Power Grid Partitioning and Monitoring Methods for Improving Resilience

Shuchismita Biswas

Dissertation submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Virgilio A. Centeno, Chair
Chen-Ching Liu
Walid Saad
Srijan Sengupta
Seemita Pal

July 30, 2021
Blacksburg, Virginia

Keywords: Power systems resilience, proactive islanding, microgrids, energy adequacy, transmission line switching, PMU data quality, temporary faults, network radiality

Copyright 2021, Shuchismita Biswas
Power Grid Partitioning and Monitoring Methods for Improving Resilience

Shuchismita Biswas

(ABSTRACT)

This dissertation aims to develop decision-making tools that aid power grid operators in mitigating extreme events. Two distinct areas are focused on: a) improving grid performance after a severe disturbance, and b) enhancing grid monitoring to facilitate timely preventive actions. The first part of the dissertation presents a proactive islanding strategy to split the bulk power transmission system into smaller self-adequate islands in order to arrest the propagation of cascading failures after an event. Heuristic methods are proposed to determine in what sequence should the island boundary lines be disconnected such that there are no operation constraint violations. The idea of optimal partitioning is further extended to the distribution network. A planning problem for determining which parts of the existing distribution grid can be converted to microgrids is formulated. This partitioning formulation addresses safety limits, uncertainties in load and generation, availability of grid-forming units, and topology constraints such as maintaining network radiality. Microgrids help maintain energy supply to critical loads during grid outages, thereby improving resilience. The second part of the dissertation focuses on wide-area monitoring using Phasor Measurement Unit (PMU) data. Strategies for data imputation and prediction exploiting the spatio-temporal correlation in PMU measurements are outlined. A deep-learning-based methodology for identifying the location of temporary power systems faults is also illustrated. As severe weather events become more frequent, and the threats from coordinated cyber intrusions increase, formulating strategies to reduce the impact of such events on the power grid becomes important; and the approaches outlined in this work can find application in this context.
The modern power grid faces multiple threats, including extreme-weather events, solar storms and potential cyber-physical attacks. Towards the larger goal of enhancing power systems resilience, this dissertation develops strategies to mitigate the impact of such extreme events. The proposed schemes broadly aim to: a) improve grid performance in the immediate aftermath of a disruptive event, and b) enhance grid monitoring to identify precursors of impending failures. To improve grid performance after a disruption, we propose a proactive islanding strategy for the bulk power grid, aimed at arresting the propagation of cascading failures. For the distribution network, a mixed integer linear program is formulated for identifying optimal sub-networks with load and distributed generators that may be retrofitted to operate as self-adequate microgrids, if supply from the bulk power systems is lost. To address the question of enhanced monitoring, we develop model-agnostic, computationally efficient recovery algorithms for archived and streamed data from Phasor Measurement Units (PMU) with data drops and additive noise. PMUs are highly precise sensors that provide high-resolution insight into grid dynamics. We also illustrate an application where PMU data is used to identify the location of temporary line faults.
Dedication

To the ones who said you must go on, when nothing seemed to make sense.
And to the ones who paved my path, whose shoulders I stand on.
Acknowledgments

I wish to convey my heartfelt gratitude to Dr. Virgilio Centeno, whose encouragement has been instrumental in sustaining me throughout my graduate studies. Without his guidance and inspiration, this dissertation would not have materialized, and I perhaps would not have pursued a doctoral degree. I am especially grateful that he fostered an environment where work was not all-consuming, and I was free to pursue ideas that I found interesting. I also thank my Ph.D. committee members Dr. Chen-Ching Liu, Dr. Seemita Pal, Dr. Srijan Sengupta, and Dr. Walid Saad for their valuable inputs and research advice that helped add diverse perspectives to this dissertation.

I am very grateful to Dr. Centeno, Dr. Liu, Dr. Jaime De La Ree and Dr. Vassilis Kekatos for the guidance I have received as a graduate student and the nurturing ambiance I have come to enjoy in the Power Lab. Their courses not only enhanced my knowledge about power systems, but also sharpened my critical thinking skills. I sincerely appreciate the research guidance and career advice I have received over the years from Dr. Seemita Pal, Dr. Xiaoyuan Fan, Dr. James Follum, Dr. Marcelo Elizondo, and Dr. Siddharth Sridhar. Their mentorship helped me realize what kind of work I would like to pursue in the future.

