

Ricerca di Sistema elettrico



Progettazione e implementazione di modelli per la riconfigurazione delle reti e microreti di distribuzione in presenza di guasti in ottica di miglioramento dell'adeguatezza, sicurezza e resilienza

Sergio Bruno, Roberto Cometa, Maria Dicorato, Massimo La Scala, Francesco Lorusso, Roberto Sbrizzai, Angelo Velini



Politecnico
di Bari

DEI DIPARTIMENTO DI
INGEGNERIA ELETTRICA
E DELL'INFORMAZIONE

LabZERO
Zero Emission Research Option

Progettazione e implementazione di modelli per la riconfigurazione delle reti e microreti di distribuzione in presenza di guasti in ottica di miglioramento dell'adeguatezza, sicurezza e resilienza

S. Bruno, R. Cometa, M. Dicorato, M. La Scala, F. Lorusso, R. Sbrizzai, A. Velini
(DEI-Politecnico di Bari, Bari, Italia)

Dicembre 2024

Report Ricerca di Sistema Elettrico

Accordo di Programma Ministero dell'Ambiente e della Sicurezza Energetica -ENEA Piano Triennale di Realizzazione 2022-2024

Obiettivo 2: Digitalizzazione ed evoluzione delle reti

Progetto 2.3: Evoluzione, pianificazione, gestione ed esercizio delle reti elettriche

Linea di attività: LA 1.10 Progettazione e implementazione di modelli per la riconfigurazione delle reti e microreti di distribuzione in presenza di guasti in ottica di miglioramento dell'adeguatezza, sicurezza e resilienza

Responsabile del Progetto: Dr. Maria Valenti

Responsabile Linea di Attività: Prof. Sergio Bruno

Mese inizio previsto: luglio 2023

Mese inizio effettivo: luglio 2023

Mese fine previsto: dicembre 2024

Mese fine effettivo: dicembre 2024

Indice

1	Risultati attesi	6
2	Risultati ottenuti.....	6
2.1	Avanzamento della ricerca rispetto allo stato dell'arte internazionale con riferimento ai risultati ottenuti.....	7
2.2	Premi	9
3	Prodotti attesi	10
4	Prodotti ottenuti.....	10
5	Analisi degli scostamenti su attività e risultati.....	10
6	Sintesi delle attività svolte	11
7	Dettaglio delle attività svolte.....	11
8	Contributo delle eventuali consulenze alle attività sopra descritte.....	13
9	Pubblicazioni scientifiche	14
10	Eventi di disseminazione	14
11	Descrizione dei risultati ottenuti	14
11.1	Nomenclatura	14
11.2	Formulazione matematica del problema di Optimal Network Reconfiguration.....	15
11.2.1	Suddivisione zonale della rete di distribuzione	16
11.2.2	Funzioni obiettivo.....	17
11.2.3	Vincoli di radialità e connettività della rete.....	17
11.2.4	Relazioni di "parentela"	18
11.3	Funzioni di Distribution Automation	19
11.3.1	Funzione Ricerca Guasti (FRG).....	19
11.3.2	Funzione Neutro Compensato (FNC)	21
11.3.3	Smart Fault Selection (SFS).....	22
11.4	Algoritmi di soluzione del problema ONR	23
11.4.1	Algoritmo MILP (Mixed-Integer Linear Programming)	24
11.4.2	Algoritmo GA (Genetic Algorithm)	24
11.5	Esempio di soluzione del problema ONR su una rete semplificata	27
11.5.1	Confronto delle prestazioni dei due algoritmi MILP e GA	31
11.6	Architettura del software open-source per ONR	32
11.6.1	Parsing dei dati	33
11.6.2	Algoritmo di suddivisione zonale.....	33
11.6.3	Inizializzazione del problema ONR	35

11.6.4	Risoluzione del problema ONR	35
11.6.5	Selezione del risolutore	36
11.6.6	Navigazione del tool e Interfaccia grafica utente (Graphic User Interface, GUI)	36
12	Software e librerie open-source.....	39
13	Riferimenti bibliografici	40

Indice delle figure

Figura 1 - Esempio di suddivisione zonale di una rete di distribuzione primaria.....	17
Figura 2 - Esempio di ciclo di richiusura di una linea MT in cavo, nel caso di guasto a terra	19
Figura 3 - Esempio di attuazione logica FRG realizzata con IMS.....	20
Figura 4 - Esempio di attuazione logica FNC [24]	21
Figura 5 - Esempio di attuazione logica SFS [25].....	23
Figura 6 - Flowchart dell'algoritmo GA	26
Figura 7 - Rete di prova per la verifica dei vincoli di radialità e connettività del problema ONR	27
Figura 8 - Topologia della rete di prova: a) topologia radiale iniziale, b) topologia radiale ottimale	28
Figura 9 - Indici di qualità monitorati e funzione obiettivo normalizzata nella configurazione iniziale (SI) e nelle configurazioni ottimali	31
Figura 10 - Architettura dello strumento software open-source per la risoluzione di ONR	33
Figura 11 - Algoritmo di suddivisione zonale della rete di distribuzione da riconfigurare	34
Figura 12 - Esempio di GUI: corrispondenza tra grafo nodale (a sinistra) e grafo zonale equivalente (a destra).....	38
Figura 13 - Esempio di GUI: grafi zonali equivalenti della rete di distribuzione prima e dopo la ONR.....	38
Figura 14 - Esempio di GUI: indici di qualità monitorati normalizzati e funzione obiettivo	38
Figura 15 - Script Python per l'importazione delle librerie necessarie per l'esecuzione di ONR	40

Indice delle tabelle

Tabella 1 - Domanda di carico zonale e numero di clienti connessi.....	28
Tabella 2 - Stato degli interruttori prima e dopo l'esecuzione di ONR	28
Tabella 3 - Valori assunti dalla variabile di flusso artificiale nella topologia radiale iniziale (a) e ottimale (b)	29
Tabella 4 - Valori assunti dalla variabile di condizione "a monte" nella topologia radiale iniziale (a) e ottimale (b)	30
Tabella 5 - Algoritmo MILP, variazione percentuale degli indicatori al variare della tecnica FDIR applicata	31
Tabella 6 - Algoritmo GA, variazione percentuale degli indicatori al variare della tecnica FDIR applicata	31
Tabella 7 - Prestazioni computazionali degli algoritmi.....	32

1 Risultati attesi

Si riporta quanto previsto in output alla attività.

La LA1.10 dovrà fornire modelli software per la Optimal Network Reconfiguration orientati al miglioramento della adeguatezza, sicurezza e resilienza delle reti e adattabili alle attuali pratiche e funzioni di protezione adottate nella distribution automation delle reti pubbliche italiane.

Elementi quantitativi per la verifica:

- *n.1 algoritmo strutturato in modo da poter integrare almeno 3 diverse funzioni obiettivo collegate agli indicatori di prestazione proposti;*
- *n.1 software basato sull'algoritmo di calcolo scelto in funzione dei risultati dei test preliminari condotti.*

La LA1.10 dovrà fornire un Rapporto Tecnico descrittivo del software sviluppato e delle scelte progettuali che hanno portato alla definizione delle metodologie identificate.

Elementi quantitativi per la verifica:

- *n.1 modello matematico corrispondente alla formulazione del problema di ottimizzazione;*
- *descrizione degli algoritmi che verranno preliminarmente testati;*
- *test di almeno due algoritmi, ad esempio uno basato su tecniche analitiche (ad es. MILP o MINLP) e uno su tecniche di metaeuristica (ad es. applicate alla parte intera in un algoritmo basato sulla two-stage optimization).*

Si ricorda inoltre che l'attività mirava altresì allo sviluppo di algoritmi per la riconfigurazione ottimale, basati su tecniche di programmazione matematica e strutturati secondo le seguenti specifiche:

- Il problema deve essere risolvibile mediante codici o utilizzo di piattaforme di calcolo e ottimizzazione open-source;
- le reti di distribuzione dovranno essere rappresentate utilizzando software di tipo open-source;
- consentire accoppiamento con altri software di ottimizzazione sviluppati nell'ambito del progetto.

2 Risultati ottenuti

L'attività di ricerca ha condotto allo sviluppo di modelli di Optimal Network Reconfiguration (ONR) che includono delle funzioni obiettivo indirizzate al miglioramento della adeguatezza, sicurezza e resilienza delle reti, e che possono inoltre essere adattati alle attuali pratiche e funzioni di protezione adottate nella *Distribution Automation* delle reti pubbliche italiane.

L'attività ha condotto allo sviluppo di un tool software per la riconfigurazione ottimale, basato su tecniche MILP di programmazione matematica e strutturati secondo le seguenti specifiche:

- il problema è formulato e risolto in un ambiente di programmazione e calcolo *open-source*;
- i modelli di rete sono compatibili con il software open-source OpenDSS;

- il tool è integrato all'intero della piattaforma di progetto ARS-Tool.

Di seguito si elencano i principali vantaggi dei modelli e degli algoritmi sviluppati:

- i modelli risolvono efficacemente i problemi di natura computazionale legati alla complessità del problema, dovuta alla sua natura combinatoriale, e che cresce con il crescere delle possibili risorse di controllo e con la diffusione di *Distribution Automation*;
- i modelli sono sufficientemente generali per poter essere adattati ad altre funzioni di ottimizzazione topologica;
- grazie alle elevate prestazioni computazionali, l'algoritmo ONR è applicabile sia in pianificazione che in esercizio nell'ambito delle funzioni SCADA/DMS;
- l'adozione di pratiche ONR abilita l'utilizzo di risorse di flessibilità *cost-free* che non impattano con altri equilibri del sistema (ad esempio mercati, piani di produzione, etc.);
- valutazione degli impatti dei guasti sulla continuità del servizio, secondo i classici indicatori di qualità (SAIDI, SAIFI, EENS, etc.);
- capacità di gestire reti di dimensioni realistiche (ad es. rete sottesa ad una intera cabina primaria);
- sviluppo del tool su ambiente open-source (OpenDSS, Python, etc.);
- formulazione analitica del problema ONR e risoluzione utilizzando Mixed Integer Linear Programming o altre tecniche che possano comunque essere utilizzate su solutori open-source;
- inclusione nella formulazione delle logiche di automazione effettivamente in uso nel panorama italiano della distribuzione;
- possibili applicazioni in fase di planning, per determinare l'impatto del deployment di *Distribution Automation* e nuove tecniche FDIR per la gestione dei guasti.

2.1 Avanzamento della ricerca rispetto allo stato dell'arte internazionale con riferimento ai risultati ottenuti

Il settore della distribuzione dell'energia elettrica svolge un ruolo fondamentale nel garantire una fornitura di energia sicura e affidabile e nel soddisfare la domanda dei consumatori. Tuttavia, interruzioni e guasti possono compromettere la continuità operativa delle reti di distribuzione. Per migliorare la qualità del servizio elettrico, i gestori dei sistemi di distribuzione (Distribution System Operators, DSO) sono chiamati a ridurre il numero e la durata delle interruzioni, impiegando apparecchiature di *Distribution Automation* e adottando strategie efficienti per identificare e isolare i guasti lungo i feeders di distribuzione [1].

Quando si verifica un guasto in un feeder, gli utenti collegati a monte, a valle e nei percorsi laterali rischiano di subire un'interruzione del servizio. Oggi l'identificazione dei guasti avviene attraverso schemi di *Distribution Automation*, che si basano su protezioni intelligenti, dispositivi elettronici intelligenti avanzati (Intelligent Electronic Devices, IED), indicatori di passaggio delle correnti di guasto (Fault Passage Indicators, FPI) e rilevatori di guasti, come ad esempio i sistemi RGDAT ed RGDM adottati nella pratica italiana [2].

Nel contesto italiano, infatti, vengono adottate diverse soluzioni per il rilevamento, l'isolamento e il ripristino dei guasti. In particolare, i DSO italiani utilizzano spesso schemi automatici come la Funzione di Ricerca Guasti (FRG), la Funzione Neutro Compensato (FNC), e la Smart Fault Selection (SFS) [3].

L'applicabilità di queste tecniche dipende dalla struttura della rete e dai dispositivi di protezione installati. Ad esempio, la tecnica FRG può essere applicata a qualsiasi sistema di distribuzione, con neutro isolato o compensato, per identificare sia i guasti monofase a terra che quelli polifase [4]. L'FRG si basa sull'uso di FPI e sulle funzioni di richiusura automatica degli interruttori. L'FNC, invece, richiede la presenza di un neutralizzatore di guasti a terra (ad esempio la bobina di Petersen) e la sua azione è limitata ai soli guasti monofase a terra [5]. La tecnica SFS costituisce la tecnica di automazione più innovativa e si basa sull'uso di IED e sistemi di comunicazione adeguati (ad esempio, fibra ottica, LTE, WiMax) [6]. L'impatto di queste funzioni sulla sicurezza e sulla qualità della fornitura dipende, ovviamente, dalla configurazione e dalla topologia della rete.

Sebbene le reti di distribuzione siano per lo più gestite utilizzando topologie fisse e predeterminate, nelle reti di distribuzione intelligenti (*smart distribution grids*) i DSO possono implementare, all'interno di sistemi SCADA/DMS, funzioni di Optimal Network Reconfiguration (ONR) [7], utili per supportare le decisioni degli operatori sia a fini di sicurezza operativa [8] che di pianificazione [9].

L'analisi dello stato dell'arte ha evidenziato come il problema di ONR sia stato tradizionalmente discusso e applicato a livello di trasmissione, ma solo di recente esteso alla pianificazione e riconfigurazione off-line dei sistemi di distribuzione. Solitamente gli operatori di sistema attuano riconfigurazioni di rete solamente in pianificazione o, al più, in condizioni di emergenza e/o di ripristino della rete.

L'adozione di strumenti ONR da parte dei gestori delle reti elettriche di distribuzione è sempre stata limitata, specialmente nella fase di esercizio della rete, per via della complessità e tempi di computazione, costi di hardware e software per implementazione della architettura di controllo in SCADA/DMS, impatto dei cambi topologici sulle logiche di protezione. In aggiunta, al fine di sviluppare funzioni compatibili con le attuali pratiche di gestione delle reti e della *Distribution Automation*, l'approccio deve includere vincoli operativi legati alle strategie di coordinamento e controllo delle protezioni, degli interruttori e degli organi di manovra/sezionatori telecomandati.

Relativamente alla sua modellazione matematica e implementazione, in letteratura esistono svariate formulazioni del problema di ONR [10], risolto utilizzando molteplici approcci, sia analitici [11], [12] che meta-euristici [13], [14]. Trattandosi di problemi di tipo Mixed-Integer Non-Linear Programming (MINLP), le principali sfide relative alla risoluzione di ONR sono legate alla gestione delle non linearità delle equazioni di rete (AC Load Flow) e alla natura combinatoriale del problema, caratterizzato da un gran numero di variabili intere.

La maggior parte delle applicazioni mira a minimizzare le perdite o a ottimizzare gli indici di qualità [15], offrendo un approccio strategico per migliorare l'efficienza operativa e l'affidabilità dei sistemi di distribuzione. Solo pochi lavori si occupano di integrare la logica dei dispositivi di protezione all'interno dell'ottimizzazione [16].

Inoltre, spesso sono necessarie efficienti tecniche di ottimizzazione per la risoluzione, motivo per cui è raro ritrovare implementazioni di ONR in ambienti open-source, soprattutto per la risoluzione di modelli che integrano anche le equazioni di *power flow* nella formulazione. Infatti, modelli complessi come quelli descritti in [11], [17] e [18] vengono comunque risolti tramite solver commerciali disponibili con licenza. In particolare, anche se [17] fornisce un approccio pratico per implementare e risolvere il problema di ONR in applicazioni real-time, l'ambiente di sviluppo software utilizzato necessita di una licenza commerciale.

