desafíos procedimentales y técnicos [2].

Dentro de los desafíos procedimentales se menciona:

- Complejidad de la red: Gran escala de los sistemas de distribución reales. El empleo de redes inteligentes debe ser gradual.
- Falta de estándares unificados: Diversidad de protocolos de telecontrol para sistemas de energía (IEEE 1815 – DNP3, Modbus, IEC 61850, IEC 60870-5-101/104, etc.)

Dentro de los desafíos técnicos se señala:

- Actualización de equipamiento existente: Incluye el remplazo de los equipos de conmutación controlados manualmente (Manually controlled switches – MCS) por equipos de conmutación controlados automáticamente (Automatic controlled switches – ACS), la instalación de modernos sistemas inteligentes como unidades terminales remotas (Remote terminal units – RTU) y

dispositivos electrónicos inteligentes (Intelligent electronic devices – IED) o al menos, repotenciar los componentes hardware/software de los dispositivos instalados permitiendo que cuenten con las capacidades inteligentes necesarias.

- Control distribuido adaptativo: Para que facilite al sistema la operación de autorreparación en tiempo real.
- Incorporación de recursos de energía distribuida (Distributed energy resources – DER) en planes de maniobras para restauración del servicio.
- Redundancia en los enlaces de comunicación de datos.
- Análisis del costo de implementación de redes autorreparables versus el costo de ahorros por indisponibilidad del servicio, reparación y mantenimiento.

El éxito de la aplicación FLISR depende de la implementación de hardware y software en los sistemas de distribución [2]. En cuanto a hardware, como son los sensores, equipamiento de conmutación, sistemas de automatización y de comunicación, y en software, como son los algoritmos para detectar de forma precisa la localización de la falla y para determinar las acciones de conmutación del equipamiento inteligente para aislar la falla o restaurar el servicio.

1.4 Beneficios de la aplicación FLISR

Existe una tendencia creciente en las empresas distribuidoras por implementar la aplicación FLISR debido a los beneficios funcionales y monetarios que provee [8].

Dentro de los beneficios funcionales se menciona:

- Mejora de los índices de confiabilidad: Como son el índice de duración de interrupción media del sistema (system average interruption duration index - SAIDI), el índice de frecuencia de interrupción media del sistema (system average interruption frequency index - SAIFI), y otras estadísticas de confiabilidad.
- Reducción de la energía no suministrada (ENS) (kWh).
- Provisión del servicio de calidad “Premium”.
- Reducción del tiempo de investigación de fallas.

Dentro de los beneficios monetarios se señala:

- Aumento de ingresos: Mayor venta de energía.
- Reducción de costos por demanda no suministrada.
- Ingresos adicionales por clientes de calidad “Premium”.

- Ahorro de mano de obra y vehículos: Debido a la ubicación rápida y precisa de la falla.
- Incentivos regulatorios (en caso de que el ente regulador lo conciba).

1.5 Restricciones a satisfacer por la aplicación FLISR

La aplicación FLISR resuelve y ejecuta un plan de maniobras ante condiciones de emergencia del sistema de distribución. La solución representa un problema computacionalmente complejo debido a que es [2]:

- Combinatorial: Número extenso de equipos de conmutación.
- No lineal: Naturaleza no lineal de las restricciones de flujos de potencia.
- No diferenciable: Cualquier cambio en el estado de un equipo de conmutación puede modificar los valores de la función objetivo o restricciones.
- Restringido: Operación en configuración radial, voltaje operativo y restricciones de corriente.
- Multiobjetivo: Minimizar el número de clientes interrumpidos, las operaciones de conmutación de los equipos y las pérdidas en la red.

1.6 Pasos de la aplicación FLISR

Al ocurrir una falla permanente en la red de distribución, la aplicación FLISR lleva a cabo una secuencia de pasos representados en la Figura 1.3.

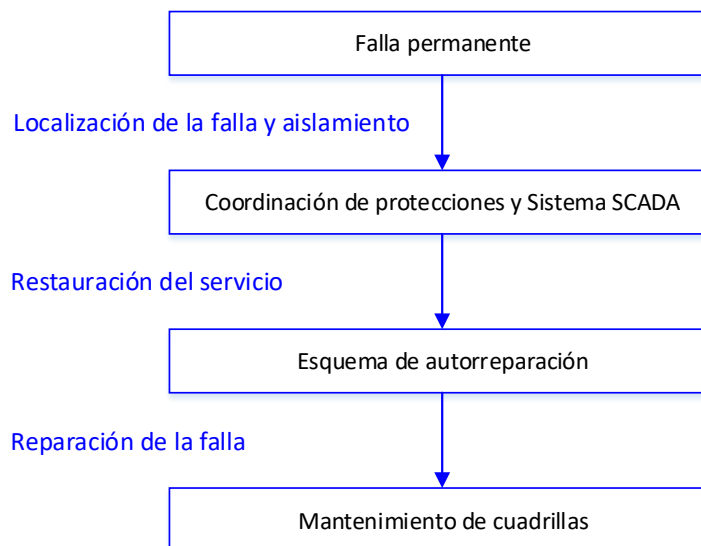


Figura 1.3 Secuencia de pasos y tecnologías usadas para la aplicación FLISR [10].

1.6.1 Detección de la falla

La detección de la falla reside en descubrir que se ha producido una falla, incluso si aún no se conoce la causa raíz [2]. La falla puede ser detectada por señales de alarma basadas en sobrecorrientes o bajos voltajes, registradas por los IED de los interruptores de alimentación (Feeder circuit breakers - FCB) e indicadores de paso de falla (Fault passage indicator - FPI). Estos dispositivos, a través de enlaces de comunicación dedicados, informan inmediatamente al sistema de automatización de distribución (Distribution automation system - DAS) quien inicia la operación FLISR. Esta aplicación debe operar únicamente tras una falla permanente en la red, y no cuando el circuito es desenergizado por maniobras manuales o por el deslastre de carga debido a baja frecuencia. La Figura 1.4 muestra este paso [6], los equipos de conmutación 1 y 2 detectan la ruta de la corriente de falla.

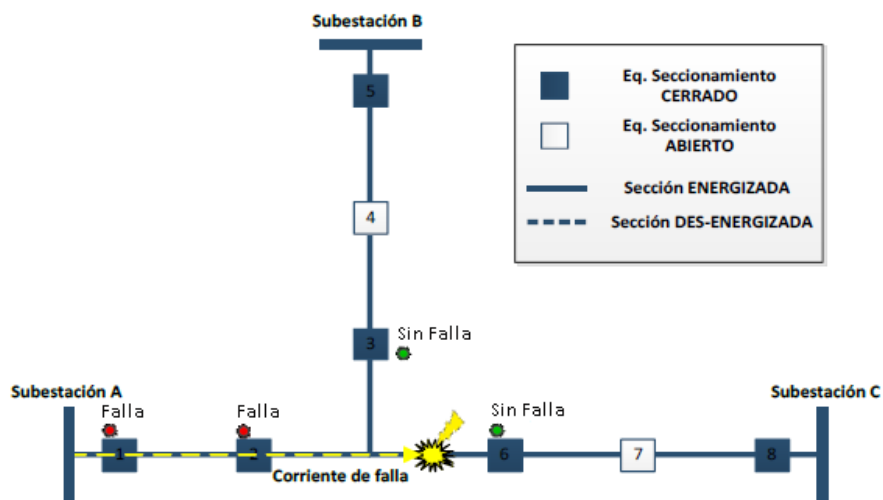


Figura 1.4 Detección de la falla, paso 1 de la aplicación FLISR [6].

1.6.2 Localización de la falla

La localización de la falla consiste en determinar la sección del alimentador que contiene la falla [6]. Se considera como sección a la porción del circuito limitado por equipos de seccionamiento telecontrolados, que incluyen indicadores de falla (FPI) que detectan si una sobrecorriente ha circulado por el tramo. La aplicación FLISR emplea las indicaciones de estado de los FPI y el conocimiento de la topología de red para determinar la sección en falla. La Figura 1.5 representa este paso [6], donde la sección en falla está delimitada por el FPI inicial que registra la indicación de falla activa (equipo 2), y los FPI que no registran alarma (equipos 3 y 6).

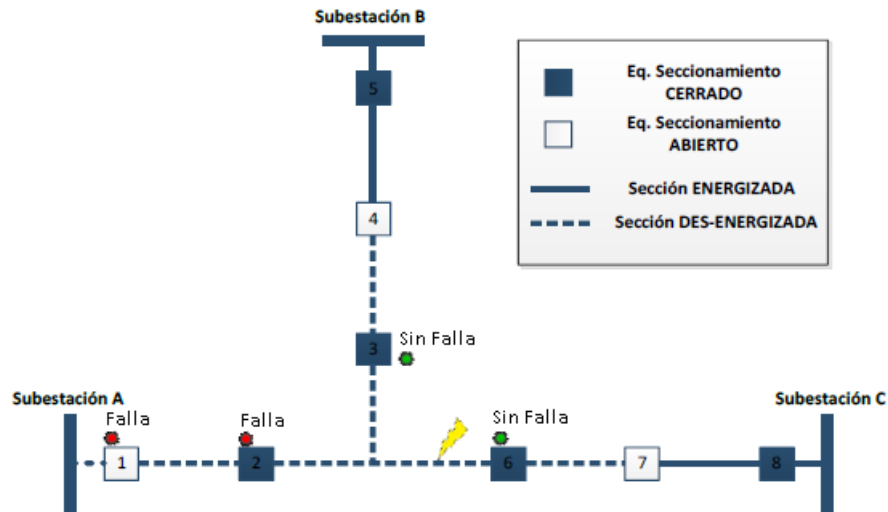


Figura 1.5 Localización de la falla, paso 2 de la aplicación FLISR [6].

1.6.3 Aislamiento de la falla

Este paso consiste en aislar la sección en falla mediante la ejecución de apertura de los primeros interruptores de alimentador (FCB) aguas arriba y aguas abajo de la sección dañada (aislar la falla en ambas direcciones) [2]. La información de falla es analizada de forma automática en el DAS, que posteriormente ejecuta la lógica FLISR para aislar la sección más pequeña del sistema. En aplicaciones prácticas, la inteligencia FLISR debe esperar hasta que las secuencias de recierre estándar se completen antes de ejecutar los comandos [6], para garantizar que la reconfiguración de la red se realice bajo condiciones de falla permanente. La Figura 1.6 escenifica este paso [6], la sección en falla queda aislada por la apertura de los equipos de conmutación 2, 3 y 6.

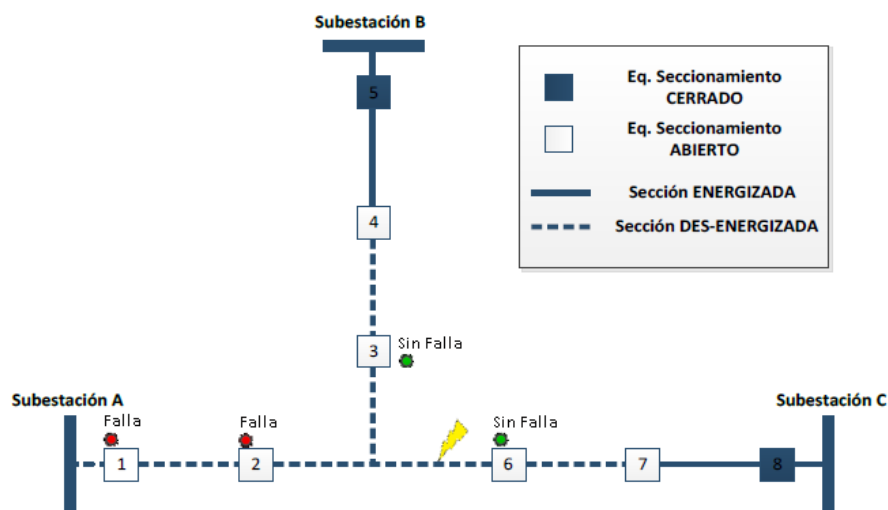


Figura 1.6 Aislamiento de la falla, paso 3 de la aplicación FLISR [6].

1.6.4 Restauración del servicio

Luego de aislada la sección en falla del alimentador, la restauración del servicio (Service restoration - SR) consiste en que se genere y ejecute de forma automática un plan para restablecer el servicio en las zonas no afectadas, empleando el menor número de maniobras [4]. De esta manera, la inteligencia FLISR realiza una *estimación de capacidad* para determinar si las cargas fuera de servicio pueden ser transferidas de forma total o parcial a los alimentadores vecinos no afectados [2]. Si existe la capacidad necesaria, los interruptores de enlace son cerrados para restablecer el servicio a las secciones no afectadas [6], caso contrario, permanecerán desenergizadas hasta que los grupos operativos arriben al sitio y reparen la falla. Si se integran las unidades de generación distribuida (Distributed generation – DG) en los planes de SR, se proporciona una fuente de energía de apoyo y aumenta la oportunidad de restaurar clientes adicionales fuera de servicios [2]. La Figura 1.7 representa este paso [6], el interruptor 1 (aguas arriba de la falla) y los interruptores de enlace (normalmente abiertos) 4 y 7 son cerrados, mientras los interruptores de alimentación 2, 3 y 6 permanecen abiertos a la espera de que se repare la falla.

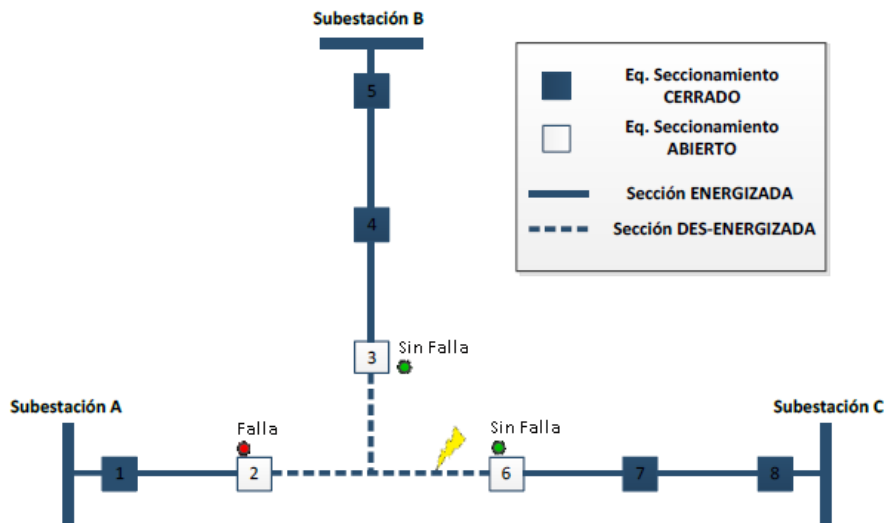


Figura 1.7 Restauración del servicio, paso 4 de la aplicación FLISR [6].

1.6.5 Secuencia de reversa

La secuencia de reversa se aplica luego de que la sección en falla es reparada por los grupos operativos, ejecutándose la conmutación de los equipos para que el sistema de distribución retorne a su configuración normal anterior a la falla [2].

Capítulo 2

2 RESTAURACIÓN DEL SERVICIO EN LOS SISTEMAS DE DISTRIBUCIÓN

2.1 Introducción

La restauración del servicio (Service restoration - SR) es un componente esencial de la aplicación FLISR [4]. Permite que luego de detectada y aislada una falla permanente en un área del sistema, se genere y ejecute de forma automática un plan para restablecer el servicio en las áreas no afectadas, empleando el menor número de maniobras [6]. De esta manera, se mejora la confiabilidad y eficiencia de operación del sistema de distribución.

Para efectuar la restauración del servicio se puede disponer de planes basados en reconfiguración de red y unidades de generación distribuida para soporte de la red principal o alimentación local de las cargas, formando microrredes [4]. Durante la reconfiguración de la red, se deben satisfacer las restricciones de operación como el balance de potencia, los límites de voltaje, la capacidad de corriente de los conductores y la configuración radial. Durante la formación de la microrred, restricciones adicionales deben considerarse como la tasa de respuesta de frecuencia, reserva de giro, la carga incremental máxima. Para ilustrar estos procedimientos se muestra la topología inicial de una red de prueba en la Figura 2.1, que consta de seis zonas y una unidad DG conectada en la Zona 6 [4]. Si ocurre una falla permanente en la Zona 1, los interruptores S1 y S2 se abren para aislar la falla. Seguido, la solución de SR reconfigura la topología de red para restaurar la Zona 2 a través del cierre del interruptor de enlace TS1, como se muestra en la Figura 2.2 [4]. En cambio, si ocurre una falla en la Zona 5, la unidad DG se desconecta para evitar impactos adversos de acuerdo con las políticas de operación tradicionales [2], [4], y los interruptores S5 y S6 se abren para aislar la falla. La ruta de restauración para la Zona 5 no está disponible. Por consiguiente, se despacha la unidad DG para alimentar a las cargas localmente, formándose una microrred en la Zona 6 que se ilustra en la Figura 2.2 [4].

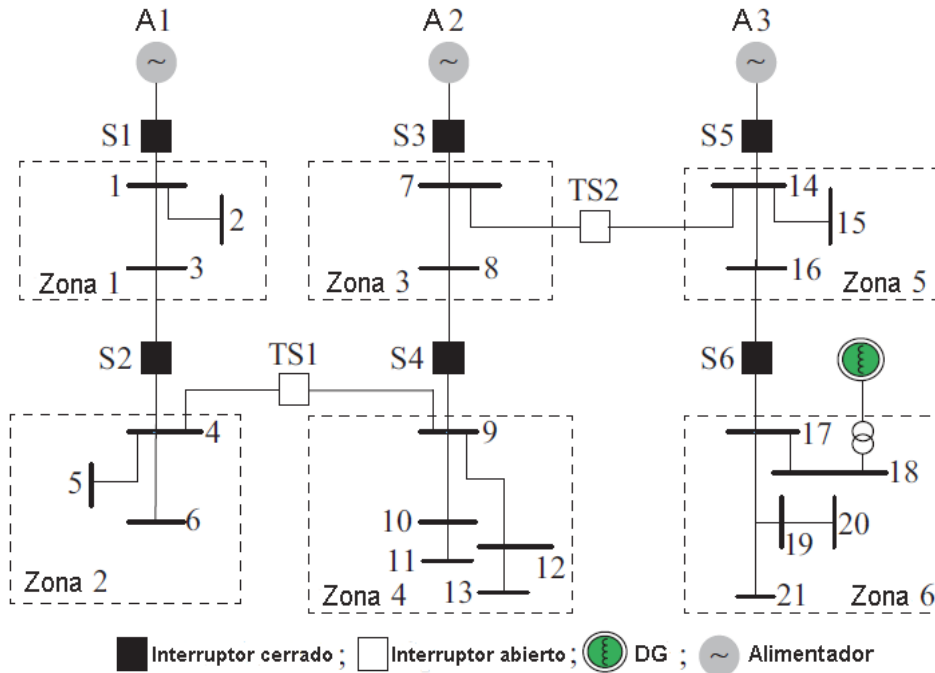


Figura 2.1 Topología inicial del sistema de prueba [4].

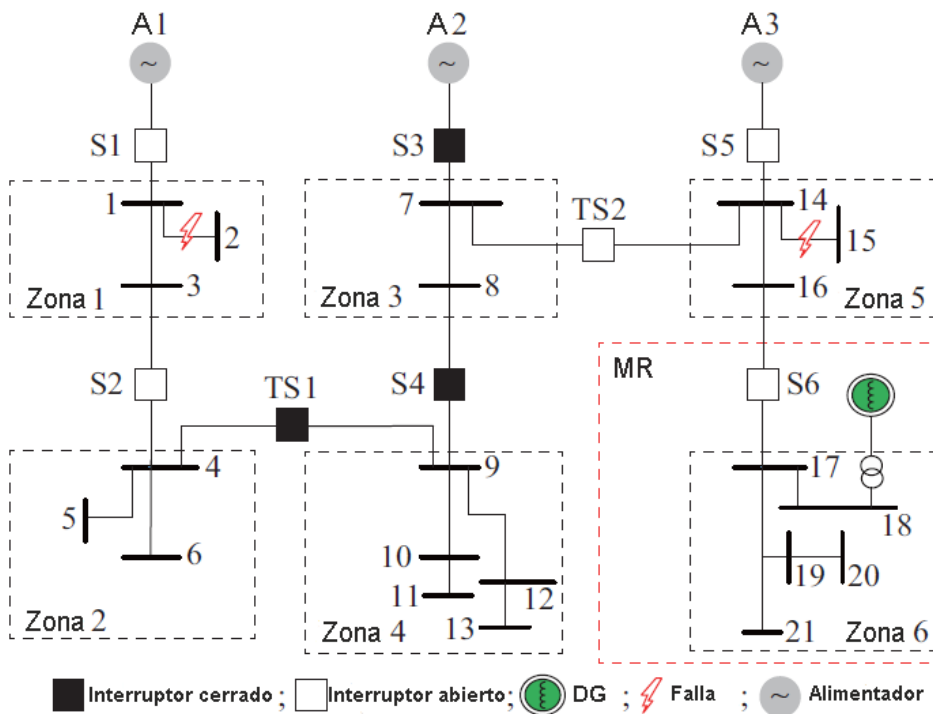


Figura 2.2 Procedimientos de restauración de servicio en el sistema de prueba [4].

El plan de restauración del servicio debe satisfacer los siguientes objetivos [4]:

- Objetivo 1: Restaurar el mayor número posible de cargas fuera de servicio.
- Objetivo 2: Minimizar el tiempo de restauración del servicio.
- Objetivo 3: Minimizar las pérdidas en la red con la nueva topología.

Los objetivos de SR mencionados tienen diferentes prioridades [4]. La prioridad esencial atiende a maximizar el número de cargas fuera de servicio restauradas, especialmente los clientes considerados prioritarios. La segunda atiende a minimizar el tiempo de restauración del servicio, es decir, reducir el número de operaciones de los equipos de conmutación que intervienen en el plan de maniobras. La última prioridad atiende a minimizar las pérdidas en la red con la nueva topología que se forme; sin embargo, al ser esta temporal, la reducción de pérdidas no representa un beneficio relevante al sistema.

2.2 Arquitecturas de Automatización para la restauración del servicio

Las arquitecturas de automatización para la restauración del servicio se pueden dividir en cuatro categorías [4], [6], descritas a continuación:

- Arquitectura centralizada: Cada agente local comunica con el centro de control (Control center – CC), quien toma las decisiones en función a la información global del sistema para determinar el plan óptimo de maniobras de restauración.
- Arquitectura distribuida: Cada agente local comunica con sus vecinos para tomar la decisión más acorde, se prescinde de un controlador central en el sistema.
- Arquitectura jerárquica: Los agentes locales comunican en una estructura jerárquica para realizar la toma de decisiones.
- Arquitectura descentralizada: Cada agente local toma la decisión en función a la información local que dispone, sin que exista comunicación entre los agentes.

La categorización de las arquitecturas responde al lugar donde reside la lógica de control o inteligencia principal del sistema [9].

2.3 Arquitectura de automatización sin sistemas de comunicación

La arquitectura de automatización para la restauración del servicio sin el empleo de sistemas de comunicación corresponde al enfoque descentralizado (local), que usa

la inteligencia local para obtener soluciones de restauración a través del control de recierre automático, control de conmutación automática de respaldo y la automatización de alimentadores (Feeder automation – FA) basada en interruptores inteligentes como seccionadores en tipo de retardo de voltaje [4]. Este enfoque es sencillo porque emplea únicamente la lógica de control integrada en los dispositivos inteligentes del alimentador [6]. La lógica se realiza mediante el análisis de condiciones externas como valores de tensión, corriente o estado de posición (abierto/cerrado) de los equipos de conmutación (Figura 2.3), para la ejecución de las acciones de automatización [10].



Figura 2.3 Interruptor seccionador para análisis de estado abierto/cerrado de red [12].

Esta arquitectura no requiere de la infraestructura de control y comunicaciones para aislar y restaurar el servicio en las secciones aguas arriba de la falla [6]. Sin embargo, al no contar con un sistema de comunicaciones no se puede informar de la falla al Centro de Control. A pesar de que este enfoque brinda soluciones rápidas, se torna difícil coordinar diferentes tipos de unidades de control local [4], que hace que se opte por otras arquitecturas de automatización.

2.4 Arquitecturas de automatización con sistemas de comunicación

Las arquitecturas de automatización para la restauración del servicio con el empleo de sistemas de comunicación involucra a los enfoques centralizado, distribuido y jerárquico, como se escenifica en la Figura 2.4 [9]. Estas arquitecturas utilizan diferentes técnicas de optimización para obtener las soluciones de restauración, que incluyen sistemas expertos, algoritmos heurísticos, algoritmos metaheurísticos, teoría de grafos, programación matemática y sistemas multi-agentes [4]. En las siguientes secciones se expone sobre las arquitecturas de automatización y técnicas de optimización empleadas.

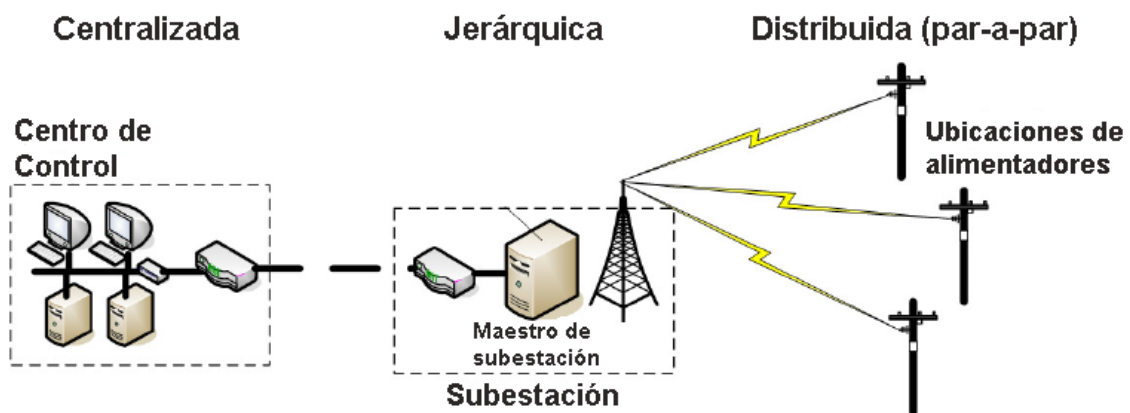


Figura 2.4 Arquitecturas de automatización con uso de sistemas de comunicación [11].

2.4.1 Arquitectura centralizada

La arquitectura centralizada se basa en un DAS conformado por un Centro de Control (CC) o estación maestra, Gateway de subestaciones, dispositivos DA (interruptores, reconectores, seccionadores, etc.) y un sistema de comunicación [4], como se representa en la Figura 2.5 [6]. Emplea la información global del sistema de distribución, incluyendo su topología y parámetros de operación [6], para obtener las soluciones óptimas de restauración. Esta arquitectura es adecuada cuando existen sistemas de CC (Supervisory control and data acquisition – SCADA y Distribution management system – DMS), que están diseñados para soportar funciones de control y monitoreo centralizado [11]. Todas las funciones de control e inteligencia residen en el CC [6].

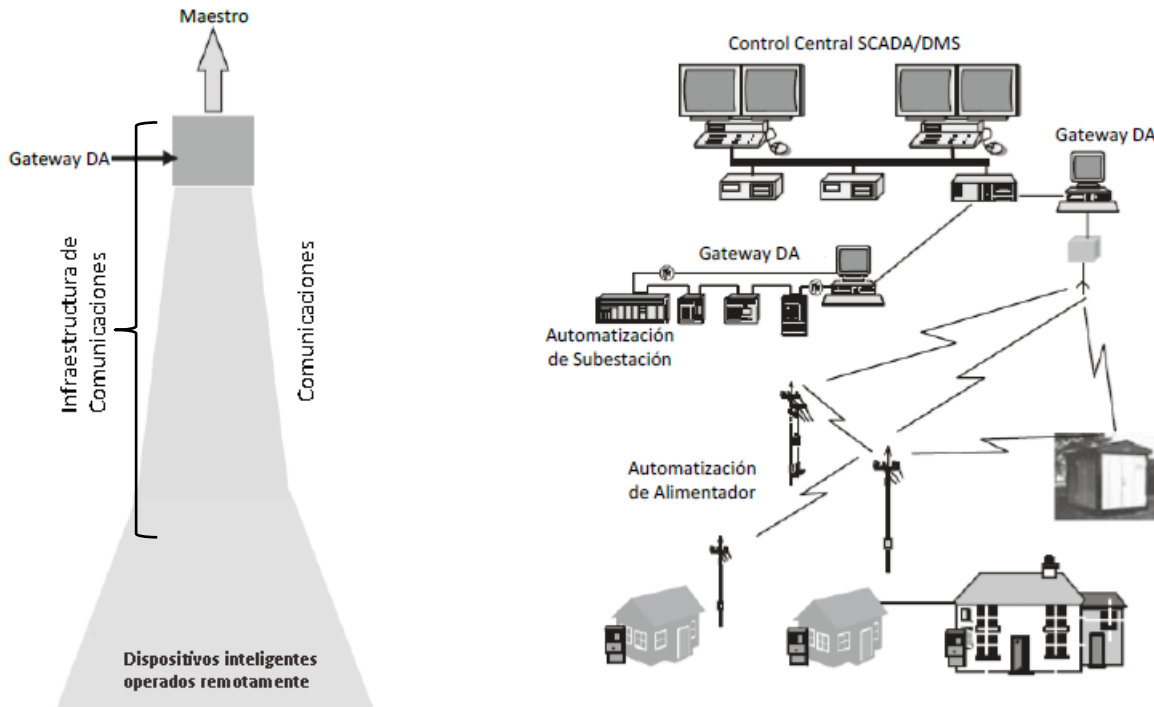


Figura 2.5 Componentes de la arquitectura de control centralizada [6].

Como ventajas que ofrece esta arquitectura se menciona:

- Al disponer de la información global del sistema, puede proporcionar la mejor solución de restauración, especialmente para sistemas de pequeña escala [2].
- Arquitectura madura, es el enfoque de automatización más aplicado para controlar un sistema de distribución [12].
- Se acopla de forma adecuada en alimentadores con un alto grado de penetración de recursos de energía distribuidos (DER) y que son reconfigurados frecuentemente por balanceo de carga u otras condiciones operativas [6].

Como desventajas que presenta esta arquitectura se señala:

- Requerimiento de alto nivel de confiabilidad y baja latencia de los sistemas de comunicación para la transferencia de grandes volúmenes de datos entre los dispositivos de campo y el CC [2].
- Requerimiento de elevada capacidad computacional del CC para procesar datos y cálculos [4], lo que significa el uso de una supermáquina costosa.
- Baja confiabilidad, está sujeto a riesgo de falla de un solo punto [2], [4]. Una falla del CC causaría un colapso de la restauración del servicio.

Las siguientes subsecciones resumen las técnicas de optimización que se usan en la arquitectura centralizada para obtener las soluciones de restauración del servicio.

2.4.1.1 Sistema Experto

El sistema experto (Expert system – ES) es una técnica basada en el conocimiento. Está compuesto de tres partes [4]: base del conocimiento, base de datos y motor de inferencia, como se ilustra en la Figura 2.6. La base del conocimiento consiste en el conocimiento del dominio utilizado para la resolución de problemas, que se traduce en un conjunto de reglas IF (condición) – THEN (acción). La base de datos incluye un conjunto de hechos que coinciden con las reglas de la base del conocimiento. El motor de inferencia enlaza los hechos con las reglas y lleva a cabo el proceso de razonamiento para obtener las soluciones.