The foundation of some works presented in this dissertation was laid during my internship at the Advanced Analytics and Surveillance department of PJM Interconnection. For this I must thank Dr. Emanuel Bernabeu and David Picarelli. Discussions with them greatly enhanced my understanding of how power markets work, and how ISOs approach the implementation of resilience improvement measures.

I am indebted to Manish, a constant cheerleader, whose insights were instrumental in formulating many of my research questions as well as their solutions. I am thankful for my friends Rounak, Tapas, Sherin, and Sangeetha, who have provided encouragement and critique to improve my work. Inputs from Shantanab and Chiranjib have also helped in the execution of some of my ideas.

A chunk of my time in graduate school was spent holed up in my apartment at 810 Cascade Ct, Blacksburg due to the coronavirus pandemic, and surviving the constant waves of panic, grief, and loss would not have been possible without the close-knit coterie of Ranit, Arit, Tuntai, Esha, Anindya, Poorna Di and Sreeyaa Di. Our dinners, adda, and rituals have kept isolation at bay, and the idea of home alive in our self-imposed exiles. Even before the pandemic, despite the geographical distance, Blacksburg had never felt far from home, because of people like Santa, Prasenjit da, GB, Pal, Srijan Da, Swarnali Di, Gupta Da, Niharika, Shreya Di, and Abhijit Da. There is a saying that the company you keep influences who you
become, and I firmly believe that the people in Blacksburg I have spent the bulk of the last five years with have greatly impacted who I am today.

I cherish my friends Sarika, Pooja, Vasu, Nikhil and Daksh, whose milestones I have missed from a different continent, but who nonetheless remain my steadfast constants. Without the help of Souransu and Saptarshi, navigating the tricky labyrinths of graduate school applications would have been far trickier. And without my friend Shantanab, I perhaps would have been a more lost, more anxious, more unsure, less receptive person.

I am thankful for the support of my family that enabled me to set sail to distant lands in pursuit of my goals. My sister Tuntai props me up, with her timely pots of tea made my marathon writing sessions bearable, and as much as I hate to admit it, I am glad she ended up in Blacksburg.
Contents

List of Figures xi

List of Tables xiv

1 Introduction 1

1.1 Infrastructure Resilience 2

1.1.1 Resilience Metrics 2

1.1.2 Grid Partitioning Schemes 3

1.2 Data-based Grid Monitoring 4

1.3 Organization 6

2 Proactive Islanding of the Power Grid to Mitigate High-Impact Low-Frequency Events 9

2.1 Introduction 9

2.2 Methodology 10

2.2.1 Graph Construction 10

2.2.2 Hierarchical Constrained Spectral Clustering 12

2.2.3 Optimal number of islands 14

2.2.4 Separation into islands 14

2.3 Results and Discussions 15

2.3.1 Case Studies 16

2.3.2 Discussions 17

2.4 Summary 20

3 Operation Sequence for Proactive Islanding of the Power Grid : Heuristic Methods 21

3.1 Introduction 21
3.2 Methodology ............................................................. 22
   3.2.1 Line Outage Distribution Factor ................................ 22
   3.2.2 Generalized LODF ............................................. 23
   3.2.3 LODF-based Forward-Approach ................................ 24
   3.2.4 GLODF-based Backward Approach ............................ 25
3.3 Numeric Results ....................................................... 25
   3.3.1 Experiment Setup .............................................. 25
   3.3.2 Operation Sequence .......................................... 27
3.4 Summary ............................................................... 31

4 Chance-Constrained Optimal Distribution Network Partitioning to Enhance Power Grid Resilience 32
   4.1 Introduction ......................................................... 32
   4.2 Power systems resilience ......................................... 34
   4.3 Preliminaries ....................................................... 36
   4.3.1 Graph Theory ................................................. 36
   4.3.2 Chance-Constrained Optimization ............................ 37
   4.4 Problem Formulation .............................................. 37
   4.4.1 Distribution Network Model .................................. 38
   4.4.2 Power Flow Model ............................................ 39
   4.4.3 Radiality Constraints ........................................ 40
   4.4.4 Connection to grid-forming generators ...................... 42
   4.4.5 Deterministic ODNP .......................................... 43
   4.4.6 Probabilistic ODNP .......................................... 43
   4.4.7 Sample Average Approximation ............................... 44
   4.5 Solution Validation ............................................... 46
   4.6 Numerical Results ................................................ 47
   4.6.1 Experiment Set-up ............................................ 48
   4.6.2 Chance-Constrained ODNP .................................. 48
4.6.3 Choice of risk parameter .................................................. 50
4.6.4 Number of Normally Open Switches ................................. 50
4.6.5 Microgrid Topology .......................................................... 51
4.6.6 Computational Performance .............................................. 52