In [19], invece, è presentata una formulazione basata su Mixed-Integer Linear Programming (MILP) implementata con una libreria di ottimizzazione open-source, tuttavia risolta con un solver a pagamento. I modelli in [12], [13] e [20], invece, sono risolti con algoritmi metaeuristici che non richiedono l'impiego di alcun risolutore esterno ma sono sviluppati con un linguaggio di programmazione non gratuito (MATLAB®). Uno dei pochi esempi in cui sono utilizzati strumenti open-source per risolvere un problema di ONR è disponibile in [21], nonostante venga impiegato un metodo di ottimizzazione globale per la risoluzione.

L'obiettivo di questa attività è stato quello di fornire uno strumento flessibile, efficiente, che si possa facilmente adattare alle funzioni di protezione attualmente in uso, garantendo pertanto un più facile trasferimento dei risultati di questa ricerca al mondo dell'industria. Pertanto, lo strumento ONR è stato progettato per avere le seguenti proprietà:

- flessibilità e adattabilità ad altre funzioni di gestione delle reti;
- efficienza computazionale;
- affidabilità della soluzione e del metodo risolutivo;
- integrazione delle pratiche operative utilizzate dai distributori italiani.

Per quanto riguarda il paragone con lo stato dell'arte si segnalano i seguenti *gap* trovati nella letteratura e che si reputano invece superati da questa ricerca:

- sviluppo di un modello matematico organico che integra al suo interno le funzioni di *Distribution Automation* con le relazioni di parentela tra elementi del sistema, i vincoli di connettività e radialità, e le funzioni obiettivo;
- tempi di soluzione e compatibilità con il controllo in esercizio delle reti (si rimanda ai risultati della LA 1.11);
- scalabilità e capacità di gestione di reti complesse (si rimanda alla validazione compiuta durante la LA 1.11 con un modello realistico di rete di distribuzione);
- sviluppo in ambiente di programmazione e calcolo open-source.

In sintesi, è stato sviluppato un modello di ONR che possa abilitare l'effettivo utilizzo di questo tipo di funzione nell'industria elettrica e nei sistemi di gestione delle reti di distribuzione elettriche italiane.

In relazione alla effettiva applicabilità dello ONR in esercizio, si reputa che la principale limitazione sia relativa alla effettiva capacità di riconfigurazione delle logiche di protezione per mezzo di schemi di protezione adattativa (*adaptive relaying*). Per questo motivo, nella LA 1.11 sono state condotte delle prove in ambiente (Power) Hardware-in-the-Loop, per la verifica di schemi adattativi di protezione basati su protocollo IEC 61850, in grado di implementare nuove logiche di selettività a seguito dei cambiamenti topologici del sistema.

2.2 Premi

In aggiunta ai risultati già descritti, e previsti da capitolato, si segnala il premio ricevuto nell'ambito di un convegno internazionale di prestigio:

Il paper "An Open-Source Optimal Network Reconfiguration Tool for Improving Distribution Grid Reliability", di R. Cometa, A. Velini, F. Lorusso, A. Ricca, R. Sbrizzai, S. Bruno, si è aggiudicato il **Best Student Paper Award** durante il convegno 2024 AEIT International Annual Conference (AEIT), Trento, Italy, 25-27 Sep. 2024.

Il premio ricevuto conferma l'originalità e la qualità tecnica dei prodotti scientifici della presente ricerca.

3 Prodotti attesi

In output alla LA1.10 erano attesi i seguenti deliverable:

- n.1 rapporto descrittivo del modello di riconfigurazione ottimale di un sistema di distribuzione;
- n.1 software (research grade) sviluppato in linguaggio Python, eventualmente integrato con software open source per rappresentazione di reti elettriche e accompagnato da un manuale d'uso.

4 Prodotti ottenuti

In output alla LA1.10 sono presentati i seguenti deliverable:

- n.1 un rapporto descrittivo del modello di riconfigurazione ottimale di un sistema di distribuzione (contenuto in questo stesso documento);
- n.1 software (research grade) sviluppato in linguaggio Python, integrato con software open-source e inserito nella piattaforma di progetto ARS-Tool, comprensiva di manualetto d'uso.

Il software è integrato nella piattaforma ARS-Tool e costituisce uno degli strumenti a disposizione dell'utente in piattaforma. Una breve guida contenuta nelle cartelle degli script permette di customizzare alcune scelte parametriche e adeguare la rete test alle esigenze del software.

Come descritto nei paragrafi successivi, il software si arricchisce anche di alcune routine necessarie per la strutturazione dell'algoritmo risolutivo. In particolare, si fa riferimento alle routine di data-parsing, necessarie per l'importazione dei dati di rete dall'ambiente OpenDSS all'ambiente di sviluppo Python, e alla routine dedicata al partizionamento della rete elettrica in "zone" e alla costruzione di un grafo zonale per la formulazione delle funzioni di *Distribution Automation* (v. descrizione dettagliata nel Paragrafo 11).

Come da capitolato, il rapporto tecnico contiene:

- n.1 modello matematico corrispondente alla formulazione del problema di ottimizzazione (v. paragrafi 11.2 e 11.3). Il modello matematico include la descrizione di una funzione multi-obiettivo basata su tre indicatori di progetto (EENS, SAIDI, SAIFI). Inoltre, il modello contiene tre diverse formulazioni delle medesime funzioni obiettivo a seconda della tecnica di automazione adottata (FRG, FNC, SFS).
- descrizione degli algoritmi preliminarmente testati (v. paragrafo 11.4);
- test di due algoritmi, uno basato su tecniche analitiche (MILP) e uno su metaeuristica (GA)(v. paragrafo 11.5).

5 Analisi degli scostamenti su attività e risultati

I principali scostamenti sono dovuti alla tarda pubblicazione del decreto direttoriale n. 495 del 6 novembre 2023 di approvazione dell'Accordo di Programma (decreto del 2 novembre 2023 tra il Ministero dell'ambiente e della sicurezza energetica e RSE S.p.A., ENEA e CNR- DIITET,

registrato alla Corte dei conti in data 28/12/2023 al n.4072), che ha rallentato l'inizio delle procedure di reclutamento e richiesto alcune modifiche nel piano economico:

- Il progetto prevedeva il reclutamento di un assegnista di ricerca per la durata di un anno. A causa dei suddetti ritardi non è stato possibile reclutare un assegnista entro la scadenza naturale del progetto. Al fine di mantenere l'equilibrio complessivo tra ore-uomo, attività previste e curriculum dei ricercatori reclutati, è stato scelto di bandire, e poi affidare, due borse post-laurea della durata di sette mesi ciascuna.
- Per gli stessi motivi richiamati in precedenza, le attività di progetto hanno richiesto un maggiore impegno temporale da parte delle risorse già presenti (personale dipendente), limitando, invece, altre capacità di spesa (ad esempio partecipazione a convegni, missioni, etc.). Ad ogni modo lo scostamento sulla voce complessiva del personale è di lieve entità (circa tremila euro).

I lievi scostamenti economici appena discussi non hanno avuto alcun impatto sui risultati attesi della LA.

6 Sintesi delle attività svolte

Nel corso di questa attività sono stati studiati dei modelli di riconfigurazione ottimale di rete (ONR) adattati alle caratteristiche topologiche e alle pratiche di gestione delle reti elettriche di distribuzione italiane. Il problema ONR è stato formulato per poter ottenere benefici in termini di adeguatezza, sicurezza e resilienza delle reti e, pertanto, ha richiesto l'integrazione nella sua struttura di alcune relazioni che possano replicare le pratiche di gestione guasti (FDIR) nella automazione dei sistemi di distribuzione italiani. Il problema include anche vincoli di connettività e radialità, e funzioni di parentela tra i nodi del sistema, necessarie per le logiche FDIR.

Il problema matematico è stato implementato su una piattaforma open-source basata su linguaggio Python, capace di gestire modelli di rete rinvenienti dal software open-source OpenDSS e di interfacciarsi con solver matematici open-source e commerciali. La validazione del software è stata compiuta nella LA1.11.

7 Dettaglio delle attività svolte

L'attività ha mirato allo sviluppo di metodologie e modelli che possano essere applicati al problema di riconfigurazione ottimale (Optimal Network Reconfiguration - ONR) delle reti di distribuzione dell'energia elettrica, nell'ambito delle funzioni di controllo in supervisione dei sistemi SCADA/DMS (Distribution Management System) e delle loro applicazioni in ambito Nazionale.

Nel seguito vengono dettagliate le attività che sono state compiute nell'ambito della LA1.10, sottolineandone i risultati, le scelte metodologiche che ne sono derivate e gli sviluppi della intera macro-attività.

La prima fase di questa attività è stata orientata alla ricerca bibliografica e alla determinazione di modelli che possano essere compatibili con le attuali pratiche nella Distribution Automation (DA) e procedure di gestione dei guasti (Fault Detection Isolation and Recovery - FDIR).

In questa fase sono stati individuati i **principali schemi di automazione** attualmente in uso nel panorama italiano della distribuzione dell'energia elettrica, individuando tre schemi principali, basati su un crescente grado di sviluppo della Distribution Automation:

- FRG (Funzione Ricerca Guasti) – tecnica FDIR ampiamente utilizzata e basata sulla presenza di sistemi di individuazione del passaggio di correnti di guasto e di assenza tensione (ad esempio RGDAT e RGDM). Il sistema si basa sulla apertura automatica di sezionatori telecomandati e su schemi di ricerca guasto basati sulla temporizzazione delle richiuse.
- FNC (Funzione Neutro Compensato) – tecnica FDIR utilizzata nei sistemi di distribuzione con neutro compensato per i soli guasti monofase a terra. La tecnica si avvale della presenza sul campo di sistemi di individuazione del passaggio di correnti di guasto e di assenza tensione (ad esempio RGDAT e RGDM) e dell'automazione di interruttori di manovra-sezionatori (IMS) in grado di interrompere le piccole correnti di guasto nel caso di cortocircuiti monofase. Per mezzo di selettività cronometrica, gli IMS sono aperti in modo da isolare il solo tronco guasto.
- SFS (Smart Fault Selection) – tecnica SFS più moderna basata sulla selettività logica ottenuta coordinando interruttori automatici distribuiti lungo-linea per mezzo di comunicazione orizzontale e l'utilizzo del protocollo IEC 61850. La possibilità di ottenere una protezione adattativa con SFS è stata discussa nella LA1.11, nella quale la logica SFS è stata riprodotta in test Power Hardware-in-the-Loop.

Al fine di tener conto delle suddette logiche di automazione è necessario definire all'interno del problema di riconfigurazione delle **relazioni topologiche** tra i diversi nodi del sistema. Per questo motivo, in una successiva fase della attività, sono stati studiati metodi per la rilevazione delle relazioni di parentela nello sviluppo del grafo di un sistema di distribuzione. Le relazioni di parentela sono necessarie, ad esempio, per determinare l'appartenenza a un determinato feeder, e quindi quale sia l'interruttore automatico in testa-linea deputato alla interruzione di un guasto. Inoltre, le relazioni di parentela consentono di determinare la posizione di ogni nodo rispetto ad un elemento guasto (upstream/downstream) o l'eventuale ricerca di nodi guasti sui laterali di uno stesso feeder.

Al fine di poter determinare con facilità queste relazioni, e contenere al tempo stesso il numero di variabili e vincoli del problema ONR, è stato scelto di utilizzare un **approccio zonale**. Le zone raccolgono elementi elettricamente connessi (sbarre e linee) che non sono connessi tra loro mediante switch telecomandati. Il raggruppamento in zone permette di associare ad elementi presenti in una stessa zona gli stessi parametri affidabilistici, quali ad esempio frequenza o durata di un guasto, e di semplificare il numero di relazioni topologiche tra gli elementi di rete (le relazioni topologiche sono costruite tra le zone e non tra elementi singoli di rete). Al fine di costruire il modello zonale del sistema è stato necessario sviluppare un **algoritmo di partizionamento zonale** descritto nel dettaglio nel Paragrafo 11.

Una volta eseguito il partizionamento nodale è possibile determinare con facilità anche le relazioni di parentela, **connettività e radialità**. In particolare, è stato sviluppato un metodo basato sui **flussi artificiali**, in grado di determinare relazioni di parentela tra i nodi e di assicurare la connettività (banalmente deve esistere almeno un percorso radiale di alimentazione tra ogni zona e la sorgente del sistema). La radialità è facilmente assicurata se è assicurata connettività e se è rispettata la condizione necessaria (ma non sufficiente) che il numero di lati sia pari al numero di zone meno uno.

Nella fase successiva di questa attività è stato **formulato matematicamente il problema di riconfigurazione di rete**. Senza perdere di generalità, il problema è stato formulato minimizzando le tipiche funzioni obiettivo di qualità del servizio utilizzate anche nell'ambito della regolazione italiana (SAIDI, SAIFI, EENS). Il problema considera come variabili, le variabili binarie on/off che rappresentano lo stato chiuso/aperto di ciascuno switch telecomandabile del sistema. Inoltre, sono definite delle variabili addizionali, sia binarie che continue, che permettono di risolvere il problema di ONR mediante tecniche di programmazione MILP (Mixed-Integer Linear Programming).

Successivamente, le attività sono state mirate alla ricerca delle tecniche di programmazione matematica più opportune per la **soluzione del problema**, privilegiando nella scelta metodologie che possano essere facilmente implementate su piattaforme di ottimizzazione open-source. Questa attività è stata svolta in stretto coordinamento con la LA1.11, poiché modellazione e validazione degli algoritmi risolutivi richiedono la definizione di un processo iterativo. In questa fase sono stati testati **due algoritmi risolutivi**, uno basato su tecniche MILP e uno su algoritmi genetici. A seguito dei test, è stato scelto di privilegiare l'approccio analitico alla soluzione del problema, utilizzando tecniche di programmazione lineare-mista interi (MILP). L'approccio analitico è stato preferito ad approcci di tipo metaeuristico poiché, essendo basato su fondamenti matematici solidi e rigorosi, conduce a delle formulazioni più facili da comprendere e interpretare, permette di analizzare in dettaglio i risultati ottenuti, e garantisce una maggiore affidabilità dei risultati. Inoltre, l'approccio analitico si è dimostrato capace di trovare soluzioni migliori, con tempi più ridotti.

Questa scelta ha permesso l'utilizzo di applicativi **open-source** facilmente integrabili nella piattaforma complessiva di sviluppo in ambiente Python. I risultati delle attività di test e di validazione sono presentati nel rapporto relativo alla LA1.11.

L'ultima fase di questa attività è stata mirata allo sviluppo dell'applicativo software da integrare nella piattaforma di progetto ARS-Tool. L'applicativo è stato sviluppato, come da specifiche di progetto, in **ambiente Python**, ed è in grado di gestire modelli di rete sviluppati in ambiente **OpenDSS**. L'OpenDSS è un software open-source dedicato alla modellazione ed analisi delle reti di distribuzione, ed è considerato uno degli standard *de facto* nell'ambito degli studi per le reti di distribuzione.

L'applicativo ONR sviluppato durante la LA1.10 consente la integrazione nella piattaforma ARS-Tool di tutti i metodi appena discussi. Pertanto, il software consente di:

- importare un modello di rete di distribuzione;
- sviluppare un modello zonale equivalente;
- associare parametri affidabilistici a ciascuno degli elementi di rete e a ciascuna zona;
- associare degli indici di qualità del servizio ad una qualsiasi topologia di rete;
- selezionare la tecnica FDIR in uso (FRG, FNC o SFS);
- trovare una topologia ottimale al fine di minimizzare la funzione obiettivo;
- presentare i risultati dell'ottimizzazione e rappresentare graficamente le scelte di modifica topologica e i grafi equivalenti del sistema.

8 Contributo delle eventuali consulenze alle attività sopra descritte

Non si è ricorso a consulenze.