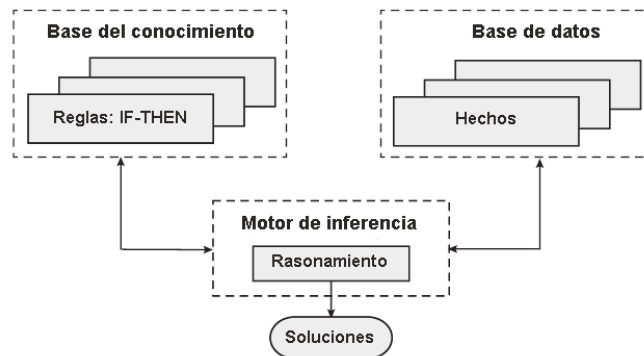


Figura 2.6 Componentes del sistema experto [4].

Existen trabajos investigativos que proponen un ES para resolver problemas de restauración del servicio. Se mencionan los siguientes:

- En [13], los autores desarrollan una base de conocimiento compuesta por 180 reglas usadas por los operadores en la planificación de restauración. Además, se ha incorporado la restauración de grupo, restauración de zona y transferencia de carga. El sistema propuesto puede restaurar la red con estrategias de restauración de una o varias zonas.
- En [14], el sistema sugerido tiene dos funciones (detección de falla y restauración del servicio) y tres motores de inferencia (detección de falla, planificación de restauración y operación de restauración) que mejoran la velocidad de razonamiento.

El sistema experto puede obtener soluciones de restauración factibles de manera rápida. Sin embargo, resulta difícil construir una base de conocimiento extensa y se

torna costoso el mantenimiento de un sistema experto a gran escala [4], dado que las reglas son específicas del sistema de distribución y deben modificarse conforme las variaciones del sistema [2]. Además, no se garantiza que las soluciones sean óptimas [4].

2.4.1.2 Algoritmos Heurísticos

La técnica basada en algoritmos heurísticos (Heuristics algorithms – HA) usa heurísticas o reglas empíricas para obtener soluciones de restauración. En esta técnica, las heurísticas se transforman en algoritmos para guiar la búsqueda de soluciones [4]. Se requiere del conocimiento y experiencia de un experto que se traduce en lógicas de programación para resolver problemas de restauración [2].

Existen trabajos investigativos que proponen un HE para resolver problemas de restauración del servicio. Se mencionan los trabajos:

- En [15], se desarrolla un algoritmo heurístico a partir de la compilación de un conjunto de reglas heurísticas a través de entrevistas con operadores experimentados de Taiwan Power Company. El algoritmo restaura las cargas desenergizadas mediante los alimentadores de apoyo. Luego, utiliza los laterales de apoyo correspondientes con el propósito de reducir las acciones de conmutación.
- En [16], los autores desarrollan un algoritmo heurístico basado en el flujo de carga. Todos los interruptores disponibles en la red se cierran para crear una malla. Los interruptores que aíslan el área afectada permanecen abiertos. Estas operaciones solo se consideran en el algoritmo y no en campo. Todos los interruptores cerrados se consideran fuentes de corriente ficticias ajustables y se realiza un flujo de potencia óptimo. Después, el interruptor que transporta la menor corriente se abre para eliminar un bucle de red, de modo que se minimice la perturbación en el patrón de flujo de potencia. Este proceso se repite hasta que no haya bucles, obteniéndose una red radial con condiciones de operación aceptables.

Los algoritmos heurísticos pueden obtener soluciones de restauración factibles de manera rápida [4]. Sin embargo, requieren conocimiento dependiente del problema, es decir, heurísticas, para buscar soluciones [4]. Los usuarios tienen grandes dificultades para el mantenimiento del algoritmo, debido a su extensión y complejidad [2]. Además, no se garantiza que las soluciones sean óptimas [2], [4].

2.4.1.3 Teoría de grafos

La técnica basada en teoría de grafos (Graph theory – GT) consiste en encontrar un árbol de expansión que represente una configuración radial de red que cumple las restricciones de operación. Las subestaciones y zonas de carga del sistema de distribución se ven como vértices (v) y los interruptores como bordes (ε), por lo que se puede representar al sistema como un gráfico abstracto $G(v, \varepsilon)$ [4].

Para representar esta técnica, se supone que una falla permanente ocurre en la Zona 1 del sistema de prueba de la Figura 2.1. Posterior al aislamiento de la falla, el gráfico abstracto del sistema se muestra en la Figura 2.7.

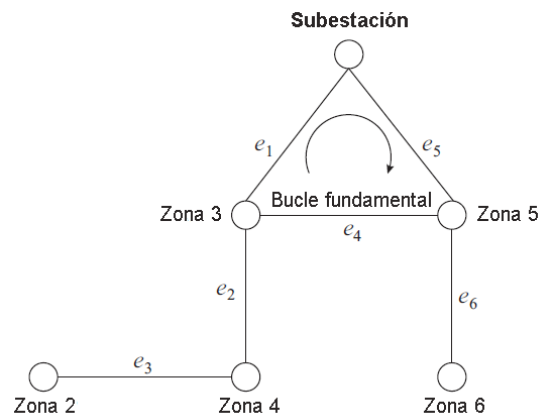


Figura 2.7 Gráfico abstracto del sistema de prueba tras el aislamiento de la falla en la Zona 1 [4].

Existen trabajos investigativos que usan GT aplicando diferentes técnicas para encontrar los árboles de expansión del sistema. Se mencionan los siguientes:

- En [17], el árbol de expansión se consigue eliminando un borde de cada bucle fundamental (por ejemplo el borde ε_1 en la Figura 2.7). Una vez obtenidas todas las configuraciones, se efectúa el análisis de flujo de potencia para comprobar la viabilidad de cada configuración, es decir, que no se violen las restricciones de operación. Con el propósito de encontrar la mejor configuración, se aplican ciertos criterios, como la cantidad de cargas desenergizadas restauradas y el número de operaciones de conmutación.
- En [18], los autores introducen el concepto de *árbol interesado* que representa un árbol en el que todas las cargas son suministradas por subestaciones (nodos raíz). Se desarrolla un algoritmo para extraer todos los *árboles interesados* a partir de los árboles de expansión factibles y

posteriormente, evaluarlos. Como resultado, se obtiene el plan de restauración de servicio completo.

Los algoritmos basados en la teoría de grafos pueden obtener soluciones de restauración factibles de forma rápida. Aunque la solución óptima puede conseguirse mediante una búsqueda exhaustiva, para sistemas a gran escala el número de árboles puede ser muy extenso, lo que hace que este enfoque sea computacionalmente costoso, resultando poco atractivo [4].

2.4.1.4 Programación Matemática

La técnica de programación matemática (Mathematical programming – MP) permite resolver los problemas de restauración mediante su formulación como un modelo de programación no lineal entera mixta (mixed integer non-linear programming – MINLP), un modelo de programación lineal entera mixta (mixed integer linear programming – MILP) y un modelo de programación cónica de segundo orden entera mixta (mixed integer second-order cone programming – MISOCP) [4].

En [19], se presenta un modelo MINLP basado en algoritmos de flujo de carga de barrido hacia atrás/ adelante para la representación matemática del problema de restauración del servicio (en la Sección 3.2 se describe su formulación matemática). Este modelo puede resolverse con el uso de solucionadores (solvers) no-lineales como BARON, obteniéndose soluciones óptimas o casi óptimas. Sin embargo, es computacionalmente costoso dado que el problema es combinatorial y no-convexo debido a la restricción no-lineal (magnitud de corriente de rama: $V_j^{sqr} I_{ij}^{sqr} = P_{ij}^2 + Q_{ij}^2 \quad \forall ij \in \Omega_l$) y a las variables binarias (estado de dispositivos de conmutación w_{ij} y estado de zona de carga x_z) [4].

Para mejorar la eficiencia en la resolución de problemas de restauración, se han aplicado técnicas de linealización y relajación convexa. En [19], se usa la técnica de linealización por partes para linealizar la expresión no lineal (magnitud de corriente de rama) y obtener un modelo MILP. Posteriormente, con las variables binarias obtenidas del modelo MILP, se resuelve un problema de programación no lineal (PNL) para obtener el punto de operación de estado estable del sistema (soluciones de variables continuas). Mientras, en [20] se utiliza la técnica de relajación convexa para transformar el modelo MINLP de [19] en un modelo MISOCP, relajando la expresión no lineal ($V_j^{sqr} I_{ij}^{sqr} = P_{ij}^2 + Q_{ij}^2 \quad \forall ij \in \Omega_l$) como una restricción cónica de segundo orden ($V_j^{sqr} I_{ij}^{sqr} \geq P_{ij}^2 + Q_{ij}^2 \quad \forall ij \in \Omega_l$). El modelo MISOCP puede resolverse empleando solvers comerciales basados en las técnicas de optimización convexa como bifurcación y límite (branch and bound – B&B) [3].

Los enfoques basados en programación matemática consiguen obtener soluciones de restauración óptimas o casi óptimas. Sin embargo, pueden tener una elevada carga computacional debido al tamaño incremental y complejidad del sistema de distribución [3], que hacen que el tiempo de cálculo incremente [2].

2.4.1.5 Algoritmos Metaheurísticos

La técnica basada en algoritmos metaheurísticos (Metaheuristics algorithms – MA) utiliza la inteligencia observada a partir de fenómenos naturales para derivar soluciones de restauración. Los algoritmos metaheurísticos poseen procedimientos de formulación del problema similares con diferentes estrategias de búsqueda y codificación. Dentro de los algoritmos usados para la resolución de problemas de restauración del servicio se incluye el algoritmo genético (Genetic algorithm – GA), búsqueda tabú (Tabu search – TS), optimización de enjambre de partículas (Particle swarm optimization – PSO), recocido simulado en paralelo (Parallel simulated annealing – PSA) [4] y optimización de colonia de hormigas (Ant colony optimization – ACO) [21].

Para mostrar el procedimiento que usa un algoritmo metaheurístico para obtener soluciones de restauración, a continuación se describen las estrategias de codificación y búsqueda del GA [3], [22]:

- *Paso 1: Representación del problema mediante cadenas.* Las variables de decisión del modelo mostrado en (1) – (12) (Ver Sección 3.2), pueden ser representadas como una cadena binaria (1, 1, 0, ..., 0, 1, 0). El valor binario en la cadena simboliza el estado de la variable binaria correspondiente, por ejemplo, “1” representa un interruptor cerrado y “0” un interruptor abierto.
- *Paso 2: Generación de la población inicial de cadenas.* Generalmente, se produce de forma aleatoria. A fin de incrementar la eficiencia del algoritmo, se pueden usar heurísticas dependientes del problema para generar la población inicial de cadenas.
- *Paso 3: Evaluación y selección de cada cadena.* Cada cadena es evaluada por una función de aptitud (función objetivo) que expresa la calidad de la cadena. Para una cadena que denota una configuración de red, se evalúa realizando un estudio de flujo de potencia óptimo con la función objetivo (1). La cadena con un mejor valor objetivo tiene más probabilidad de ser seleccionada para producir la siguiente generación de cadenas.
- *Paso 4: Actualización de cada cadena.* El algoritmo se encarga de actualizar las cadenas mediante operaciones de cruce y mutación. Una cadena descendiente se crea a partir de dos cadenas principales a través de una operación de cruce que expresa un intercambio de una parte de los números

entre dos cadenas principales. En las crías seleccionadas, algunos de sus números están sujetos a cambios, es decir, "1" reemplaza a "0" y viceversa.

- *Paso 5: Iteración y convergencia.* Los pasos 3 y 4 se repiten hasta que el número de iteraciones alcanza un máximo preestablecido o hasta que se satisfacen otras condiciones de parada. Si no se cumple ninguna de las condiciones de parada, se retorna al paso 3 y el número de iteraciones se incrementa en uno.

Existen trabajos investigativos que abordan técnicas de MA para resolver problemas de restauración del servicio. Se mencionan los siguientes:

- En [22], se realiza un estudio comparativo entre los algoritmos GA, TS, TS Reactivo y PSA con respecto al tiempo de cálculo promedio y la calidad de soluciones de restauración. Como resultado, se concluye que el algoritmo TS reactivo genera la mejor solución al requerir menos tiempo de cálculo.
- En [23], el algoritmo propuesto basado en TS se prueba aplicándolo en un sistema de distribución real en Corea. Los resultados muestran que el algoritmo puede obtener la solución óptima en 45 s.

Los algoritmos metaheurísticos no requieren conocimiento dependiente del problema (heurísticas) y pueden ser usados para varios tipos de problemas [3]. Sin embargo, estos algoritmos requieren un tiempo de cálculo extenso cuando se aplican a sistemas de distribución prácticos y se necesita un motor de flujo de potencia para estimar los valores numéricos de la función objetivo y verificar las restricciones operativas [2]. Además, no se garantiza que las soluciones sean óptimas [3].

2.4.2 Arquitecturas distribuida y jerárquica

Las arquitecturas distribuida y jerárquica usan la inteligencia distribuida para obtener las soluciones de restauración del servicio [2], [4], es decir, que las funciones de control se encuentran dispersas en diferentes áreas. El área o grupo es controlado por un agente local, y puede abarcar desde un simple segmento de circuito hasta una región en la que diversos circuitos interconectan varias subestaciones [10]. Los dispositivos DA son controlados localmente y no se requiere de un Centro de Control [6]. El software y la comunicación compartidos distribuyen los datos relacionados con el evento o la condición entre los dispositivos para obtener las soluciones de restauración [4].

La inteligencia distribuida utiliza varias entradas de datos (estados de posición abierto/cerrado, valores de corriente, voltaje, etc.) de los agentes locales para tomar

decisiones y obtener las soluciones [10]. El principio de operación de la inteligencia distribuida consiste en que cada grupo es responsable de un segmento de red, y debe garantizar el suministro a través de las fuentes disponibles (interruptores de enlace, unidades DG) [6]. Este método usa comunicaciones par-a-par (peer-to-peer) entre los controladores de un grupo y entre grupos vecinos [2], [6]. El intercambio de información entre grupos garantiza la compartición del exceso de capacidad y la coordinación de las decisiones de restauración [6].

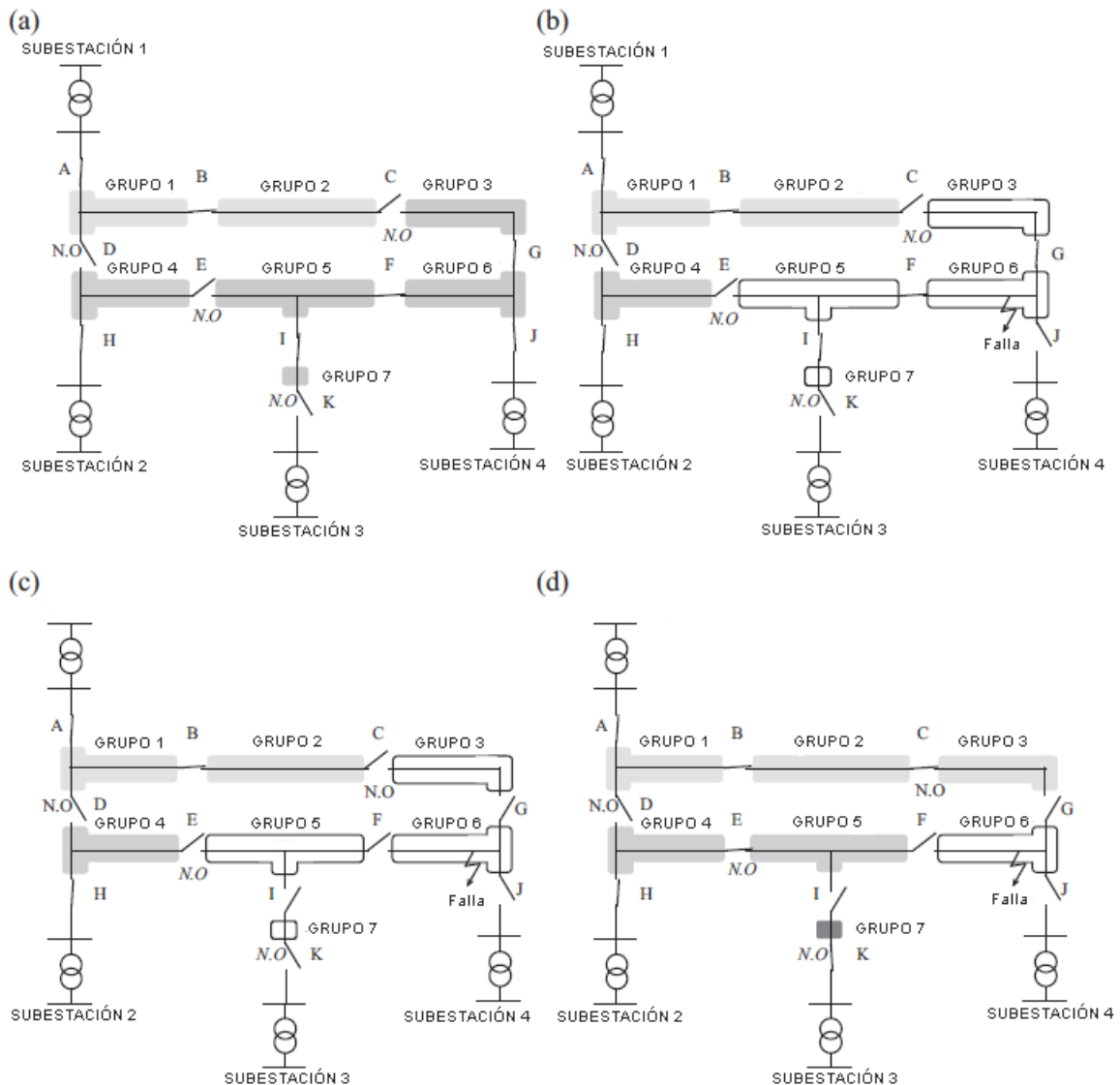


Figura 2.8 Escenario de aplicación de inteligencia distribuida para la restauración del servicio [14].

La Figura 2.8 ilustra un escenario donde se aplica la inteligencia distribuida para la restauración del servicio [12]. La Figura 2.7 (a) muestra la configuración normal del

sistema de distribución organizada en siete grupos de control. La Figura 2.7 (b) presenta una falla dentro del grupo 6 entre los interruptores F, G y J. Todos los interruptores del grupo afectado experimentan una sobrecorriente seguida por una pérdida de tensión. Por consiguiente, el interruptor J se abre para aislar la falla. La Figura 2.7 (c) muestra cómo el resto de interruptores del grupo 6 (F y G) y del grupo 5 (I) detectan la pérdida de tensión, y entonces se abren para aislar la falla. Finalmente, la Figura 2.7 (d) presenta la solución de restauración del servicio para las secciones no afectadas a través del cierre de los interruptores K del grupo 7, E de los grupos 4 y 5, y C de los grupos 2 y 3.

Como ventajas que ofrece la inteligencia distribuida se menciona:

- Disminución de la capacidad computacional de los controladores para procesar datos y cálculos. Ningún dispositivo debe procesar la información global del sistema [24], lo que significa el uso de controladores más económicos.
- Mejora la eficiencia de operación del sistema al obtener soluciones de restauración en menor tiempo [25]. Cualquier perturbación en un área de control individual se procesará localmente para minimizar su impacto en otras áreas [2].
- Mejora la confiabilidad del sistema al no existir un único punto de falla [2]. Una falla en un controlador no causaría un colapso en la restauración del servicio.
- Mejora la escalabilidad del sistema. Es posible la expansión incremental de agentes, dado que se requiere que estos únicamente mantengan información consistente y actualizada sobre su área de control y no de toda la red [26].
- Su aplicación se está popularizando para sistemas con redes de distribución extensas [12]. Se divide la tarea en varias sub-tareas, conforme con el número de áreas de control, que se procesan simultáneamente de manera distribuida (en paralelo) [2].

Como desventajas que presenta la inteligencia distribuida se menciona:

- No considera la información global del sistema, por lo que es posible que la solución de restauración no sea la óptima para todo el sistema [12].
- La introducción de inteligencia distribuida requiere inversiones considerables y su implementación es compleja desde el punto de vista técnico [25]. Se requiere de sistemas de comunicación de alta velocidad [27], que permitan comunicaciones par-a-par entre los dispositivos, todos los controladores

posean capacidad computacional y el estándar de comunicación adecuado (IEC 61850) para el intercambio de datos [6], [27], [28].

- Problema de seguridad de los datos [24], debido a que la lógica de control se encuentra distribuida en cada agente local, cuya ubicación geográfica podría estar expuesta.

Las siguientes subsecciones resumen las técnicas de optimización que se usan en las arquitecturas distribuida y jerárquica para obtener las soluciones de restauración del servicio.

2.4.2.1 Sistema Multi-agente (MAS)

La técnica de sistema multi-agente (multi-agent system - MAS) consiste en que los agentes se comuniquen y coordinen para resolver problemas de restauración en función de reglas heurísticas específicas [4]. Cada agente es considerado como una entidad de software capaz de percibir y reaccionar a los cambios en su entorno de manera autónoma [2], [4].

Los enfoques de restauración del servicio basados en MAS pueden clasificarse conforme la arquitectura de automatización en [4]:

a) MAS Jerárquico

La arquitectura de automatización sigue una estructura jerárquica y se necesita de agentes de negociación (ej. Controlador de alimentador) que coordinen la operación de los agentes locales (dispositivos DA) [4].

Existen trabajos investigativos que usan MAS jerárquico para resolver problemas de restauración del servicio. Se menciona el trabajo en [29], los autores diseñan un sistema multi-agente de dos capas: zona y alimentador. Los agentes de zona se desarrollan en la capa inferior y los agentes alimentadores (agentes de negociación) en la capa superior. El alimentador se divide en varias zonas delimitadas por interruptores y cada una cuenta con un agente de zona que detecta el estado de la zona y se comunica con su agente alimentador. El agente alimentador coordina a los agentes de zona asociados que se comunican con los agentes alimentadores vecinos.

b) MAS Distribuido

La arquitectura de automatización es distribuida y los agentes locales se comunican directamente con sus vecinos para efectuar la toma de decisiones, por lo que no se requiere de agentes de negociación [4], [6].

Existen trabajos investigativos que proponen MAS distribuido para resolver problemas de restauración del servicio. Se menciona el trabajo en [30], los autores categorizan los agentes en cuatro tipos de zonas a lo largo del alimentador: agentes de zona defectuosa, agentes de zona desenergizada, agentes de zona de transferencia y agentes de zona no afectada. Cada agente identifica su rol en función de la localización de la falla y se comunica con otros agentes para tomar decisiones.

2.4.2.2 Programación matemática

La técnica de programación matemática permite formular el problema de restauración del servicio como un modelo MISCOP [31], que se descompone en sub-problemas para cada nodo aplicando el método de multiplicadores de dirección alterna (alternating direction method of multipliers – ADMM) y lo resuelven los agentes de manera distribuida (en paralelo), a través del intercambio de información con los nodos vecinos en un procedimiento iterativo. El algoritmo basado en ADMM combina la capacidad de la descomposición dual y la especificación de convergencia para el método de multiplicadores.

2.5 Comparativa entre las arquitecturas de automatización

La elección de una arquitectura de automatización para la restauración del servicio debe realizarse considerando un punto de equilibrio entre el tiempo de respuesta y el grado de complejidad del sistema de distribución [6], [9], como se muestra en la Figura 2.9.

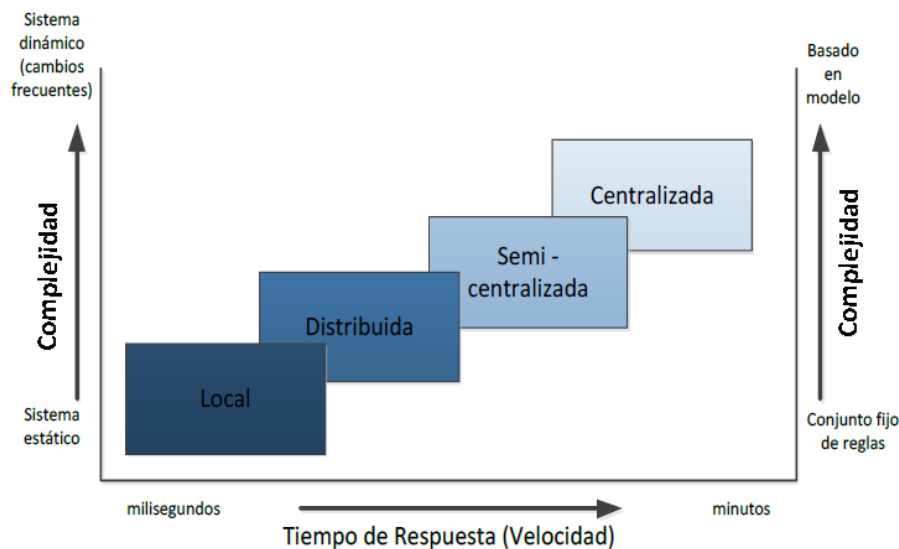


Figura 2.9 Comparativa entre el Tiempo de Respuesta y la Complejidad de las arquitecturas de automatización [6].

Los requerimientos básicos que deben tomarse en cuenta para la selección e implantación de una arquitectura de automatización se presentan en la Tabla 2.1.

Tabla 2.1 Criterios de selección de las arquitecturas de automatización [6], [11].

Criterios	Arquitecturas de automatización			
	Centralizada	Jerárquica	Distribuida	Local
Número de alimentadores DA	Alto porcentaje (%) de alimentadores automatizados	Todos los alimentadores de la subestación seleccionada automatizados	Reducido número de alimentadores automatizados, dispersos geográficamente	Reducido número de alimentadores automatizados, dispersos geográficamente
Visibilidad del operador	Todas las acciones se notifican al operador	Algunas acciones de control no se notifican al operador	Algunas acciones de control no se notifican al operador	Ninguna acción se notifica al operador
Disponibilidad de infraestructura de comunicaciones adecuadas	Requiere comunicaciones de gran cobertura y alta velocidad	Requiere comunicaciones en áreas locales (subestación-poste, subestación-subestación)	Requiere de comunicaciones en áreas locales (poste-poste, subestación-subestación)	No requiere comunicaciones
Disponibilidad de soluciones DA	VVO, FLISR, ONR	VVC, FLISR, DFA	FLISR	FLISR
Procesamiento de grandes cantidades de datos	Limitado por la capacidad de transferencia de datos	Aplicable para análisis de forma de onda	Poca o ninguna capacidad	Poca o ninguna capacidad

Las principales aplicaciones DA que pueden implementarse en cada arquitectura de automatización se resumen en la Tabla 2.2. La aplicación FLISR puede ser desplegada con todas las arquitecturas; sin embargo, las soluciones de restauración óptimas o casi óptimas se garantizan solo con la arquitectura centralizada.

Tabla 2.2 Aplicaciones DA en arquitecturas de automatización [6], [11].

Aplicación	Arquitecturas de automatización			
	Centralizada	Jerárquica	Distribuida	Local
FLISR	✓	✓	✓	✓
Volt/Var Control VVC (basado en reglas)	✓	✓		

Volt/Var Optimización VVO (basado en modelo)	✓			
VVO adaptable	✓	✓		
VVC dinámico		✓	✓	
Reconfiguración óptima de red ONR	✓			
Anticipador de falla		✓		
Monitoreo de condición de equipamiento ECM	✓	✓		

2.6 Aplicaciones prácticas de restauración del servicio

Para las arquitecturas de automatización con sistemas de comunicación, existen algunos proyectos pilotos para aplicaciones reales de SR. Se mencionan:

- El proyecto UHENPAL se ha desplegado en un sistema de distribución en Brasil [32], siguiendo una arquitectura centralizada. El módulo de software de autorreparación, que se integra en el sistema SCADA, evalúa las soluciones de SR mediante un algoritmo heurístico y el análisis multicriterio. La mejor solución se lleva a cabo operando los interruptores telecontrolados de acuerdo con los comandos enviados desde el Centro de Control.
- El proyecto Stedin se ha desplegado en un sistema de distribución en Países Bajos [33], con una arquitectura distribuida. Cada alimentador se divide en varios segmentos y cada segmento está controlado por un controlador de autorreparación (agente local). Cada controlador se comunica con sus vecinos para realizar la toma de decisiones en función de reglas predefinidas, que actúan operando los interruptores telecontrolados.
- El proyecto Guangdong se ha desplegado en un sistema de distribución en China [4], combinando las arquitecturas centralizada (back-up) y distribuida (principal). En la arquitectura distribuida, cada interruptor controlable está equipado con un controlador local que se comunica con sus vecinos para tomar decisiones basándose en reglas predefinidas. Si esta solución de SR genera una violación de las restricciones operativas (ej. sobrecarga), la arquitectura centralizada entra en operación para efectuar las acciones correctivas.

Para implementar la inteligencia de SR, los algoritmos de optimización distribuidos están en auge, en especial para redes extensas y modernas [6], [26], [28]. Sin embargo, con sistemas de automatización y comunicación heredados de las empresas distribuidoras puede resultar no viable [6], [27], debido a que estos sistemas cuentan con bajas velocidades de transmisión y protocolos de telecontrol anteriores (ej. DNP3) que no permiten comunicaciones par-a-par entre los dispositivos DA [27]. En aplicaciones prácticas, se prefieren los enfoques centralizados que usan algoritmos basados en reglas lógicas, obteniéndose soluciones de SR de buena calidad en un tiempo de cómputo razonable [4]. De hecho, con una arquitectura de automatización centralizada, el tiempo de restauración podría tomar hasta 5 minutos, y su implementación resulta menos compleja desde el punto de vista técnico, en especial para sistemas de pequeña escala [2].

3 METODOLOGÍA DEL ALGORITMO DE RESTAURACIÓN DEL SERVICIO

3.1 Introducción

La restauración del servicio (SR) se formula como un problema de optimización multiobjetivo, combinatorio, no lineal y restringido [22], [34], que puede ser resuelto aplicando diferentes arquitecturas de automatización y técnicas de optimización. Los algoritmos de optimización distribuidos resultan idóneos para implementar la inteligencia de SR en modernas redes de distribución. Sin embargo, con sistemas de automatización y comunicación heredados de las empresas distribuidoras, este enfoque puede no resultar viable desde el punto de vista técnico [6], [27], por lo que se opta por los enfoques centralizados.

Dentro de las técnicas de optimización que se usan en una arquitectura de automatización centralizada, el método de programación matemática tiende a garantizar la optimización de la solución final [19]. Sin embargo, dada la naturaleza compleja del problema de SR, se requiere del uso de modelos que pueden tener una gran carga de cálculo debido al tamaño y complejidad de la red. En aplicaciones prácticas, resulta oportuno y preferible los enfoques que emplean algoritmos basados en reglas lógicas (heurísticas/metaheurísticas) [4], [34]. Si bien las soluciones obtenidas no garantizan ser el óptimo global, estas técnicas permiten estudiar problemas de gran complejidad de forma más sencilla, brindando soluciones de buena calidad en tiempos de cómputo razonables [35], [36].

Para resolver el problema de SR en sistemas de distribución, se propone la aplicación de un algoritmo de optimización centralizado basado en técnicas metaheurísticas, cuya metodología incluye dos componentes principales: i) un modelo matemático para representar la formulación del problema y ii) una técnica de solución aplicada para resolver el modelo matemático.

Las siguientes secciones describen la metodología e implementación del algoritmo de SR, que incluye la extensión de los escenarios de restauración considerando los clientes prioritarios y el despacho de unidades de DG para soporte de la red principal o alimentación local de las cargas, operando en modo microrred.

3.2 Formulación matemática

El modelo matemático detallado para resolver el problema de restauración del servicio se presenta en [19]. En esta formulación se hacen los siguientes supuestos:

- Las cargas se representan como cargas de potencia activa y reactiva constantes en cada nodo.
- Se asume que el sistema eléctrico es balanceado, por consiguiente, se representa como circuitos equivalentes monofásicos.
- Los interruptores se consideran como circuitos de corta longitud con muy baja impedancia.