4.7 Summary .............................................................................. 52

5 A Model-Agnostic Method for PMU Data Recovery Using Optimal Singular Value Thresholding 54
5.1 Introduction ........................................................................... 54
5.2 Problem Set-up ....................................................................... 56
  5.2.1 PMU Data Recovery Using Matrix Estimation ....................... 56
  5.2.2 Matrix Transformation .................................................... 57
  5.2.3 Optimal Singular Value Thresholding (OSVT) ....................... 58
  5.2.4 Multivariate Time-Series Recovery .................................... 59
5.3 Data Recovery Algorithms ..................................................... 60
  5.3.1 Offline Data Imputation .................................................. 60
  5.3.2 Online Data Prediction ................................................... 61
  5.3.3 Choice of Hyperparameters ............................................ 61
  5.3.4 Scalability ....................................................................... 62
5.4 Numerical Results ................................................................... 63
  5.4.1 Simulated Measurements Dataset .................................... 63
  5.4.2 Data transformation ....................................................... 65
  5.4.3 Error Metric ..................................................................... 65
  5.4.4 Offline Data Imputation .................................................. 65
  5.4.5 Online Data Prediction ................................................... 67
  5.4.6 Real PMU measurements dataset ..................................... 70
5.5 Summary .............................................................................. 72

6.1 Introduction ................................................................. 73
6.2 Methodology ................................................................. 75
  6.2.1 Recurrent Plots ...................................................... 75
  6.2.2 Variational Autoencoder .......................................... 76
  6.2.3 DeVLearn Framework ............................................... 77
6.3 Results and Discussions .................................................. 78
  6.3.1 Experimental Setup .................................................. 79
  6.3.2 Recurrent Plots for Faults ........................................... 79
  6.3.3 Training the VAE ..................................................... 82
  6.3.4 Determining Fault Location ....................................... 82
6.4 Summary ................................................................. 82

7 Conclusion ................................................................. 84
  7.1 Contributions ............................................................. 84
  7.2 Future Work .............................................................. 86