9 Pubblicazioni scientifiche

- [A] R. Cometa, M. R. Nasab, F. Lorusso, A. Velini, M. La Scala, and S. Bruno, "Integration of FDIR Practices in Distribution Optimal Network Reconfiguration Routines," *2024 IEEE International Conference on Environment and Electrical Engineering and 2024 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*, Rome, Italy, 2024, pp. 1-6, doi: 10.1109/EEEIC/ICPSEurope61470.2024.10751156.
- [B] R. Cometa, A. Velini, F. Lorusso, A. Ricca, R. Sbrizzai, and S. Bruno, "An Open-Source Optimal Network Reconfiguration Tool for Improving Distribution Grid Reliability," *2024 AEIT International Annual Conference (AEIT)*, Trento, Italy, 2024, pp. 1-6, doi: 10.23919/AEIT63317.2024.10736725
L'articolo si è aggiudicato il Best Student Paper Award del convegno AEIT 2024.

10 Eventi di disseminazione

- 2024 IEEE International Conference on Environment and Electrical Engineering and 2024 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe), Roma, Italia, 17-20 Giugno 2024.
- 2024 AEIT International Annual Conference (116th Edition), Trento, Italia, 25-27 Settembre 2024.
- Evento Finale di disseminazione dei risultati del Progetto di Ricerca 2.3 "Evoluzione, pianificazione, gestione ed esercizio delle reti elettriche" (Piano triennale di realizzazione 2022-2024), C. R. ENEA, Portici (NA), Italia, 3 Dicembre 2024.

11 Descrizione dei risultati ottenuti

Nel seguito si fornisce una descrizione della formulazione matematica del problema di ONR e della strutturazione del software ONR integrato nella piattaforma ARS-Tool.

11.1 Nomenclatura

Indici

z, i, j, k, m	Indici per zone generiche
h	Indice per le "zone ancestrali" di derivazione
s	Indice per un generico interruttore automatico

Insiemi

Ω_z	Zone
$\Omega_h \subset \Omega_z$	Zone ancestrali di derivazione
$\Omega_{SS} \subset \Omega_z$	Cabine primarie
Ω_{sw}	Interruttori automatici di protezione

Parametri

$\varphi_{k,m}^G, \varphi_{k,m}^D$	Generazione (G) o domanda (D) artificiale alla zona k rispetto alla zona m
P_z^D	Domanda di potenza attiva alla zona z
λ	Tasso di guasto [guasti/anno]
N	Numero di clienti
μ	Tasso di riparazione [h]
$\bar{\mu}$	Tempo di completamento della richiusura automatica [h]

Variabili reali continue

$\varphi_{s,m}$	Flusso artificiale attraverso l'interruttore s calcolato per ogni zona m
f_z	Frequenza di guasti nella zona z [h/anno]
U_z	Indisponibilità della zona z [h/anno]
$EENS$	Expected Energy Not Served [kWh/anno]
$SAIDI$	System Average Interruption Duration Index [h/anno]
$SAIFI$	System Average Interruption Frequency Index [guasti/anno]

Variabili binarie

x_s	Stato di connessione dell'interruttore automatico: 0 (aperto), 1 (chiuso)
$z_{k,m}^{up}$	Condizione di connessione "a monte": è uguale a 1 se la zona k si trova a monte della zona m , (altrimenti 0).
$y_{k,m}$	Condizione di parentela tra zone: è uguale a 1 se le zone k ed m appartengono allo stesso feeder (altrimenti 0).
$u_{h,k,m}$	Condizione di ancestralità: è uguale a 1 se sia la zona k che la zona m sono discendenti della stessa zona ancestrale di derivazione h (altrimenti 0).

11.2 Formulazione matematica del problema di Optimal Network Reconfiguration

La soluzione di un problema di Optimal Network Reconfiguration (ONR) necessita di una attenta scelta delle tecniche di programmazione e di calcolo, poiché il problema è intrinsecamente combinatoriale e caratterizzato da un grande numero di variabili binarie. Solitamente, questo tipo di problema è risolto con metodi misto-interi (Mixed-Integer Linear Programming o Mixed-Integer Non Linear Programming) o con approcci metaeuristici.

In questo progetto è stato formulato un modello matematico, sufficientemente generale per poter essere adattato sia a una soluzione per mezzo di approcci metaeuristici (algoritmi genetici), che mediante tecniche di programmazione MILP.

Senza perdere di generalità, il problema ONR è stato sviluppato per minimizzare tre funzioni obiettivo legate ai più comuni indici di qualità del servizio del servizio elettrico, quali EENS, SAIDI e SAIFI, tenendo conto delle relazioni funzionali tra gli elementi di manovra del sistema, dovute alle tecniche di automazione di rilevamento, isolamento e ripristino di eventi di guasto (Fault Detection Isolation and Recovery, FDIR).

11.2.1 Suddivisione zonale della rete di distribuzione

Dato il numero molto elevato di possibili variabili (interi e continue) necessarie per modellare le azioni di switching degli interruttori e descrivere le relazioni di connessione tra i nodi di una rete elettrica, la complessità computazionale di un problema ONR può essere drasticamente ridotta considerando che le reti di distribuzione sono tipicamente sistemi debolmente magliati, e che inoltre la maggior parte degli elementi di ramo non può essere commutata automaticamente. Durante i guasti, infatti, solo pochi interruttori sono chiamati a intervenire per attuare le funzioni di rilevamento e isolamento dei guasti.

Data una generica rete di distribuzione, è possibile definire una zona z come un insieme di nodi e rami interconnessi tra loro e delimitati solo da interruttori (Figura 1). Non essendoci interruttori automatici all'interno di una zona, tramite il partizionamento della rete in zone, si può ottenere un grafo zone-interruttori equivalente.

In questo grafo, ogni zona con tutti i suoi elementi rappresenta un vertice, mentre gli interruttori rappresentano gli archi che interconnettono le zone. In genere, non vi sono maglie all'interno di una zona e, di conseguenza, se il grafo zonale è radiale, allora anche l'intera rete risulta altrettanto radiale. Ugualmente, se tutti all'interno di ciascuna zona sono connessi, e il grafo zone-interruttori è connesso, allora tutti gli elementi di rete sono connessi.

Questa operazione di partizionamento consente non solo di ridurre il numero di variabili intere, ma anche il numero di vincoli di connettività e di condizioni di "parentela" tra le diverse porzioni della rete.

Come osservato, dato che le zone non contengono interruttori al loro interno, qualsiasi guasto all'interno di una zona può essere isolato ed eliminato solo da un interruttore esterno alla zona e presente nel percorso a monte verso la sorgente di alimentazione (cabina primaria).

Per una generica zona z , è possibile definire un tasso di guasto zonale λ_z e un tasso di riparazione μ_z . Questi vengono valutati in accordo con l'ipotesi di elevata riparabilità, secondo cui il tempo medio di riparazione è significativamente inferiore al tempo medio di guasto. Per quanto concerne λ_z , esso si ottiene sommando i tassi di guasto di ciascun componente all'interno di una zona (ad esempio sbarre, rami, terminali, ecc.). Questo è possibile perché tutti gli eventi di guasto all'interno di una zona sono eventi in serie. Per quanto riguarda μ_z , invece, esso si ottiene dividendo la somma di tutte le ore di interruzioni in serie (cioè i prodotti $\lambda \cdot \mu$) per λ_z .

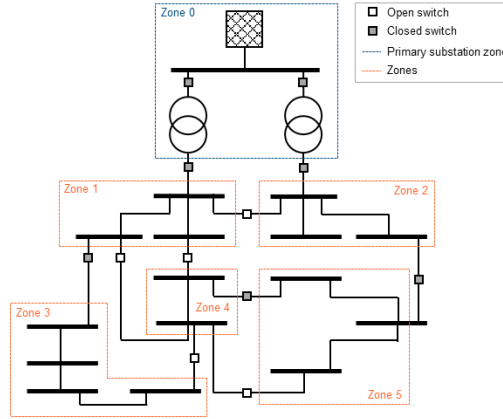


Figura 1 - Esempio di suddivisione zonale di una rete di distribuzione primaria

11.2.2 Funzioni obiettivo

Il problema di ottimizzazione MILP considera lo stato di connessione degli interruttori x_s come variabile di controllo. Per ottimizzare gli indici di qualità del servizio elettrico, si è formulata una funzione obiettivo composta. Questa funzione è ottenuta dalla combinazione ponderata di tre funzioni obiettivo individuali, ciascuna basata rispettivamente sugli indici EENS, SAIDI e SAIFI:

$$\min(w_1 \cdot EENS + w_2 \cdot SAIDI + w_3 \cdot SAIFI) \quad (1)$$

Gli indici in (1) sono ottenuti aggregando i risultati di ciascuna zona:

$$EENS = \sum_{z \in \Omega_z} P_z^D \cdot U_z \quad (2)$$

$$SAIDI = \frac{\sum_{z \in \Omega_z} U_z \cdot N_z}{\sum_{z \in \Omega_z} N_z} \quad (3)$$

$$SAIFI = \frac{\sum_{z \in \Omega_z} f_z \cdot N_z}{\sum_{z \in \Omega_z} N_z} \quad (4)$$

I pesi w_1 , w_2 e w_3 possono essere utilizzati per abilitare/disabilitare le funzioni obiettivo o per pesare in misura diversa i tre contributi. Come illustrato nel seguito, senza perdere di generalità, l'approccio solitamente seguito è quello di pesare in maniera uniforme i tre contributi normalizzati (v. paragrafo 11.6.3).

La minimizzazione della funzione obiettivo (1) è soggetta a vincoli di uguaglianza e disuguaglianza per garantire che la rete rimanga radiale e connessa dopo l'ONR.

Inoltre, nel seguito, saranno formulate delle funzioni di *Distribution Automation* che permettono di introdurre le relazioni funzionali tra switch dovute alle diverse tecniche FDIR (FRG, FNC, o SFS) nella formulazione di (2)-(4). A seguito di questa distinzione, seguiranno tre diverse formulazioni MILP del problema.

11.2.3 Vincoli di radialità e connettività della rete

Se la rete di distribuzione viene esercita con una topologia radiale, allora anche il suo grafo zonale equivalente deve essere radiale. La condizione necessaria ma non sufficiente per la radialità viene quindi formulata con l'approccio zonale come:

$$\sum_{s \in \Omega_{sw}} x_s = |\Omega_z| - |\Omega_{ss}| \quad (5)$$

dove $|\Omega_z|$ e $|\Omega_{ss}|$ indicano le cardinalità dei rispettivi insiemi, ossia il loro numero di elementi.

Per assicurare la connettività, si fa riferimento al metodo dei "flussi artificiali". Supponendo che una potenza unitaria sia generata in una qualsiasi zona sorgente $z \in \Omega_{ss}$, ossia in cabina primaria, e che vi sia una domanda di carico unitaria in una generica zona m , ogni interruttore s (orientato dalla zona i alla zona j) sarà interessato dalla circolazione di un flusso di potenza "artificiale" $\varphi_{s,m}$.

Il flusso artificiale $\varphi_{s,m}$ rappresenta la potenza che transita attraverso l'interruttore s quando una domanda di carico unitaria risulta applicata in m . Quindi, $\varphi_{s,m}$ è diverso da zero solo l'interruttore s si trova nel percorso radiale tra la sorgente e la zona m ed esso risulta chiuso. In tal caso, $\varphi_{s,m}$ assume valore $+1$ o -1 , a seconda dell'orientazione dell'interruttore s .

Pertanto, la connettività della rete è garantita se, per ogni zona m , sono rispettate le seguenti equazioni di bilancio zonale:

$$\sum_{\substack{s \in \Omega_{sw} \\ i=k}} \varphi_{s,m} - \sum_{\substack{s \in \Omega_{sw} \\ j=k}} \varphi_{s,m} + \varphi_{k,m}^G = \varphi_{k,m}^D \quad \forall k, m \in \Omega_z \quad (6)$$

dove:

$$\varphi_{k,m}^D = \begin{cases} 1 & \text{se } k = m \\ 0 & \text{se } k \neq m \end{cases} ; \quad \varphi_{k,m}^G = \begin{cases} 1 & \text{se } m \in \Omega_{ss} \\ 0 & \text{se } m \notin \Omega_{ss} \end{cases} \quad (7)$$

Il flusso artificiale $\varphi_{s,m}$ è inoltre legato allo stato di connessione di s attraverso i seguenti vincoli di disuguaglianza:

$$|\varphi_{s,m}| \leq x_s \quad \forall s \in \Omega_{sw}, \forall m \in \Omega_z \quad (8)$$

11.2.4 Relazioni di "parentela"

Per una generica zona m , è possibile determinare se qualsiasi altra zona si trovi a monte o a valle, attraverso i seguenti vincoli:

$$z_{i,m}^{up} \geq |\varphi_{s,m}| \quad \forall s \in \Omega_{sw}, \forall m \in \Omega_z \quad (9)$$

$$z_{j,m}^{up} \geq |\varphi_{s,m}| \quad \forall s \in \Omega_{sw}, \forall m \in \Omega_z \quad (10)$$

Finché l'interruttore s fa parte del percorso radiale tra due zone, allora $z_{i,m}^{up} = z_{j,m}^{up} = |\varphi_{s,m}| = 1$.

Grazie alle condizioni di connessione "a monte", è possibile determinare se due zone generiche k ed m appartengono allo stesso feeder introducendo le variabili binarie $u_{h,k,m}$ e $y_{k,m}$ e i seguenti vincoli:

$$u_{h,k,m} \geq z_{h,k}^{up} + z_{h,m}^{up} - 1 \quad \forall k, m \in \{\Omega_z \setminus \Omega_{ss}\}, \forall h \in \Omega_h \quad (11)$$

$$y_{k,m} = \sum_{h \in \Omega_h} u_{h,k,m} \quad \forall k, m \in \{\Omega_z \setminus \Omega_{ss}\} \quad (12)$$

Si osservi che $u_{h,k,m}$, e quindi $y_{k,m}$, saranno diverse da zero solo se k ed m sono discendenti della stessa zona ancestrale di derivazione h .

La dimensione del problema, e quindi l'onere computazionale richiesto per la sua soluzione, può essere ridotto considerando le relazioni di simmetria delle variabili $y_{k,m}$ e $y_{m,k}$. Al fine di costruire la matrice $|\Omega_z| \times |\Omega_z|$ contenente gli elementi $y_{k,m}$, è sufficiente definire solo gli

elementi che appartengono alla parte triangolare inferiore, compresa la diagonale principale. Pertanto, (11) e (12) possono essere formulate anche solo per $m \leq k$.

11.3 Funzioni di Distribution Automation

I termini in (2)-(4) devono essere espressi come funzioni delle variabili $z_{k,m}^{up}$ e $y_{k,m}$. Tali funzioni sono formulate in accordo con le tecniche di automazione FDIR precedentemente elencate e discusse nel seguito.

11.3.1 Funzione Ricerca Guasti (FRG)

La tecnica FRG è la più semplice tecnica di *Distribution Automation* adottata per isolare i guasti. Essa può essere attuata sia su linee in cavo che su linee aeree; tuttavia, tra i due casi variano i tempi di intervento. Inoltre, è possibile utilizzarla sia per reti esercite con neutro isolato che per reti esercite con neutro compensato.

Questa tecnica presuppone che gli interruttori automatici siano installati solo in cabina primaria, mentre tutti gli altri organi di protezione sono semplici sezionatori. Poiché i sezionatori sono in grado di aprire soltanto circuiti non alimentati, l'isolamento del guasto si ottiene manovrando i sezionatori entro il ciclo di apertura/chiusura della funzione di richiusura automatica degli interruttori [23], [24].

L'interruttore in cabina primaria interviene quando viene rilevata una corrente di guasto. Dopo un periodo di attesa, viene attivata una Richiusura Rapida (RR) al fine di eliminare eventuali guasti transitori autoestinguenti. Nel caso in cui, al termine della RR, il guasto persista, vengono avviate le funzioni di Richiusura Lenta (RL) in base alla tipologia della linea considerata, così come mostrato nell'esempio in Figura 2 [22]. L'intero ciclo operativo ha una durata massima di 180 secondi, rispettando così il limite temporale per le interruzioni brevi.