Considerando los objetivos que debe satisfacer la restauración del servicio (Ver Sección 2.1), y los modelos matemáticos detallados en trabajos previos [19], [37], [38], se presenta la formulación matemática del problema de SR que contiene las subsecciones:

- Reconfiguración del sistema
- Unidades de generación distribuida
- Prioridades de clientes

3.2.1 Reconfiguración del sistema para restauración del servicio

- **Problema maestro:** Reconfiguración del sistema aplicada para la restauración del servicio.

$$\min \left[c_z^R \sum_{z \in \Omega_z} (1 - x_z) P_i^D + c^{SW} \sum_{ij \in \Omega_{SW} : s_{ij}^{ini}=1} (1 - w_{ij}) + c^{SW} \sum_{ij \in \Omega_{SW} : s_{ij}^{ini}=0} w_{ij} + c^{lss} \sum_{ij \in \Omega_l} R_{ij} I_{ij}^{sqr} \right] \quad (1)$$

- **Subproblema:** Operación del sistema en condiciones adecuadas (configuración radial y con límites aceptables de las variables eléctricas).

Sujeto a:

$$\sum_{ji \in \Omega_l} P_{ji} - \sum_{ij \in \Omega_l} (P_{ij} + R_{ij} I_{ij}^{sqr}) + P_i^S = P_i^D x_{z_i}^b \quad \forall i \in \Omega_b \quad (2)$$

$$\sum_{ji \in \Omega_l} Q_{ji} - \sum_{ij \in \Omega_l} (Q_{ij} + X_{ij} I_{ij}^{sqr}) + Q_i^S = Q_i^D x_{z_i}^b \quad \forall i \in \Omega_b \quad (3)$$

$$V_i^{sqr} - 2(R_{ij}P_{ij} + X_{ij}Q_{ij}) - Z_{ij}^2 I_{ij}^{sqr} - V_j^{sqr} = 0 \quad \forall ij \in \Omega_l \quad (4)$$

$$V_j^{sqr} I_{ij}^{sqr} = P_{ij}^2 + Q_{ij}^2 \quad \forall ij \in \Omega_l \quad (5)$$

$$0 \leq I_{ij}^{sqr} \leq \bar{I}_{ij}^2 x_{z_{ij}^l} \quad \forall ij \in \Omega_l \quad (6)$$

$$(P_i^S)^2 + (Q_i^S)^2 \leq \bar{S}_i^2 \quad \forall i \in \Omega_b^S \quad (7)$$

$$\underline{V}^2 x_{z_i^b} \leq V_i^{sqr} \leq \bar{V}^2 x_{z_i^b} \quad \forall i \in \Omega_b \quad (8)$$

$$|\Omega_l| + \sum_{ij \in \Omega_{SW}} w_{ij} = |\Omega_b| - |\Omega_b^S| - \sum_{z \in \Omega_z} (1 - x_z) \quad (9)$$

$$\sum_{ij \in \Omega_{SW} : z_i^b = z} w_{ij} + \sum_{ji \in \Omega_{SW} : z_i^b = z} w_{ji} \geq x_z \quad \forall z \in \Omega_z \quad (10)$$

$$w_{ij} \in \{0,1\} \quad \forall ij \in \Omega_{SW} \quad (11)$$

$$x_z \in \{0,1\} \quad \forall z \in \Omega_z \quad (12)$$

Donde:

Conjuntos: Ω_b conjunto de nodos, Ω_b^S conjunto de nodos fuente, Ω_l conjunto de ramas (líneas), Ω_{SW} conjunto de interruptores, y Ω_z conjunto de zonas de carga.

Parámetros: c_z^R costo de zona desenergizada $z \in \Omega_z$ [US\$/kW], c^{lss} costo de pérdidas de potencia activa [US\$/kW], c^{SW} costo de operación de interruptores [US\$/kW], \bar{I}_{ij} magnitud de corriente máxima a través de rama ij [A], P_i^D demanda de potencia activa [kW], Q_i^D demanda de potencia reactiva [kVAr], R_{ij} resistencia de rama ij [mΩ], \bar{S}_i potencia aparente máxima generada en nodo i [kVA], s_{ij}^{ini} estado inicial de interruptor ij ($s_{ij}^{ini} = 1$ si el interruptor está cerrado; de otra forma, $s_{ij}^{ini} = 0$ si está abierto), \bar{V} magnitud de voltaje máxima [kV], \underline{V} magnitud de voltaje mínima [kV], X_{ij} reactancia de rama ij [mΩ], Z_{ij} impedancia de rama ij [mΩ], z_i^b zona de nodo i , y z_{ij}^l zona de rama ij .

Variables continuas: I_{ij}^{sqr} cuadrado de magnitud de corriente de rama ij [A²], P_i^S potencia activa generada en nodo i [kW], P_{ij} flujo de potencia activa a través de rama ij [kW], Q_i^S potencia reactiva generada en nodo i [kVAr], Q_{ij} flujo de potencia reactiva a través de rama ij [kVAr], y V_i^{sqr} cuadrado de magnitud de voltaje en nodo i [V²].

Variables binarias: w_{ij} estado de interruptor ij ($w_{ij} = 1$ si el interruptor está cerrado; de otra forma, $w_{ij} = 0$ si está abierto), y x_z estado de zona z ($x_z = 1$ si la zona está energizada; de otra forma, $x_z = 0$ si está desenergizada).

La función objetivo se representa en (1) y está conformada de cuatro partes: 1) minimización de costo de zonas desenergizadas, 2) y 3) minimización de costo de operación de los interruptores después de la reconfiguración y 4) minimización de costo de pérdidas de potencia activa de red.

Las restricciones del problema se hallan en (2) a (12). Las expresiones (2) y (3) representan el balance de flujo de potencia activa y reactiva por cada nodo. La ecuación (4) calcula la caída de voltaje para sistemas de distribución radial. La expresión (5) calcula la magnitud de corriente a lo largo de las ramas del circuito. La restricción (6) limita la magnitud de corriente a lo largo de las ramas; si la zona de rama es energizada ($z_{ij}^l = 1$). La expresión (7) representa la capacidad de generación de los nodos fuente acorde el parámetro \bar{S}_i establecido para cada subestación o nodo fuente de alimentador. La restricción (8) limita la magnitud de voltaje en cada nodo; si la zona de nodo es energizada ($x_{z_i}^b = 1$). Las ecuaciones (9), (2) y (3) establecen las condiciones necesarias y suficientes para generar una topología final radial. La restricción (10) es una expresión auxiliar para modelar las zonas desenergizadas. Las restricciones (11) y (12) definen la naturaleza binaria de las variables de estado de interruptor w_{ij} y estado de zona x_z .

3.2.2 Unidades de generación distribuida para restauración del servicio

El desarrollo de redes inteligentes (Smart grids) permite que tecnologías de generación distribuida (Distributed Generation - DG) se integren en la red, causando un impacto en la operación del sistema de distribución [39]. Enfocado en la restauración del servicio, las unidades DG brindan una fuente de energía de soporte cuando la red de distribución no posee la suficiente capacidad para restaurar todas las zonas fuera de servicio aguas abajo de la localización de falla. Estas pueden despacharse para conseguir restablecer el área fuera de servicio, o en su defecto, para alimentar localmente a las cargas desenergizadas, operando en modo microrred. La integración de unidades DG disminuye la cantidad requerida de operaciones de conmutación [2] y además, puede mejorar significativamente la confiabilidad del sistema [39].

Para integrar el despacho de DG en la solución de restauración del servicio, se modifica la expresión (1) para que considere la minimización del costo de generación de las unidades DG [4]. Adicionalmente, se modifica las restricciones de balance de potencia en (2) y (3) para añadir una inyección de potencia constante en

cada nodo donde esté conectada una unidad DG [19]. Por consiguiente, se rempazan las expresiones (1) a (3) por (13) a (15), respectivamente.

$$\min \left[c_z^R \sum_{z \in \Omega_z} (1 - x_z) P_i^D + c_i^{DG} \sum_{i \in \Omega_b} P_i^{DG} + c^{SW} \sum_{ij \in \Omega_{SW} : s_{ij}^{ini} = 1} (1 - w_{ij}) + c^{SW} \sum_{ij \in \Omega_{SW} : s_{ij}^{ini} = 0} w_{ij} + c^{LSS} \sum_{ij \in \Omega_l} R_{ij} I_{ij}^{sqr} \right] \quad (13)$$

$$\sum_{ji \in \Omega_l} P_{ji} - \sum_{ij \in \Omega_l} (P_{ij} + R_{ij} I_{ij}^{sqr}) + P_i^S = (P_i^D - P_i^{DG}) x_{z_i^b} \quad \forall i \in \Omega_b \quad (14)$$

$$\sum_{ji \in \Omega_l} Q_{ji} - \sum_{ij \in \Omega_l} (Q_{ij} + X_{ij} I_{ij}^{sqr}) + Q_i^S = (Q_i^D - Q_i^{DG}) x_{z_i^b} \quad \forall i \in \Omega_b \quad (15)$$

Donde: c_i^{DG} es el costo de generación [\$/kWh] de la unidad DG. Mientras P_i^{DG} y Q_i^{DG} representan la cantidad de potencia activa [kW] y reactiva [kVAr] en cada nodo i del conjunto de nodos Ω_b donde está conectada la unidad DG.

3.2.3 Clientes prioritarios para restauración del servicio

En todo sistema de distribución, existen ciertas cargas que no pueden mitigar una pérdida de electricidad y, por tanto, requieren que se restaure el servicio lo antes posible. Estas cargas denominadas *Clientes Prioritarios* (Priority Customers – PC) incluyen hospitales, bancos, departamentos de seguridad, grandes fábricas, entre otros [3]. Al ocurrir una falla en el sistema que imposibilita la restauración total de las zonas fuera de servicio aguas abajo de la falla, los planes de SR deben estar diseñados para priorizar la alimentación de los PC considerando sus niveles de prioridad [40], desconectando los clientes no prioritarios mediante el deslastre de carga (load shedding) [19].

La primera parte de la expresión (13), minimiza el deslastre de carga considerando únicamente la cantidad de demanda no suministrada. Para incorporar la prioridad de clientes, se reemplaza la demanda de potencia por el peso de prioridad de carga. El cálculo integral de este peso se realiza incluyendo múltiples factores característicos. Este peso representa la prioridad asociada para cada carga del sistema y constituye una medida cuantitativa que puede ser empleada en el proceso de priorización [3]. Se efectúa este ajuste en (13), obteniéndose la función objetivo de SR:

$$\min \left[c_z^R \sum_{z \in \Omega_z} (1 - x_z) PL_i + c_i^{DG} \sum_{i \in \Omega_b} P_i^{DG} + c^{SW} \sum_{ij \in \Omega_{SW} : s_{ij}^{ini} = 1} (1 - w_{ij}) + c^{SW} \sum_{ij \in \Omega_{SW} : s_{ij}^{ini} = 0} w_{ij} + c^{LSS} \sum_{ij \in \Omega_l} R_{ij} I_{ij}^{sqr} \right] \quad (16)$$

Donde: $PL_i = L_{critic}^i * P_i^D$, es el peso de prioridad de carga en el nodo i . En este trabajo se considera dos factores: la criticidad de carga L_{critic}^i y la demanda de potencia P_i^D .

La criticidad atiende únicamente a la naturaleza de la actividad de la carga, sin incluir o medir ningún otro factor. A estas cargas críticas se les da la mayor prioridad, independiente de su consumo o costo de interrupción [3]. El factor de criticidad para cada carga del sistema se modela en función del nivel de criticidad, que puede clasificarse en 3 grupos [39], [3], [40], como se muestra en la Tabla 3.1. La mayoría de las cargas se clasifican como de baja criticidad ($L_{critic}^i = 1$) y, posteriormente, se priorizan en función del nivel más alto ($L_{critic}^i = 3$).

Tabla 3.1 Niveles de criticidad de carga [3].

Nivel de criticidad	L_{critic}^i	Ejemplos
Alto	3	Hospitales, estaciones de bomberos, bancos, industrias
Medio	2	Escuelas, centros comerciales
Bajo	1	Residencial

La incorporación de prioridad de los clientes hace que el problema de SR sea más complejo y difícil de resolver [34]. Sin embargo, su implementación permite que el proceso de restauración se vuelva más efectivo y versátil. El resultado no solo se refleja en una reducción de las pérdidas financieras asociadas a fallas; sino también en un aumento en la confiabilidad del sistema [3], incrementando su capacidad de autorreparación.

3.3 Técnicas metaheurísticas

Para la resolución del problema de SR, se propone la aplicación de una metaheurística híbrida que combina las técnicas de Evolución Diferencial (Differential evolution – DE) y Aprendizaje Incremental Basado en Población Continua (Continuous Population Based Incremental Learning – PBILc). Esta técnica híbrida se presenta en [35] para resolver el problema de Planificación de la

Expansión de la Red de Transmisión (Transmission network expansion planning - TNEP), obteniéndose un desempeño sobresaliente en robustez y eficiencia.

Las siguientes subsecciones describen los fundamentos de las técnicas metaheurísticas aplicadas en el algoritmo de optimización para obtener las soluciones de restauración del servicio.

3.3.1 Evolución diferencial (DE)

El algoritmo DE se basa en poblaciones, donde cada individuo dentro de la población es un vector de n-dimensiones que representa una solución candidata y que se somete a las operaciones de cruce y mutación antes de ser comparado con los individuos de la generación actual [35].

La idea básica de mutación consiste en tomar el vector diferencia entre dos individuos, y agregarle un tercer individuo para crear una nueva solución denominada vector mutante v_i . Este proceso se ilustra en la Figura 3.1 [41].

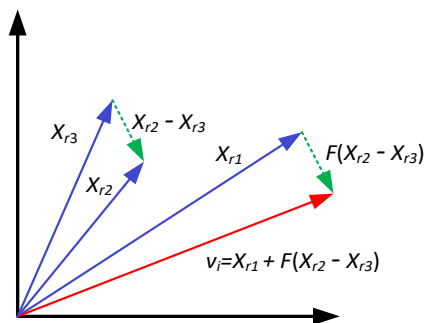


Figura 3.1 Generación del vector mutante v_i en DE [41].

Luego, la operación de cruce combina al vector mutante v_i con un individuo x_i de la población actual, para producir un vector de prueba u_i . Esta operación se implementa mediante la expresión:

$$u_{ij} = \begin{cases} v_{ij} & \text{Si } (r < C_r) \mid (J = J_{rand}) \\ x_{ij} & \text{caso contrario} \end{cases} \quad (17)$$

Después de que los N (tamaño de la población) vectores de prueba u_i son generados, el proceso de selección compara los vectores u_i y x_i , manteniéndose el vector más apto para la siguiente generación DE.

A continuación, se describe el pseudocódigo del algoritmo DE [35]:

Paso 1: Población inicial. Partir aleatoriamente de una población inicial $x^0 = [x_1^0, x_2^0, x_i^0, \dots, x_m^0]$ de m individuos. Cada individuo en la población es un vector de variables de estado que representa una solución candidata y se expresa como $x_i^0 = [x_{i,1}^0, x_{i,2}^0, x_{i,j}^0, \dots, x_{i,n}^0]$, donde n corresponde a la dimensión del problema.

Enfocado al problema de SR, x_i^0 representa el vector de estado de los elementos del sistema considerados para la restauración: interruptores de línea, unidades DG y clientes desenergizados por peso de prioridad.

Mientras el criterio de parado no se satisfaga **hacer**

Para $i = 1 : m$ **hacer** (m es el tamaño de población)

Paso 2: Mutación. Generar el vector mutante v_i^k de una perturbación en cada individuo i :

$$v_{i,j}^k = x_b^k + F \cdot (x_{1,j}^k - x_{2,j}^k + x_b^k - x_{3,j}^k)$$

Paso 3: Cruce. Generar el vector de prueba u_i^k de la combinación del individuo de la población actual x_i^k y el vector mutante v_i^k :

Si ($r < C_r$) | ($J = J_{rand}$) **luego**

$$u_{i,j}^k = v_{i,j}^k$$

Caso contrario

$$u_{i,j}^k = x_{i,j}^k$$

Fin Si

Paso 4: Selección. Comparar el vector de prueba u_i^k con cada individuo de la población actual x_i^k , para determinar qué individuo sobrevive para la siguiente generación:

Si $f(u_i) < f(x_i)$ **luego**

$$x_i^{k+1} = u_i^k$$

Caso contrario

$$x_i^{k+1} = x_i^k$$

Fin Si

Fin Para

Fin Mientras

Donde: $x_1^k, x_2^k, x_3^k \in [1, m]$ con $x_1 \neq x_2 \neq x_3$, $F \in [0, 2]$ es el factor de escala, x_b^k es el individuo con la mejor solución, r es un número aleatorio entre $[0, 1]$, $C_r \in [0, 2]$ es la tasa de cruce, $J = J_{rand}$ es un número aleatorio entre $[0, n]$, que garantiza que al menos un componente del vector mutante v_i^k intercambie información al vector de prueba u_i^k .

3.3.2 Aprendizaje incremental basado en población continua (PBILc)

El algoritmo PBILc se basa en un aprendizaje competitivo. En cada generación, el algoritmo determina las características más relevantes e irrelevantes de los individuos (soluciones candidatas) de la población, y aprende de estas características para generar nuevos individuos a través del ajuste de un vector de probabilidad p , que hace que la siguiente generación sea más similar a los individuos más aptos [35].

La idea de esta técnica consiste en crear elementos $x_i(k)$ de cada individuo x_i , generando un número aleatorio Gaussiano que tiene una media de p_k y una desviación estándar σ_k . A medida que aumenta el recuento de generaciones, disminuye la desviación estándar del generador de números aleatorios gaussianos, haciendo que el vector p converja hacia su solución óptima. Los elementos de las soluciones candidato se limitan a un dominio $x_i(k) \in [x_{min}(k), x_{max}(k)]$. La Figura 3.2 ilustra este proceso, se observa que la función densidad de probabilidad (PDF) tiene una desviación estándar más pequeña en las generaciones siguientes, que permite que el algoritmo converja hacia la solución óptima [41]. A continuación, se describe el pseudocódigo del algoritmo PBILc [35]:

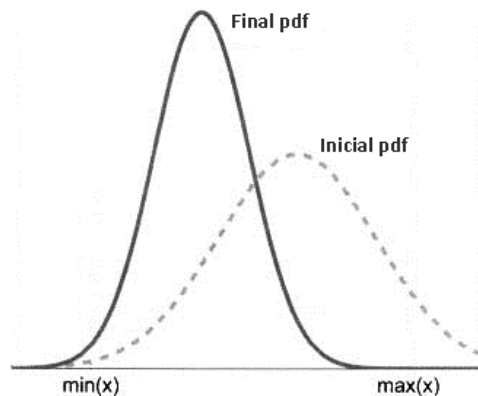


Figura 3.2 Ilustración de evolución de PDF en PBILc [42].

Paso 1: Población inicial (similar al algoritmo DE) y modelo probabilístico. El vector de probabilidad n -dimensional p se basa en un modelo de función de distribución normal $p(\mu, \sigma)$. El valor inicial de la media $p(\mu^0) = \mu_1^0, \dots, \mu_n^0$ se genera de una distribución aleatoria normal con un dominio de espacio $[x^{min}, x^{max}]$, y una desviación estándar $p(\sigma^0) = \sigma_1^0, \dots, \sigma_n^0$ que se ajusta para obtener diversidad de individuos x_i .

Mientras el criterio de parado no se satisfaga **hacer**

Paso 2: Actualizar el vector de probabilidad p . El vector de probabilidad $p(\mu^k, \sigma^k)$ se actualizará en función de:

Para $j = 1 : n$ hacer

$$\mu_j^k = (1 - \eta) \cdot \mu_j^k + \eta \cdot (x_{(1,j)}^k + x_{(2,j)}^k - x_{(m,j)}^k)$$

$$\sigma_i^k = (1 - \eta) \cdot \sigma_i^k + \eta \cdot \sqrt{\sum_{i=1}^{N_{best}} \frac{(x_{(i,j)} - \mu_{best}^k)^2}{N_{best}^k}}$$

Fin Para

Paso 3: Generar la siguiente población. La siguiente población se genera usando una distribución aleatoria normal:

Para $i = 1 : m$ hacer (m es el tamaño de la población)

$$x_{i,j}^{k+1} = N(p(\mu_j^k), p(\sigma_j^k))$$

Fin Para

Fin Mientras

Donde: $\eta \in ([0, 1])$ es la tasa de aprendizaje, N_{best} representa el número de individuos con mejores soluciones, μ_{best} es la media de los individuos N_{best} , $x_{(1,j)}^k, x_{(2,j)}^k$ representan el primero y segundo individuo con las mejores soluciones y $x_{(m,j)}^k$ corresponde al individuo con peor solución.

3.3.3 Algoritmo DE-PBILc

Los operadores tradicionales de cruce y selección no son tan eficientes para obtener soluciones óptimas o casi-óptimas y presentan un mal desempeño para problemas de alta dimensión. El algoritmo DE-PBILc combina ambas técnicas para realizar una búsqueda más eficiente y así cubrir ciertas desventajas individuales. Integra DE que utiliza la diferencia de vectores de parámetros para explorar el espacio de búsqueda y PBILc que usa modelos probabilísticos basados en aprendizaje competitivo. Esta metaheurística híbrida aplica una mutación diferencial doble que permite acelerar el proceso de optimización sin perder robustez o eficiencia [35].

El proceso de optimización inicia definiéndose una probabilidad de combinación $p_{comb} \in [0, 1]$. Después, cada individuo u_i se crea usando DE si un número aleatorio $r \in [0, 1]$ es menor a p_{comb} , caso contrario, se emplea PBILc para generarlo. Si se usa DE, se puede escoger entre dos tipos de mutación diferencial, dependiendo de si r es menor a una probabilidad de mutación $p_{mut} \in [0, 1]$ se aplica mutación trigonométrica, caso contrario, el enfoque DE/best. Dentro de este último, se efectúa una modificación al algoritmo DE-PBILc de [35] para que combine un par de métodos de como generar aleatoriamente el vector mutante v_i . Con este paso se

obtienen más permutaciones para generar v [41]. Se aplica el esquema DE/best/1 si r es menor a una probabilidad de doble mutación $p_{double-mut} \in [0, 1]$, caso contrario, el esquema DE/best/2. Se seleccionó estos esquemas debido a que tienen los mejores desempeños conforme a los resultados registrados en [36].

A continuación, se presenta el pseudocódigo del algoritmo DE-PBILc [35] acoplado para la restauración del servicio:

Paso 1: Población inicial. Partir aleatoriamente de una población inicial (similar al algoritmo DE).

Mientras el criterio de parada no se satisfaga **hacer**

Para $i = 1 : m$ **hacer** (m es el tamaño de la población)

Si $r \leftarrow N[0, 1] < p_{comb}$ **luego**

Paso 2: Aplicar mutación diferencial doble.

$x_1^k, x_2^k, x_3^k, x_4^k \rightarrow$ Seleccionado aleatoriamente

Si $r \leftarrow N[0, 1] < p_{mut}$ **luego** (Mutación trigonométrica)

$$v_{i,j}^k = \frac{x_{1,j}^k - x_{2,j}^k - x_{3,j}^k}{3} + (p_2 - p_1) * (x_{1,j}^k - x_{2,j}^k) + (p_3 - p_2) * (x_{2,j}^k - x_{3,j}^k) + (p_1 - p_3) * (x_{3,j}^k - x_{1,j}^k)$$

Caso contrario

Si $r \leftarrow N[0, 1] < p_{double-mut}$ **luego** (DE/best/1)

$$v_{i,j}^k = x_b^k + F * (x_{1,j}^k - x_{2,j}^k)$$

Caso contrario (DE/best/2)

$$v_{i,j}^k = x_b^k + F * (x_{1,j}^k - x_{2,j}^k + x_{3,j}^k - x_{4,j}^k)$$

Fin Si

Fin Si

Paso 3: Aplicar cruce DE.

Si ($r < C_r$) | ($J = J_{rand}$) **luego**

$$u_{i,j}^k = v_{i,j}^k$$

Caso contrario

$$u_{i,j}^k = x_{i,j}^k$$

Fin Si

Caso contrario

Paso 4: Aplicar PBILc.

$$u_{i,j}^k = N(p(\mu_j^k), p(\sigma_j^k))$$

Fin Si

Paso 5: Aplicar selección DE.

Si $f(u_i) < f(x_i)$ **luego**

$$x_i^{k+1} = u_i^k$$

Caso contrario

$$x_i^{k+1} = x_i^k$$

Fin Si

Fin Para

Paso 6: Actualizar el vector de probabilidad p . Ver algoritmo PBILc.

Fin Mientras

Donde: $p_1 = f(x_1)^k/p'$, $p_2 = f(x_2)^k/p'$, $p_3 = f(x_3)^k/p'$, $p' = f(x_1) + f(x_2) + f(x_3)$, y x_b^k es el individuo con la mejor solución.

3.4 Implementación del algoritmo de Restauración del Servicio

La inteligencia de restauración del servicio (SR) ha sido desarrollado mediante la aplicación de un algoritmo de optimización centralizado que se basa en una metaheurística híbrida que combina las técnicas DE y PBILc. En presencia de una falla permanente, el algoritmo determina y lleva a cabo las soluciones de restauración sobre los equipos inteligentes correspondientes para restablecer el servicio al mayor número de secciones no afectadas del sistema de distribución. La inteligencia SR se establece en torno a tres procesos generales [5], que se aplican en el algoritmo:

1. Actualización dinámica de la configuración del sistema de distribución.
2. Lógica para evaluar las soluciones de restauración.
3. Reconfiguración automática del sistema de distribución.

El proceso 1 adquiere la topología actual del sistema de distribución (estado de los equipos de conmutación, despacho de generación distribuida, variables eléctricas, falla(s), etc.). El proceso 2 efectúa la inteligencia sobre la base de la información del punto anterior para determinar el plan de maniobras idóneo de restauración del servicio. Finalmente, el proceso 3 reconfigura automáticamente el sistema de distribución basándose en el plan de restauración.

Las siguientes subsecciones describen la implementación del algoritmo para resolver el problema de SR, esquematizado en el diagrama de flujo de Figura 3.3.

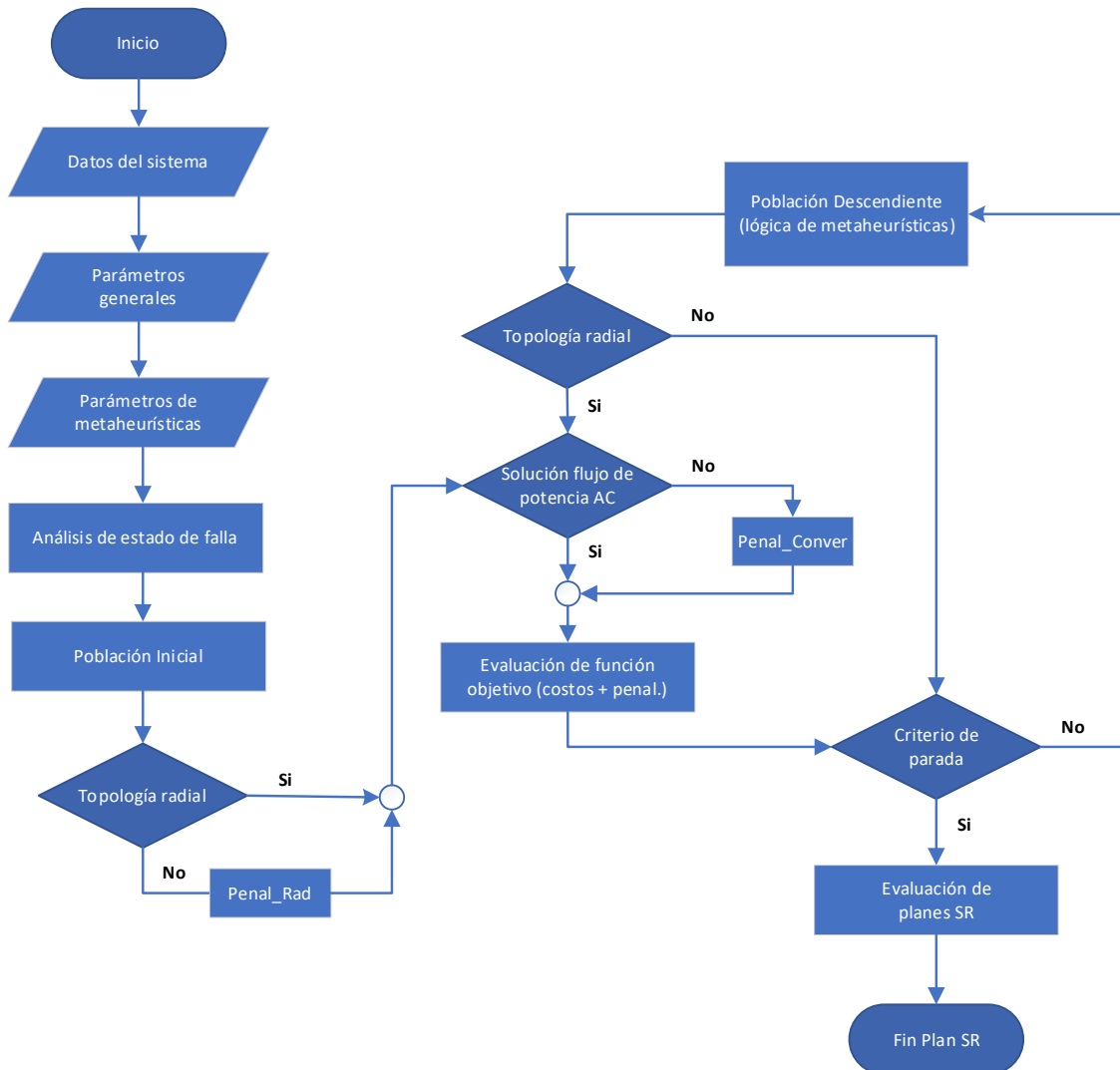


Figura 3.3 Diagrama de flujo del algoritmo para la resolución del problema de SR.

Datos del sistema: Bloque que adquiere la configuración del sistema de distribución. Esta información incluye los nodos de carga, nodos fuente, impedancia de líneas, estado de interruptores de líneas (abierto - cerrado), sección(es) en falla, prioridades de las cargas, estado de las unidades DG. La dimensión del problema n está dada por el número de los elementos del sistema dispuestos para la restauración del servicio: interruptores de líneas (n_L), unidades DG (n_{DG}) y el deslastre de carga por prioridad de clientes (n_{LC}). El dominio de búsqueda $[x_{min}, x_{max}]$ comprende desde la topología inicial x_{min} (estado de interruptores de líneas y

unidades DG post falla) hasta x_{max} (todos los interruptores de líneas cerrados y las unidades DG en funcionamiento - excepto los elementos que aíslan la falla).

Parámetros generales: Valores definidos previo a la ejecución de la lógica de metaheurísticas y que se requieren para la convergencia del algoritmo. Esta información incluye el número de individuos de la población m , el número máximo de iteraciones I_{max} , el número máximo de evaluaciones de función objetivo FO_{max} , el número de repeticiones o ensayos n_{exp} .

Parámetros de metaheurísticas: Inicialización de los parámetros de metaheurísticas DE-PBILc para la resolución del problema de SR. Esta información incluye el factor de escala F , la tasa de cruce Cr , el número de mejores individuos N_{best} , la tasa de aprendizaje η , la media inicial μ^0 , la desviación estándar inicial σ^0 , la probabilidad de combinación p_{comb} , la probabilidad de mutación p_{mut} y la probabilidad de doble mutación $p_{double-mut}$. Los valores se tomaron de [34] y tras pruebas de ensayo-error, se ajustaron para obtener mejores resultados. La Tabla 3.2 resume los parámetros DE-PBILc utilizados para todos los sistemas y casos de estudio.

Tabla 3.2 Parámetros de metaheurísticas DE-PBILc.