Bibliography ................................................................. 88
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Multi-phase resilience trapezoid describing system behavior during a disturbance</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>PMU deployment in the North American power grid</td>
<td>5</td>
</tr>
<tr>
<td>2.1</td>
<td>Timeline of the proposed proactive islanding strategy</td>
<td>10</td>
</tr>
<tr>
<td>2.2</td>
<td>PJM backbone transmission network</td>
<td>11</td>
</tr>
<tr>
<td>2.3</td>
<td>Proposed islanding methodology</td>
<td>11</td>
</tr>
<tr>
<td>2.4</td>
<td>Case study: summer peak-load conditions</td>
<td>15</td>
</tr>
<tr>
<td>2.5</td>
<td>Dendrogram for summer peak-load conditions</td>
<td>16</td>
</tr>
<tr>
<td>2.6</td>
<td>Maximum load generation imbalance</td>
<td>16</td>
</tr>
<tr>
<td>2.7</td>
<td>Number of tie-lines to be disconnected</td>
<td>16</td>
</tr>
<tr>
<td>2.8</td>
<td>Two islands: summer peak</td>
<td>17</td>
</tr>
<tr>
<td>2.9</td>
<td>Three islands: summer peak</td>
<td>17</td>
</tr>
<tr>
<td>2.10</td>
<td>Dendrogram: winter peak</td>
<td>17</td>
</tr>
<tr>
<td>2.11</td>
<td>Dendrogram: spring light</td>
<td>17</td>
</tr>
<tr>
<td>2.12</td>
<td>Two islands for winter peak conditions</td>
<td>18</td>
</tr>
<tr>
<td>2.13</td>
<td>Three islands for winter peak conditions</td>
<td>18</td>
</tr>
<tr>
<td>2.14</td>
<td>Two islands for spring light-load conditions</td>
<td>18</td>
</tr>
<tr>
<td>2.15</td>
<td>Three islands for spring light-load conditions</td>
<td>18</td>
</tr>
<tr>
<td>3.1</td>
<td>Modified IEEE 39-bus test case</td>
<td>28</td>
</tr>
<tr>
<td>3.2</td>
<td>Bus voltages before and after separation into islands</td>
<td>29</td>
</tr>
<tr>
<td>3.3</td>
<td>Line loading before and after separation into islands</td>
<td>29</td>
</tr>
<tr>
<td>3.4</td>
<td>Load bus voltages following a random sequence of operations</td>
<td>29</td>
</tr>
<tr>
<td>3.5</td>
<td>Load bus voltages following the operation sequence chosen by algorithm 2</td>
<td>30</td>
</tr>
<tr>
<td>3.6</td>
<td>Load bus voltages following the operation sequence chosen by algorithm 3</td>
<td>30</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.1</td>
<td>Resilience trapezoid</td>
<td>35</td>
</tr>
<tr>
<td>4.2</td>
<td>Spanning forest and spanning tree</td>
<td>41</td>
</tr>
<tr>
<td>4.3</td>
<td>Modified IEEE 37-bus benchmark feeder</td>
<td>45</td>
</tr>
<tr>
<td>4.4</td>
<td>Median computation time increases with number of scenarios</td>
<td>49</td>
</tr>
<tr>
<td>4.5</td>
<td>2-D visualization of scenario clusters</td>
<td>49</td>
</tr>
<tr>
<td>4.6</td>
<td>Variation in cc-ODNP performance with number of scenarios considered</td>
<td>50</td>
</tr>
<tr>
<td>4.7</td>
<td>Variation in cc-ODNP performance with $\gamma$ considering 50 scenarios</td>
<td>51</td>
</tr>
<tr>
<td>4.8</td>
<td>Variation in performance with number of normally open switches. Number of scenarios considered is 50, $\gamma = 0.1$.</td>
<td>51</td>
</tr>
<tr>
<td>4.9</td>
<td>Optimal microgrid topology considering all switches, 100 scenarios and $\gamma = 0.1$. Average load served is 0.1935 p.u. $U_{0.05,1000}(x) = 0.08005$.</td>
<td>52</td>
</tr>
<tr>
<td>4.10</td>
<td>Optimal microgrid topology for the base radial network, 100 scenarios and $\gamma = 0.1$. Average load served is 0.1483 p.u. $U_{0.05,1000}(x) = 0.06465$.</td>
<td>53</td>
</tr>
<tr>
<td>5.1</td>
<td>PMU data flow from point-of-measurement to point-of-use</td>
<td>55</td>
</tr>
<tr>
<td>5.2</td>
<td>Page and hankel matrix construction from PMU measurements</td>
<td>58</td>
</tr>
<tr>
<td>5.3</td>
<td>Stacked page matrix for multivariate data recovery</td>
<td>60</td>
</tr>
<tr>
<td>5.4</td>
<td>Visual description of matrices $G$, $H$, $G'$ and $H'$ for a page matrix using data from a single channel</td>
<td>61</td>
</tr>
<tr>
<td>5.5</td>
<td>PMU measurements: Positive sequence voltage magnitude</td>
<td>64</td>
</tr>
<tr>
<td>5.6</td>
<td>PMU measurements: Positive sequence voltage angle (unwrapped)</td>
<td>64</td>
</tr>
<tr>
<td>5.7</td>
<td>PMU measurements: Frequency</td>
<td>64</td>
</tr>
<tr>
<td>5.8</td>
<td>Error in $G_2$ PMU channels with simultaneous data drop in all PMU channels (median over 20 runs)</td>
<td>66</td>
</tr>
<tr>
<td>5.9</td>
<td>Error in $G_2$ PMU channels with zero-mean additive noise on all $G_2$ PMU channels (median over 20 runs)</td>
<td>66</td>
</tr>
<tr>
<td>5.10</td>
<td>Imputed measurements with 50% simultaneous missing data and 2% additive noise on all $G_2$ PMU channels</td>
<td>67</td>
</tr>
<tr>
<td>5.11</td>
<td>Predicted measurements with 50% simultaneous missing data and 2% additive noise on all $G_2$ PMU channels</td>
<td>67</td>
</tr>
<tr>
<td>5.12</td>
<td>Prediction error in $G_2$ PMU channels with simultaneous missing data on all $G_2$ PMU channels (median over 20 runs)</td>
<td>68</td>
</tr>
</tbody>
</table>
5.13 Prediction error in $G_2$ PMU channels with additive noise on all $G_2$ PMU channels (median over 20 runs) .................................................. 68
5.14 Prediction error in $G_2$ PMU channels with simultaneous missing data on all $G_2$ PMU channels (median over 20 runs) ........................................ 69
5.15 Approximate matrix rank for different data windows .......................... 69
5.16 Data quality in the real PMU dataset .................................................. 70
5.17 Field PMU dataset ............................................................................. 71
5.18 Imputed PMU data using page matrix .................................................. 71
5.19 Imputed PMU data using hankel matrix ............................................. 71