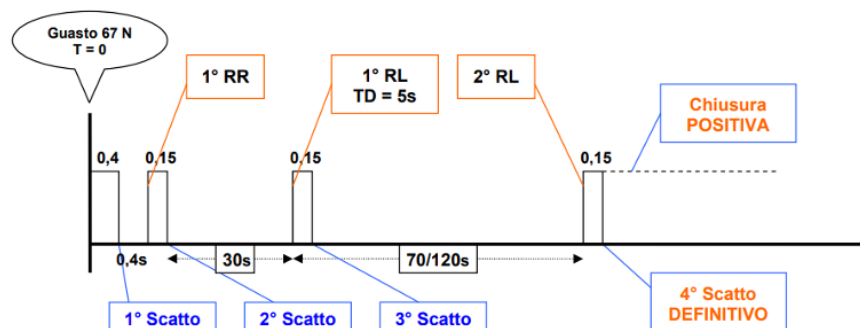


Figura 2 - Esempio di ciclo di richiusura di una linea MT in cavo, nel caso di guasto a terra

La logica di intervento è riassunta nei seguenti passi, oltre che descritta nell'esempio in Figura 3 [24]:

- al verificarsi di una mancanza di tensione a seguito dello scatto dell'interruttore di cabina primaria per un tempo prestabilito, l'automa di cabina secondaria (UP) comanda l'apertura degli IMS di cabina secondaria, i cui i rilevatori di guasto e mancanza tensione (RGDAT o RGDGM) abbiano rilevato un guasto polifase (51) o monofase (67N);
- gli IMS precedentemente aperti, avendo avvertito il ripristino della tensione per effetto di una chiusura dell'interruttore in cabina primaria e/o dell'IMS immediatamente a monte, richiudono secondo un ordine prestabilito, con un ritardo crescente da monte verso valle e avviano il conteggio di un Tempo di Discriminazione [TD];
- se durante il tempo TD è presente tensione, l'IMS entra in uno stato di inibizione all'apertura automatica successiva.

Al termine del ciclo RR/RL dell'interruttore di cabina primaria, solo l'IMS direttamente a monte del guasto risulterà aperto, mentre l'alimentazione su tutti i rami a monte sarà stata ripristinata.

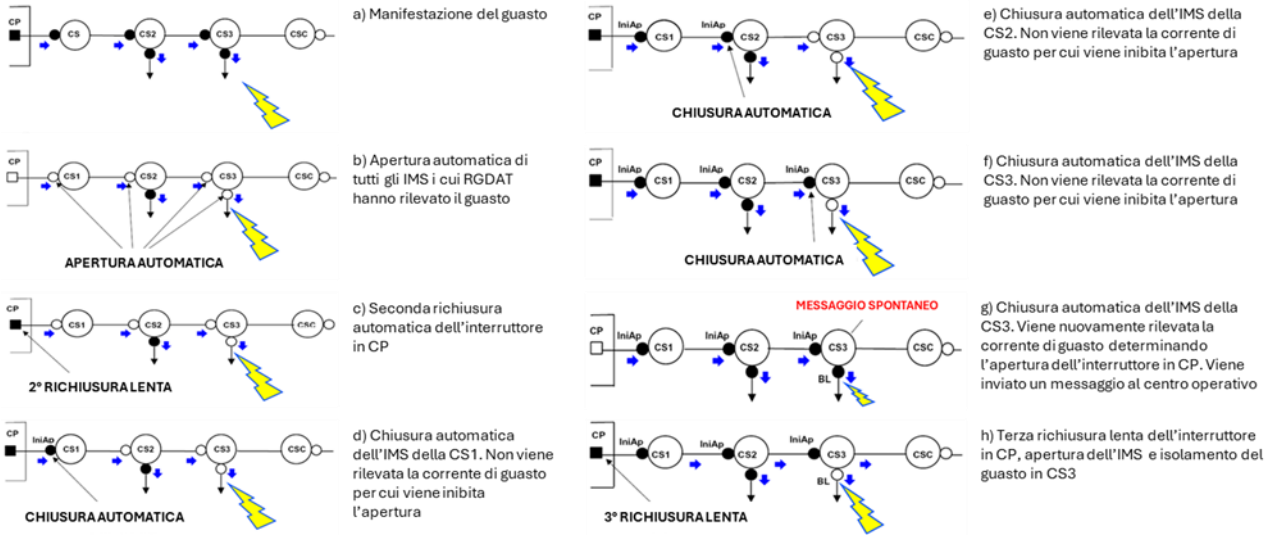


Figura 3 - Esempio di attuazione logica FRG realizzata con IMS

Relativamente al problema di ONR, in generale, la frequenza di guasto di una generica zona z è data dalla somma dei tassi di guasto di tutte le altre zone dello stesso feeder, e quindi:

$$f_z^{FRG} = \sum_{k \in \Omega_z} y_{k,z} \cdot \lambda_k \quad \forall z \in \Omega_z \quad (13)$$

La suddetta equazione afferma che qualsiasi guasto che si verifichi in un feeder causerà l'interruzione di tutte le altre zone dello stesso feeder.

Tuttavia, grazie alla funzione FRG, il guasto verrà isolato entro la fine della chiusura automatica dell'interruttore in cabina primaria. Questo significa che solo i guasti che si verificano nel percorso direttamente a monte fino alla sorgente causerebbero una lunga interruzione.

Tutti gli altri guasti, nei percorsi a valle o laterali, verranno eliminati entro un certo tempo $\bar{\mu}$ che dipende dal tempo necessario per isolare il guasto secondo la tecnica FRG (i.e. i tre minuti necessari per completare il ciclo RR/RL).

Pertanto, l'indisponibilità per ogni zona z può essere formulata come:

$$U_z^{FRG} = \sum_{k \in \Omega_z} [z_{k,z}^{up} \cdot \lambda_k \cdot \mu_k + (y_{k,m} - z_{k,z}^{up}) \cdot \lambda_k \cdot \bar{\mu}] \quad \forall z \in \Omega_z \quad (14)$$

Nella (14), U_z^{FRG} è la somma di due componenti: il primo termine $z_{k,z}^{up} \cdot \lambda_k \cdot \mu_k$ tiene conto dell'interruzione dovuta a un guasto nel percorso a monte.

Il secondo termine $(y_{k,m} - z_{k,z}^{up}) \cdot \lambda_k \cdot \bar{\mu}$ considera ciò che accade se il guasto dovesse trovarsi su un percorso a valle o laterale. Questo secondo termine è nullo nel caso di un guasto a monte, in quanto sia $y_{k,m}$ che $z_{k,z}^{up}$ sarebbero entrambi uguali a 1.

11.3.2 Funzione Neutro Compensato (FNC)

La tecnica FNC, anch'essa applicabile sia per linee in cavo che per linee aeree di reti esercite a neutro compensato, si basa sulla selettività cronometrica per i guasti monofase a terra. La selettività è ottenuta usando interruttori di manovra-sezionatori (IMS) al posto degli interruttori. Gli IMS, infatti, sono in grado di interrompere piccole correnti di guasto monofase a terra grazie alla compensazione effettuata tramite bobina di Petersen.

Nel caso di guasto monofase a terra, gli interruttori che rilevano il passaggio della corrente di guasto scattano selettivamente con un ritardo che aumenta risalendo sul feeder fino alla testa-linea.

In caso di guasto non monofase a terra, invece, gli IMS non sono in grado di interrompere la corrente di cortocircuito. Pertanto, per tutti questi guasti la tecnica FDIR applicata è analoga alla FRG [23], [24].

La logica di intervento è riassunta nei seguenti due passi e descritta nell'esempio in Figura 4:

1. Gli IMS vengono aperti se viene rilevata una corrente di guasto a terra a valle per un tempo TD, valutato rispetto alla posizione della cabina secondaria lungo il feeder in maniera tale che, tra gli IMS il cui RGDAT/RGDM individua il guasto, si apra soltanto quello più distante dalla CP nel rispetto della selettività cronometrica.
2. L'interruttore in cabina primaria, nel caso in cui il guasto non si verifichi sul primo tronco, rimane chiuso per l'intera durata della fase di selezione. In questo modo, l'intera sezione di linea a monte del tronco guasto non subisce alcuna interruzione. Se, al contrario, il guasto coinvolge il primo tronco, ossia quello presente prima del primo interruttore, l'interruttore esegue tutto il ciclo di richiusura fin quando non interviene, definitivamente, l'interruttore in cabina primaria.

Il blocco di un interruttore in posizione di "aperto" (nel primo caso), o la disconnessione definitiva (nel secondo caso), genera un messaggio automatico al centro operativo per avvisare l'operatore e consentirgli di gestire il guasto in maniera appropriata.

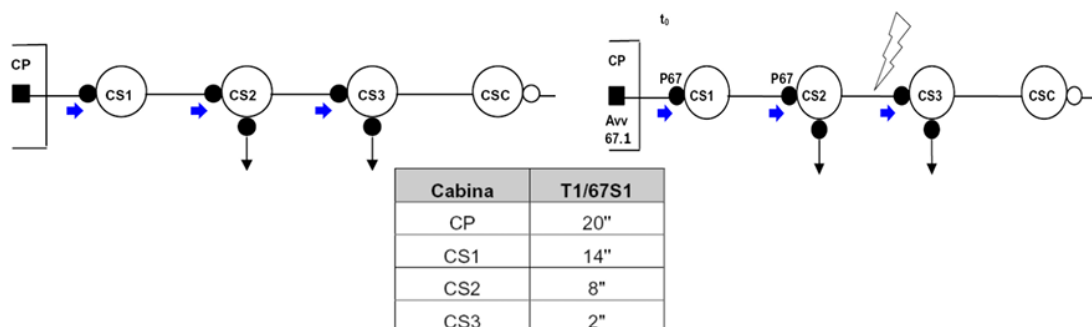


Figura 4 - Esempio di attuazione logica FNC [24]

Relativamente al problema di ONR, per valutare la frequenza di guasto e l'indisponibilità, è necessario introdurre tassi di guasto diversi per i guasti monofase a terra (λ'_k) e non monofase a terra (λ''_k). Nel primo caso, la tecnica FNC garantisce la selettività e quindi una zona viene interrotta solo qualora il guasto si verifichi nel percorso direttamente a monte verso la sorgente.

I guasti che si manifestano in un percorso laterale o a valle rispetto alla zona affetta da guasto non causano interruzioni. Per i guasti non monofase a terra, invece, valgono le stesse ipotesi fatte per la FRG, e quindi:

$$f_z^{FNC} = \sum_{k \in \Omega_z} (z_{k,z}^{up} \cdot \lambda'_k + y_{k,z} \cdot \lambda''_k) \quad \forall z \in \Omega_z \quad (15)$$

$$U_z^{FNC} = \sum_{k \in \Omega_z} [z_{k,z}^{up} \cdot (\lambda'_k + \lambda''_k) \cdot \mu_k + (y_{k,m} - z_{k,z}^{up}) \cdot \lambda''_k \cdot \bar{\mu}] \quad \forall z \in \Omega_z \quad (16)$$

11.3.3 Smart Fault Selection (SFS)

Questo tipo di automazione, di recente implementazione sulle reti di distribuzione italiane, presuppone l'adozione di veri e propri interruttori lungo i feeder. La selettività logica è ottenuta utilizzando le funzioni di comunicazione orizzontale tra gli Intelligent Electronic Device (IED) deputati alle funzioni di protezione. Tale tecnica è stata introdotta per far fronte a interruzioni molto brevi (entro 5 secondi), difatti non richiede alcuna azione remota da parte dell'operatore o del sistema e si attua in meno di 1 secondo.

Questo tipo di automazione sfrutta la struttura del protocollo IEC 61850 che consente, grazie alla sua organizzazione client-server bidirezionale e alla capacità di mandare messaggi interrupt (ad. esempio GOOSE message), l'implementazione di selettività logica tra le protezioni.

Ogni interruttore attraversato da una corrente di cortocircuito invia un messaggio di *blind* agli interruttori del percorso a monte. L'interruttore più vicino al guasto, e che quindi non riceve messaggi di *blind*, interverrà per primo, evitando l'interruzione dei percorsi a monte o laterali.

Affinché la SFS possa essere implementata è necessario che i dispositivi di comunicazione rispettino le seguenti condizioni:

- Stabilire comunicazioni tra CS-CS e CS-CP;
- Trasportare ed elaborare in modo efficiente i messaggi GOOSE su una rete estesa;
- Garantire una bassa latenza tra i dispositivi (< 100 ms);
- Assicurare il recapito sicuro delle informazioni e in particolare dei messaggi GOOSE;
- Gestire un elevato numero di dispositivi, approssimativamente cento per ciascuna sottostazione primaria.

Inoltre, è necessaria la presenza, in CS, dei Rilevatori di Guasto Direzionali e Misura (RGDM).

La logica di intervento è riassunta nei seguenti tre passi e descritta nell'esempio in Figura 5:

1. Individuazione del guasto;
2. Isolamento del ramo guasto;
3. Chiusura dell'interruttore della cabina di confine.

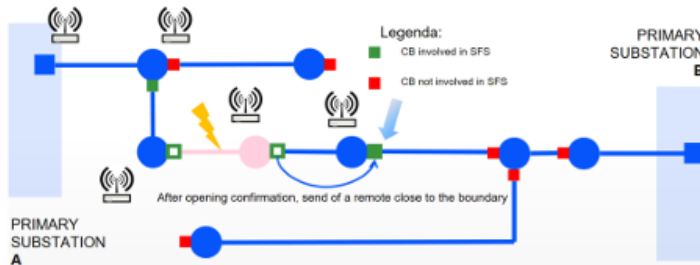
In presenza di un guasto, ogni RGDM che non rileva il guasto invia un segnale di *blind* per inibire l'azione dei dispositivi a monte. Il rivelatore più vicino al guasto, invece, invia un segnale di *trip*, determinando l'apertura dell'interruttore automatico a monte (prima manovra). Al termine di questa prima manovra, l'interruttore situato subito a valle del ramo colpito dal guasto si apre (seconda manovra), isolando completamente il ramo coinvolto. Al termine della seconda manovra, l'RGDM invia un messaggio di *Remote Close* all'interruttore posto al confine che chiudendosi ripristina l'alimentazione ai rami sani della rete [25], [26].



(a) Si presenta il guasto.



(b) Selezione del guasto.



(c) Contro-alimentazione della porzione di rete sana a seguito dell'isolamento del guasto.

Figura 5 - Esempio di attuazione logica SFS [25]

Relativamente al problema di ONR, in questo caso, i vincoli di "parentela" (11) e (12) non sono necessari e le funzioni di affidabilità risultano formulate semplicemente come segue:

$$f_z^{SFS} = \sum_{k \in \Omega_z} z_{k,z}^{up} \cdot \lambda_k \quad \forall z \in \Omega_z \quad (17)$$

$$U_z^{SFS} = \sum_{k \in \Omega_z} z_{k,z}^{up} \cdot \lambda_k \cdot \mu_k \quad \forall z \in \Omega_z \quad (18)$$

11.4 Algoritmi di soluzione del problema ONR

Il problema di ottimizzazione descritto nei paragrafi precedenti può essere risolto sia ricorrendo a un approccio metaeuristico che, in maniera più rigorosa, mediante tecniche di programmazione MILP.

Nel corso del progetto sono stati testati i due approcci mediante due algoritmi separati. Entrambi gli algoritmi risolutivi sono stati basati su applicativi open-source, sviluppati in ambiente Python. I risultati ottenuti, e presentati in questa relazione e nella relazione relativa alla attività di test LA1.11, hanno condotto alla **scelta dell'approccio MILP**.

La soluzione analitica del problema permette infatti di mantenere un fondamento matematico rigoroso e di giungere a delle formulazioni e dei risultati facilmente interpretabili. In aggiunta, come dimostrato dai risultati dei test condotti, l'approccio MILP permette di ottenere migliori risultati, e con tempi di computazione inferiori.