DE-PBILc
$F = 1$
$Cr = 0.3$
$\eta = 0.05$
$N_{best} = m/2$
μ^0 aleatorio
$\sigma^0 = 2$
$p_{comb} = 0.9$
$p_{mut} = 0.4$
$p_{double-mut} = (p_{comb} + p_{mut})/2$

Análisis de estado de falla: Bloque que evalúa el posible camino para restaurar el servicio a las zonas desenergizadas aguas abajo de la localización de la falla. Esta estrategia puede contribuir a disminuir el tiempo de cálculo y mejorar el desempeño del algoritmo [39]. A partir de este análisis, se obtiene la opción adecuada para generar la población inicial, reduciéndose el espacio de búsqueda de soluciones del problema. El diagrama de flujo de la Figura 3.4 escenifica el análisis de estado de falla realizado para obtener el camino de restauración. Se evalúa el flujo de potencia AC óptimo considerando:

- *Reconfiguración del sistema:* Es el camino más económico y sencillo para la restauración del servicio. Se verifica si existe éxito de convergencia y el voltaje mínimo del sistema sea igual o mayor al límite permisible, al disponerse todos los interruptores de línea cerrados (excepto los elementos que aíslan la falla). En caso afirmativo, el siguiente bloque (población inicial) generará soluciones iniciales centradas únicamente en la conmutación de los interruptores de línea. Caso contrario, se evalúa el siguiente camino.
- *Unidades de generación distribuida:* Al disponer de unidades de generación distribuida localizadas en distintos puntos del sistema, se puede despachar estos elementos para conseguir restablecer el servicio [39]. Se comprueba si existe éxito de convergencia y el voltaje mínimo del sistema sea igual o mayor al límite permisible, al disponerse todos los interruptores de línea cerrados y el despacho de las unidades DG (excepto los elementos que aíslan la falla). En caso afirmativo, el siguiente bloque (población inicial) generará soluciones iniciales considerando la conmutación de los interruptores de línea y la operación de las unidades DG. De otro modo, se opta por el camino restante.
- *Prioridades de clientes:* En caso de que no sea posible la restauración total de las cargas fuera de servicio a partir de los elementos disponibles del sistema (conmutación de interruptores de línea y despacho de unidades DG), se efectúa el deslastre de carga para lograr la restauración parcial del servicio. El siguiente bloque (población inicial) generará soluciones iniciales considerando la conmutación de los interruptores de línea, el despacho de las unidades DG y el deslastre de carga en atención al peso de prioridad (LP_i) de los clientes del sistema.

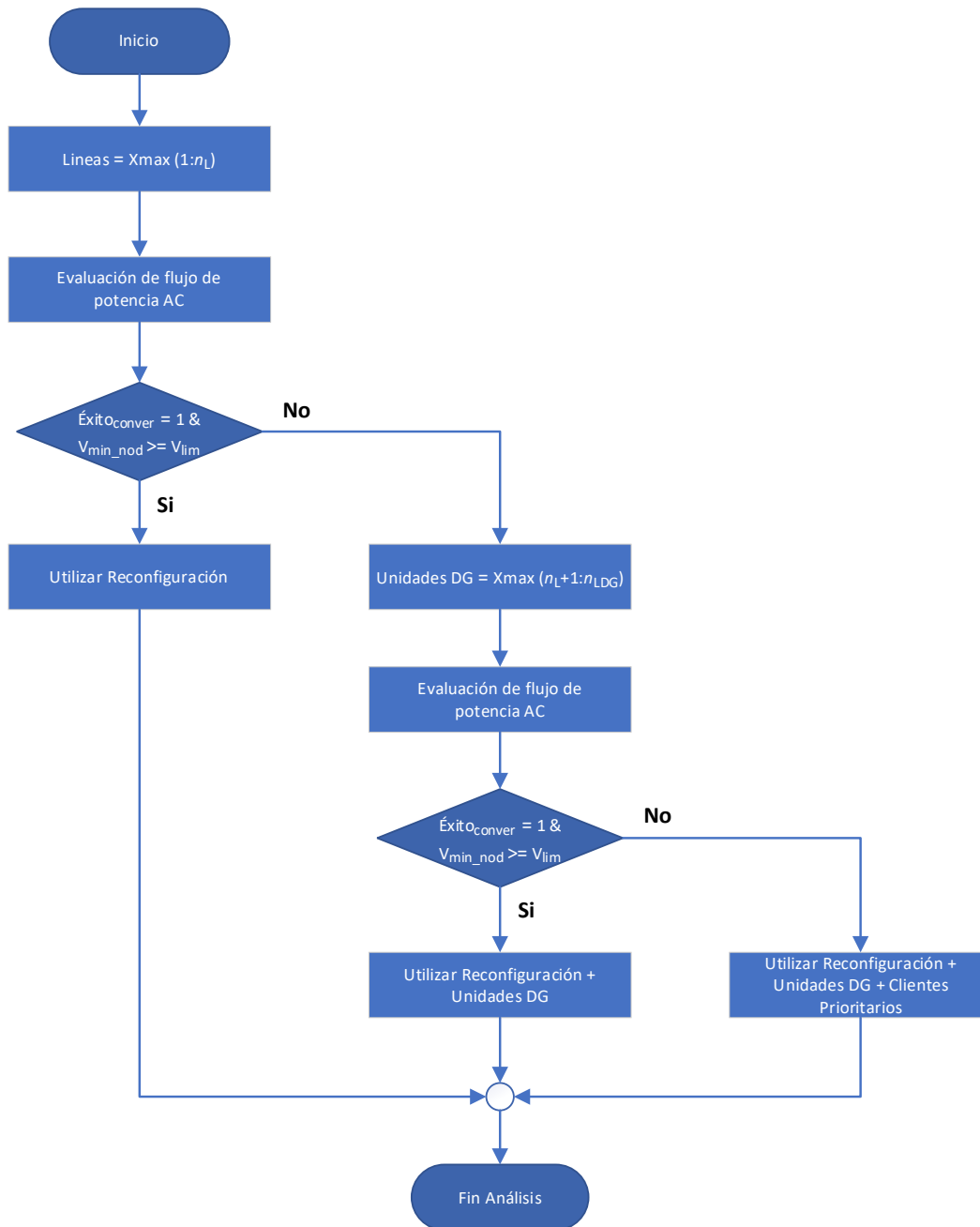


Figura 3.4 Diagrama de flujo del análisis de estado de falla para la obtención del camino de restauración.

Para el caso del deslastre de carga, si se considera que todas las L cargas del sistema son candidatos, el espacio de búsqueda se incrementa en un factor de 2^L . Considerando la configuración de operación radial del sistema de distribución, la combinación aleatoria de los elementos (interruptores, unidades DG y el deslastre de carga), incrementará la cantidad de soluciones inviables, ocasionando que el

tiempo de procesamiento y desempeño del algoritmo resulten afectados. Para facilitar el cálculo de un plan de restauración, se puede emplear un método de búsqueda heurística para construir una lista de deslastre con los posibles candidatos [42]. Se utiliza una estrategia heurística similar a [40], que se enfoca en determinar y registrar las zonas del sistema con los valores más bajos de voltaje, inferiores al límite permisible. Los elementos de este registro se ordenan conforme el nivel de criticidad de carga, obteniéndose la lista de deslastre *Lista_LC*. Luego, se determina la matriz con las diferentes combinaciones de los candidatos de *Lista_LC*, asumiendo el estado 1 si el candidato es desenergizado y 0 en caso contrario. Posteriormente, se evalúa el flujo de potencia óptimo considerando las columnas de la matriz resultante y los elementos del sistema dispuestos (interruptores de línea cerrados y unidades DG operativas). Para los casos donde se tiene éxito de convergencia y el voltaje mínimo del sistema sea igual o mayor al límite permisible, se calcula el peso de prioridad de clientes *PL* (según demanda de potencia y criticidad de carga). Finalmente, se selecciona el caso con el menor peso como el deslastre de carga adecuado. El diagrama de flujo de la Figura 3.5 resume la metodología para determinar el deslastre de carga según la prioridad de clientes.

El deslastre de carga conlleva la actuación de interruptores de línea, por lo que se puede determinar esta operación. Si el deslastre de carga involucra dejar fuera de servicio a varios clientes, se debe determinar la existencia de secuencia(s) de clientes a desenergizar, a fin de reducir el número de interruptores de líneas a conmutar. Este registro de interruptores operados *Lin_off_LC*, se actualiza en los dominios de búsqueda, como estrategia para facilitar la obtención de soluciones viables. En la Figura 3.6 se presenta el diagrama de flujo que escenifica la metodología para determinar los interruptores de líneas a conmutar según el deslastre de carga.

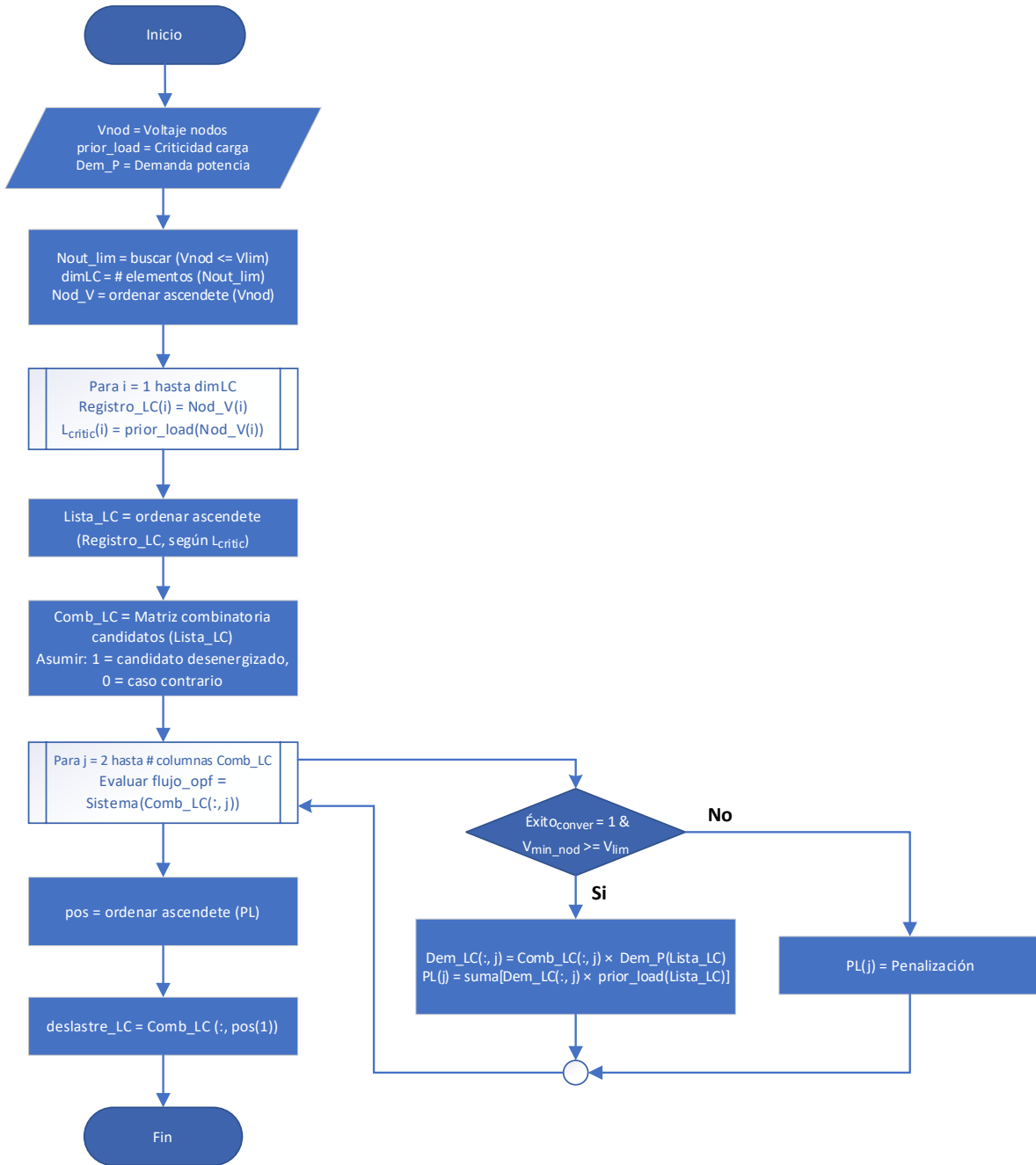


Figura 3.5 Diagrama de flujo de metodología de deslastre de carga según la prioridad de clientes.

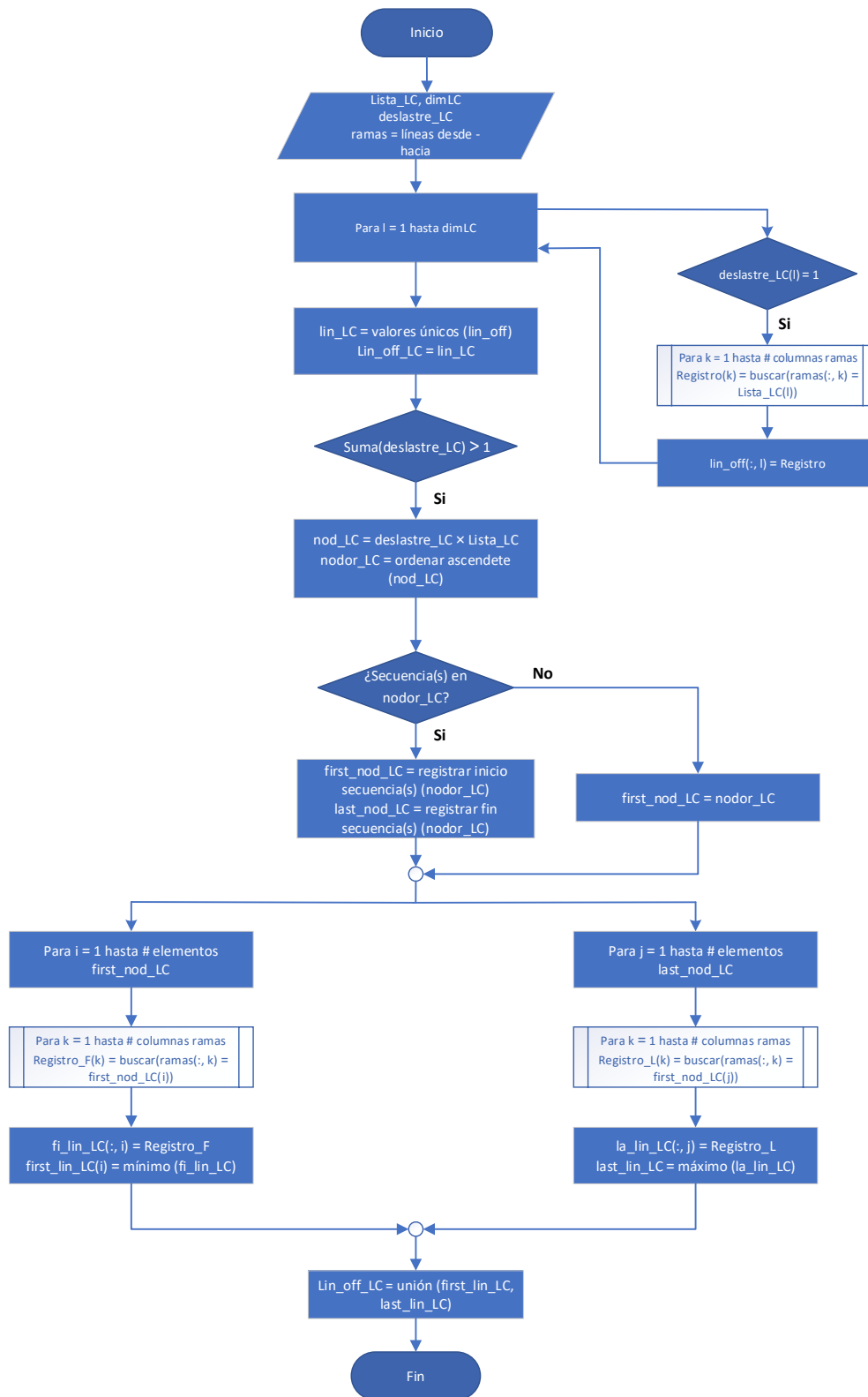


Figura 3.6 Diagrama de flujo de metodología de interruptores de línea a conmutar según el deslastre de carga.

Población Inicial: Bloque de función que genera el conjunto de individuos (soluciones candidatas) para ser evaluadas por la lógica de metaheurísticas y posteriormente, ajustar el espacio de búsqueda. Cada individuo de la población es un vector n dimensional que representa el estado de los elementos del sistema disponibles para la restauración del servicio: interruptores de línea, unidades DG y deslastre de carga por prioridad de clientes. El estado cerrado de los interruptores se representa como 1 y 0 para el estado abierto. El funcionamiento de las unidades DG se representa como 1, o 0 de otra forma. Para el deslastre de carga, se asume 1 si el candidato de la lista se desenergiza y 0 en caso contrario.

La población inicial se genera de forma aleatoria (algoritmo tradicional) [22], [34], [37]. Una forma común de generarla se describe en [36], a través de la expresión:

$$x_{i,j}^0 = x_{j,min} + (x_{j,max} - x_{j,min}) * rand_{i,j}[0,1] \quad (18)$$

Mediante (18) se crean topologías aleatorias en las que se agregan líneas (cierre de interruptores) respecto al estado inicial de red. Sin embargo, no se generan topologías en las que se quitan líneas (apertura de interruptores). Para mejorar el proceso de generación de las soluciones iniciales se combina con reglas heurísticas, tomando en cuenta la opción proveniente del bloque anterior (análisis de estado de falla). Las Figuras 3.7, 3.8 y 3.9 esquematizan los diagramas de flujo para la generación de la población inicial (x^0). Los bloques previos a x^0 , permiten que se modifique x_{min} y haya topologías en las que los interruptores inicialmente cerrados pasen a abrir, proporcionando una población inicial diversa.

Donde: x_{min} es la topología inicial, x_{max} es la topología resultante al estar en operación todos los elementos del sistema (interruptores de línea cerrados, unidades DG en funcionamiento y el deslastre de clientes de ser necesario), $Rango$ es la cantidad total de elementos disponibles (interruptores y unidades DG) para las maniobras de restauración, n_L es número de líneas (ramas), n_{DG} es la cantidad de unidades DG, n_{LDG} es el número de elementos del sistema (interruptores y unidades DG), n es la dimensión del problema, m es el tamaño de la población, $Disp_elemen_L$ es el número de interruptores de línea disponibles (interruptores de transferencia), $Disp_fault$ es la cantidad de interruptores de línea abiertos por la falla, Lin_ini es la cantidad de interruptores de línea que estuvieron funcionales pre-falla, $Conmut_open_L$ es el número aleatorio de interruptores de línea a abrir en cada individuo (x_i^0) de la población, $Conmut_close_L$ es el número aleatorio de interruptores de línea a cerrar en cada individuo, $Conmut_elemen_{DG}$ es la cantidad aleatoria de unidades DG en operación en cada individuo, Lin_off_LC son las líneas desactivadas conforme secuencia(s) de deslastre de clientes y x^0 es la población inicial generada.

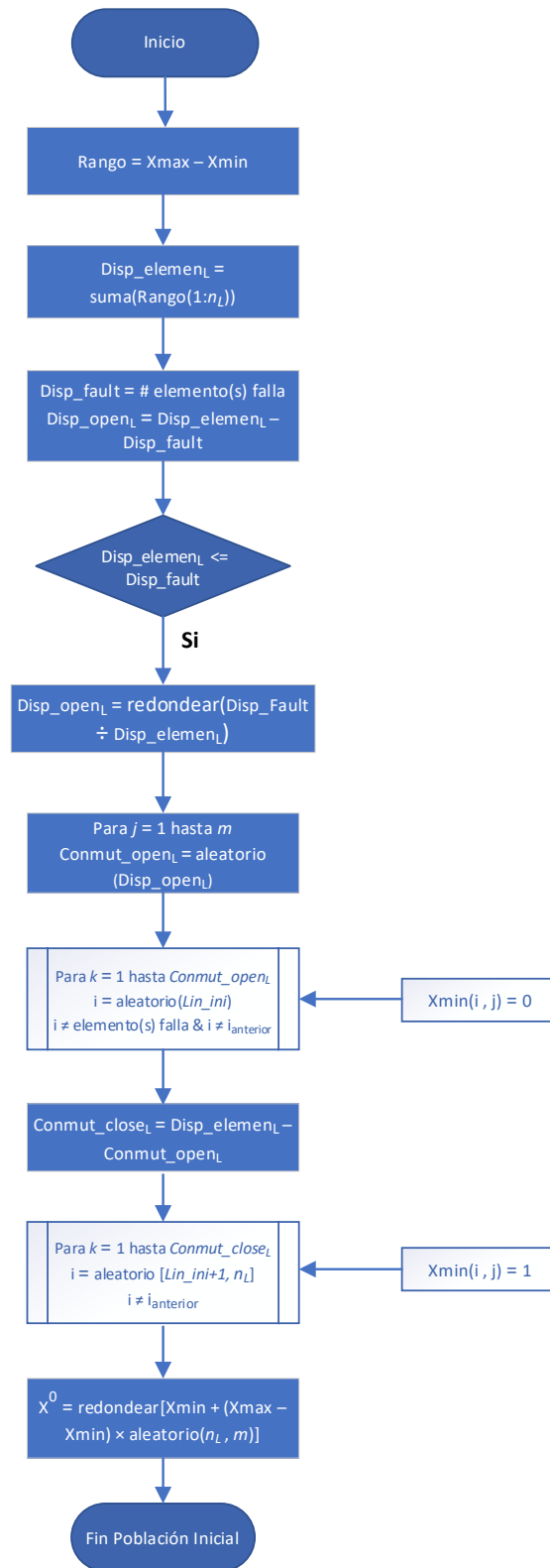


Figura 3.7 Diagrama de flujo de generación de la población inicial considerando la conmutación de interruptores de línea del sistema.

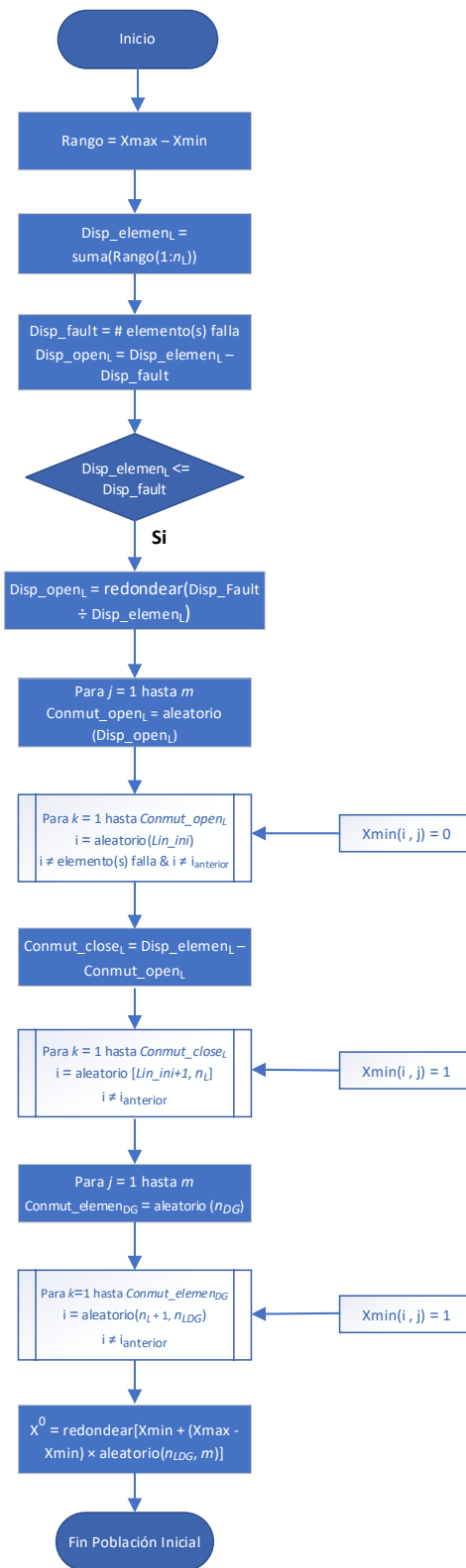


Figura 3.8 Diagrama de flujo de generación de la población inicial considerando la conmutación de interruptores de línea y las unidades DG del sistema.

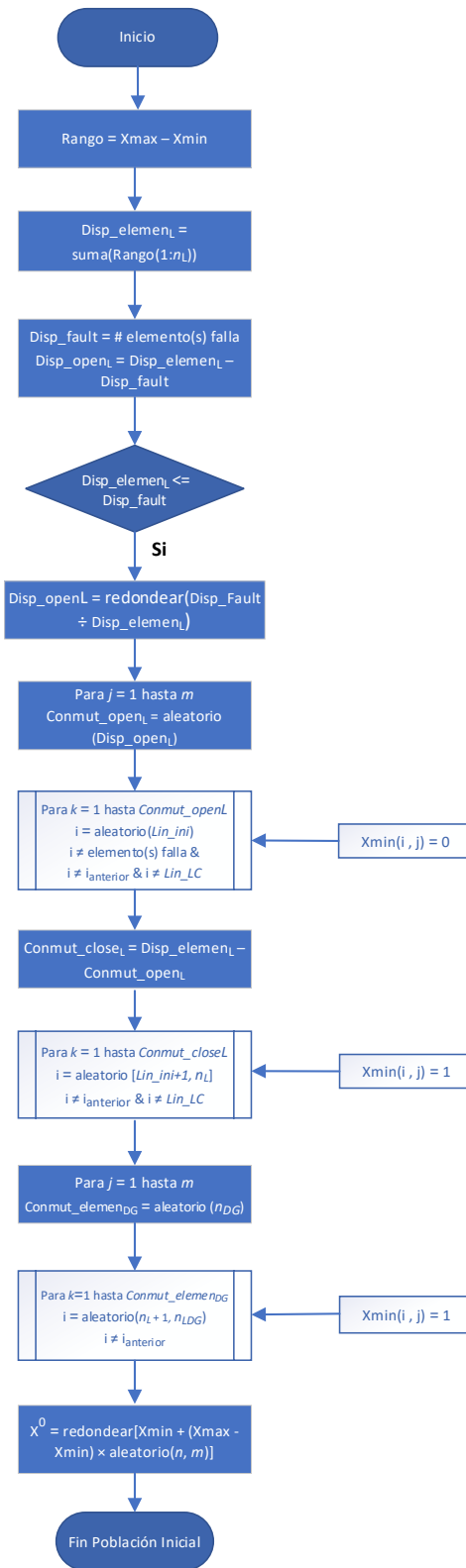


Figura 3.9 Diagrama de flujo de generación de la población inicial considerando la conmutación de interruptores de línea, las unidades DG y el deslastre de clientes.

Topología radial: Bloque de función que verifica que las topologías de red (individuos) provenientes de la población inicial y posteriormente, de la lógica de metaheurísticas (proceso de búsqueda de mejores soluciones), operen en configuración radial [35], [36], [43], es decir, que se permita un máximo de una entrada de flujo de potencia positivo en cada nodo del sistema [44].

Para representar la restricción de operación radial del sistema de distribución se emplea la expresión generalizada (*Condición 1*):

$$Rad \rightarrow M = n_b - n_{b_s} \quad (19)$$

Donde: M es el número de líneas operativas (ramas) de la topología, n_b es el número de nodos energizados y n_{b_s} es el número de nodos de subestación (fuentes) del sistema [44].

Al pasar M al segundo miembro de la ecuación (19) e igualar esta expresión para Rad , se puede clasificar la topología de red en atención al valor resultante:

- Para $Rad < 0$, la topología de red opera en configuración mallada.
- Si $Rad = 0$, la topología de red opera en configuración radial.
- Cuando $Rad > 0$, la topología de red posee nodos desconectados (aislados).

En caso de que la topología de red no cumpla la condición de radialidad, se penalizará con un valor alto al resultado de la función objetivo de esta para que se considere como una solución de baja calidad en el proceso de selección de la lógica de metaheurísticas (Ver Sección 3.3 Algoritmo DE-PBILc, paso 5). Adicionalmente, se puede mejorar los individuos de la Población Inicial (bloque anterior) usando criterios que permitan que el algoritmo de optimización inicialice en regiones más cercanas a los valores óptimos [36], mejora que se conoce como *población evolutiva* [34]. En este trabajo, se reemplaza las topologías de red que poseen nodos aislados ($Rad > 0$) por topologías radiales ($Rad = 0$) provenientes de una Población Nueva/Descendiente. Con esta estrategia se intenta acelerar el proceso de convergencia y, además, mejorar el desempeño del algoritmo.

Solución de flujo de potencia AC óptimo: La *Condición 1* no es suficiente para asegurar que la topología de red (solución), opera en configuración radial. Adicional, se requiere que sea conexa (*Condición 2*), es decir, que los nodos de la solución deben estar conectados [39]. La convergencia de flujo de potencia AC óptimo garantiza que la topología de red es factible [36], es decir que, considera las restricciones operativas y eléctricas del sistema de distribución. En caso de no convergencia, se penalizará con un valor alto al resultado de la función objetivo de

la topología de red para que se considere como una solución de baja calidad en el proceso de selección de la lógica de metaheurísticas (Ver Sección 3.3, paso 5).

Conforme lo expuesto en [34] y [36], se puede mejorar el tiempo de cálculo del algoritmo al implementar alguna estrategia que permita reducir el cómputo de flujos de potencia AC óptimos en el proceso de evaluación de la función objetivo. Se consideró que en las topologías que proceden de la lógica de metaheurísticas, únicamente se calcule el flujo de potencia AC óptimo para aquellas que cumplen la condición 1, y, además, sean diferentes a las generadas en la Población Inicial.

Evaluación de la función objetivo: Bloque que determina el valor de restauración del servicio (función objetivo) al aplicar cada individuo (topología) de la población (inicial/descendiente). La evaluación de la función objetivo se determina empleando la ecuación 16 (Ver Sección 3.2.3), donde las variables (potencia desenergizada, pérdidas de potencia) se obtienen al resolver el flujo de potencia AC óptimo y los costos operativos (pesos), que se muestran en la Tabla 3.3, se referenciaron en orden a las prioridades de restauración [9], [38]. Adicionalmente, se considera un valor alto de penalización (no radialidad o no convergencia) para las topologías de red que no permiten operar al sistema de distribución en condiciones adecuadas.

Tabla 3.3 Costos de operaciones de restauración en la función objetivo de SR.

Peso en función objetivo	Valor
Costo de zona desenergizada (c_z^R)	3 \$/kW
Costo de operación de interruptor (c^{sw})	1 \$
Costo de generación DG (c_p^{DG})	0.1 \$/kWh
Costo de pérdida de potencia activa (c^{lss})	0.02 \$/kW

Población Descendiente: En cada iteración se genera una nueva población resultante de la lógica de metaheurísticas (Ver Sección 3.3, pasos 2 - 6), que se encuentra ajustada al dominio de búsqueda [x_{min} , x_{max}]. En cada generación, el algoritmo determina las características más significativas de los individuos (topologías) en la población, y aprende de esas características para generar nuevos individuos, lo que hace que la siguiente generación sea más similar a los individuos más aptos [34]. Este proceso se realiza de forma iterativa y se conoce como evolución del enjambre.

Criterios de parada: El proceso iterativo del algoritmo de SR se detiene cuando se satisface alguno de los criterios descritos. Los dos primeros se tomaron conforme [35]. El tercero se considera en atención a lo sugerido en [36], donde se menciona

que, en la práctica, se puede emplear el estancamiento de la búsqueda como un criterio alternativo.