6.1 The DeVLearn framework ................................................................. 74
6.2 Procedure for constructing unthresholded RP images from time-series data .................................................. 74
6.3 VAE architecture .............................................................................. 76
6.4 Pipeline showing DeVLearn operation ............................................. 77
6.5 IEEE 68-bus transmission network ................................................... 79
6.6 Voltage at Gen. 1 for faults at lines A and B ................................. 80
6.7 Voltage at Gen. 4 for faults at lines A and B ................................. 80
6.8 Unthresholded RP images for measurements shown in Fig. 6.6 and Fig. 6.7. 80
6.9 Evolution of latent attribute distribution in the compressed two dimensional space over different training epochs .................................................. 81
6.10 SVM classifier applied to fault location ........................................... 83
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Clustering performance metric $p$</td>
<td>19</td>
</tr>
<tr>
<td>2.2</td>
<td>Electrical performance when the network is divided into two islands</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>Electrical performance when the network is divided into three islands</td>
<td>19</td>
</tr>
<tr>
<td>3.1</td>
<td>Load and Gen. for the IEEE-39 Bus Test System</td>
<td>26</td>
</tr>
<tr>
<td>3.2</td>
<td>Operation Sequences Determined by Various Methods</td>
<td>28</td>
</tr>
<tr>
<td>3.3</td>
<td>Violations Observed for Various Switching Sequences</td>
<td>30</td>
</tr>
<tr>
<td>4.1</td>
<td>Comparison of SAA and clustering based approaches</td>
<td>49</td>
</tr>
</tbody>
</table>
# List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>Auto-Encoders</td>
</tr>
<tr>
<td>ANSI</td>
<td>American National Standards Institute</td>
</tr>
<tr>
<td>AUC</td>
<td>Area Under the Curve</td>
</tr>
<tr>
<td>CCO</td>
<td>Chance-Constrained Optimization</td>
</tr>
<tr>
<td>CHP</td>
<td>Combined Heat and Power</td>
</tr>
<tr>
<td>CNN</td>
<td>Convolutional Neural Network</td>
</tr>
<tr>
<td>DER</td>
<td>Distributed Energy Resources</td>
</tr>
<tr>
<td>DG</td>
<td>Diesel Generator</td>
</tr>
<tr>
<td>DL</td>
<td>Deep Learning</td>
</tr>
<tr>
<td>EI</td>
<td>Eastern Interconnect</td>
</tr>
<tr>
<td>EV</td>
<td>Electric Vehicle</td>
</tr>
<tr>
<td>fps</td>
<td>Frames per second</td>
</tr>
<tr>
<td>GLODF</td>
<td>Generalized Line Outage Distribution Factor</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>HILF</td>
<td>High-Impact Low-Frequency</td>
</tr>
<tr>
<td>HVDC</td>
<td>High Voltage Direct Current</td>
</tr>
<tr>
<td>ICI</td>
<td>Intentional Controlled Islanding</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>ISF</td>
<td>Injection Shift Factor</td>
</tr>
<tr>
<td>ISO</td>
<td>Independent System Operator</td>
</tr>
<tr>
<td>LDF</td>
<td>Linearized Distribution Flow</td>
</tr>
<tr>
<td>LODF</td>
<td>Line Outage Distribution Factor</td>
</tr>
<tr>
<td>LRF</td>
<td>Linear Recurrent Function</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>LSE</td>
<td>Linear State Estimation</td>
</tr>
<tr>
<td>MAPE</td>
<td>Mean Absolute Percentage Error</td>
</tr>
<tr>
<td>ME</td>
<td>Matrix Estimation</td>
</tr>
<tr>
<td>MILP</td>
<td>Mixed-Integer Linear Program</td>
</tr>
<tr>
<td>ML</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>NASPI</td>
<td>North American Synchrophasor Initiative</td>
</tr>
<tr>
<td>NN</td>
<td>Neural Network</td>
</tr>
<tr>
<td>NREL</td>
<td>National Renewable Energy Laboratory</td>
</tr>
<tr>
<td>ODNP</td>
<td>Optimal Distribution Network Partitioning</td>
</tr>
<tr>
<td>OSVT</td>
<td>Optimal Singular Value Thresholding</td>
</tr>
<tr>
<td>PAA</td>
<td>Piecewise Aggregate Approximation</td>
</tr>
<tr>
<td>PMU</td>
<td>Phasor Measurement Unit</td>
</tr>
<tr>
<td>PTDF</td>
<td>Power Transfer Distribution Factor</td>
</tr>
<tr>
<td>PV</td>
<td>Photo-Voltaic</td>
</tr>
<tr>
<td>RAS</td>
<td>Remedial Action Schemes</td>
</tr>
<tr>
<td>RP</td>
<td>Recurrent Plots</td>
</tr>
<tr>
<td>SAA</td>
<td>Sample Average Approximation</td>
</tr>
<tr>
<td>SCADA</td>
<td>Supervisory Control and Data Acquisition</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to Noise Ratio</td>
</tr>
<tr>
<td>SPS</td>
<td>Special Protection Schemes</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular Value Decomposition</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
<tr>
<td>TSC</td>
<td>Time Series Classification</td>
</tr>
<tr>
<td>VAE</td>
<td>Variational Auto-Encoder</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