Sulla base di queste osservazioni, la scelta finale è ricaduta sull'algoritmo MILP che si avvantaggia anche della disponibilità di numerosi solver, liberamente scaricabili e già compatibili con l'ambiente Python di sviluppo della piattaforma ARS-Tool.

La struttura di questo algoritmo è dettagliata nella sezione 11.6.

11.4.1 Algoritmo MILP (Mixed-Integer Linear Programming)

La soluzione del problema ONR è ottenuta implementando l'algoritmo risolutivo sulla piattaforma Pyomo. Pyomo è una libreria open-source che permette di formalizzare facilmente, mediante una programmazione simbolica, le funzioni obiettivo e i vincoli lineari descritti in precedenza, specificando altresì il set di variabili continue ed intere.

Per la soluzione di un problema misto-intero, Pyomo si interfaccia con un risolutore (commerciale come Gurobi o open-source come SCIP o HIGHS), che implementa l'algoritmo di *branch and bound*. Inizialmente, il risolutore semplifica il problema risolvendolo nel dominio dei numeri reali trattando tutte le variabili come continue (*continuous relaxation*) e risolvendo, di fatto, un problema di programmazione lineare. Se la soluzione di questo problema rilassato è tale per cui le variabili intere assumono valori frazionari, il risolutore divide il problema in due sottoproblemi, aggiungendo vincoli che forzano le variabili intere ad assumere valori interi. Per ogni sottoproblema, il risolutore calcola un limite (*bound*) sulla funzione obiettivo, che permette di escludere rami dell'albero di ricerca che non possono contenere la soluzione ottima. Se viene trovata una soluzione intera valida, questa diventa la soluzione corrente, e il risolutore può escludere i sottoproblemi con limiti peggiori. Questo processo di divisione e limitazione continua fino a quando non viene trovata la soluzione ottima o viene raggiunto un criterio di arresto. Il *MIP gap*, che è la differenza percentuale tra la soluzione corrente e il miglior limite globale, è uno dei criteri di arresto comuni, insieme a limiti di tempo per la risoluzione e al numero di nodi dell'albero di ricerca esplorati.

L'algoritmo, quindi, esplora lo spazio delle soluzioni, escludendo quelle non promettenti e concentrandosi su quelle che possono contenere la soluzione ottima, fino a raggiungere una soluzione ottima o una soluzione sub-ottimale accettabile entro i limiti stabiliti.

Il risolutore HIGHS rappresenta un'alternativa open-source ad alte prestazioni, in grado di competere con i risolutori commerciali in termini di velocità ed efficienza.

11.4.2 Algoritmo GA (Genetic Algorithm)

Questa sezione descrive la procedura seguita per l'implementazione di un algoritmo genetico (GA) per la soluzione del problema di ONR. Il vantaggio principale nell'utilizzo di un approccio metaeuristico, come il GA, è quello di poter semplificare la struttura del problema matematico. Nel caso di ONR, lo spazio delle soluzioni è infatti formato dai soli set di possibili combinazioni aperto/chiuso (0/1) degli interruttori di manovra.

A ciascun set (individuo), selezionato come possibile soluzione dall'algoritmo genetico, possono essere associati in maniera univoca le relazioni di parentela ed ancestralità, le condizioni upstream/downstream zonali, e gli indici di affidabilità. Mediante la soluzione di semplici sotto-problemi è pertanto possibile associare ad ogni configurazione topologica, un determinato valore della funzione obiettivo da minimizzare. In particolare, la funzione obiettivo e le relazioni di vincolo vengono calcolati per ciascun individuo di ogni generazione se e solo se

ad ogni individuo corrisponde una topologia radiale e connessa. Soluzioni caratterizzate da configurazioni magliate o non connesse vengono penalizzate secondo un fattore proporzionale al numero della generazione corrente in modo tale da impedirne l'esplorazione nelle generazioni successive.

Come in tutti gli algoritmi GA, la routine ONR inizia con un campionamento iniziale della popolazione di N_{pop} individui, dove ogni individuo è un vettore contenente le variabili di controllo (ad esempio, lo stato degli switch manovrabili). Per ogni individuo, viene calcolato il valore della funzione obiettivo e verificato il rispetto dei vincoli di connettività e radialità. Nelle fasi successive, la popolazione viene aggiornata in base agli operatori genetici di selezione, crossover e mutazione, generando nuovi individui in cui quelli caratterizzati da un valore migliore della funzione obiettivo sostituiscono i peggiori, fino al raggiungimento di un criterio di stop.

Di solito l'inizializzazione di una popolazione di individui è basata su meccanismi di campionamento casuale, ma questo può portare a soluzioni lontane dalla soluzione ottimale desiderata, richiedendo un gran numero di iterazioni per raggiungere la convergenza. Per questo motivo, è stato utilizzato un metodo semplice per scartare rapidamente soluzioni che sono peggiori della condizione operativa iniziale durante l'esecuzione del GA. Partendo da una topologia iniziale connessa e radiale, la popolazione iniziale è ottenuta considerando N_{pop} topologie connesse e radiali ottenibili con una sola manovra di apertura e una sola manovra di chiusura rispetto alla configurazione iniziale, in modo da tale da ricavare una popolazione costituita da individui unici.

In un classico algoritmo GA, gli individui vengono confrontati in base alle loro funzioni di *fitness*, tuttavia, nella transizione da una generazione di individui alla successiva, gli individui devono essere adeguatamente selezionati per partecipare all'accoppiamento. Per migliorare la convergenza, è stato utilizzato un criterio di "torneo binario" (*binary tournament*) [27]: dato un sottoinsieme di individui della popolazione, gli individui vengono confrontati casualmente in coppie sulla base del loro valore di *fitness*. Per ciascuna coppia, l'individuo migliore viene selezionato come vincitore del torneo per poi essere sottoposto alla fase successiva di crossover nel ruolo di "genitore".

Quando i genitori sono stati selezionati, può iniziare la fase di crossover. Questo operatore combina le informazioni genetiche di due genitori per creare uno o più individui che possano avere una *fitness* migliore di quello dei genitori. Tra le tecniche esistenti, è stato scelto il *two point crossover* [28]: un tipo di crossover standard basato sulla frammentazione del gene in due punti casuali del cromosoma. Il metodo prevede la selezione casuale di due punti nel cromosoma (vettore di variabili binarie rappresentante lo stato degli switch manovrabili) di una coppia di individui (genitori), le informazioni comprese tra questi due punti vengono invertiti e dati alla coppia di individui della generazione successiva (figli). La nuova coppia di individui presenta in questo modo delle caratteristiche comuni alla coppia di individui della generazione precedente.

La fase di mutazione aiuta ad aumentare la diversità della popolazione evitando l'inconveniente di trovare ottimi locali. Essa viene eseguita dopo che i nuovi individui vengono creati attraverso il crossover. In particolare, la *bit flip mutation* (BFM) è stata adottata perché opera con variabili di tipo binario [29]. La BFM è una delle tecniche di mutazione più semplici e più comunemente utilizzate negli algoritmi genetici. Opera su individui di tipo binario, dove ogni individuo è rappresentato da un singolo bit (0 o 1). Durante la mutazione, ogni bit ha una piccola probabilità di essere mutato, cioè di cambiare il suo stato da 0 a 1 o viceversa. La probabilità di mutazione di un bit è determinata dal tasso di mutazione, che di solito è impostato su un valore basso per

mantenere un equilibrio tra esplorazione e valorizzazione dell'individuo. La BFM introduce piccole modifiche localizzate all'individuo, consentendo all'algorithmo genetico di esplorare soluzioni vicine nello spazio di ricerca.

Nell'algorithmo GA le nuove generazioni di individui sono ottenute alternando crossover e mutazione secondo delle funzioni di distribuzione di probabilità. Il crossover promuove la valorizzazione delle migliori soluzioni correnti, combinandole al fine di ottenere delle soluzioni potenzialmente migliori, mentre la mutazione è utile per evitare di rimanere bloccati in una soluzione ottima locale, mediante l'introduzione di variazioni casuali.

L'algorithmo GA si interrompe dopo un determinato numero massimo di generazioni stabilito a priori o al verificarsi di un criterio di stop. Il criterio di stop scelto è basato sul raggiungimento di un certo numero di occorrenze di una soluzione ottima accettabile, compatibilmente con le tolleranze imposte sul tasso di variabilità degli individui, della funzione obiettivo tra una generazione e la successiva, e della violazione dei vincoli del problema [30].

L'algorithmo GA testato nel corso di questa attività è stato implementato sulla piattaforma open-source Pymoo, operante in ambiente di programmazione Python. La struttura base dell'algorithmo è mostrata in Figura 6 in forma di *flowchart*.

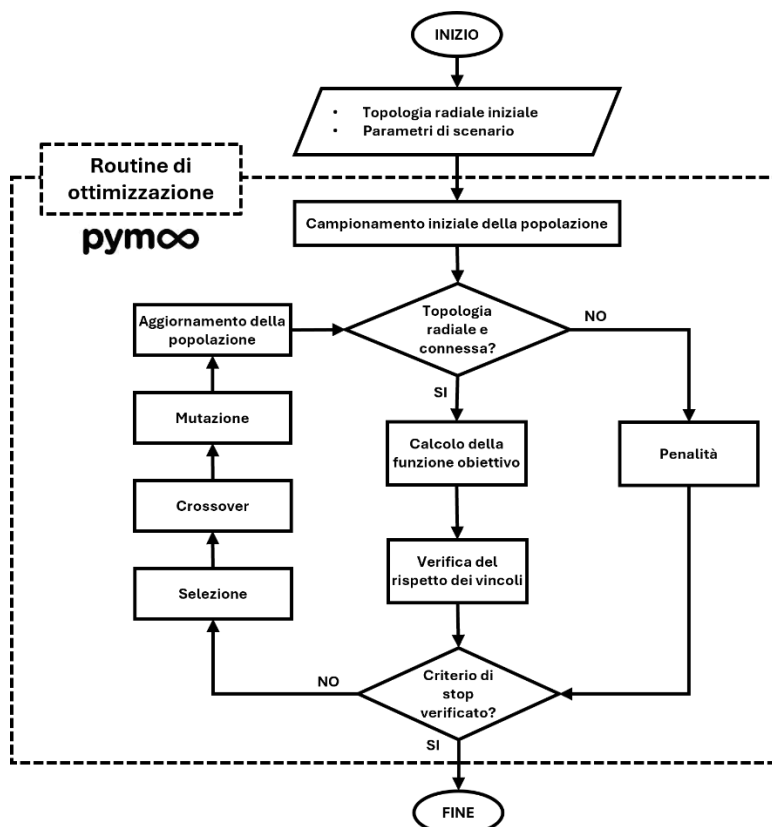


Figura 6 - Flowchart dell'algorithmo GA

11.5 Esempio di soluzione del problema ONR su una rete semplificata

A fini esplicativi, in questa sezione, il problema ONR viene risolto su una rete di prova semplificata, rappresentata in Figura 7. La soluzione permetterà anche di verificare la correttezza della formulazione matematica, e rendere più intellegibile la formulazione delle relazioni di radialità e connettività formulate in 11.2.3 e 11.2.4.

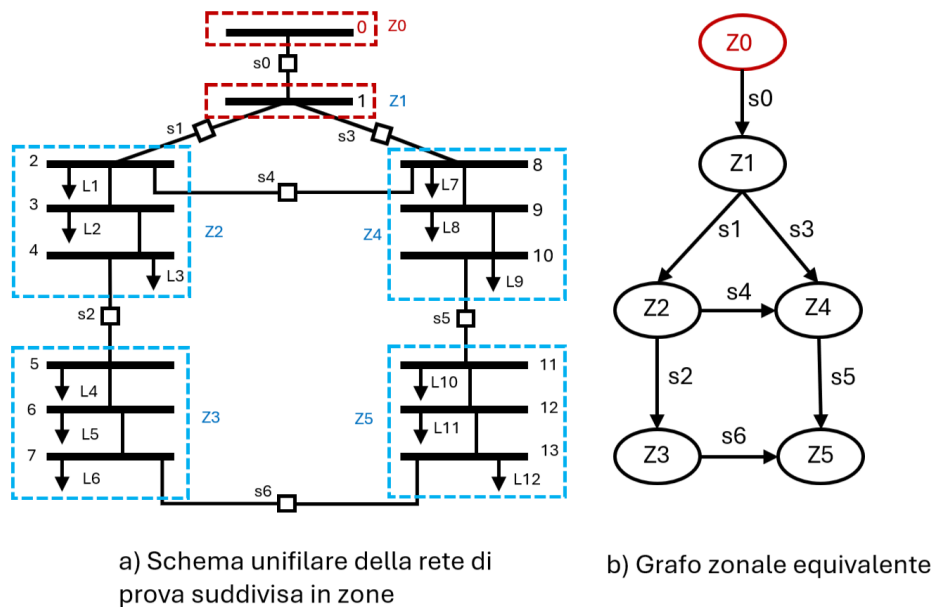


Figura 7 - Rete di prova per la verifica dei vincoli di radialità e connettività del problema ONR

La zona z_0 (in rosso nella Figura 2b) rappresenta la sorgente del sistema, che assume anche la funzione di nodo *slack*. Questa zona contiene anche lo switch s_0 che rappresenta l'unico interruttore presente nella cabina primaria. L'ipotesi semplificativa è che ci sia un solo feeder, protetto da s_0 , e che due laterali si dipartano da una cabina di smistamento rappresentata dalla zona z_1 .

Le altre zone (z_2 - z_5) rappresentano invece degli aggregati di altri elementi del sistema (cabine secondarie e linee di collegamento) nei quali sono presenti i clienti del sistema di distribuzione. Per semplicità, i carichi all'interno di ciascuna zona saranno rappresentati attraverso un carico aggregato equivalente.

Per semplicità, si è ipotizzato che tutte le zone abbiano gli stessi valori di λ_k e μ_k , e che $\lambda_k \ll \mu_k$ in accordo con l'ipotesi di alta riparabilità.

In seguito alla suddivisione zonale della rete, si riportano in Tabella 1 le potenze assorbite dai carichi equivalenti zonali (P_z^D) e il numero di clienti connessi (N_z). Inoltre, si è assunto $\lambda_z = 0.02$ e $\mu_z = 2$ per $z \in \{Z_2, Z_3, Z_4, Z_5\}$.

Assumendo che la topologia radiale iniziale sia quella in Figura 8a, l'algoritmo di ONR riconfigura la rete collocando le zone z_2 e z_3 , precedentemente connesse rispettivamente con la zona z_4 e la zona z_5 , entrambe a valle della zona z_1 (Figura 8b).

La soluzione trovata è certamente corretta giacché essa permette di associare allo switch s_1 , non utilizzato nella configurazione 3a, le funzioni di protezione per le zone z_2 e z_3 .

Lo stato iniziale (SI) e finale degli interruttori sono riportati in Tabella 2, in cui si osserva che, per conservare una topologia radiale, alla manovra di apertura degli interruttori s_4 e s_5 (1/0), corrisponde la manovra di chiusura degli interruttori s_1 e s_2 (0/1).

La condizione necessaria ma non sufficiente di radialità è rispettata giacché gli interruttori chiusi sono in numero pari al numero di zone meno uno.