- Si el contador de iteraciones alcanza su número máximo (I_{max}),
- Si el contador de evaluaciones de función objetivo alcanza su número máximo (FO_{max})
- Si la diferencia entre el resultado de mejor calidad ($f_{x_{opt(1)}}$) de la función objetivo de la última iteración y el promedio de los x anteriores resultados de iteraciones es igual o inferior a una precisión deseada (ej. 0.001). En este trabajo se usa el número de mejores individuos (N_{best}) como valor de x .

Evaluación de planes SR: Para cada ensayo, se registra la solución de menor costo de función objetivo (F_{opt}) conseguida por el algoritmo de SR tras satisfacerse alguno de los criterios de parada (convergencia del algoritmo). Posteriormente, el algoritmo selecciona la solución de menor costo del registro ($F_{opt_{best}}$).

Fin Plan SR: Se imprime el plan de restauración (conmutación de interruptores de línea, despacho de unidades DG, deslastre de carga) que representa el menor costo operativo ($F_{opt_{best}}$) para que se restaure el servicio al mayor número de zonas no afectadas del sistema de distribución posterior a la falla.

4 RESULTADOS DEL ALGORITMO DE RESTAURACIÓN DEL SERVICIO

4.1 Introducción

En el presente capítulo se verifica el desempeño del algoritmo metaheurístico DE-PBILc aplicado para resolver el problema de restauración del servicio en sistemas de distribución. Se utiliza como red de prueba un sistema estándar del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE). Diferentes casos de estudio se definen para comprobar su desempeño y eficiencia. La calidad de los resultados se evalúa conforme algunos criterios [35]. El primero es la *tasa de restauración*, que establece la proporción de ensayos para los que se alcanza una solución de buena calidad, es decir, una topología de red que satisface las restricciones operativas y eléctricas para restablecer el servicio. El siguiente es el *éxito de respuesta*, que es un indicador de robustez que determina el porcentaje de ensayos en los que la solución obtenida resulta ser la idónea, complementa al indicador anterior y se considera aceptable con un valor de al menos el 70 %. El tercero es la *desviación estándar* del total de las iteraciones efectuadas, que es un indicador de la variabilidad de convergencia del algoritmo (componente de búsqueda aleatoria alta o baja). El cuarto es el *promedio de iteraciones*, que es la media estadística del número de iteraciones necesarias en cada ensayo hasta conseguir la solución, es un indicador de la eficiencia del algoritmo. Finalmente, el quinto es el *tiempo de cálculo* que le toma al algoritmo efectuar los ensayos y determinar la solución idónea, es un indicador de la rapidez de convergencia. En las siguientes secciones se presenta el sistema de prueba, los escenarios analizados y los resultados obtenidos.

4.2 Sistema de prueba estándar

La red de prueba empleada para evaluar el desempeño del algoritmo de SR es el sistema de distribución estándar de 33 nodos de IEEE que se muestra en la Figura 4.1 [43]. El sistema radial consta de: un alimentador, 37 líneas, 32 interruptores de línea normalmente cerrados, 5 interruptores normalmente abiertos (interruptores de transferencia) ubicados en las líneas 33 a 37, y 32 puntos de carga. La potencia base (S_{base}) es 100 MVA, el voltaje base (V_{base}) es 12.66 kV, la carga total del sistema es 3715 kW y 2300 kVAr. Los datos completos del sistema de prueba estándar se encuentran disponibles en [45].

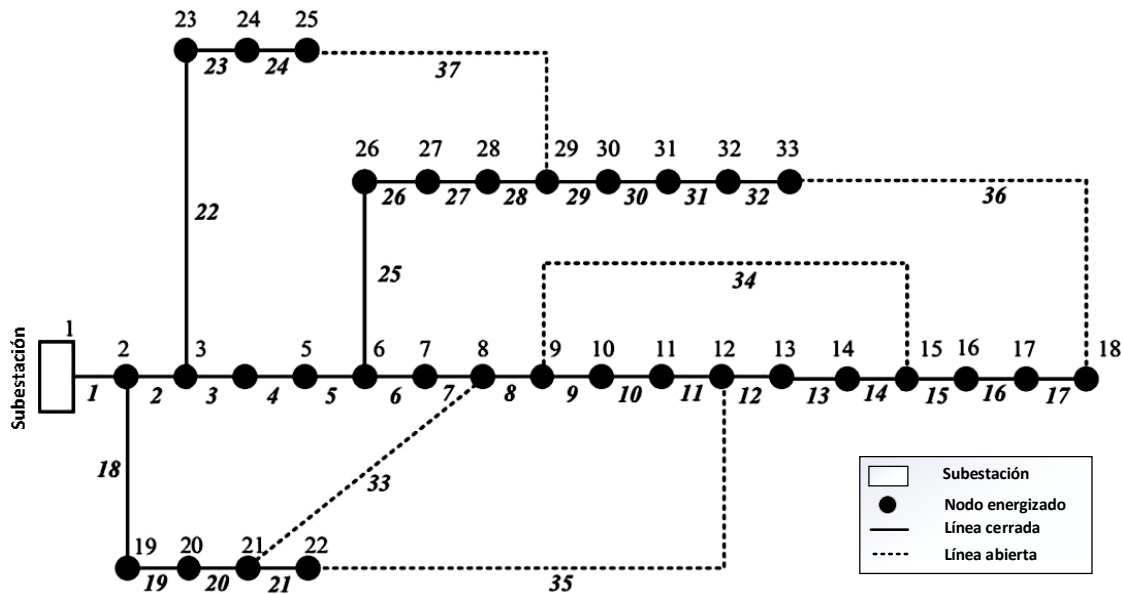


Figura 4.1 Diagrama unifilar del sistema de distribución de 33 nodos de IEEE [43]

Existen pocos trabajos investigativos publicados de restauración del servicio en sistemas de distribución aplicando técnicas metaheurísticas. En algunos, la metodología propuesta se evalúa en sistemas de prueba que no se dispone de la información de la red [19], [39]. Se eligió este sistema de distribución estándar debido a la facilidad de acceso a los datos de la red y a la existencia de información en trabajos relacionados: reconfiguración de red para reducción de pérdidas [43], reconfiguración de red considerando unidades DG para reducción de pérdidas [46], restauración del servicio en sistemas de distribución [37], [47]. Adicionalmente, se asume que existe un interruptor seccionador en cada línea del sistema que es controlado remotamente.

4.3 Casos de estudio

Para evaluar el desempeño y eficiencia del algoritmo de SR se definen algunos casos de estudio en los que se somete al sistema de prueba estándar a distintos escenarios de falla. Los casos de estudio se plantearon con el objetivo de verificar los caminos existentes para restaurar el servicio en las zonas desenergizadas aguas abajo de la localización de la falla [39] (Ver Sección 3.4 – apartado: *Análisis de estado de falla*). Los casos considerados son:

- Caso 1: Reconfiguración del sistema para la restauración.
- Caso 2: Integración de unidades de generación distribuida para la restauración.
- Caso 3: Consideración de clientes prioritarios para la restauración.

4.3.1 Caso 1: Reconfiguración del sistema para la restauración

Se somete al sistema de prueba estándar a determinadas fallas en las que el algoritmo de SR obtiene la mejor solución a partir de la conmutación (apertura / cierre) de los interruptores de líneas del sistema. Los escenarios planteados son:

4.3.1.1 Localización de falla en líneas 7, 9, 14, 28 y 32

En [46] se presenta la apertura de los interruptores de líneas 7, 9, 14, 28 y 32, como resultado de la reconfiguración de red para reducir las pérdidas del sistema. Enfocado en la restauración del servicio, se considera la localización de falla en múltiples zonas del sistema, asumiendo que dichos interruptores de líneas operan para despejar las fallas, dejando fuera de servicio a los nodos 8 a 18 y 29 a 33. Con este escenario se pretende evaluar el desempeño del algoritmo de SR y validar la solución resultante al compararla con los resultados de [46]. En la Figura 4.2 se escenifica la falla múltiple en el sistema, los símbolos en forma de rayo representan la localización de falla.

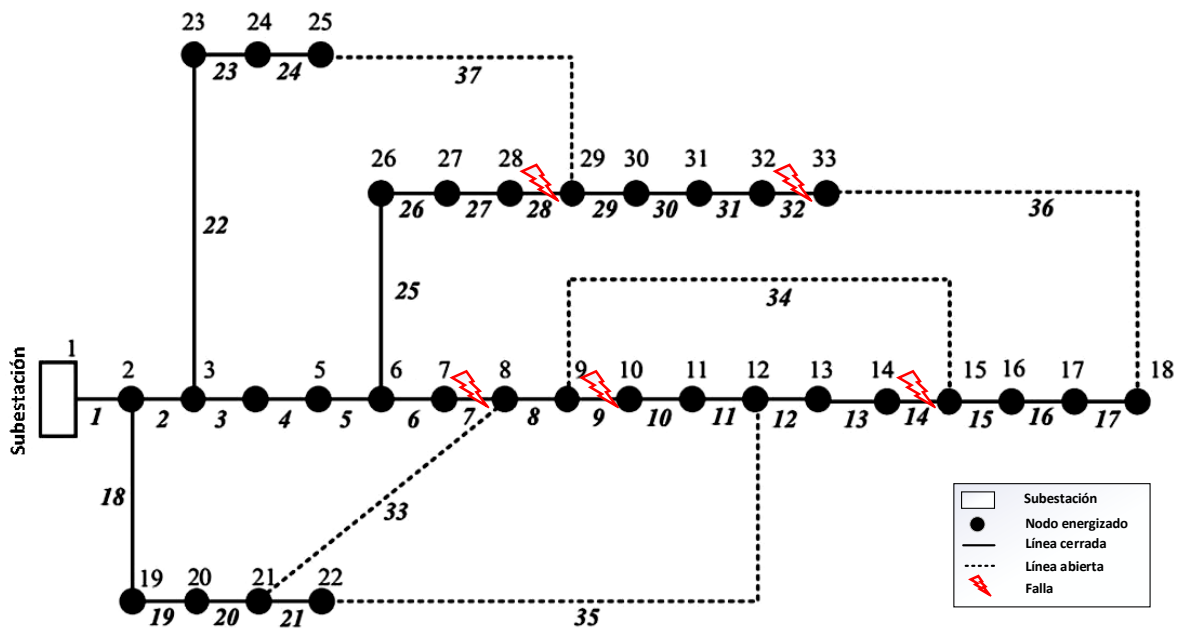


Figura 4.2 Localización de falla en líneas 7, 9, 14, 28 y 32 en el sistema de prueba estándar.

4.3.1.2 Localización de falla en líneas 3, 4 y disparo en línea 9

El caso de estudio se plantea en [37] y [47]. Se asume una falla localizada en las líneas 3 y 4. Además, el interruptor de la línea 9 dispara. Los nodos 4 a 18 y 26 a 33 quedan fuera de servicio tras la falla. Con este escenario se pretende evaluar el desempeño del algoritmo de SR al ocurrir una falla cercana a la cabecera del

alimentador que involucra la pérdida del servicio de gran parte del sistema. La solución resultante se compara con los resultados presentados en [37] y [47]. En la Figura 4.3 se escenifica la falla en el sistema, los símbolos en forma de rayo representan la localización de falla.

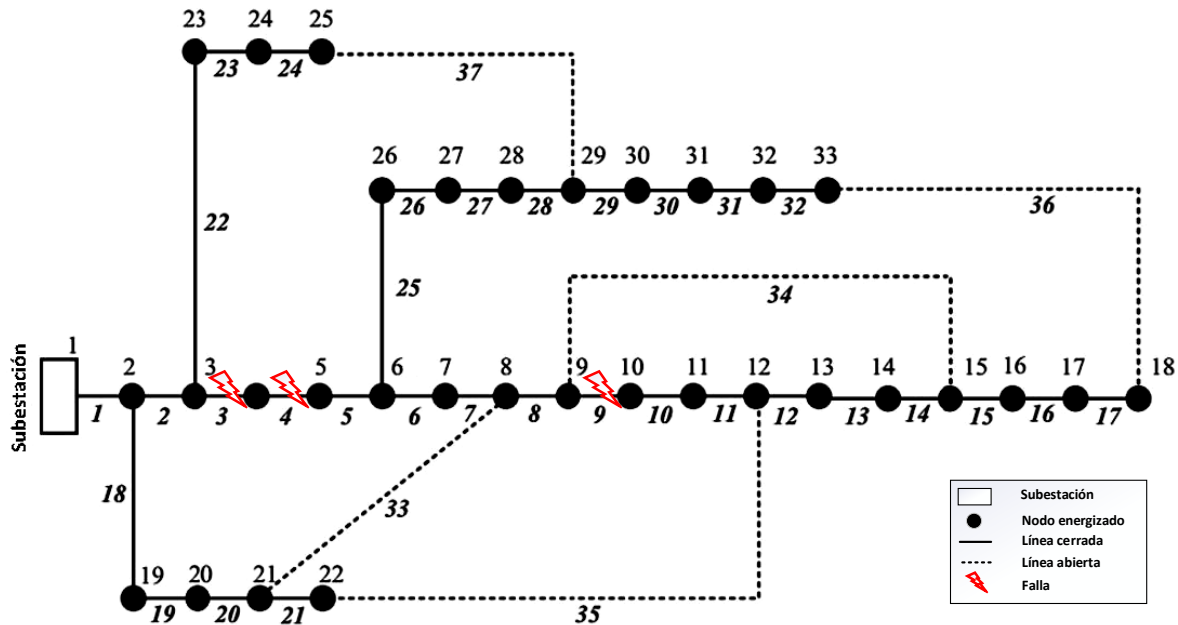


Figura 4.3 Localización de falla en líneas 3, 4 y disparo en línea 9 en el sistema de prueba estándar.

4.3.2 Caso 2: Integración de unidades DG para la restauración

El sistema de prueba estándar es sometido a ciertos escenarios en los que no es posible restaurar el servicio a partir de la reconfiguración de red. Para lograr una solución, el algoritmo considera la integración de las unidades DG existentes del sistema. Para este caso, se considera la incorporación de 4 unidades DG instaladas y localizadas en los nodos 6, 12, 25 y 32, respectivamente. La localización y capacidad de las unidades DG en el sistema es fija y se obtiene de [46], datos que se resumen en la Tabla 4.1. Las unidades DG generan solamente potencia activa, por lo que se asume la integración de unidades de Energía Renovable (ej. energía eólica, solar). A continuación, se plantea el escenario.

Tabla 4.1 Localización y capacidad de unidades DG añadidas al sistema estándar [47].

No. de unidades DG	Localización (nodo)	Capacidad DG (MW)
DG 1	6	1.6725
DG 2	12	0.3798
DG 3	25	0.6255
DG 4	32	0.6560

4.3.2.1 Localización de falla en líneas 4, 28 y 29 con integración DG

Se asume una falla simultánea en distintas localizaciones del sistema. Para despejar las fallas, los interruptores de las líneas 4, 28 y 29 operan, dejando fuera de servicio a los nodos 5 a 18 y 26 a 33. Con este escenario se pretende evaluar el desempeño del algoritmo de SR al considerar la reconfiguración de red con la integración de las unidades DG del sistema. En la Figura 4.4 se representa la falla simultánea en el sistema, los símbolos en forma de rayo escenifican la localización de falla y los círculos con la letra G las unidades DG.

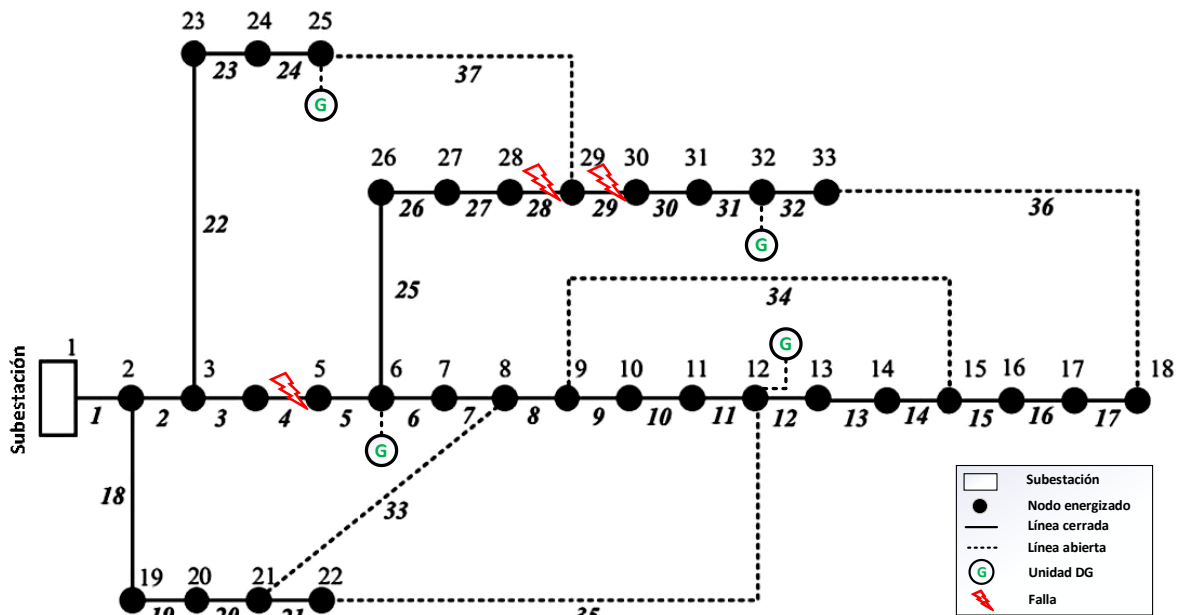


Figura 4.4 Localización de falla en líneas 4, 28 y 29 en el sistema de prueba estándar con la integración de unidades DG.

4.3.3 Caso 3: Consideración de clientes prioritarios para la restauración

Se somete al sistema de prueba estándar a determinados escenarios en los que no es posible restaurar toda el área fuera de servicio a partir de los elementos disponibles (conmutación de interruptores de línea y despacho de unidades DG). Para lograr restablecer parcialmente el servicio, el algoritmo efectúa el deslastre de carga según la prioridad de los clientes del sistema. El nivel de criticidad de las distintas cargas se asigna en concordancia con la localización de las unidades DG, clasificación que se presenta en la Tabla 4.2. En función a este factor y a la demanda de potencia, se calcula el peso de prioridad de los clientes (Ver Sección 3.2.3). A continuación, se exponen los escenarios.

Tabla 4.2 Niveles de criticidad de carga asignados en el sistema de prueba estándar.

Carga	Nivel de criticidad	L_{critic}^i
Nodo 2	Alto	3
Nodo 3	Bajo	1
Nodo 4	Bajo	1
Nodo 5	Bajo	1
Nodo 6	Medio	2
Nodo 7	Bajo	1
Nodo 8	Bajo	1
Nodo 9	Bajo	1
Nodo 10	Bajo	1
Nodo 11	Alto	3
Nodo 12	Medio	2
Nodo 13	Bajo	1
Nodo 14	Bajo	1
Nodo 15	Bajo	1
Nodo 16	Bajo	1
Nodo 17	Bajo	1
Nodo 18	Medio	2
Nodo 19	Bajo	1
Nodo 20	Bajo	1
Nodo 21	Bajo	1
Nodo 22	Medio	2
Nodo 23	Bajo	1
Nodo 24	Medio	2
Nodo 25	Medio	2
Nodo 26	Bajo	1
Nodo 27	Bajo	1
Nodo 28	Bajo	1
Nodo 28	Bajo	1
Nodo 29	Bajo	1
Nodo 30	Bajo	1
Nodo 31	Bajo	1
Nodo 32	Medio	2
Nodo 33	Alto	3

4.3.3.1 Localización de falla en líneas 4, 28 y 29

Se asume la falla del caso 2. Para este escenario no se considera la integración de las unidades DG en el sistema. Los interruptores de líneas operados y los nodos fuera de servicio son los expuestos anteriormente. Con este escenario se pretende evaluar el desempeño del algoritmo de SR al considerar la reconfiguración de red con el deslastre de carga según la prioridad de los clientes. En la Figura 4.5 se escenifica la falla simultánea en el sistema, los símbolos en forma de rayo representan la localización de falla.

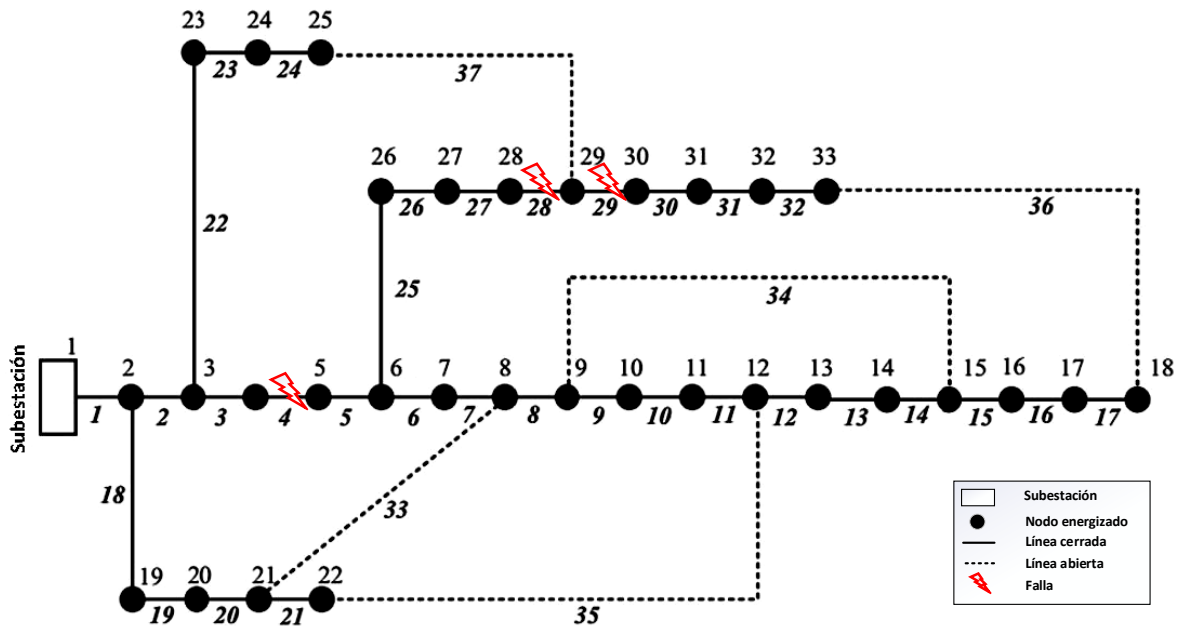


Figura 4.5 Localización de falla en líneas 4, 28 y 29 en el sistema de prueba estándar.

4.3.3.2 Localización falla en líneas 4, 28 y 29 con adición DG parcial

Se considera la falla del caso 2. Para este escenario se incluye la integración de las unidades DG 2 a 4 en el sistema, mientras la unidad DG 1 se asume que no está disponible. Los interruptores de líneas operados y los nodos fuera de servicio son los expuestos en el caso 2. Con este escenario se pretende evaluar el desempeño del algoritmo de SR al considerar la reconfiguración de red con la integración de las unidades DG del sistema y el deslastre de carga según la prioridad de los clientes. En la Figura 4.6 se representa la falla simultánea en el sistema, los símbolos en forma de rayo escenifican la localización de falla y los círculos con la letra G las unidades DG.

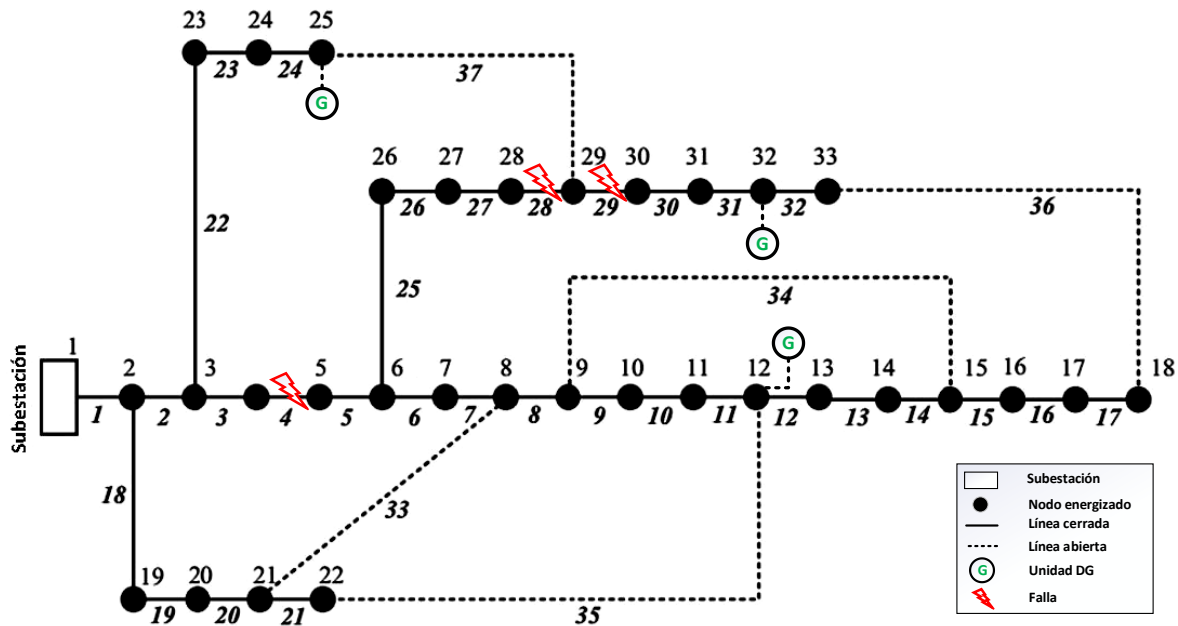


Figura 4.6 Localización de falla en líneas 4, 28 y 29 en el sistema de prueba estándar con la integración de unidades DG 2 a 4.

4.4 Resultados de la simulación

El algoritmo metaheurístico DE-PBILc aplicado para la restauración del servicio se implementó en MATLAB R2018a (versión de prueba gratuita) y fue computado en una plataforma hardware (computador) con las siguientes características: Intel Core i7-11800H, 2.3 GHz, 16 GB RAM. Se utilizó la herramienta gratuita MATPOWER V7.1 para resolver la formulación del flujo de potencia AC óptimo.

Para los casos de estudio, se consideró una población N de 60 individuos, un máximo de 100 iteraciones y un número de 10 ensayos conforme [48]. Mientras el valor de convergencia (precisión deseada) se tomó como 0.0001 de acuerdo a [46].

Los resultados proporcionados por el algoritmo de SR se presentan y discuten en las siguientes subsecciones. Conforme [34], para propósitos de simplificación, las maniobras de apertura ejecutadas para aislar la sección en falla no se consideran en los cálculos de la función objetivo. Para evaluar el desempeño se consideró los criterios de: tasa de restauración, éxito de respuesta, desviación estándar de iteraciones, promedio de iteraciones y tiempo de cálculo.

4.4.1 Resultados para Caso 1

4.4.1.1 Plan SR - Falla en líneas 7, 9, 14, 28 y 32

Se ejecutó la simulación en reiteradas ocasiones, obteniéndose el mismo plan de maniobras de restauración (topología de red con el costo más bajo). En la Tabla 4.3 se proporciona el resultado del algoritmo de SR de una simulación. Al ocurrir falla múltiple en las líneas 7, 9, 14, 28 y 32 del sistema de prueba estándar, el algoritmo determina que para restaurar el servicio se debe cerrar los interruptores de las líneas 33, 34, 35, 36 y 37. Esta topología de red involucra la conmutación de 5 interruptores, genera 139.98 kW de pérdidas de potencia y tiene un costo de 7.80 U\$ (conforme a costos de Tabla 3.2) para restaurar el servicio en las zonas desenergizadas aguas abajo de la localización de falla.

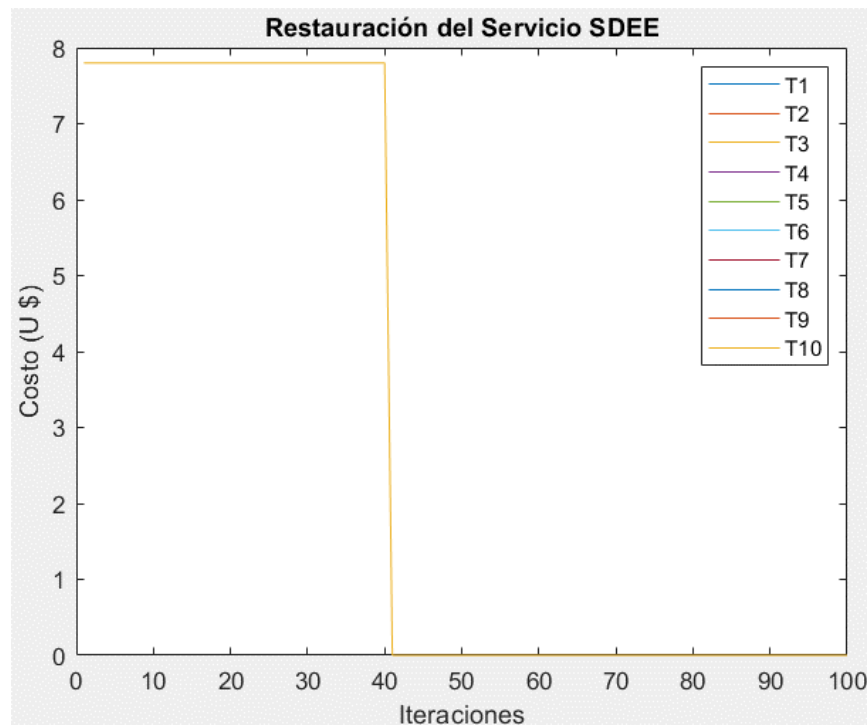


Figura 4.7 Proceso de convergencia del algoritmo de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.

La Figura 4.7 representa la gráfica del proceso de convergencia de la simulación. Se observa una única solución mediante reconfiguración de red. Para los distintos ensayos, la gráfica parte desde el costo de restauración (7.80 U\$) en la iteración 1 hasta que completa un determinado número de iteraciones (40) y luego sale, resultando un escalón. En cada ensayo el proceso iterativo se detiene en un valor similar dado que se satisface el tercer criterio de parada (Ver Sección 3.4 –

apartado: *Criterios de parada*), es decir, que el resultado de las últimas x iteraciones coincide, siendo x mayor que el número de mejores individuos ($x > N_{best}$).

Tabla 4.3 Plan de restauración del algoritmo de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.

Ítem	Resultado
Costo de restauración (U\$)	7.80
Zonas fuera de servicio (z)	0
Carga fuera de servicio (kW)	0.0
Pérdidas de potencia (kW)	139.98
Potencia de unidades DG (kW)	0.0
Número de interruptores a operar	5
Apertura de interruptor de línea	-
Cierre de interruptores de líneas	33 (21-8), 34 (9-15), 35 (12-22), 36 (18-33), 37 (25-29)
Número de unidades DG a operar	0

El desempeño del algoritmo de SR se evalúa en atención a los criterios expuestos. Los resultados se muestran en la Tabla 4.4.

- **Tasa de restauración:** Para cada ensayo, se considera acierto si la topología de red obtenida como mejor solución permite operar al sistema en condiciones adecuadas (sin penalizaciones de no radialidad o no convergencia) para restaurar el servicio. La simulación tiene una tasa de 1.
- **Éxito de respuesta:** El porcentaje de ensayos para los que se obtiene la solución idónea como respuesta es del 100 %.
- **Desviación estándar de iteraciones:** Se tiene un valor de 0, que corresponde a una baja variabilidad de convergencia (desviación estándar menor que doce, $\sigma < 12$) [34], es decir, que, para cada ensayo, el algoritmo converge hacia la mejor solución usando un número similar de iteraciones.
- **Iteraciones promedio:** El número promedio de iteraciones requeridas en cada ensayo hasta que el algoritmo converge a la mejor solución es de 40.
- **Tiempo de cálculo:** El tiempo empleado por el algoritmo para efectuar cada ensayo, registrar la mejor solución (local), precisar la solución idónea (global) y mostrarla como el plan de maniobras de restauración es de 16.53 s.