Tabella 1 - Domanda di carico zonale e numero di clienti connessi

Zona	P_z^D (kW)	N_z
Z2	115	110
Z3	160	135
Z4	200	135
Z5	150	135

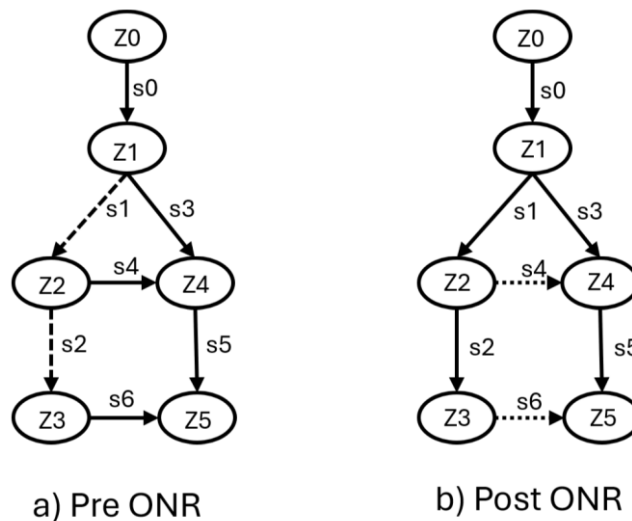


Figura 8 - Topologia della rete di prova: a) topologia radiale iniziale, b) topologia radiale ottimale

Tabella 2 - Stato degli interruttori prima e dopo l'esecuzione di ONR

	s_0	s_1	s_2	s_3	s_4	s_5	s_6
Pre ONR	1	0	0	1	1	1	1
Post ONR	1	1	1	1	0	1	0

In questo caso estremamente semplificato, la soluzione ottenuta non dipende dalla tecnica di automazione FDIR adottata. Infatti, la riconfigurazione trovata riduce sempre frequenza e/o entità dei guasti, perché permette di salvaguardare i carichi presenti sotto il laterale sinistro (protetto da s_1) dai guasti nel laterale destro (protetto da s_3), e viceversa.

Per comprendere meglio come sono determinate le variabili relative ai flussi artificiali $\varphi_{s,m}$, si prenda come esempio la zona z_2 in Figura 8a: il percorso radiale più breve da z_0 a z_2 comprende gli switch s_0, s_3, s_4 , i quali rispettivamente collegano $z_0-z_1, z_1-z_4, z_2-z_4$.

In questo modo la variabile φ_{s,z_2} assume valore pari a 1 in corrispondenza degli interruttori facenti parte del percorso più breve ($\varphi_{s_0,z_2}, \varphi_{s_3,z_2}$), mentre per l'interruttore s_4 , la variabile φ_{s_4,z_2} assume valore pari a -1, data l'orientazione dell'interruttore opposta rispetto alla direzione del flusso artificiale. Gli interruttori non facenti parti del percorso radiale più breve assumono, in corrispondenza di z_2 , un valore di φ_{s,z_2} pari a 0 ($\varphi_{s_1,z_2}, \varphi_{s_2,z_2}, \varphi_{s_5,z_2}, \varphi_{s_6,z_2}$), indipendentemente dallo stato di connessione dell'interruttore (aperto o chiuso).

I valori assunti dalle variabili di flusso artificiale $\varphi_{s,m}$ prima della riconfigurazione sono riportati in forma matriciale in Tabella 3.

Si osservi come la variabile assuma valore pari a ± 1 solamente nel caso in cui lo switch si trovi nel percorso radiale della zona in esame. Prendendo come esempio lo switch s_0 , la variabile assume sempre un valore pari ad 1 in corrispondenza delle zone z_1, z_2, z_3, z_4, z_5 , esclusa la z_0 , in quanto lo switch si trova immediatamente sotto di essa. Guardando invece lo switch s_3 , il quale connette la z_1 con la z_4 , nella topologia iniziale la variabile del flusso artificiale ($\varphi_{s_3,z_2}, \varphi_{s_3,z_3}, \varphi_{s_3,z_4}, \varphi_{s_3,z_5}$) assume valore pari a 1 in corrispondenza delle zone z_2, z_3, z_4 e z_5 , in quanto facente parte del percorso radiale di tutte e tre le zone. Qualora lo switch fosse orientato in maniera opposta (come nel caso di s_4 che connette z_2 a z_4 nella topologia iniziale radiale), rispetto alla direzione di un flusso di potenza unitaria in uscita dalla z_0 , la variabile φ_{s_4,z_2} , assume valore pari a -1 in corrispondenza della zona per la quale il medesimo switch fa parte del percorso radiale.

Tabella 3 - Valori assunti dalla variabile di flusso artificiale nella topologia radiale iniziale (a) e ottimale (b)

(a)	z0	z1	z2	z3	z4	z5
s_0	0	1	1	1	1	1
s_1	0	0	0	0	0	0
s_2	0	0	0	0	0	0
s_3	0	0	1	1	1	1
s_4	0	0	-1	0	0	0
s_5	0	0	0	1	0	1
s_6	0	0	0	-1	0	0

(b)	z0	z1	z2	z3	z4	z5
s_0	0	1	1	1	1	1
s_1	0	0	1	1	0	0
s_2	0	0	0	1	0	0
s_3	0	0	0	0	1	1
s_4	0	0	0	0	0	0
s_5	0	0	0	0	0	1
s_6	0	0	0	0	0	0

La Tabella 3b (destra) descrive i flussi artificiali relativi alla soluzione del problema ONR. La z_3 viene connessa a valle della z_2 , che a sua volta è connessa direttamente sotto la z_1 , ne consegue che lo switch s_3 , rimane parte del percorso radiale solamente per le zone z_4 e z_5 , per le quali la variabile del flusso artificiale assume valore pari a 1 ($\varphi_{s_3,z_4}, \varphi_{s_3,z_5}$).

Per quanto concerne il valore assunto dalle variabili $z_{k,m}^{up}$, osservando la Tabella 4, è evidente che per la zona z_0 tale variabile assuma sempre valore 1 rispetto alle altre zone, in quanto z_0 è la zona più a monte rispetto alle altre. Inoltre, ogni zona è assunta essere a monte rispetto a sé stessa, quindi la variabile $z_{k,m}^{up}$, con $k = m$ assume sempre valore pari a 1.

Osservando le altre zone al di sotto di z_1 , si può notare come il valore della variabile di condizione a "monte" vari dalla topologia radiale iniziale e a quella ottimale, a causa della loro differente connessione. In particolare, i valori evidenziati in rosso e verde evidenziano proprio questo tipo di cambiamento: la z_2 , inizialmente a fondo feeder, si trova al termine dell'ottimizzazione a monte della z_3 , ne consegue che z_{z_2,z_3}^{up} sia pari a 1. La z_4 , inizialmente a monte di z_2, z_5, z_3 , al termine dell'ottimizzazione si trova a monte solamente di z_5 , quindi, z_{z_4,z_5}^{up} assume valore pari a 1. Infine, z_5 si trova a termine dell'ottimizzazione a valle della z_4 , senza più essere a monte di z_3 , la quale viene connessa sotto la z_2 ; z_{z_5,z_3}^{up} passa da 1 a 0.

Tabella 4 - Valori assunti dalla variabile di condizione "a monte" nella topologia radiale iniziale (a) e ottimale (b)

(a)	Z0	Z1	Z2	Z3	Z4	Z5
Z0	1	1	1	1	1	1
Z1	0	1	1	1	1	1
Z2	0	0	1	0	0	0
Z3	0	0	0	1	0	0
Z4	0	0	0	0	1	1
Z5	0	0	0	0	0	1

(b)	Z0	Z1	Z2	Z3	Z4	Z5
Z0	1	1	1	1	1	1
Z1	0	1	1	1	1	1
Z2	0	0	1	1	0	0
Z3	0	0	0	1	0	0
Z4	0	0	0	0	1	1
Z5	0	0	0	0	0	1

I valori di EENS, SAIDI, SAIFI e la funzione obiettivo uniformemente normalizzata sono rappresentati in Figura 9. Come si può notare, l'unico indice che non subisce riduzioni è il SAIFI in corrispondenza della tecnica di automazione FRG.

La motivazione è data dal fatto che la relazione (4) tiene conto della frequenza di guasto zonale calcolata in (13). Tale valore risulta invariato anche a seguito dell'ottimizzazione, in quanto le zone z_2, z_3, z_4 e z_5 continuano ad appartenere allo stesso feeder (infatti la variabile $u_{h,k,m}$ assume valore pari a 1 per $h = z_1$ e $k, m \in \{z_1, z_2, z_3, z_4, z_5\}$).

Diversamente, nel caso delle altre due tecniche di automazione FNC e SFS, la frequenza di guasto zonale in (15) e (17) dipende dalla variabile $z_{k,m}^{up}$, rappresentante le relazioni upstream tra zone. In questi due casi, la suddivisione delle zone sui due laterali permette di separare gli eventi di guasto, che potranno essere interrotti selettivamente dagli organi di manovra corrispondenti a ciascun feeder laterale.

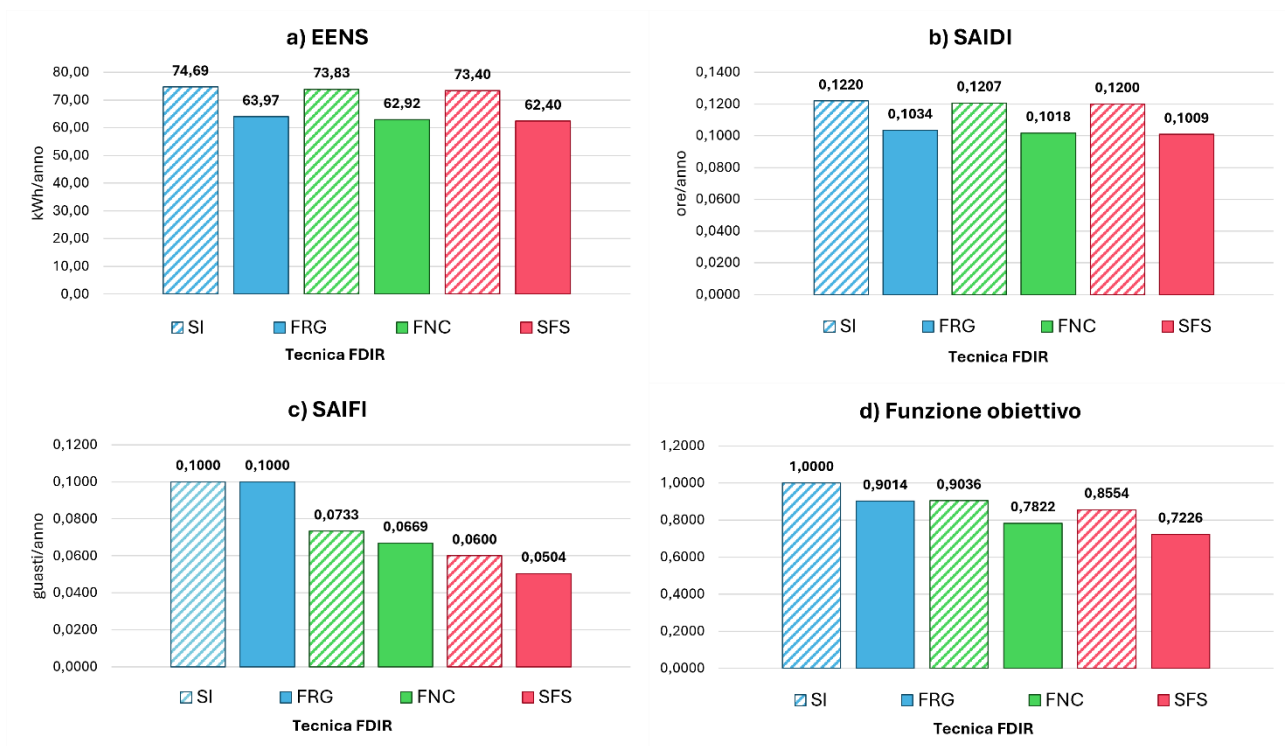


Figura 9 - Indici di qualità monitorati e funzione obiettivo normalizzata nella configurazione iniziale (SI) e nelle configurazioni ottimali

11.5.1 Confronto delle prestazioni dei due algoritmi MILP e GA

In questa sottosezione, si confrontano i risultati ottenuti risolvendo il problema ONR per il caso studio appena descritto con i due algoritmi sotto test (MILP e GA).

Dal confronto della Tabella 5 con la Tabella 6, è possibile verificare che, in questo caso estremamente semplificato, i due algoritmi hanno condotto alla medesima soluzione in tutti i casi.

Tabella 5 - Algoritmo MILP, variazione percentuale degli indicatori al variare della tecnica FDIR applicata

FDIR	$\Delta EENS$	$\Delta SAIDI$	$\Delta SAIFI$	Δf
FRG	-14,35%	-15,25%	0%	-9,86%
FNC	-14,78%	-15,66%	-8,73%	-13,44%
SFS	-14,99%	-15,92%	-16,00%	-15,52%

Tabella 6 - Algoritmo GA, variazione percentuale degli indicatori al variare della tecnica FDIR applicata

FDIR	$\Delta EENS$	$\Delta SAIDI$	$\Delta SAIFI$	Δf
FRG	-14,35%	-15,25%	0%	-9,86%
FNC	-14,78%	-15,66%	-8,73%	-13,44%
SFS	-14,99%	-15,92%	-16,00%	-15,52%

I tempi di computazione necessari per raggiungere la convergenza e la soluzione ottimale sono indicati in Tabella 7. Benché estremamente ridotti in entrambi i casi è possibile verificare che la soluzione viene raggiunta molto più rapidamente dall'algoritmo MILP.

La superiorità computazionale dell'algoritmo MILP rispetto al GA è molto più evidente nel caso di problemi ONR complessi. I risultati relativi all'applicazione dei due algoritmi su modelli di rete realistici sono presentati nel rapporto tecnico relativo alla attività LA1.11.

Tabella 7 – Prestazioni computazionali degli algoritmi

FDIR	Algoritmo MILP [s]	Algoritmo GA [s]
FRG	0.11	2.18
FNC	0.08	1.93
SFS	0.08	1.81

11.6 Architettura del software open-source per ONR

Questa sezione descrive in dettaglio le routine utilizzate per l'implementazione del problema ONR, fornendo informazioni sull'architettura software e gli ambienti di sviluppo impiegati per la realizzazione di un tool open-source.

Il software si compone di diversi elementi, tutti basati su software e applicativi open-source. Per la rappresentazione della rete e lo scambio di dati è stato utilizzato OpenDSS (software di simulazione delle reti di distribuzione), mentre l'intero codice è stato sviluppato utilizzando il linguaggio Python e alcune librerie open-source.

Il problema di ottimizzazione ONR è stato formulato attraverso la libreria Pyomo, che supporta l'uso di risolutori esterni, che siano prodotti commerciali con licenza oppure open-source. Lo schema generale dell'architettura software è riportato in Figura 10, dove è possibile distinguere diverse subroutine, il cui dettaglio è riportato nel seguito.

architettura

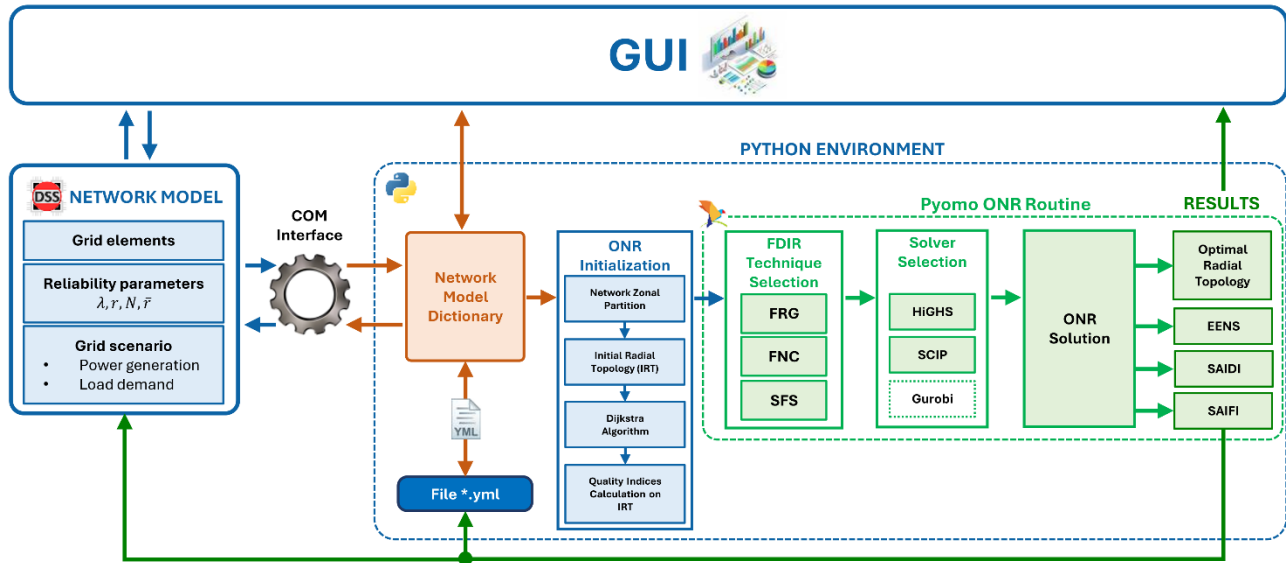


Figura 10 - Architettura dello strumento software open-source per la risoluzione di ONR

11.6.1 Parsing dei dati

La prima parte del tool assicura l'importazione/esportazione dei dati di rete. Per garantire un accesso funzionale alle caratteristiche e alle proprietà dei modelli di rete OpenDSS, è stata sviluppata in linguaggio Python una routine personalizzata per la gestione dei modelli di rete.