Tabla 4.4 Desempeño del algoritmo de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.

Criterio	Desempeño
Tasa de restauración	1.0
Éxito de respuesta (%)	100
Desviación estándar de iteraciones	0.0
Iteraciones promedio	40.0

Evaluación promedio de función objetivo	65.0
Tiempo de cálculo (s)	16.53

La respuesta del algoritmo de SR se compara con los resultados de [46]: El algoritmo GA presenta un tiempo de cálculo de 36 s y un promedio de iteraciones de 130. Mientras el algoritmo PSO presenta un tiempo de cálculo de 16.2 s y un promedio de iteraciones de 88. Los algoritmos proporcionan la misma secuencia de conmutación de los interruptores de línea: 7, 9, 14, 28 y 32 en estado abierto y el resto en estado cerrado. El promedio de iteraciones del algoritmo de SR es inferior, el tiempo de cálculo es menor al algoritmo GA y cercano al algoritmo PSO. No se menciona en [46] el tamaño de población, el número máximo de iteraciones y los ensayos que ocuparon para obtener los resultados.

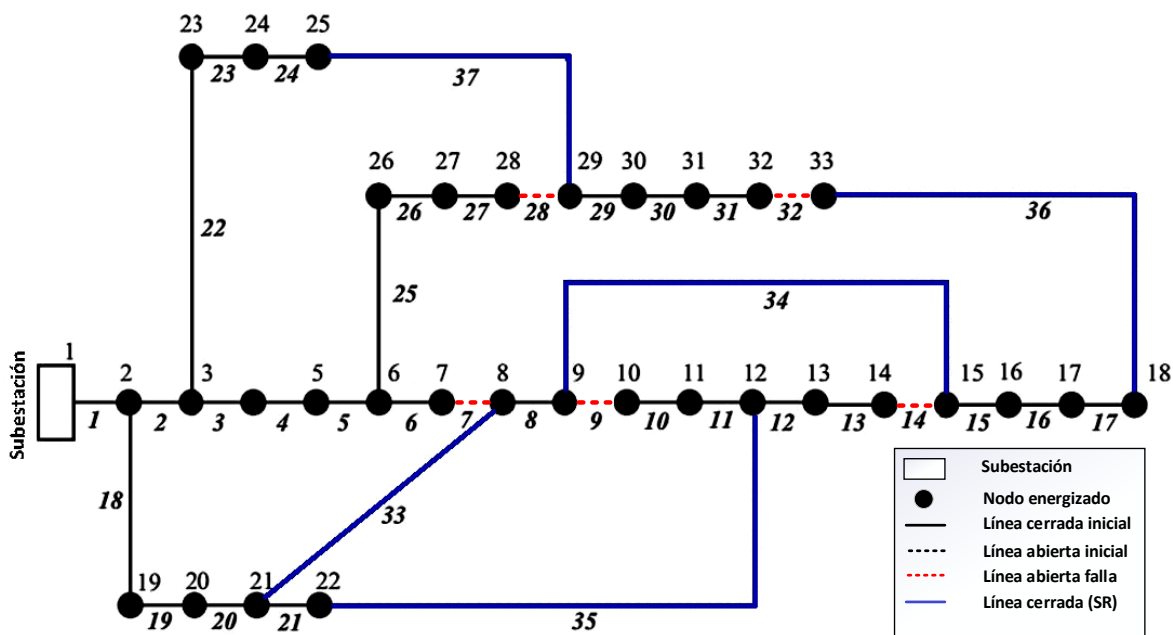


Figura 4.8 Topología de red resultante del algoritmo de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.

La Figura 4.8 escenifica la topología de red resultante para la restauración del servicio. La falla múltiple analizada en este caso de estudio no implica mayor esfuerzo computacional para el algoritmo de SR, que se ve reflejado en un reducido tiempo de cálculo. El servicio se restaura a todas las zonas desenergizadas al cerrar los interruptores de línea disponibles del sistema (excepto aquellos que aíslan la falla). Este escenario permite validar la solución resultante al compararla con los resultados de [42].

4.4.1.2 Plan SR - Falla en líneas 3, 4 y disparo en línea 9

Se corrió la simulación en reiteradas ocasiones, obteniéndose el mismo plan de maniobras de restauración (solución idónea). En la Tabla 4.5 se proporciona el resultado del algoritmo de SR de una simulación. La zona en falla contiene al nodo 4 por lo que queda fuera de servicio. Las maniobras de apertura que aíslan la sección en falla no fueron consideradas en los cálculos del algoritmo. El plan de restauración, que proporciona la menor cantidad de interruptores a conmutar y genera menos pérdidas de potencia, conlleva cerrar los interruptores de las líneas 35 y 37. Esta topología de red implica la conmutación de 2 interruptores, genera 221.80 kW de pérdidas de potencia y tiene un costo de 6.44 U\$ (conforme a costos de Tabla 3.2) para restaurar el servicio en las zonas desenergizadas aguas abajo de la localización de falla. Queda fuera de servicio una zona correspondiente al nodo 4 con demanda de 120 kW no suministrados.

Tabla 4.5 Plan de restauración del algoritmo de SR para Caso 1 – Falla en líneas 3, 4 y disparo en línea 9.

Ítem	Resultado
Costo de restauración (U\$)	6.44
Zonas fuera de servicio (z)	1
Carga fuera de servicio (kW)	120.0
Pérdidas de potencia (kW)	221.80
Potencia de unidades DG (kW)	0.0
Número de interruptores a operar	2
Apertura de interruptor de línea	-
Cierre de interruptores de líneas	35 (12-22), 37 (25-29)
Número de unidades DG a operar	0

La gráfica del proceso de convergencia de la simulación se presenta en la Figura 4.9. Se observa que, para ciertos ensayos, la gráfica parte desde un costo de restauración alto (topología con penalización) en la iteración 1. Luego, en el lazo de evolución del enjambre se disminuye los costos (topologías sin penalización) para las siguientes iteraciones hasta que se alcanza cierto valor, donde queda fijo. El proceso iterativo del algoritmo de SR en cada ensayo se detiene en un determinado valor de iteraciones (≥ 40) dado que se satisface el tercer criterio de parada (Ver Sección 3.4 – apartado: *Criterios de parada*).



(a)



(b)

Figura 4.9 Proceso de convergencia del algoritmo de SR para Caso 1 – Falla en líneas 3, 4 y disparo en 9. (a) Gráfica en vista original. (b) Gráfica aumentada.

El desempeño del algoritmo de SR se evalúa en atención a los criterios expuestos. Los resultados se muestran en la Tabla 4.6.

Tabla 4.6 Desempeño del algoritmo de SR para Caso 1 – Falla en líneas 3, 4 y disparo en línea 9.

Criterio	Desempeño
Tasa de restauración	1.0
Éxito de respuesta (%)	100
Desviación estándar de iteraciones	3.33
Iteraciones promedio	42.20
Evaluación promedio de función objetivo	384.40
Tiempo de cálculo (s)	30.67

La respuesta del algoritmo de SR se compara con los resultados de [37] y [47]. El algoritmo GA-ACO de [37] posee un tiempo de cálculo de 49.673 s. Mientras el algoritmo CP-ACO de [47] presenta un tiempo de cálculo de 41.314 s. Los algoritmos proporcionan el mismo número de interruptores de línea a conmutar (2), pertenecientes al grupo de elementos de transferencia. El tiempo de cálculo que emplea el algoritmo de SR disminuye aproximadamente en un 38.25 % y 25.75 %, respecto a los trabajos mencionados.

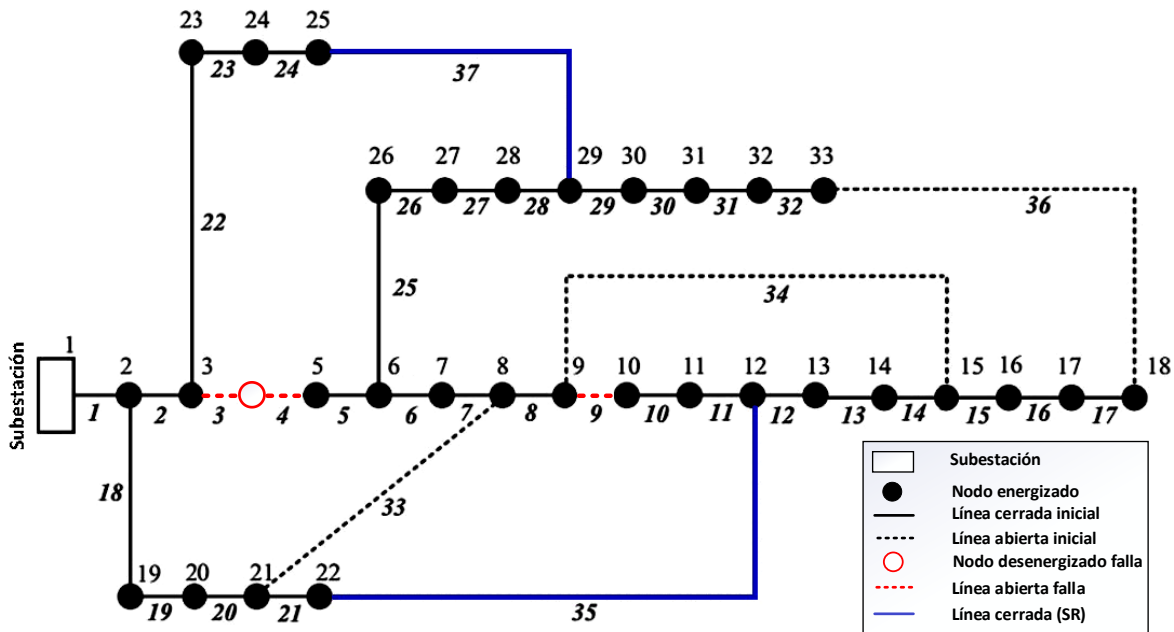


Figura 4.10 Topología de red resultante del algoritmo de SR para Caso 1 – Falla en líneas 3, 4 y disparo en línea 9.

La Figura 4.10 representa la topología de red resultante para la restauración del servicio. La falla analizada en este caso de estudio implica cierto grado de dificultad

para el algoritmo de SR (mayor tiempo de cálculo), debido a la falla (línea 3) cercana a la cabecera del alimentador, que provoca la pérdida del servicio de gran parte del sistema. El plan de maniobras para restaurar las zonas desenergizadas (excepto la sección en falla) se obtiene al cerrar los interruptores de las líneas 35 y 37. Este escenario permite validar la solución al compararla con los resultados de [37] y [47].

4.4.2 Resultados para Caso 2

4.4.2.1 Plan SR - Falla en líneas 4, 28 y 29 con integración DG

La simulación se ejecutó en reiteradas ocasiones obteniéndose el mismo plan de maniobras de restauración (solución idónea). En la Tabla 4.7 se proporciona el resultado del algoritmo de SR de una simulación. Al ocurrir falla simultánea en las líneas 4, 28 y 29 del sistema de prueba estándar, el algoritmo determina que el plan de maniobras de restauración, que proporciona la menor cantidad de interruptores a conmutar, involucra el menor despacho de unidades DG y genera menos pérdidas de potencia, conlleva: Cerrar los interruptores de las líneas 33, 34, 36, 37; abrir el interruptor de la línea 14 (permite que el sistema opere en configuración radialidad) y operar las unidades DG ubicadas en los nodos 6 y 32, respectivamente. Esta topología de red implica la conmutación de 5 interruptores, el despacho de 2 unidades DG que aportan 2328.50 kW, genera 154.59 kW de pérdidas de potencia y tiene un costo de 240.94 U\$ (conforme a costos de Tabla 3.2) para restaurar el servicio en las zonas desenergizadas aguas abajo de la localización de falla.

Tabla 4.7 Plan de restauración del algoritmo de SR para Caso 2 – Falla en líneas 4, 28 y 29 con integración DG.

Ítem	Resultado
Costo de restauración (U\$)	240.94
Zonas fuera de servicio (z)	0
Carga fuera de servicio (kW)	0.0
Pérdidas de potencia (kW)	154.59
Potencia de unidades DG (kW)	2328.50
Número de interruptores a operar	5
Apertura de interruptor de línea	14 (14-15)
Cierre de interruptores de líneas	33 (21-8), 34 (9-15), 36 (18-33), 37 (25-29)
Número de unidades DG a operar	2
Operación de unidades DG	1 (nodo 6), 4 (nodo 32)

La Figura 4.11 muestra el proceso de convergencia de la simulación. Se observa que, para varios ensayos, la gráfica parte desde un costo de restauración elevado en la iteración 1. Luego, en el lazo de evolución del enjambre se disminuye los costos (topologías sin penalización) para las siguientes iteraciones hasta que se

alcanza cierto valor, donde queda fijo. En cada ensayo el proceso iterativo se detiene en un determinado valor de iteraciones (≥ 40) dado que se satisface el tercer criterio de parada (Ver Sección 3.4 – apartado: *Criterios de parada*).



Figura 4.11 Proceso de convergencia del algoritmo de SR para Caso 2 – Falla en líneas 4, 28 y 29 con integración DG.

El desempeño del algoritmo de SR se evalúa en atención a los criterios expuestos. Los resultados se muestran en la Tabla 4.8.

Tabla 4.8 Desempeño del algoritmo de SR para Caso 2 - Falla en líneas 4, 28 y 29 con integración DG.

Criterio	Desempeño
Tasa de restauración	1.0
Éxito de respuesta (%)	100
Desviación estándar de iteraciones	3.79
Iteraciones promedio	41.20
Evaluación promedio de función objetivo	584.30
Tiempo de cálculo (s)	34.59

La Figura 4.12 escenifica la topología de red resultante para la restauración del servicio. La falla simultánea analizada en este caso de estudio implica mayor grado de dificultad para el algoritmo de SR (mayor tiempo de cálculo), al ocurrir una falla cercana a la cabecera del alimentador y otra intermedia, que provoca la pérdida del servicio de gran parte del sistema. Para restaurar todas las zonas desenergizadas

es necesario la reconfiguración de red con la integración de las unidades DG del sistema. El plan de maniobras de restauración con el menor costo se obtiene al abrir el interruptor de la línea 14, cerrar los interruptores de las líneas 33, 34, 36, 37 y despachar las unidades DG de los nodos 6 y 32, respectivamente. No se ha comparado la calidad de respuesta del algoritmo, ya que en los trabajos investigativos publicados que usan este sistema como red de prueba no se ha evaluado este caso de estudio.

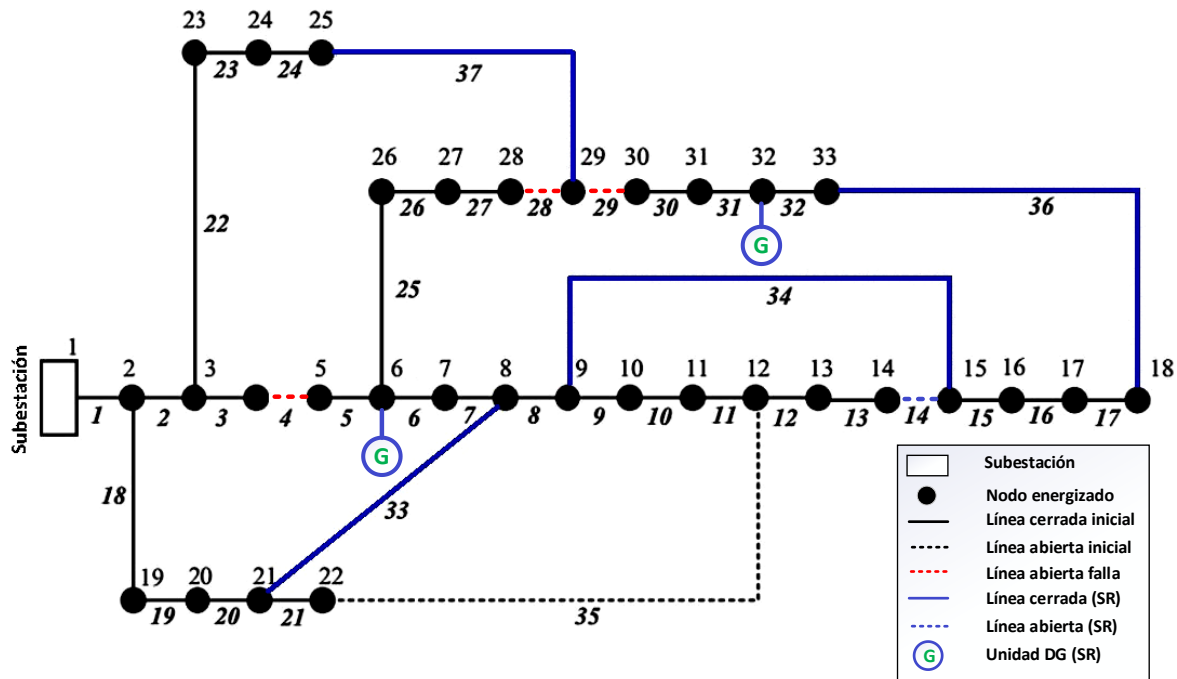


Figura 4.12 Topología de red resultante del algoritmo de SR para Caso 2 - Falla en líneas 4, 28 y 29 con integración DG.

4.4.3 Resultados para Caso 3

4.4.3.1 Plan SR - Falla en líneas 4, 28 y 29

Se corrió la simulación en reiteradas ocasiones, obteniéndose el mismo plan de maniobras de restauración (solución idónea). En la Tabla 4.9 se presenta el resultado del algoritmo de SR de una simulación. Al ocurrir falla simultánea en las líneas 4, 28 y 29 y no disponer de las unidades DG, el sistema no posee la capacidad para restaurar todas las zonas fuera de servicio aguas abajo de la localización de falla con los elementos disponibles. Para lograr restablecer parcialmente el servicio, el algoritmo de SR efectúa el deslastre de carga según la prioridad de los clientes del sistema. La lista de deslastre que establece comprende los clientes de los nodos 16 a 18 y 30 a 33. Luego del proceso de simulación, el algoritmo determina que el plan de maniobras de restauración que proporciona la

menor cantidad de interruptores a conmutar, involucra el menor deslastre de carga y genera menos pérdidas de potencia, conlleva: Cerrar los interruptores de las líneas 33, 35, 36, 37, y abrir los interruptores de las líneas 11 (permite la operación del sistema en configuración radialidad) y 31 (efectúa el deslastre de carga). Esta topología de red implica la conmutación de 6 interruptores, genera 140.86 kW de pérdidas de potencia y tiene un costo de 1058.82 U\$ (conforme a costos de Tabla 3.2) para restaurar el servicio a la mayor cantidad de zonas desenergizadas aguas abajo de la localización de falla. Queda fuera de servicio 2 zonas correspondientes a los nodos 30 y 31 con demanda de 350 kW no suministrados. Los clientes desenergizados poseen el nivel de criticidad bajo y, por ende, un menor peso de prioridad.



Figura 4.13 Proceso de convergencia del algoritmo de SR para Caso 3 – Falla en líneas 4, 28 y 29.

La gráfica del proceso de convergencia de la simulación se muestra en la Figura 4.13. Se observa que, para ciertos ensayos, la gráfica parte desde un costo de restauración superior en la iteración 1. Luego, en el lazo de evolución del enjambre se disminuye los costos (topologías sin penalización) para las siguientes iteraciones hasta que se alcanza cierto valor, donde queda fijo. En cada ensayo el proceso iterativo se detiene en un determinado valor de iteraciones (> 40) dado que se satisface el tercer criterio de parada (Ver Sección 3.4 – apartado: *Criterios de parada*).

Tabla 4.9 Plan de restauración del algoritmo de SR para Caso 3 – Falla en líneas 4, 28 y 29.

Ítem	Resultado
Costo de restauración (U\$)	1058.82
Zonas fuera de servicio (z)	2
Carga fuera de servicio (kW)	350.0
Pérdidas de potencia (kW)	140.86
Potencia de unidades DG (kW)	0.0
Número de interruptores a operar	6
Apertura de interruptor de línea	11 (11-12), 31(31-32)
Cierre de interruptores de líneas	33 (21-8), 35 (12-22), 36 (18-33), 37 (25-29)
Número de unidades DG a operar	0

El desempeño del algoritmo de SR se evalúa en atención a los criterios expuestos. Los resultados se muestran en la Tabla 4.10.

Tabla 4.10 Desempeño del algoritmo para Caso 3 - Falla en líneas 4, 28 y 29.

Criterio	Desempeño
Tasa de restauración	1.0
Éxito de respuesta (%)	100
Desviación estándar de iteraciones	2.83
Iteraciones promedio	41.30
Evaluación promedio de función objetivo	305.90
Tiempo de cálculo (s)	30.58

La Figura 4.14 representa la topología de red resultante para la restauración del servicio. La falla simultánea analizada en este caso de estudio implica mayor grado de dificultad para el algoritmo de SR (mayor tiempo de cálculo), al ocurrir una falla cercana a la cabecera del alimentador y otra intermedia, que deja al sistema sin la capacidad para restaurar el servicio a todas las zonas desenergizadas aguas abajo de la localización de falla. Para restablecer parcialmente el servicio, es necesario la reconfiguración de red con el deslastre de carga según la prioridad de los clientes del sistema. El plan de maniobras de restauración con el menor costo se obtiene al abrir los interruptores de las líneas 11 y 31, y cerrar los interruptores de las líneas 33, 35, 36, 37. Queda fuera de servicio 2 zonas correspondientes a los nodos 30 y 31. En los trabajos investigativos publicados de restauración del servicio que emplean este sistema como red de prueba no se ha evaluado este caso de estudio.

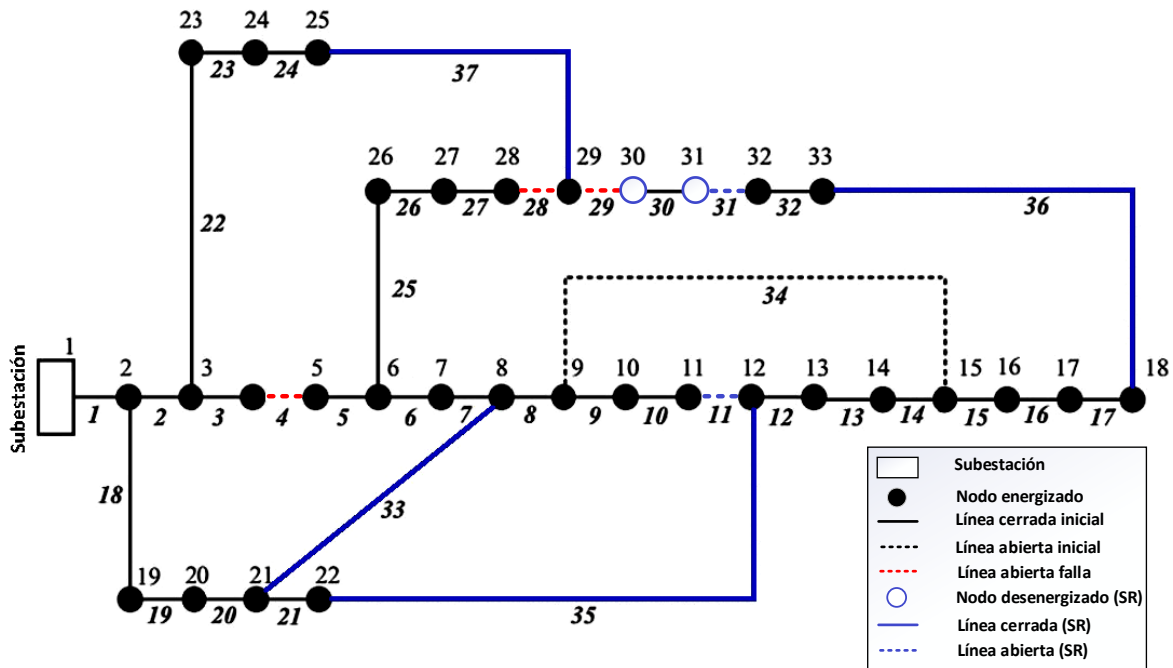


Figura 4.14 Topología de red resultante del algoritmo de SR para Caso 3 - Falla en líneas 4, 28 y 29.

4.4.3.2 Plan SR - Falla en líneas 4, 28 y 29 con adición DG parcial

Se ejecutó la simulación en reiteradas ocasiones, obteniéndose el mismo plan de maniobras de restauración (solución idónea). En la Tabla 4.11 se presenta el resultado del algoritmo de SR de una simulación. Al ocurrir falla simultánea en las líneas 4, 28 y 29 y no contar con todas las unidades DG (unidad DG 1 no disponible), el sistema no posee la capacidad para la restauración completa de las zonas fuera de servicio aguas abajo de la localización de falla con los elementos disponibles. Para lograr restablecer parcialmente el servicio, el algoritmo de SR efectúa el deslastre de carga según la prioridad de los clientes del sistema. La lista de deslastre que establece comprende los clientes de los nodos 30 a 32. Luego del proceso de simulación, el algoritmo determina que el plan de maniobras de restauración, que proporciona la menor cantidad de interruptores a conmutar, involucra el menor despacho de unidades DG, ejecuta el menor deslastre de carga y genera menos pérdidas de potencia, conlleva: Cerrar los interruptores de las líneas 33, 36, 37; abrir el interruptor de la línea 30 (efectúa el deslastre de carga) y operar la unidad DG ubicada en el nodo 32. Esta topología de red implica la conmutación de 4 interruptores, el despacho de 1 unidad DG que aporta 656 kW, genera 113.86 kW de pérdidas de potencia y tiene un costo de 671.88 U\$ (conforme a costos de Tabla 3.2) para restaurar el servicio a la mayor cantidad de zonas desenergizadas aguas abajo de la localización de falla. Queda fuera de servicio una

zona correspondiente al nodo 30 con demanda de 200 kW no suministrados. El cliente desenergizado posee el nivel de criticidad bajo y, por ende, un menor peso de prioridad.

Tabla 4.11 Plan de restauración del algoritmo para Caso 3 – Falla en líneas 4, 28 y 29 con integración DG parcial.

Ítem	Resultado
Costo de restauración (U\$)	671.88
Zonas fuera de servicio (z)	1
Carga fuera de servicio (kW)	200.0
Pérdidas de potencia (kW)	113.86
Potencia de unidades DG (kW)	656.0
Número de interruptores a operar	4
Apertura de interruptor de línea	30 (30-31)
Cierre de interruptores de líneas	33 (21-8), 36 (18-33), 37 (25-29)
Número de unidades DG a operar	1
Operación de unidades DG	4 (nodo 32)

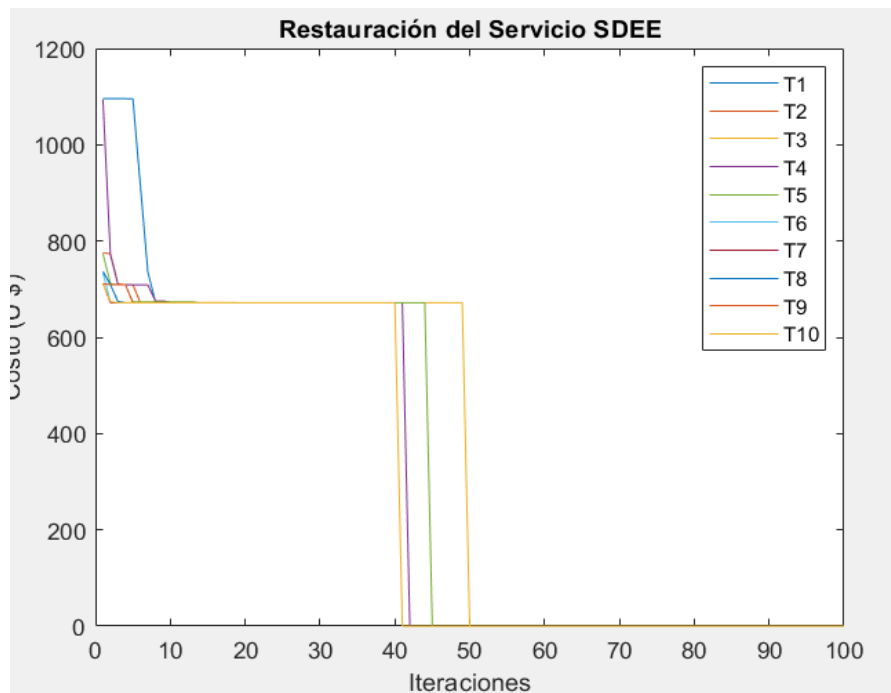


Figura 4.15 Proceso de convergencia del algoritmo de SR para Caso 3 – Falla en líneas 4, 28 y 29 con integración DG parcial.

La Figura 4.15 muestra el proceso de convergencia de la simulación. Se observa que, para varios ensayos, la gráfica parte desde un costo de restauración superior en la iteración 1. Luego, en el lazo de evolución del enjambre se disminuye los

costos (topologías sin penalización) para las siguientes iteraciones hasta que se alcanza cierto valor, donde queda fijo. En cada ensayo el proceso iterativo se detiene en un determinado valor de iteraciones (> 40) dado que se satisface el tercer criterio de parada (Ver Sección 3.4 – apartado: *Criterios de parada*).

El desempeño del algoritmo de SR se evalúa en atención a los criterios expuestos. Los resultados se muestran en la Tabla 4.12.

Tabla 4.12 Desempeño del algoritmo de SR para Caso 3 - Falla en líneas 4, 28 y 29 con integración DG parcial.

Criterio	Desempeño
Tasa de restauración	1.0
Éxito de respuesta (%)	100
Desviación estándar de iteraciones	2.92
Iteraciones promedio	41.50
Evaluación promedio de función objetivo	481.0
Tiempo de cálculo (s)	32.87

La Figura 4.16 representa la topología de red resultante para la restauración del servicio. La falla simultánea analizada en este caso de estudio implica mayor grado de dificultad para el algoritmo de SR (mayor tiempo de cálculo), al ocurrir una falla cercana a la cabecera del alimentador y otra intermedia, que deja al sistema sin la capacidad para restaurar el servicio a todas las zonas desenergizadas aguas abajo de la localización de falla. Para restablecer parcialmente el servicio, es necesario la reconfiguración de red con la integración de unidad DG y el deslastre de carga según la prioridad de los clientes del sistema. El plan de maniobras de restauración con el menor costo se obtiene al abrir el interruptor de la línea 30, cerrar los interruptores de las líneas 33, 36, 37, y despachar la unidad DG del nodo 32. Queda fuera de servicio una zona correspondiente al nodo 30. Respecto a los resultados del caso anterior, se observa que la integración de unidades DG en el sistema de distribución disminuye la cantidad requerida de operaciones de interruptores, las zonas desenergizadas por el deslastre de carga y las pérdidas de potencia en la red, que se reflejan en una reducción del costo de restauración. En los trabajos investigativos publicados de restauración del servicio que ocupan este sistema como red de prueba no se ha evaluado este caso de estudio.