Lo scopo della routine è quello di gestire le proprietà dei componenti della rete elettrica da studiare e di offrire un accesso sistematico alle funzionalità di calcolo di OpenDSS, all'interno di un'interfaccia grafica dedicata (Graphic User Interface, GUI).

Attraverso una procedura assistita, è possibile importare modelli DSS personalizzati, richiamare modelli di benchmark predefiniti o definire una nuova rete con la propria topologia, i suoi componenti e i rispettivi parametri. Tutti i parametri degli elementi sono accessibili nella GUI e possono essere modificati in base alle esigenze dell'utente, oltre che esportati per scopi futuri e ulteriori analisi.

Questa routine è stata sviluppata per supportare tutti gli strumenti software per la valutazione dell'affidabilità, resilienza, sicurezza e adeguatezza precedentemente sviluppati da ENEA, compresa la ONR. Tutte le proprietà degli elementi (categoria, dati topologici, parametri e risultati di studio) vengono memorizzati in un dizionario dedicato nell'ambiente Python, al fine di ottenere un accesso rapido a tutte le caratteristiche del modello di rete.

Il dizionario può essere esportato come file YAML (*.yaml) e importato in sessioni future per richiamare il modello di rete e il caso di studio precedente.

11.6.2 Algoritmo di suddivisione zonale

Per poter formulare e risolvere il problema ONR, è stato necessario sviluppare alcune routine di inizializzazione.

In una prima fase, i dati di rete sono stati estratti dal modello OpenDSS e importati nell'apposito dizionario. La struttura della rete è nota attraverso le informazioni sui nodi (bus), rami e interruttori.

Come descritto nella sezione precedente, la metodologia su cui si fonda la ONR impone che la rete di distribuzione sia partizionata in zone, dove ciascuna zona contiene un insieme di nodi e rami ed è connessa ad altre zone per mezzo di interruttori. Il modello di dati OpenDSS non contiene informazioni relative ad una ripartizione in zone. Pertanto, è stato sviluppato un codice che consente di creare automaticamente il grafo zonale dal modello di rete.

L'algoritmo implementato per effettuare automaticamente la suddivisione in zone è riportato nello pseudo-codice in Figura 11.

```

Set  $b \leftarrow 1, n \leftarrow 1, z \leftarrow 1, \Omega_z \leftarrow \emptyset$ 
Part 1: Zones creation
while  $b \leq$  number of branches do
  select branch  $b$ 
  if branch  $b$  is not a switch then
    Set  $i \leftarrow$  bus-sending,  $j \leftarrow$  bus-receiving of branch  $b$ 
    if  $i, j$  are already assigned respectively to zone  $z_i$  and  $z_j$  then
      if  $z_i \neq z_j$  then
        move all buses in  $z_j$  into  $z_i$ 
        empty  $z_j$ 
      end if
    else if only  $i$  is assigned to a zone then
      assign  $j$  to the same zone as  $i$ 
    else if only  $j$  is assigned to a zone then
      assign  $i$  to the same zone as  $j$ 
    else if both  $i, j$  are not assigned to a zone then
      define a new zone  $z$  in  $\Omega_z$ , assign  $i$  and  $j$  to  $z$ 
    end if
  end if
   $b \leftarrow b + 1$ 
end while
Part 2: Control of nodes not assigned to zones
while  $n \leq$  number of nodes do
  if node  $n$  is not assigned to any zone then
    define a new zone  $z$  in  $\Omega_z$  and assign node  $n_{th}$  to  $z$ 
  end if
   $n \leftarrow n + 1$ 
end while
Part 3: Control of zones left empty
while  $z \leq$  number of zones do
  if zone  $z$  is empty then
    remove  $z$  from  $\Omega_z$ 
  end if
   $z \leftarrow z + 1$ 
end while
renumber all zones left in  $\Omega_z$ 

```

Figura 11 - Algoritmo di suddivisione zonale della rete di distribuzione da riconfigurare

L'algoritmo si basa sulla struttura complessiva del sistema elettrico e non richiede la conoscenza degli stati (aperto/chiuso) degli organi di manovra.

Le zone vengono create operando un ciclo *for* su tutti i branch della rete, e tramite semplici istruzioni *if*. L'algoritmo si basa sul concetto secondo cui ogni ramo (e i relativi nodi di partenza e arrivo) debba essere associato a una singola zona. Pertanto, gli interruttori sono gli unici componenti della rete che non appartengono a una zona, ma ne collegano due.

Per ogni elemento branch di rete l'algoritmo verifica se l'elemento è collegato ad altri elementi di rete già assegnati ad una zona. Se l'elemento è collegato ad una sola zona già esistente, allora viene associato a quella zona. Se l'elemento è collegato a due zone, viene creata una nuova zona che contiene tutti gli elementi delle due zone, e l'elemento è associato a quest'ultima. Se l'elemento non è connesso ad alcuna zona, allora viene creata una nuova zona alla quale viene associato l'elemento.

Nella Parte 2, vengono individuati potenziali nodi isolati che non sono stati assegnati ad alcuna zona e una nuova zona viene definita per ciascuno di essi.

Infine, nella Parte 3, vengono eliminate eventuali zone vuote. Le informazioni sulla zona vengono infine salvate come proprietà nel dizionario degli elementi.

11.6.3 Inizializzazione del problema ONR

La topologia iniziale della rete è nota attraverso la posizione di ogni interruttore (aperto/chiuso). Le posizioni degli interruttori sono importate da OpenDSS durante il parsing dei dati iniziali, ma, in ogni caso, il tool sviluppato consente anche di generare una configurazione radiale casuale qualora la rete da riconfigurare fosse inizialmente magliata.

La routine di inizializzazione controlla innanzitutto se il modello di rete importato è inizialmente radiale e connesso, successivamente verifica se il grafo zonale equivalente è anch'esso radiale. Si noti che, dato il tipo di tecniche di automazione FDIR, la radialità del grafo zonale è un requisito, mentre le maglie all'interno di una zona non influiscono sull'analisi degli indici di qualità del servizio.

A questo punto, i valori iniziali di EENS, SAIDI e SAIFI possono essere calcolati risolvendo il problema ONR dopo aver fissato le variabili binarie x_s per tutti gli interruttori e ottenuto i valori zonal di λ_z e U_z relativi alla topologia radiale iniziale. Tuttavia, si è optato per un approccio alternativo che evita di eseguire uno stadio preliminare di ottimizzazione.

Esso consiste nell'applicare un algoritmo di Dijkstra sulla topologia radiale iniziale, al termine del quale risultano note tutte le relazioni di connessione tra le zone, necessarie per il calcolo dei valori iniziali degli indici di qualità. L'algoritmo di Dijkstra consente quindi di verificare la connettività e la radialità della rete ed è implementato all'interno della libreria open-source NetworkX.

Questa routine consente anche di normalizzare i pesi della equazione (1). Assumendo di applicare la tecnica FRG, che è sempre caratterizzata da peggiori performance in termini di qualità del servizio rispetto a FNC e SFS, è possibile considerare come base per il calcolo in (1) i valori degli indici di qualità assunti nella configurazione iniziale.

Supponendo che i valori calcolati per la topologia iniziale siano $EENS_0$, $SAIDI_0$, $SAIFI_0$, allora è possibile definire:

$$w_1 = \frac{\alpha}{EENS_0}, \quad w_2 = \frac{\beta}{SAIDI_0}, \quad w_3 = \frac{\gamma}{SAIFI_0} \quad (19)$$

$$\alpha + \beta + \gamma = 1 \quad (20)$$

Ipotizzando che gli indici abbiano uno stesso peso, è possibile assegnare ai parametri α , β e γ il valore di 1/3.

La normalizzazione di (1) permette di confrontare facilmente gli impatti che l'algoritmo di ONR comporta in termini di affidabilità del sistema al variare della tecnica di automazione impiegata per riconfigurare la topologia della rete.

11.6.4 Risoluzione del problema ONR

La routine ONR è stata sviluppata sulla piattaforma Pyomo, una libreria open-source di Python che consente la definizione, la risoluzione e l'analisi di modelli di ottimizzazione. La sua funzione principale è quella di tradurre la formulazione matematica del problema di ottimizzazione nel linguaggio di programmazione utilizzato dai risolutori commerciali.

I principali vantaggi risiedono nella semplicità di implementazione del modello, in quanto Pyomo si basa sulla programmazione orientata agli oggetti (Object Oriented Programming, OOP), dove il problema di ottimizzazione è costruito come una classe i cui oggetti sono le variabili, i vincoli e le funzioni obiettivo. Questa caratteristica favorisce il riutilizzo del codice, la modularità e la manutenibilità, consentendo agli utenti di utilizzare diversi risolutori per la

risoluzione dei loro modelli senza la necessità di cambiare la sintassi utilizzata per scrivere lo script.

Questa interoperabilità si rivela essenziale quando è necessario testare lo stesso modello utilizzando diverse tecniche risolutive, potendo così utilizzare il solver che si ritiene più adatto per il tipo di problema che si intende risolvere.

11.6.5 Selezione del risolutore

Nonostante la libreria Pyomo consenta di implementare il problema di ottimizzazione utilizzando diversi risolutori, solamente pochi possono essere effettivamente utilizzati, poiché ognuno di essi può affrontare solo formulazioni specifiche. I due solver open-source individuati in grado di risolvere in modo efficiente il problema ONR proposto sono HiGHS e SCIP, entrambi capaci di risolvere modelli MILP e disponibili in *amplpy*, una libreria che consente di accedere ad AMPL (A Mathematical Programming Language), un tool che collega diversi solver open-source all'ambiente Python per risolvere problemi di ottimizzazione. In questo caso, *amplpy* è stato utilizzato solo come utility per accedere a due dei risolutori open-source di AMPL disponibili come pacchetti Python.

Va notato che, per confrontare le prestazioni di HiGHS e SCIP, il tool di ONR sviluppato è stato programmato per trovare soluzioni anche con il solver commerciale Gurobi 11.0, che non è open-source, ma risulta utilizzabile grazie a licenze accademiche gratuite, disponibili e rinnovabili annualmente.

11.6.6 Navigazione del tool e Interfaccia grafica utente (Graphic User Interface, GUI)

In questa sezione, si fornisce una descrizione degli step che compongono il tool di analisi e ottimizzazione ONR, e della interfaccia grafica utilizzata per presentare i risultati all'utente.

I risultati dell'ottimizzazione possono essere infatti facilmente letti e confrontati con la soluzione iniziale attraverso una GUI, in cui vengono rappresentati i grafi ottenuti tramite le librerie open-source NetworkX e Graphviz (Figura 12, Figura 13). Sempre all'interno della GUI sono disponibili altre tipologie di grafici, come ad esempio *barplot* utili per confrontare gli indici di qualità monitorati e la funzione obiettivo rispetto alla topologia iniziale e al variare della tecnica di *Distribution Automation* utilizzata (Figura 14).

Di seguito, sono elencate i passaggi che costituiscono la routine di ONR e le informazioni testuali o grafiche che la GUI permette di visualizzare a schermo:

❖ **Lettura e importazione del modello di rete OpenDSS all'interno del tool:**

- Informazioni topologiche (numero di nodi, linee, trasformatori, generatori, carichi, relazioni di connessione, matrici di adiacenza e di incidenza, ecc.);
- Informazioni di scenario (potenze generate/assorbite, tensioni, transiti di potenza e correnti nei rami, perdite).

❖ **Algoritmo di partizionamento zonale:**

- Numero complessivo di zone in cui viene suddivisa la rete al termine dell'algoritmo di suddivisione zonale (output testuale);
- Creazione del grafo equivalente zonale (output grafico).

❖ **Ottenimento di una topologia radiale iniziale:**

(questo step viene eseguito automaticamente solo se la rete importata da OpenDSS è inizialmente magliata)

- Operazioni di smagliatura della topologia iniziale della rete finalizzate all'ottenimento di un grafo radiale e connesso;
- Numero di nodi (zone) e linee (archi) di cui è composto il grafo nodale (zonale)(output testuale);
- Controllo della connettività del grafo nodale e del grafo zonale equivalente (output testuale);
- Rappresentazione nodale e zonale della rete nella configurazione radiale iniziale (output grafico).

❖ **Calcolo degli indici di reliability sulla topologia radiale iniziale:**

- Indici di affidabilità (EENS, SAIDI, SAIFI) e funzione obiettivo calcolati sulla topologia radiale iniziale (output testuale e grafico; i valori sono forniti sia in valore assoluto che normalizzato);

❖ **Esecuzione routine iniziale di Load Flow:**

(questo step viene eseguito dopo aver impostato in OpenDSS lo stato degli interruttori corrispondente alla topologia iniziale radiale)

- Potenza attiva e reattiva erogate/assorbite della slack bus equivalente (output testuale);
- Totale potenza attiva e reattiva assorbite dei carichi equivalente (output testuale);
- Totale potenza attiva e reattiva erogate dai generatori equivalente (output testuale);
- Perdite totali di potenza attiva e reattiva della rete equivalente (output testuale);
- Rilevamento di eventuali sovraccarichi di linee e trasformatori (output testuale: nome dell'elemento di ramo interessato da sovraccarico e relativa entità in termini percentuali)
- Rilevamento di eventuali violazioni di tensione sui nodi (output testuale: nome del nodo interessato da violazione e relativa entità in p.u.);
- Profilo di tensione su tutti i nodi di rete (output grafico);
- Transiti di potenza apparente sui rami in sovraccarico finalizzato a quantificare il superamento della portata (output grafico).

❖ **Esecuzione routine di ONR:**

- Selezione della tecnica di automazione da implementare (FRG, FNC, SFS) mediante scelta (input) da parte dell'utente (output testuale);
- Indici di affidabilità e della funzione obiettivo ottimizzati e confronto rispetto ai valori iniziali (output testuale);
- Transizioni degli interruttori che hanno cambiato stato di connessione rispetto alla condizione iniziale (output testuale);
- Numero totale di aperture e chiusure degli interruttori, numero totale di rami che risultano chiusi e aperti al termine dell'ottimizzazione (output testuale);
- Numero di zone e archi di cui è composto il grafo zonale al termine dell'ottimizzazione, controllo della connettività della rete in seguito alle manovre di riconfigurazione (output testuale);
- Rappresentazione nodale e zonale della rete nella configurazione radiale ottimale (output grafico);

❖ **Esecuzione routine finale di Load Flow** (questo step viene eseguito impostando in OpenDSS lo stato degli interruttori corrispondente alla topologia radiale ottimale).