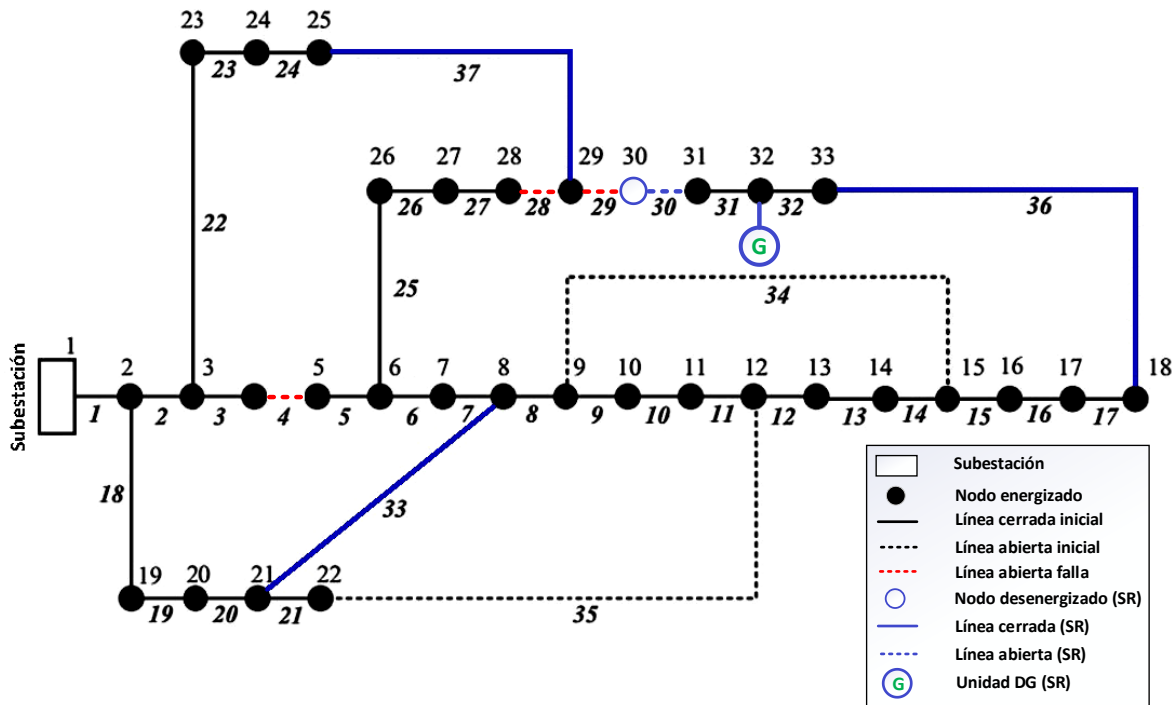


Figura 4.16 Topología de red resultante del algoritmo de SR para Caso 3 - Falla en líneas 4, 28 y 29 con integración DG parcial.

4.5 Comparación de desempeño del algoritmo de SR

En la sección anterior se presentó los resultados del algoritmo de SR para los casos de estudio evaluados. La información existente en la literatura permitió verificar el plan de restauración y un par de criterios de desempeño de las soluciones de los escenarios del Caso 1; sin embargo, el resto de criterios de desempeño y resultados de los otros Casos no fueron contrastados. Con el propósito de comparar el desempeño y comprobar las respuestas, se utilizó las técnicas metaheurísticas DE y PBILc por separado en el algoritmo de SR. Los resultados proporcionados por estos algoritmos se presentan y discuten en las siguientes subsecciones.

4.5.1 Comparación de resultados para Caso 1

4.5.1.1 Plan SR - Falla en líneas 7, 9, 14, 28 y 32

Se ejecutó la simulación en reiteradas ocasiones, obteniéndose el mismo plan de restauración mostrado en la Tabla 4.3. La Figura 4.17 representa la gráfica del proceso de convergencia de la simulación de los algoritmos, que es igual a la obtenida en la Figura 4.7.



(a)



(b)

Figura 4.17 Proceso de convergencia de los algoritmos de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32. (a) Técnica DE. (b) Técnica PBILc.

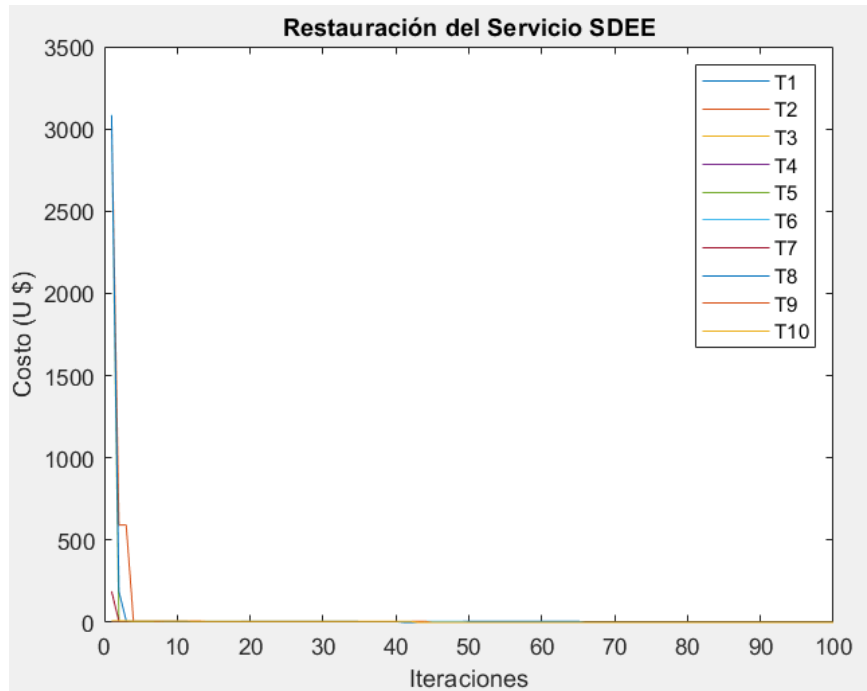
El desempeño de los algoritmos se evalúa en atención a los criterios expuestos. Los resultados se muestran en la Tabla 4.13. Se incluyó los valores proporcionados en la Tabla 4.4 para facilitar la didáctica de comparación. Se observa que las 3 técnicas metaheurísticas permiten al algoritmo que alcance la solución idónea para todos los ensayos (éxito de respuesta de 100 %). El número promedio de iteraciones requeridas en cada ensayo para converger hacia la solución es el mismo (40), por lo que no existe variabilidad de convergencia (desviación estándar de cero). El promedio de evaluaciones de función objetivo difiere, siendo superior con la técnica PBILc (84.90), que se ve reflejado en un mayor tiempo de cálculo (17.59 s). Para este caso sencillo de falla, las 3 técnicas presentan desempeños adecuados, siendo el algoritmo con técnica híbrida DE-PBILc el de mejor desempeño.

Tabla 4.13 Desempeño de los algoritmos de SR para Caso 1 – Falla en líneas 7, 9, 14, 28 y 32.

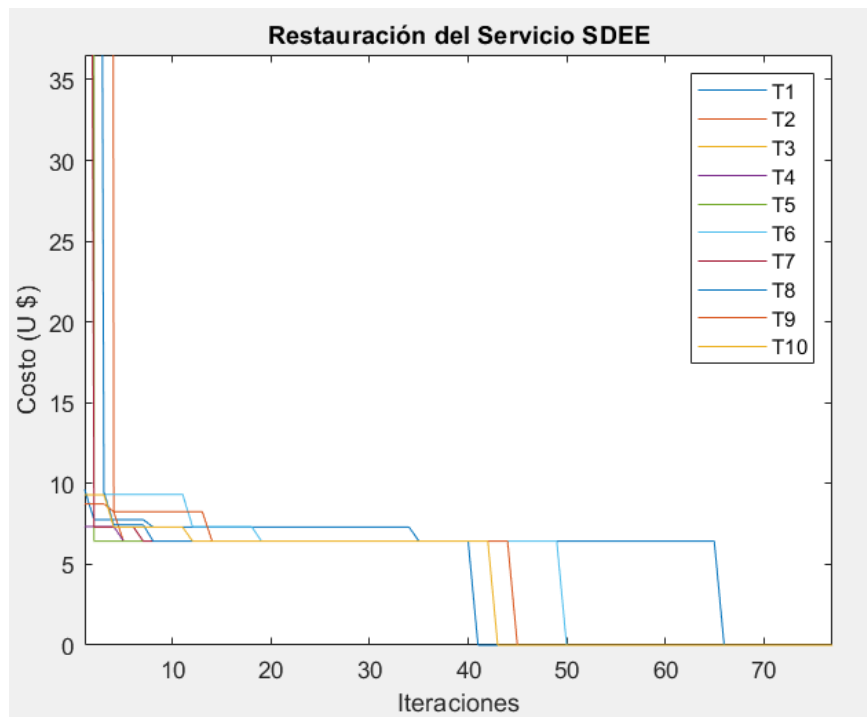
Criterio	Desempeño		
	DE-PBILc	DE	PBILc
Tasa de restauración	1.0	1.0	1.0
Éxito de respuesta (%)	100	100	100
Desviación estándar de iteraciones	0.0	0.0	0.0
Iteraciones promedio	40.0	40.0	40.0
Evaluación promedio de función objetivo	65.0	62.60	84.90
Tiempo de cálculo (s)	16.53	16.64	17.59

4.5.1.2 Plan SR - Falla en líneas 3, 4, y disparo en línea 9

Se corrió la simulación en reiteradas ocasiones, obteniéndose el mismo plan de restauración de la Tabla 4.5. Las Figuras 4.18 y 4.19 representan las gráficas del proceso de convergencia de la simulación de los algoritmos. El algoritmo con técnica DE presenta un proceso de convergencia semejante al obtenido en la Figura 4.9, cada ensayo converge en un valor similar de costo; sin embargo, en algunos ensayos se ocupa más iteraciones. El algoritmo con técnica PBILc tiene un proceso de convergencia un tanto disperso, se observa que para ciertos ensayos no se logra el valor idóneo; además que, se emplea un mayor número de iteraciones.

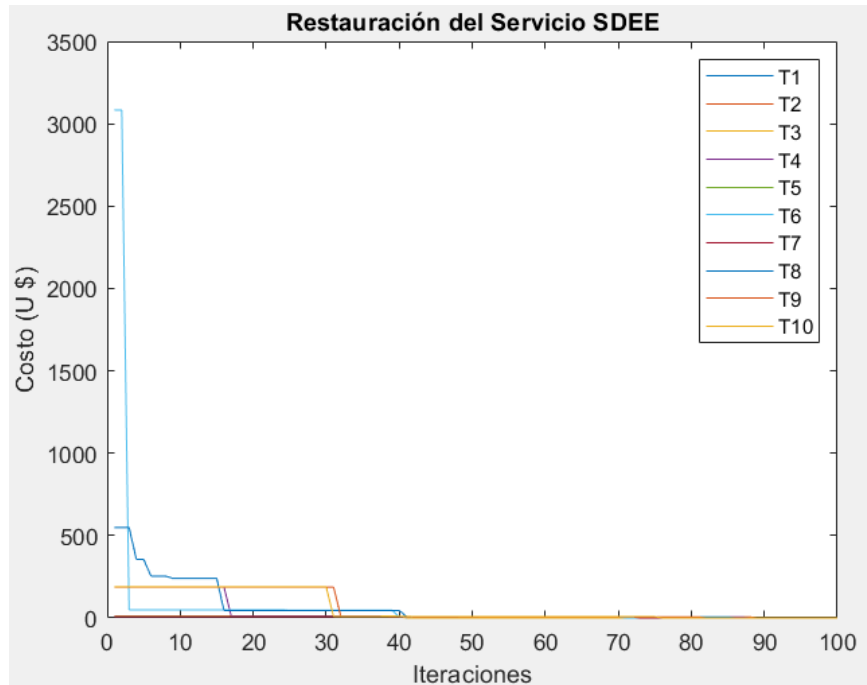


(a)



(b)

Figura 4.18 Proceso de convergencia del algoritmo con DE para Caso 1 – Falla en líneas 3, 4 y disparo en 9. (a) Gráfica en vista original. (b) Gráfica aumentada.



(a)



(b)

Figura 4.19 Proceso de convergencia del algoritmo con PBILc para Caso 1 – Falla en líneas 3, 4 y disparo en 9. (a) Gráfica en vista original. (b) Gráfica aumentada.

Se evalúa el desempeño de los algoritmos conforme a los criterios expuestos. Los resultados se muestran en la Tabla 4.14. Se incluyó los valores de la Tabla 4.6 para facilitar la didáctica de comparación. Se observa que las 2 primeras técnicas metaheurísticas permiten al algoritmo que alcance la solución idónea para todos los ensayos (éxito de respuesta de 100 %), mientras su porcentaje disminuye con la tercera; sin embargo, se encuentra dentro del límite aceptable (éxito de respuesta de 70 %). El número promedio de iteraciones requeridas en cada ensayo para converger hacia la solución es menor con la técnica híbrida (42.2) y aumenta con las otras técnicas. Además, existe una baja variabilidad de convergencia (desviación estándar menor que doce, $\sigma < 12$) con las dos primeras técnicas; pero crece de forma considerable con la última. El promedio de evaluaciones de función objetivo difiere, siendo superior con la técnica PBILc (484.10), que se ve reflejado en un mayor tiempo de cálculo (44.09 s). Para este caso de falla, las 2 primeras técnicas presentan desempeños adecuados, mientras la tercera se encuentra en el límite aceptable. Tras la comparativa, se determina que el algoritmo con técnica híbrida DE-PBILc presenta el mejor desempeño.

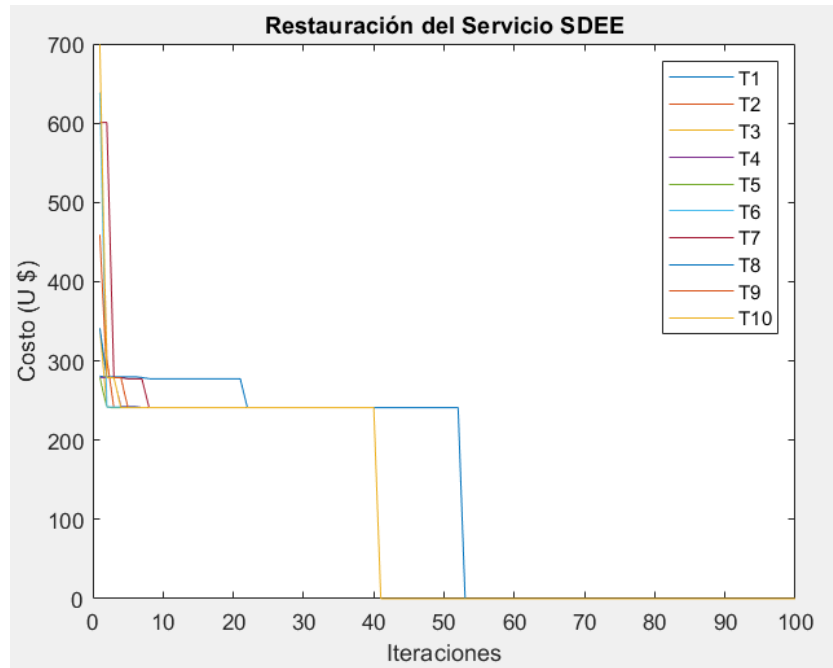
Tabla 4.14 Desempeño de los algoritmos de SR para Caso 1 – Falla en líneas 3, 4 y disparo en línea 9.

Criterio	Desempeño		
	DE-PBILc	DE	PBILc
Tasa de restauración	1.0	1.0	1.0
Éxito de respuesta (%)	100	100	70
Desviación estándar de iteraciones	3.33	7.93	17.58
Iteraciones promedio	42.20	44.0	70.80
Evaluación promedio de función objetivo	384.40	409.50	484.10
Tiempo de cálculo (s)	30.67	32.29	44.09

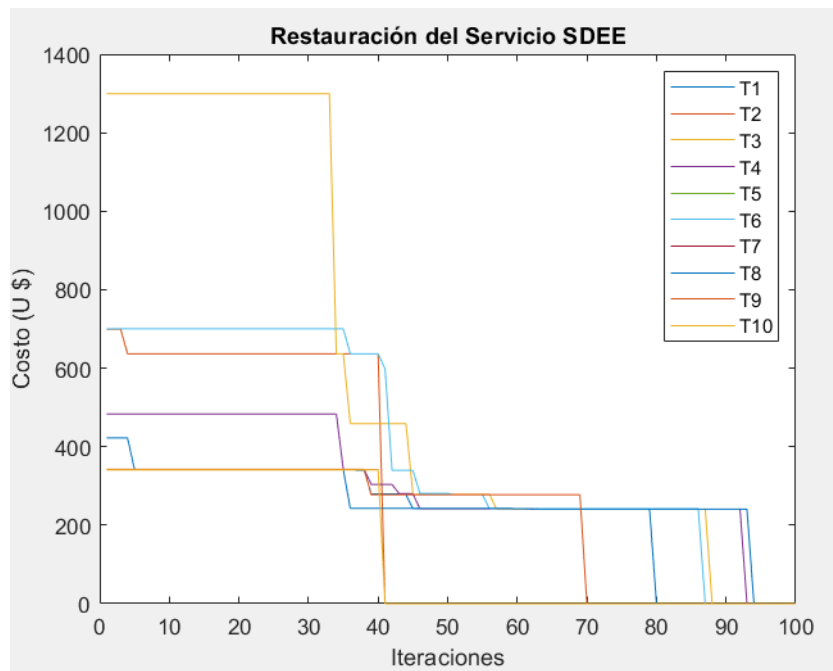
4.5.2 Comparación de resultados para Caso 2

4.5.2.1 Plan SR - Falla en líneas 4, 28 y 29 con integración DG

La simulación fue ejecutada en reiteradas ocasiones, obteniéndose el mismo plan de restauración de la Tabla 4.7. La Figura 4.20 muestra la gráfica del proceso de convergencia de la simulación de los algoritmos. El algoritmo con técnica DE tiene un proceso de convergencia semejante al obtenido en la Figura 4.11, cada ensayo converge en un valor similar de costo; sin embargo, en ciertos ensayos se ocupa más iteraciones. El algoritmo con técnica PBILc presenta un proceso de convergencia disperso, se observa que para varios ensayos no se logra el valor idóneo; además que, se emplea un mayor número de iteraciones.



(a)



(b)

Figura 4.20 Proceso de convergencia de los algoritmos de SR para Caso 2 – Falla en líneas 4, 28 y 29 con integración DG. (a) Técnica DE. (b) Técnica PBILc.

El desempeño de los algoritmos se evalúa en atención a los criterios expuestos. Los resultados se presentan en la Tabla 4.15. Se incluyó los valores de la Tabla 4.8 para facilitar el proceso de comparación. Se observa que la primera técnica permite al algoritmo que alcance la solución idónea para todos los ensayos (éxito de respuesta de 100 %), su porcentaje disminuye con la segunda técnica, y se encuentra fuera del límite aceptable con la tercera (éxito de respuesta menor de 70 %). El número promedio de iteraciones requeridas en cada ensayo para converger hacia la solución es menor con la técnica híbrida (41.2) y crece con las otras técnicas. Además, existe una baja variabilidad de convergencia ($\sigma < 12$) con las dos primeras técnicas; pero aumenta de forma considerable con la última. El promedio de evaluaciones de función objetivo varía, siendo inferior con la técnica híbrida (584.30), que se ve reflejado en un menor tiempo de cálculo (34.59 s). Para este caso de falla, la primera técnica presenta un desempeño adecuado, la segunda se encuentra en el límite aceptable, mientras la tercera tiene un bajo desempeño.

Tabla 4.15 Desempeño de los algoritmos de SR para Caso 2 – Falla en líneas 4, 28 y 29 con integración DG.

Criterio	Desempeño		
	DE-PBILc	DE	PBILc
Tasa de restauración	1.0	1.0	1.0
Éxito de respuesta (%)	100	80	30
Desviación estándar de iteraciones	3.79	5.06	23.86
Iteraciones promedio	41.20	41.60	66.60
Evaluación promedio de función objetivo	584.30	661.90	472.10
Tiempo de cálculo (s)	34.59	36.85	41.55

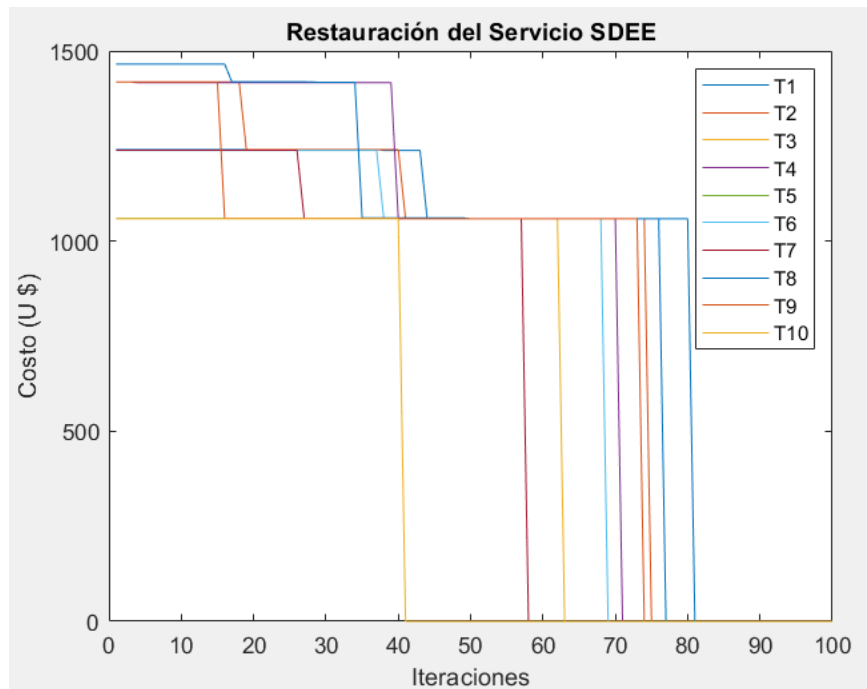
4.5.3 Comparación de resultados para Caso 3

4.5.3.1 Plan SR - Falla en líneas 4, 28 y 29

Se corrió la simulación en reiteradas ocasiones, obteniéndose el mismo plan de restauración de la Tabla 4.9. La Figura 4.21 muestra la gráfica del proceso de convergencia de la simulación de los algoritmos. El algoritmo con técnica DE presenta un proceso de convergencia semejante al obtenido en la Figura 4.13, cada ensayo converge en un valor similar de costo. El algoritmo con técnica PBILc presenta un proceso de convergencia un tanto disperso; sin embargo, se visualiza que los ensayos convergen en un valor similar, aunque se emplea un mayor número de iteraciones.



(a)



(b)

Figura 4.21 Proceso de convergencia de los algoritmos de SR para Caso 3 – Falla en líneas 4, 28 y 29. (a) Técnica DE. (b) Técnica PBILc.

Se evalúa el desempeño de los algoritmos de acuerdo a los criterios expuestos. Los resultados se muestran en la Tabla 4.16. Se incluyó los valores de la Tabla 4.10 para facilitar el proceso de comparación. Se observa que la primera técnica permite al algoritmo que alcance la solución idónea para todos los ensayos (éxito de respuesta de 100 %), su porcentaje disminuye con las siguientes; sin embargo, se encuentra dentro del límite aceptable (éxito de respuesta mayor o igual a 70 %). El número promedio de iteraciones requeridas en cada ensayo para converger hacia la solución es similar con las dos primeras técnicas y se incrementa con la última. Además, existe una baja variabilidad de convergencia ($\sigma < 12$) con las dos primeras técnicas; pero supera este umbral con la última. El promedio de evaluaciones de función objetivo difiere, siendo inferior con la técnica híbrida (305.90), que se ve reflejado en un menor tiempo de cálculo (30.58 s). Para este caso de falla, el algoritmo con técnica híbrida DE-PBILc presenta un desempeño adecuado, mientras los restantes se encuentran dentro del límite aceptable.

Tabla 4.16 Desempeño de los algoritmos de SR para Caso 3 – Falla en líneas 4, 28 y 29.

Criterio	Desempeño		
	DE-PBILc	DE	PBILc
Tasa de restauración	1.0	1.0	1.0
Éxito de respuesta (%)	100	80	70
Desviación estándar de iteraciones	2.83	3.16	14.29
Iteraciones promedio	41.30	41.0	64.0
Evaluación promedio de función objetivo	305.90	316.60	400.20
Tiempo de cálculo (s)	30.58	31.61	43.39

4.5.3.2 Plan SR - Falla en líneas 4, 28 y 29 con adición DG parcial

La simulación se ejecutó en reiteradas ocasiones, obteniéndose el mismo plan de restauración de la Tabla 4.11. La Figura 4.22 representa la gráfica del proceso de convergencia de la simulación de los algoritmos. El algoritmo con técnica DE presenta un proceso de convergencia semejante al obtenido en la Figura 4.15, cada ensayo converge en un valor similar de costo. El algoritmo con técnica PBILc presenta un proceso de convergencia un tanto disperso; sin embargo, se visualiza que los ensayos convergen en un valor similar, aunque se emplea un mayor número de iteraciones.



(a)



(b)

Figura 4.22 Proceso de convergencia de los algoritmos de SR para Caso 3 – Falla en líneas 4, 28 y 29 con integración DG parcial. (a) Técnica DE. (b) Técnica PBILc.

El desempeño de los algoritmos se evalúa en atención a los criterios expuestos. Los resultados se presentan en la Tabla 4.17. Se incluyó los valores de la Tabla 4.12 para facilitar la didáctica de comparación. Se observa que las 2 primeras técnicas permiten al algoritmo que alcance la solución idónea para todos los ensayos (éxito de respuesta de 100 %), mientras que con la tercera su porcentaje disminuye; no obstante, se encuentra dentro del límite aceptable (éxito de respuesta mayor o igual a 70 %). El número promedio de iteraciones requeridas en cada ensayo para converger hacia la solución es cercano con las dos primeras técnicas y se incrementa con la última. Además, existe baja variabilidad de convergencia con las tres técnicas ($\sigma < 12$); aunque con la última se tiene un valor próximo al umbral. El promedio de evaluaciones de función objetivo varía, siendo inferior con la técnica PBILc (408.7); sin embargo, con la técnica híbrida se consigue el menor tiempo de cálculo (32.87 s). Para este caso de falla, las 2 primeras técnicas presentan desempeños adecuados, mientras la tercera se encuentra en el límite aceptable. Tras la comparativa, se determina que el algoritmo con técnica híbrida DE-PBILc presenta el mejor desempeño.

Tabla 4.17 Desempeño de los algoritmos de SR para Caso 3 – Falla en líneas 4, 28 y 29 con integración DG parcial.

Criterio	Desempeño		
	DE-PBILc	DE	PBILc
Tasa de restauración	1.0	1.0	1.0
Éxito de respuesta (%)	100	100	80
Desviación estándar de iteraciones	2.92	1.08	11.33
Iteraciones promedio	41.50	40.50	70.20
Evaluación promedio de función objetivo	481.0	510.90	408.70
Tiempo de cálculo (s)	32.87	34.26	39.78

4.6 Extensión de análisis de desempeño del algoritmo de SR

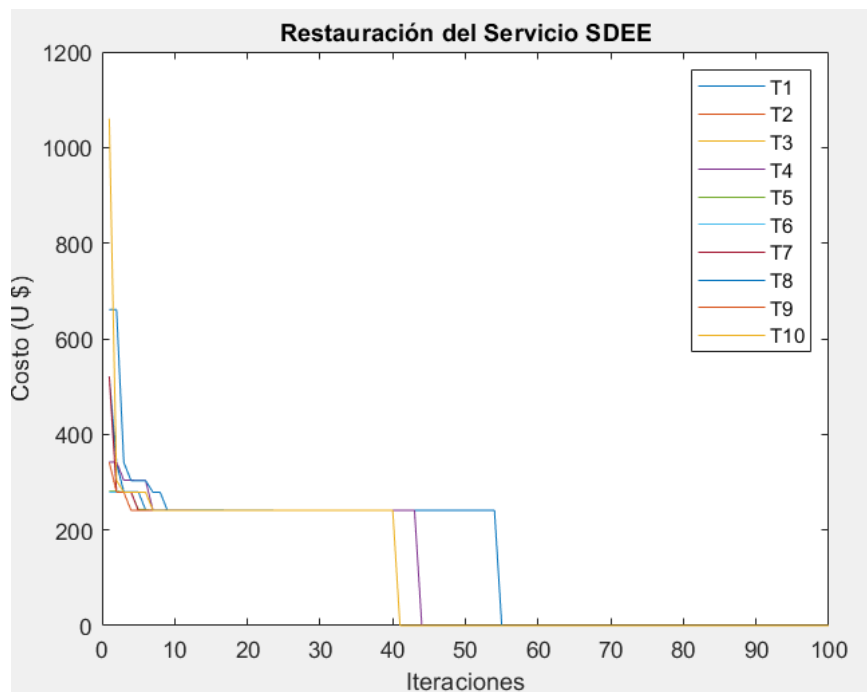
En este apartado se evalúa el desempeño del algoritmo de SR con técnica híbrida DE-PBILc para un par de situaciones adicionales, a fin de constatar su impacto en la calidad de respuesta. Para este análisis se eligió el segundo caso de estudio. En las siguientes subsecciones se exponen las situaciones y se discuten los resultados.

4.6.1 Influencia del procesamiento en paralelo

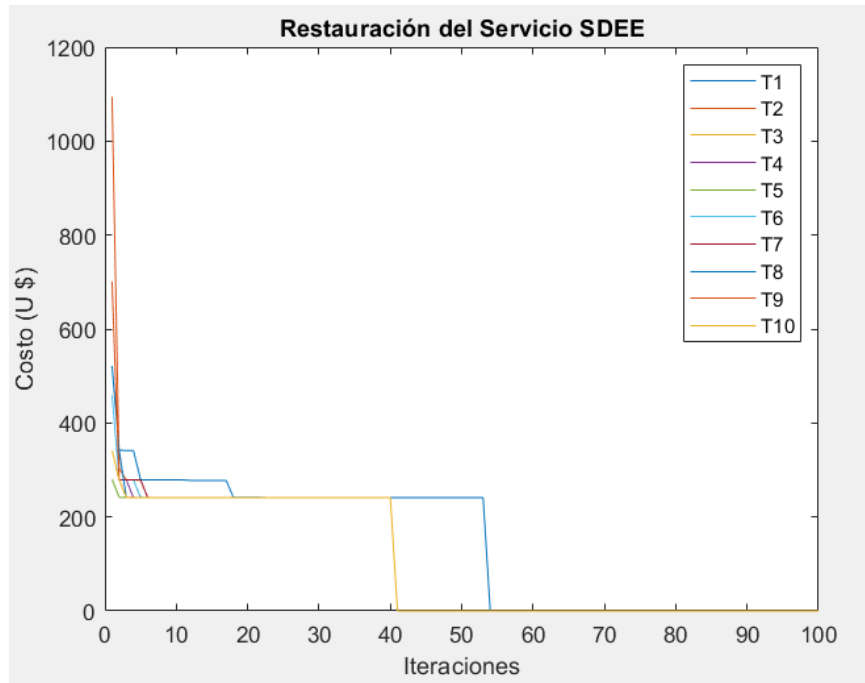
Cada individuo (topología) de la población (inicial/descendiente) de las metaheurísticas se evalúa para calcular el valor de la función objetivo (costo de restauración). Se puede utilizar el procesamiento en paralelo (multinúcleos) para

resolver problemas operativos simultáneos. En este trabajo, se usa la herramienta Parallel Computing Toolbox de MATLAB para ejecutar en paralelo los cálculos de evaluación de la población entre los recursos disponibles. Cada núcleo (worker) de la herramienta calcula un determinado número de flujos de potencia AC óptimos, lo que acelera todo el proceso de evaluación de la función objetivo [35]. Los resultados de la Sección 4.4 se obtuvieron al emplear en paralelo los 8 núcleos del procesador. En esta subsección se analiza los resultados del algoritmo de SR al variar en un número par los núcleos desde 2 hasta 8, utilizados en el procesamiento en paralelo.

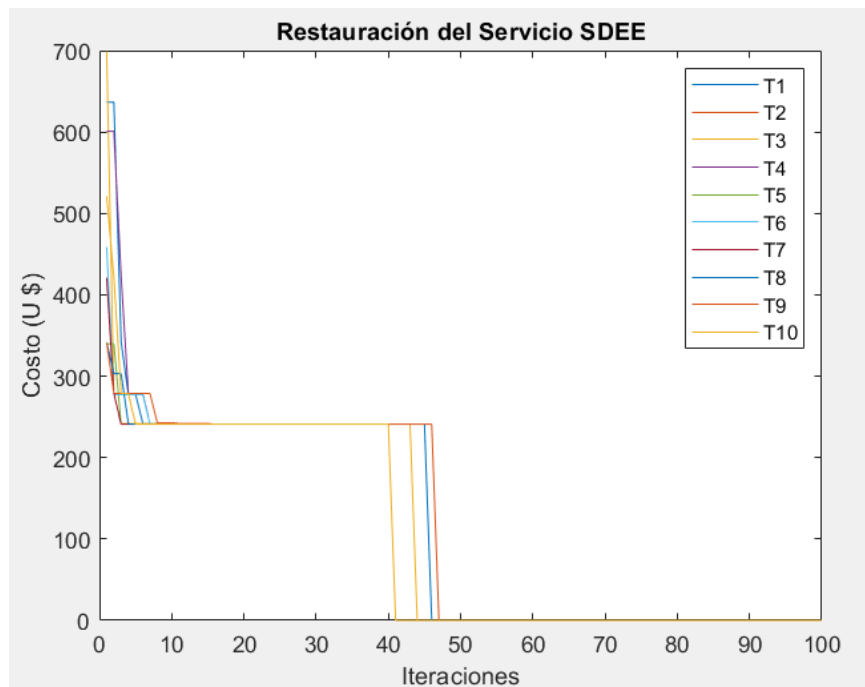
Las simulaciones se ejecutaron en reiteradas ocasiones, obteniéndose el mismo plan de restauración de la Tabla 4.7. La Figura 4.23 muestra la gráfica del proceso de convergencia de la simulación al utilizarse en paralelo 2, 4 y 6 núcleos del procesador, respectivamente. Se observa un proceso de convergencia semejante al obtenido en la Figura 4.11, cada ensayo converge en un valor similar de costo.



(a)



(b)



(c)

Figura 4.23 Proceso de convergencia del algoritmo de SR para Caso 2 al variar los núcleos del procesamiento en paralelo. (a) 2 núcleos. (b) 4 núcleos. (c) 6 núcleos.

El desempeño del algoritmo al modificarse el número de núcleos del procesamiento en paralelo se evalúa en atención a los criterios expuestos. Los resultados se presentan en la Tabla 4.18. Se incluyó los valores de la Tabla 4.8 para facilitar el proceso de análisis. No se observa variaciones relevantes en los valores de los criterios; excepto, en el tiempo de cálculo, que es menor cuando la simulación emplea en paralelo los 8 núcleos del procesador, existiendo una reducción de aproximadamente el 57.21 %, 21.16 % y 7.60 % con respecto al uso en paralelo de 2, 4 y 6 núcleos.

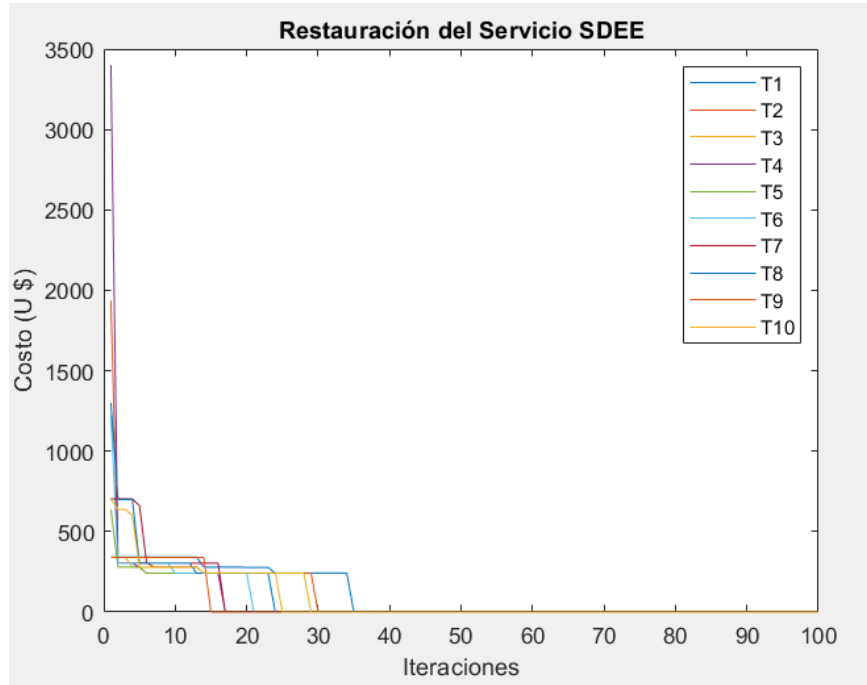
Tabla 4.18 Desempeño del algoritmo de SR para Caso 2 al variar los núcleos del procesamiento en paralelo.

Criterio	Desempeño			
	2 núcleos	4 núcleos	6 núcleos	8 núcleos
Tasa de restauración	1.0	1.0	1.0	1.0
Éxito de respuesta (%)	100	100	100	100
Desviación estándar de iteraciones	4.42	4.11	2.37	3.79
Iteraciones promedio	41.70	41.30	41.40	41.20
Evaluación promedio de función objetivo	639.50	602.50	626.30	584.30
Tiempo de cálculo (s)	54.38	41.91	37.22	34.59

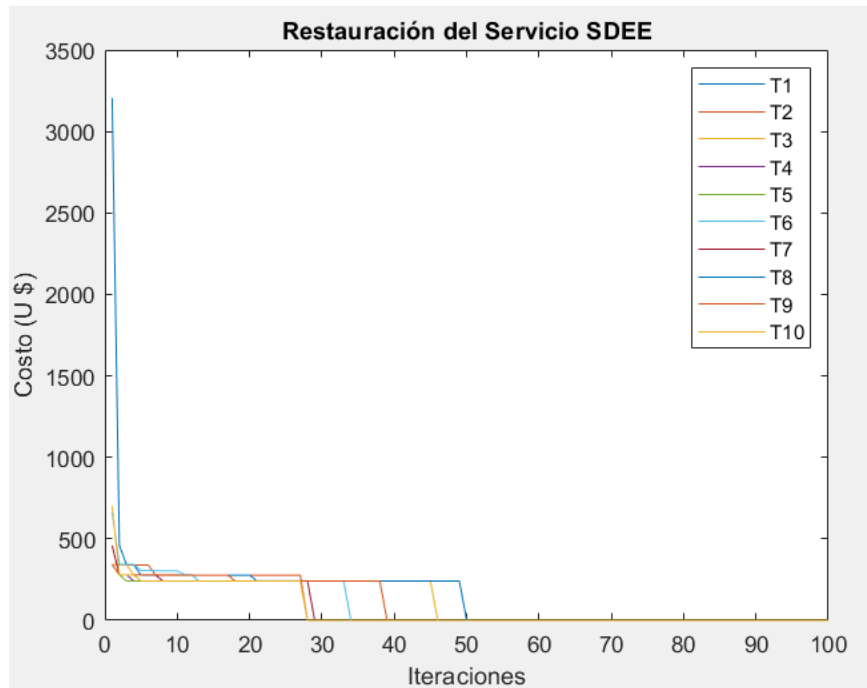
4.6.2 Impacto del tamaño de la población

Los resultados de la Sección 4.4 se obtuvieron con una población N de 60 individuos. En esta subsección se analiza los resultados del algoritmo de SR al variar el número de individuos de 20 en 20 hasta los 80.

Las simulaciones se ejecutaron en reiteradas ocasiones, obteniéndose el mismo plan de restauración de la Tabla 4.7. La Figura 4.24 muestra la gráfica del proceso de convergencia de la simulación al emplearse 20, 40 y 80 individuos, respectivamente. Con una población de 80 individuos, se tiene un proceso de convergencia semejante al obtenido en la Figura 4.11, cada ensayo converge en un valor similar de costo; sin embargo, se ocupa más iteraciones. Con una población de 40 individuos, el algoritmo presenta un proceso de convergencia un tanto disperso, se observa que para algunos ensayos no se logra el valor idóneo. Mientras que, con 20 individuos, se tiene un proceso de convergencia disperso, se visualiza que para varios ensayos no se alcanza el valor idóneo. Sin embargo, para estos dos últimos escenarios, se emplea un menor número de iteraciones.



(a)



(b)



(c)

Figura 4.24 Proceso de convergencia del algoritmo de SR para Caso 2 al variar el tamaño de la población. (a) 20 individuos. (b) 40 individuos . (c) 80 individuos.

Tabla 4.19 Desempeño del algoritmo de SR para Caso 2 al variar el tamaño de la población.

Criterio	Desempeño			
	20 individuos	40 individuos	60 individuos	80 individuos
Tasa de restauración	1.0	1.0	1.0	1.0
Éxito de respuesta (%)	30	80	100	100
Desviación estándar de iteraciones	6.41	8.36	3.79	2.85
Iteraciones promedio	22.70	32.80	41.20	54.90
Evaluación promedio de función objetivo	131.40	355.10	584.30	992.60
Tiempo de cálculo (s)	13.42	23.96	34.59	54.19

El desempeño del algoritmo al modificarse el tamaño de la población se evalúa en atención a los criterios expuestos. Los resultados se presentan en la Tabla 4.19. Se consideró los valores de la Tabla 4.8 para facilitar el proceso de análisis. Se observa que, una población de 60 y 80 individuos permite al algoritmo que alcance la solución idónea para todos los ensayos (éxito de respuesta de 100 %), su porcentaje disminuye cuando se emplea 40 individuos y se encuentra fuera del límite aceptable con 20 (éxito de respuesta menor de 70 %). El número promedio de iteraciones

requeridas en cada ensayo para converger hacia la solución es menor cuando se emplea 20 individuos (22.70) y crece conforme incrementa el tamaño de la población. Además, existe una baja variabilidad de convergencia ($\sigma < 12$) cuando se tiene un mayor número de individuos y aumenta cuando se reduce esta cantidad. El promedio de evaluaciones de función objetivo varía, siendo inferior al emplearse menos individuos (131.40), que se ve reflejado en un menor tiempo de cálculo (13.42 s). Para este análisis, el algoritmo con una población de 60 individuos presenta el mejor desempeño, si se incrementa su tamaño, se asegura el éxito de respuesta; sin embargo, el tiempo de cálculo aumenta. En caso de reducirse el tamaño de la población, disminuye el tiempo de cálculo; pero, el éxito de respuesta decrece hasta caer en un bajo desempeño.

5 CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

La operación de los sistemas de distribución en la práctica se realiza en configuración radial. Esta topología de red ofrece una baja confiabilidad; esto es, una falla en cualquier componente producirá una interrupción para todas las zonas aguas abajo de la localización de falla. Actualmente, la instalación de equipamiento de automatización de distribución (Distribution automation - DA) permite implementar la aplicación avanzada FLISR (Fault Location, Insolation and Service Restoration), que proporciona a la red la capacidad de autorreparación (self-healing) para mejorar su resiliencia.

La restauración del servicio (Service restoration - SR) es un componente esencial de la aplicación FLISR. Permite que luego de detectada y aislada una falla permanente en un área del sistema, se genere y ejecute automáticamente un plan para restablecer el servicio en las áreas no afectadas, empleando el menor número de maniobras. De esta manera, se mejora la confiabilidad y eficiencia de operación del sistema de distribución.

La restauración del servicio se formula como un problema de optimización multiobjetivo, combinatorio, no lineal y restringido, que puede ser resuelto con diferentes arquitecturas de automatización y técnicas de optimización. Desde el punto de vista tecnológico de los sistemas de distribución y en atención a aplicaciones prácticas, se desarrolló la metodología e implementación de un algoritmo de optimización centralizado para la resolución del problema de SR. El algoritmo se basa en una metaheurística híbrida que combina las técnicas de Evolución Diferencial (Differential evolution – DE) y el Aprendizaje Incremental Basado en Población Continua (Continuous Population Based Incremental Learning – PBILc).

En presencia de una falla permanente, el algoritmo de SR determina y lleva a cabo las soluciones de restauración sobre los elementos disponibles que generan el menor costo operativo y a su vez, permiten restablecer el servicio al mayor número de áreas no afectadas del sistema de distribución. La inteligencia de SR se integró en torno a la disposición de los siguientes elementos: la conmutación de los interruptores de líneas para reconfiguración, el despacho de las unidades de

generación distribuida y el deslastre de carga según la prioridad de los clientes del sistema. La metodología desarrollada para la integración de estos elementos permitió que el proceso de restauración del algoritmo se vuelva más efectivo y versátil. De esta forma, se proporciona al sistema la capacidad de autorreparación para atender a más escenarios de restauración.

Con el propósito de evaluar el desempeño y eficiencia del algoritmo de SR, se definieron algunos casos de estudio en los que se sometió al sistema de prueba estándar a distintos escenarios de falla. La calidad de los resultados se analizó conforme a los criterios de tasa de restauración, éxito de respuesta, desviación estándar de las iteraciones, promedio de iteraciones y el tiempo de cálculo. Se realizó una comparativa de rendimiento de los resultados del algoritmo con metaheurística híbrida contra los obtenidos al utilizar las técnicas por separado. Adicionalmente, se demostró la ventaja de la metaheurística al emplear el procesamiento en paralelo para reducir el esfuerzo computacional. Los resultados numéricos ilustran la robustez y eficiencia de la técnica de optimización implementada que proporciona planes de restauración adecuados, respetando todas las restricciones operativas y eléctricas, y en tiempos de cómputo aceptables.

En el presente trabajo de titulación se logró implementar un algoritmo de optimización centralizado basado en técnicas metaheurísticas DE-PBILc para resolver el problema de restauración del servicio en sistemas de distribución. Este algoritmo consideró la integración de distintos elementos de la red en la obtención de soluciones. El empleo de una técnica de optimización distinta a las aplicadas en investigaciones previas y la integración de más escenarios en la solución de restauración son contribuciones que este trabajo intenta brindar en el área, en pro de lograr una restauración del servicio efectiva bajo condiciones de operación de los futuros sistemas de distribución.

5.2 Recomendaciones

La restauración del servicio es un paso esencial hacia una red inteligente; sin embargo, no existe un número considerable de investigaciones publicadas sobre esta área. En el presente trabajo se desarrolló la metodología e implementación de un algoritmo de optimización metaheurístico para resolver el problema de SR, que incluye en las soluciones la integración de elementos como reconfiguración de red, generación distribuida y clientes prioritarios. No obstante, se requieren de más estudios que extiendan los escenarios con el propósito de lograr una red inteligente capaz de autorrepararse de eventos extremos. Desde el punto de vista funcional, se considera que se puede efectuar mejoras al algoritmo implementado. Se presenta algunos enfoques a manera de recomendaciones para trabajos futuros:

- Afinar la técnica de generación de soluciones iniciales. Esta estrategia puede permitir que el algoritmo de optimización inicialice en regiones más cercanas a los valores óptimos, mejorando el proceso de convergencia y su desempeño.
- Complementar la metodología de priorización de las cargas del sistema. El proceso de priorización de los clientes puede afinarse al incorporar múltiples factores para el cálculo del peso de prioridad de carga, tales como: criticidad de carga, efecto sobre los índices de confiabilidad total del sistema (Energy not supplied - ENS, System average interruption duration index - SAIDI, System average interruption frequency index - SAIFI) y el costo de interrupción de carga. Con esta metodología los pesos se pueden calcular y modelar sistemáticamente, lo que magnifica la controlabilidad de la restauración.
- Extender la operación de microrredes. Si el procedimiento de restauración del algoritmo no dispone de rutas de restauración, las unidades DG pueden alimentar localmente a las cargas desenergizadas, por lo que la totalidad o parte de la red puede operar en modo microrred. Es necesario incorporar una estrategia integral que extienda la inteligencia de SR y considere la operación coordinada óptima de las microrredes en el sistema de distribución.
- Incluir las incertidumbres de DER y cargas. En la práctica, la generación de energía desde recursos renovables y el consumo de las cargas son fluctuantes, lo que introduce incertidumbres al proceso de restauración. La solución de SR necesita incorporar la distribución de probabilidad y técnicas de pronóstico para permitir una toma de decisiones realista.

Anexo A

ANEXO A DATOS DE RED DEL SISTEMA DE PRUEBA ESTÁNDAR

A.1 Datos de nodos

Nodo	Tipo	Pd (kW)	Qd (kVAr)
1	3	0	0
2	1	100	60
3	1	90	40
4	1	120	80
5	1	60	30
6	1	60	20
7	1	200	100
8	1	200	100
9	1	60	20
10	1	60	20
11	1	45	30
12	1	60	35
13	1	60	35
14	1	120	80
15	1	60	10
16	1	60	20
17	1	60	20
18	1	90	40
19	1	90	40
20	1	90	40
21	1	90	40
22	1	90	40
23	1	90	50
24	1	420	200
25	1	420	200
26	1	60	25
27	1	60	25
28	1	60	20
29	1	120	70
30	1	200	600
31	1	150	70
32	1	210	100
33	1	60	40

A.2 Datos de líneas

A.2.1. Líneas operativas

Línea	Nodo desde	Nodo hacia	r (Ω)	x (Ω)
1	1	2	0,057526	0,029324
2	2	3	0,307595	0,156668
3	3	4	0,228357	0,1163
4	4	5	0,237778	0,121104
5	5	6	0,510995	0,441115
6	6	7	0,116799	0,386085
7	7	8	0,44386	0,146685
8	8	9	0,642643	0,461705
9	9	10	0,651378	0,461705
10	10	11	0,122664	0,040555
11	11	12	0,233598	0,077242
12	12	13	0,915922	0,720634
13	13	14	0,337918	0,444796
14	14	15	0,36874	0,328185
15	15	16	0,465635	0,340039
16	16	17	0,80424	1,073775
17	17	18	0,456713	0,358133
18	2	19	0,102324	0,097644
19	19	20	0,938508	0,845668
20	20	21	0,255497	0,298486
21	21	22	0,442301	0,584805
22	3	23	0,281515	0,192356
23	23	24	0,560285	0,442425
24	24	25	0,559037	0,437434
25	6	26	0,126657	0,064514
26	26	27	0,17732	0,090282
27	27	28	0,660737	0,582559
28	28	29	0,501761	0,437122
29	29	30	0,316642	0,161285
30	30	31	0,607953	0,60084
31	31	32	0,193729	0,225799
32	32	33	0,212759	0,330805

A.2.2. Líneas de enlace

Línea	Nodo desde	Nodo hacia	r (Ω)	x (Ω)
33	21	8	1,247851	1,247851
34	9	15	1,247851	1,247851
35	12	22	1,247851	1,247851
36	18	33	0,311963	0,311963
37	25	29	0,311963	0,311963

REFERENCIAS

- [1] J. D. Juárez, *Sistemas de distribución de energía eléctrica*, 1.^a ed. México: Sans Serif Editores, 1995.
- [2] A. Zidan *et al.*, «Fault Detection, Isolation, and Service Restoration in Distribution Systems: State-of-the-Art and Future Trends», *IEEE Trans. Smart Grid*, vol. 8, n.º 5, pp. 2170-2185, sep. 2017, doi: 10.1109/TSG.2016.2517620.
- [3] M. AlOwaifeer y M. AlMuhaini, «Load Priority Modeling for Smart Service Restoration», *Can. J. Electr. Comput. Eng.*, vol. 40, n.º 3, pp. 217-228, 2017, doi: 10.1109/CJECE.2017.2705174.
- [4] F. Shen, Q. Wu, y Y. Xue, «Review of Service Restoration for Distribution Networks», *Journal of Modern Power Systems and Clean Energy*, vol. 8, n.º 1, pp. 1-14, ene. 2020, doi: 10.35833/MPCE.2018.000782.
- [5] A. F. Zobaa y T. J. Bihl, Eds., *Big data analytics in future power systems*. Boca Raton: Taylor & Francis, a CRC title, part of the Taylor & Francis imprint, a member of the Taylor & Francis Group, the academic division of T&F Informa, plc, 2018.
- [6] C. E. Landázuri, «Diseño e implementación de un sistema piloto para la automatización de alimentadores de la red de distribución de la Empresa Eléctrica Regional Centro Sur C.A.», Universidad de Cuenca, Cuenca, 2016.
- [7] Y. D. Benitez, «Metodología de diseño conceptual de la automatización de red de distribución de energía que permita la integración de recursos energéticos distribuidos (DER) e implementación de estrategias de gestión de demanda (DSM)», Universidad Nacional de Colombia, Bogotá, 2017.
- [8] J. Romero, «Appying Self-Healing Schemes to Modern Power Distribution Systems», en *IEEE Power and Energy Society General Meeting*, jul. 2012, pp. 1-4. doi: 10.1109/PESGM.2012.6344960.
- [9] B. Uluski, «Distribution Management Systems», Cleveland Ohio, 2011, p. 178.
- [10] F. Friend *et al.*, «Effect of Distribution Automation on Protective Relaying», en *2014 67th Annual Conference for Protective Relay Engineers*, College Station, TX, USA, mar. 2014, pp. 193-228. doi: 10.1109/CPRE.2014.6799002.
- [11] N. Higgins, V. Vyatkin, N.-K. C. Nair, y K. Schwarz, «Distributed Power System Automation With IEC 61850, IEC 61499, and Intelligent Control», *IEEE Trans. Syst.*,

Man, Cybern. C, vol. 41, n.º 1, pp. 81-92, ene. 2011, doi: 10.1109/TSMCC.2010.2046322.

[12] J. M. Gers, *Distribution system analysis and automation*. London: Institution of Engineering and Technology, 2014.

[13] C.-C. Liu, S. J. Lee, y S. S. Venkata, «An expert system operational aid for restoration and loss reduction of distribution systems», *IEEE Transactions on Power Systems*, vol. 3, n.º 2, pp. 619-626, may 1988, doi: 10.1109/59.192914.

[14] K. Hotta, H. Nomura, H. Takemoto, K. Suzuki, S. Nakamura, y S. Fukui, «Implementation of a real-time expert system for a restoration guide in a dispatching center», en *Conference Papers Power Industry Computer Application Conference*, may 1989, pp. 172-178. doi: 10.1109/PICA.1989.38992.

[15] Y.-Y. Hsu *et al.*, «Distribution system service restoration using a heuristic search approach», *IEEE Transactions on Power Delivery*, vol. 7, n.º 2, pp. 734-740, abr. 1992, doi: 10.1109/61.127075.

[16] D. Shirmohammadi, «Service restoration in distribution networks via network reconfiguration», *IEEE Transactions on Power Delivery*, vol. 7, n.º 2, pp. 952-958, abr. 1992, doi: 10.1109/61.127104.

[17] E. Drayer, N. Kechagia, J. Hegemann, M. Braun, M. Gabel, y R. Caire, «Distributed Self-Healing for Distribution Grids With Evolving Search Space», *IEEE Transactions on Power Delivery*, vol. 33, n.º 4, pp. 1755-1764, ago. 2018, doi: 10.1109/TPWRD.2017.2762090.

[18] N. D. R. Sarma, V. C. Prasad, K. S. Prakasa Rao, y V. Sankar, «A new network reconfiguration technique for service restoration in distribution networks», *IEEE Transactions on Power Delivery*, vol. 9, n.º 4, pp. 1936-1942, oct. 1994, doi: 10.1109/61.329526.

[19] P. L. Cavalcante *et al.*, «Centralized Self-Healing Scheme for Electrical Distribution Systems», *IEEE Trans. Smart Grid*, vol. 7, n.º 1, pp. 145-155, ene. 2016, doi: 10.1109/TSG.2015.2454436.

[20] S. Khushalani, J. M. Solanki, y N. N. Schulz, «Optimized Restoration of Unbalanced Distribution Systems», *IEEE Trans. Power Syst.*, vol. 22, n.º 2, pp. 624-630, may 2007, doi: 10.1109/TPWRS.2007.894866.

[21] R. Karn, Y. Kumar, y G. Agnihotri, «Development of ACO Algorithm for Service Restoration in Distribution system», p. 7.

- [22] S. Toune, H. Fudo, T. Genji, Y. Fukuyama, y Y. Nakanishi, «Comparative study of modern heuristic algorithms to service restoration in distribution systems», *IEEE Trans. Power Delivery*, vol. 17, n.º 1, pp. 173-181, ene. 2002, doi: 10.1109/61.974205.
- [23] K.-J. Mun, H.-S. Kim, J. H. Park, J.-I. Seo, y G.-H. Hwang, «Practical Application of Tabu Search to Distribution System Service Restoration in Korea», *IFAC Proceedings Volumes*, vol. 36, n.º 20, pp. 523-527, sep. 2003, doi: 10.1016/S1474-6670(17)34522-6.
- [24] A. S. Tanenbaum y M. van Steen, *Distributed systems: principles and paradigms*, 2nd ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2007.
- [25] I.-H. Lim *et al.*, «Design and Implementation of Multiagent-Based Distributed Restoration System in DAS», *IEEE Trans. Power Delivery*, vol. 28, n.º 2, pp. 585-593, abr. 2013, doi: 10.1109/TPWRD.2013.2244923.
- [26] M. Eriksson, M. Armendariz, O. O. Vasilenko, A. Saleem, y L. Nordstrom, «Multiagent-Based Distribution Automation Solution for Self-Healing Grids», *IEEE Trans. Ind. Electron.*, vol. 62, n.º 4, pp. 2620-2628, abr. 2015, doi: 10.1109/TIE.2014.2387098.
- [27] B. Pham, C. Huff, P. E. Nick Vendittis, A. Smit, A. Stinskiy, y S. Chanda, «Implementing Distributed Intelligence by Utilizing DNP3 Protocol for Distribution Automation Application», en *2018 IEEE/PES Transmission and Distribution Conference and Exposition (T&D)*, Denver, CO, abr. 2018, pp. 1-7. doi: 10.1109/TDC.2018.8440305.
- [28] P. Parikh, I. Voloh, y M. Mahony, «Fault Location, Isolation, and Service Restoration (FLISR) Technique using IEC 61850 GOOSE», p. 6.
- [29] A. Zidan y E. F. El-Saadany, «A Cooperative Multiagent Framework for Self-Healing Mechanisms in Distribution Systems», *IEEE Trans. Smart Grid*, vol. 3, n.º 3, pp. 1525-1539, sep. 2012, doi: 10.1109/TSG.2012.2198247.
- [30] A. Abel Hafez, W. A. Omran, y Y. G. Hegazy, «A Decentralized Technique for Autonomous Service Restoration in Active Radial Distribution Networks», *IEEE Trans. Smart Grid*, vol. 9, n.º 3, pp. 1911-1919, may 2018, doi: 10.1109/TSG.2016.2602541.
- [31] R. Roofegari Nejad y W. Sun, «Distributed Load Restoration in Unbalanced Active Distribution Systems», *IEEE Trans. Smart Grid*, vol. 10, n.º 5, pp. 5759-5769, sep. 2019, doi: 10.1109/TSG.2019.2891419.

- [32] D. P. Bernardon, L. N. Canha, A. R. Abaide, V. J. Garcia, M. Sperandio, y G. S. Lopes, «Smart grid concepts applied to self-healing in distribution system», en *2014 49th International Universities Power Engineering Conference (UPEC)*, Cluj-Napoca, Romania, sep. 2014, pp. 1-6. doi: 10.1109/UPEC.2014.6934588.
- [33] E. Coster, W. Kerstens, y T. Berry, «Self healing distribution networks using smart controllers», en *22nd International Conference and Exhibition on Electricity Distribution (CIRED 2013)*, jun. 2013, pp. 1-4. doi: 10.1049/cp.2013.0603.
- [34] W. P. Mathias-Neto y J. R. S. Mantovani, «A Node-Depth Encoding-Based Tabu Search Algorithm for Power Distribution System Restoration», *J Control Autom Electr Syst*, vol. 27, n.º 3, pp. 317-327, jun. 2016, doi: 10.1007/s40313-016-0234-6.
- [35] E. G. Morquecho, S. P. Torres, y C. A. Castro, «An efficient hybrid metaheuristics optimization technique applied to the AC electric transmission network expansion planning», *Swarm and Evolutionary Computation*, vol. 61, p. 100830, mar. 2021, doi: 10.1016/j.swevo.2020.100830.
- [36] S. P. Torres y C. A. Castro, «Specialized differential evolution technique to solve the alternating current model based transmission expansion planning problem», *International Journal of Electrical Power & Energy Systems*, vol. 68, pp. 243-251, jun. 2015, doi: 10.1016/j.ijepes.2014.12.016.
- [37] R. K. Karn, Y. Kumar, y G. Agnihotri, «Multiobjective Service Restoration Considering Primary Customers using Hybrid GA-ACO Algorithm», *IJCA*, vol. 64, n.º 3, pp. 1-10, feb. 2013, doi: 10.5120/10611-5327.
- [38] Xianchao Huang, G. Taylor, M. Irving, Lizi Zhang, y Wenbin Tao, «Service restoration of distribution systems using enhanced NDE with LCO», en *2013 IEEE Power & Energy Society General Meeting*, Vancouver, BC, 2013, pp. 1-5. doi: 10.1109/PESMG.2013.6672400.
- [39] Wei Wei, Mingjun Sun, Ran Ren, y Yong Wang, «Service restoration of distribution system with priority customers and distributed generation», en *IEEE PES Innovative Smart Grid Technologies*, Tianjin, China, may 2012, pp. 1-6. doi: 10.1109/ISGT-Asia.2012.6303400.
- [40] L. T. Marques, A. C. B. Delbem, J. B. A. London, y M. H. M. Camillo, «Service restoration in large-scale distribution systems considering three levels of priority customers», en *2015 IEEE Eindhoven PowerTech*, Eindhoven, jun. 2015, pp. 1-6. doi: 10.1109/PTC.2015.7232702.

- [41] D. Simon, *Evolutionary Optimization Algorithms*. New Jersey: John Wiley & Sons, 2013.
- [42] M. Kleinberg, K. Miu, y H.-D. Chiang, «Service restoration of power distribution systems incorporating load curtailment», en *2009 IEEE International Symposium on Circuits and Systems*, Taipei, Taiwan, may 2009, pp. 1709-1712. doi: 10.1109/ISCAS.2009.5118104.
- [43] V. Vai, S. Suk, R. Lorm, C. Chhlonh, S. Eng, y L. Bun, «Optimal Reconfiguration in Distribution Systems with Distributed Generations Based on Modified Sequential Switch Opening and Exchange», p. 13, 2021.
- [44] M. Lavorato, J. F. Franco, M. J. Rider, y R. Romero, «Imposing Radiality Constraints in Distribution System Optimization Problems», *IEEE Trans. Power Syst.*, vol. 27, n.º 1, pp. 172-180, feb. 2012, doi: 10.1109/TPWRS.2011.2161349.
- [45] M. E. Baran y F. F. Wu, «Network reconfiguration in distribution systems for loss reduction and load balancing», *IEEE Transactions on Power Delivery*, vol. 4, n.º 2, pp. 1401-1407, abr. 1989, doi: 10.1109/61.25627.
- [46] W. M. Dahalan y H. Mokhlis, «Network reconfiguration for loss reduction with distributed generations using PSO», en *2012 IEEE International Conference on Power and Energy (PECon)*, Kota Kinabalu, Malaysia, dic. 2012, pp. 823-828. doi: 10.1109/PECon.2012.6450331.
- [47] R. K. Karn, Y. Kumar, y G. Agnihotri, «Analysis of Crowding Population Based Ant Colony Optimization for Service Restoration Problem in Distribution System», *International Journal of Advanced Research in Computer Science*, p. 9, 2010.
- [48] N. E. Matute, S. P. Torres, E. G. Morquecho, F. Astudillo-Salinas, J. C. Lopez, y W. C. Flores, «Improving the AC Transmission Expansion Planning by Using Initial Solutions Algorithms», en *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, The Hague, Netherlands, oct. 2020, pp. 494-498. doi: 10.1109/ISGT-Europe47291.2020.9248778.