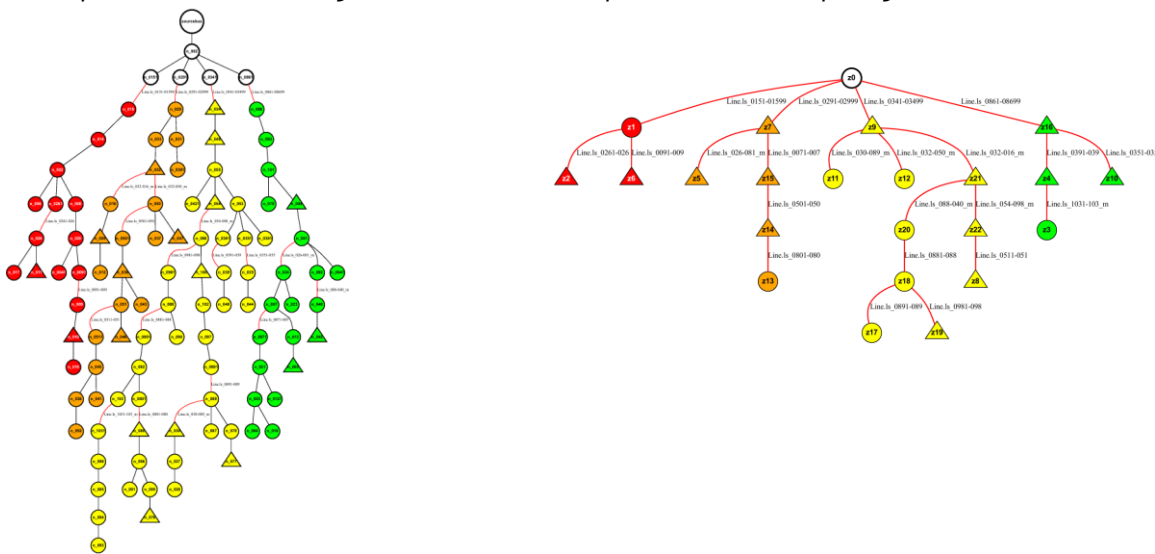


Figura 12 - Esempio di GUI: corrispondenza tra grafo nodale (a sinistra) e grafo zonale equivalente (a destra)

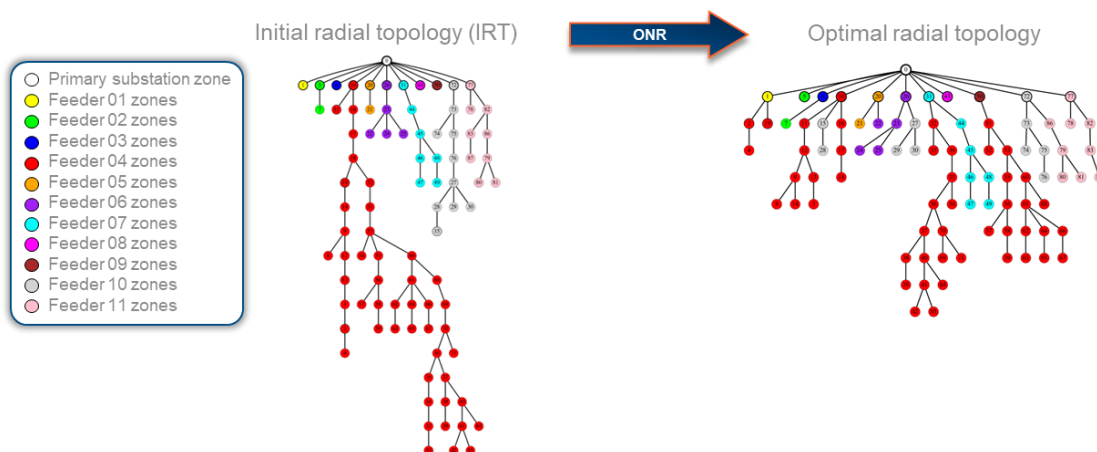


Figura 13 - Esempio di GUI: grafi zonali equivalenti della rete di distribuzione prima e dopo la ONR

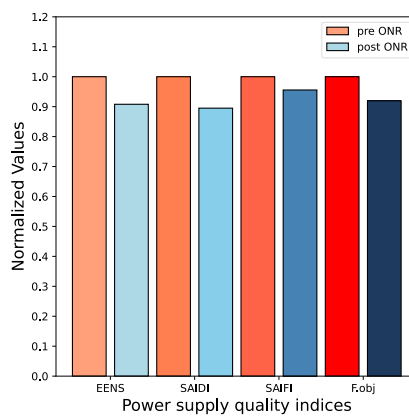


Figura 14 - Esempio di GUI: indici di qualità monitorati normalizzati e funzione obiettivo

12 Software e librerie open-source

Oltre a un interprete Python (Spyder, PyCharm, ecc.) e alle librerie convenzionalmente utilizzate per la gestione dei dati (*math*, *numpy*, *pandas*, *collections*) e la visualizzazione di grafici (*matplotlib*), il tool di ONR richiede l'installazione dei seguenti pacchetti software:

1) OpenDSS

Software open-source per l'analisi dei sistemi di distribuzione elettrica.

Versione: 9.6.1.1

Download: <https://sourceforge.net/projects/electricdss/>

Documentazione: <https://www.epri.com/pages/sa/openss>



N.B.: Per poter utilizzare OpenDSS, è necessario, oltre il download del software sul proprio computer, anche l'importazione della libreria *win32com.client* mediante comando `import win32com.client`. Questo permette di accedere al modulo OpenDSS COM, la cui interfaccia permette la comunicazione del software da ambiente esterno. La comunicazione viene avviata mediante l'istanza:
`dssObj = win32com.client.Dispatch("OpenDSSEngine.DSS")`

2) NetworkX

Libreria Python open-source per la creazione, la manipolazione e l'analisi di grafi e reti complesse.

Versione: 3.2.1

Download: `pip install networkx`

Documentazione: <https://networkx.org/>



3) Graphviz

Software e libreria Python open-source per la visualizzazione e il rendering di grafi.

Versione: 0.20.1 (libreria Python), 12.2.1 (applicazione esterna)

Download: `pip install graphviz` (libreria Python), <https://graphviz.org/download/> (versione esterna)

Documentazione: <https://graphviz.org/documentation/>



N.B.: per poter utilizzare Graphviz, oltre l'installazione della libreria in ambiente Python, mediante il comando `pip install graphviz`, è richiesto il download dell'applicazione da installare direttamente sul computer. Una volta scaricata e installata, la cartella deve essere aggiunta al percorso di sistema, o mediante la definizione di una nuova variabile di ambiente, o direttamente mediante l'aggiunta del percorso nello script Python.

4) Pyomo

Libreria Python open-source utilizzata per la modellazione matematica e l'ottimizzazione

Versione: 6.7.3

Download: `pip install pyomo`

Documentazione: <https://www.pyomo.org/documentation>



5) Amplpy

Libreria Python open-source che fornisce un'interfaccia per utilizzare AMPL (A Mathematical Programming Language), un potente ambiente per la modellazione e la risoluzione di problemi di ottimizzazione matematica.

Versione: 0.14.0

Download: `pip install amplpy`

Documentazione: <https://amplpy.ampl.com/en/latest/>



N.B.: una volta installata la libreria, i solver disponibili possono essere scaricati digitando `amplpy.modules.install('solver da installare')` nella console dell'interprete Python. Digitando `amplpy.modules.installed()`, si possono controllare i solver installati.

6) Gurobi



Solver su licenza che permette di risolvere problemi di ottimizzazione complessi.

Versione: 11.0.1

Download: <https://www.gurobi.com/downloads/gurobi-software/>

Documentazione: <https://docs.gurobi.com/current/>

Per un'esecuzione senza errori dello script, nel seguente listato sono riportate tutte le librerie Python da importare:

```
import platform
import os
import sys
from math import pi, sqrt, cos, sin, tan, atan2, exp, acos, isnan, log
import numpy as np
import pandas as pd
import plotly.graph_objects as go
import networkx as nx
import graphviz
from pyomo.environ import *
import random
from timeit import default_timer as timer
from itertools import chain
import copy
from itertools import combinations
from collections import defaultdict
from pprint import pprint
import re
import warnings
import time
import amplpy
import highspy
from amplpy import modules
import matplotlib.pyplot as plt
from matplotlib.font_manager import FontProperties
import matplotlib.gridspec as gridspec
from matplotlib import rc
from matplotlib.lines import Line2D
from matplotlib.ticker import ScalarFormatter, FuncFormatter
import matplotlib.patches as mpatches
import matplotlib.colors as colors
import win32com.client
```

Figura 15 - Script Python per l'importazione delle librerie necessarie per l'esecuzione di ONR

13 Riferimenti bibliografici

- [1] F. Bignucolo, R. Caldon, M. Berton, and L. Michielutti, "Fault isolation in overhead distribution networks: New developments in outdoor fault detectors," in *CIGRE 2009 - 20th International Conference and Exhibition on Electricity Distribution - Part 1*, 2009, pp. 1-4.
- [2] A. Cerretti, G. Scrosati, and L. Consiglio, "Upgrade of ENEL MV network automation to improve performances in presence of faults and to deal DG," in *21th International Conference on Electricity Distribution, CIGRE2011*, 2011, pp. 6-9.

- [3] A. Bosisio, A. Berizzi, D. Lupis, A. Morotti, B. Greco, and G. Iannarelli, "Network automation planning in distribution networks: A feeders coupling method to implement IEC 61850-based smart automation system logic," in *2021 AEIT International Annual Conference (AEIT)*. IEEE, 2021, pp. 1–6.
- [4] H. Bentarzi, A. Ouadi, and A. Abdelmoumene, "A new framework of smart auto-recloser," in *2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018)*. IEEE, 2018, pp. 1–5.
- [5] M. A. F. Boaski, C. dos Santos, M. Sperandio, D. P. Bernardon, M. J. Ramos, and D. S. Porto, "Coordination and selectivity of protection devices with reliability assessment in distribution systems," *System Reliability*, 2017.
- [6] C. D'adamo, G. Valtorta, L. Consiglio, A. Cerretti, L. D'orazio, A. Malerba, and F. Marmeggi, "Smart fault selection: New operational criteria and challenges for the large-scale deployment in e distribuzione's network," *CIREN - Open Access Proceedings Journal*, vol. 2017, no. 1, pp. 1475–1478, 2017.
- [7] S. Bruno, S. Lamonaca, M. La Scala, and U. Stecchi, "Integration of optimal reconfiguration tools in advanced distribution management system," in *2012 3rd IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*. IEEE, 2012, pp. 1–8.
- [8] N. C. Koutsoukis, D. O. Siagkas, P. S. Georgilakis, and N. D. Hatziargyriou, "Online reconfiguration of active distribution networks for maximum integration of distributed generation," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 437–448, 2016.
- [9] N. N. Mansor and V. Levi, "Operational planning of distribution networks based on utility planning concepts," *IEEE Transactions on Power Systems*, vol. 34, no. 3, pp. 2114–2127, 2019.
- [10] S. Mishra, D. Das et al., "A comprehensive review on power distribution network reconfiguration," *Energy Systems*, vol. 8, pp. 227–284, 2017.
- [11] R. A. Jabr, R. Singh, and B. C. Pal, "Minimum loss network reconfiguration using mixed-integer convex programming," *IEEE Transactions on Power Systems*, vol. 27, no. 2, pp. 1106–1115, 2012.
- [12] P. A. Karafotis, I. K. Bazionis, and P. S. Georgilakis, "Optimal distribution network reconfiguration for reliability improvement in presence of DG," in *The 12th Mediterranean Conference on Power Generation, Transmission, Distribution and Energy Conversion (MEDPOWER 2020)*, vol. 2020. IET, 2020, pp. 354–360.
- [13] S. Bruno and M. La Scala, "A MINLP Approach for Network Reconfiguration and Dispatch in Distribution Systems," *Computational Intelligence Applications in Smart Grids: Enabling Methodologies for Proactive and Self-organizing Power Systems*, p. 51, 2014.

- [14] A. M. Helmi, R. Carli, M. Dotoli, and H. S. Ramadan, "Efficient and sustainable reconfiguration of distribution networks via metaheuristic optimization," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 1, pp. 82–98, 2021.
- [15] B. Sultana, M. Mustafa, U. Sultana, and A. R. Bhatti, "Review on reliability improvement and power loss reduction in distribution system via network reconfiguration," *Renewable and Sustainable Energy Reviews*, vol. 66, pp. 297–310, 2016.
- [16] B. Khorshid-Ghazani, H. Seyedi, B. Mohammadi-Ivatloo, K. Zare, and S. Shargh, "Reconfiguration of distribution networks considering coordination of the protective devices," *IET Generation, Transmission & Distribution*, vol. 11, no. 1, pp. 82–92, 2017.
- [17] N. C. Koutsoukis, D. O. Siagkas *et al.*, "Online reconfiguration of active distribution networks for maximum integration of distributed generation," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 437–448, 2016.
- [18] S. Lei, C. Chen *et al.*, "Radiality constraints for resilient reconfiguration of distribution systems: Formulation and application to microgrid formation," *IEEE Transactions on Smart Grid*, vol. 11, no. 5, pp. 3944–3956, 2020.
- [19] L. A. Gallego, J. M. López-Lezama *et al.*, "A mixed-integer linear programming model for simultaneous optimal reconfiguration and optimal placement of capacitor banks in distribution networks," *IEEE Access*, vol. 10, pp. 52655–52673, 2022.
- [20] R. Srinivasa Rao, S. V. L. Narasimham *et al.*, "Optimal network reconfiguration of large-scale distribution system using harmony search algorithm," *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 1080–1088, 2011.
- [21] H. Hahmadi, "Distribution network reconfiguration using differential evolution algorithm," [Online]. Available: <https://github.com/Hahmadi1842/Distribution-Network-Reconfiguration-DE-Algorithm> (ultimo accesso: 2024-07-15).
- [22] Areti S.p.A., "Dispositivi di selezione automatica del tronco guasto." [Online]. Available: https://www.aretis.it/content/dam/acea-aretis/documenti/area-servizi/gestione-punto-di-fornitura/caratteristiche-rete-alta-media-tensione/selezione_automatica_tronco_guasto.pdf
- [23] A. Cerretti, G. Di Lembo, G. Di Primio, A. Gallerani, and G. Valtorta, "Automatic fault clearing on mv networks with neutral point connected to ground through impedance," in *Proceedings of the CIGRE 17th International Conference on Large Electricity Distribution*, Barcelona, Spain, 2003, pp. 12–15.
- [24] E-Distribuzione, "La qualità del servizio elettrico," 2016, capitolo 10. [Online]. Available: <https://bari.ordinequadrocloud.it/ISFormazione-Bari/uploads?f=fotoutenti/bari/E-Distribuzione26102016.pdf>
- [25] L. Delli Carpini, A. Cammarota, G. Sapienza, and P. Paulon, "Smart Fault Selection through smart protection devices using IEC61850," 2019.

- [26] ComEsto, "Definizione di criteri per l'automazione evoluta per unità periferiche di cabina secondaria nella rete di e-distribuzione rapporto tecnico di ricerca industriale d6.3a," 2020. [Online]. Available: <http://www.comesto.eu/wp-content/uploads/2022/12/ComESto-D6.3a-Definizione-di-criteri-per-l%E2%80%99Automazione-Evoluta.pdf>
- [27] Z. Michalewicz, "Genetic algorithms + Data Structures = Evolution Programs". Springer Science & Business Media, 2013.
- [28] A.J. Umbarkar and P.D. Sheth, "Crossover operators in genetic algorithms: A review", ICTAT journal on soft computing, October 2015, Volume 06, Issue 01.
- [29] Francisco Chicano, Andrew M Sutton, L Darrell Whitley, Enrique Alba, "Fitness Probability Distribution of Bit-Flip Mutation", Evolutionary Computation, September 2013, doi: 10.1162/EVCO_a_00130.
- [30] J. Blank and K. Deb, "Pymoo: Multi-objective optimization in python," IEEE Access, vol. 8, pp. 89497-89509, 2020.