The modern power grid is rapidly evolving to accommodate changing ways in which energy is being generated and consumed. Over the past decade, technological advancements, changing consumer preferences, and government policies have brought about sweeping changes in the energy sector. Integration of renewable energy resources is being encouraged to reduce fossil-fuel reliance; the proliferation of communication-enabled smart devices is changing how consumers interact with utilities; and load is growing due to several factors such as increasing population, transportation electrification, and cryptocurrency mining. Grid operation strategies must account for these changes without jeopardizing the reliability of service.

The power grid is integral to the normal functioning of modern society. Disruptions in grid service can cause immense hardships and the loss of life and property, as witnessed during several recent extreme-weather events like Hurricane Maria in Puerto Rico (September 2017) and winter storm Uri in Texas (February 2021). Today the grid faces challenges on multiple fronts—severe-weather events are becoming more extreme and frequent, the aging infrastructure remains exposed to acts of physical destruction, and the threat of malicious cyber attacks loom. Therefore, effective measures for mitigating these threats and enhancing system resilience are needed. Such measures could include weatherizing infrastructure, increasing reserve capacity to meet contingencies, and deploying wide-area situational awareness tools that help identify precursors to impending failures.

Responding to disasters is an extremely complex problem that requires coordinated actions to be executed by multiple agencies. In this context, this dissertation aims to develop tools that aid decision-making in the face of extreme events. We propose novel strategies for limiting the impact of disturbances on power systems, with particular attention towards accommodating the practitioner’s constraints. The focus is on two distinct areas— a) leveraging existing resources to improve grid performance in the immediate aftermath of a disruptive event, and b) enhancing grid monitoring capabilities using data-driven strategies. Better grid performance after a disruption can help maintain energy supply to customers in affected areas before adequate restorative activities can commence. High-resolution insight into grid conditions can help utilities identify indications of impending failures, thereby facilitating timely preventive actions.
1.1 Infrastructure Resilience

Resilient infrastructure helps reduce the magnitude and/or duration of disruptive events. The bulk power system is vulnerable to a variety of threats such as wildfires, hurricanes, geomagnetic disturbances, and coordinated cyber-physical attacks \([3]\). Different infrastructure networks are interdependent, and hence failures in one can propagate to another. However, different networks may face different challenges due to the same event, and hence sector-specific resilience frameworks may need to be drawn up.

1.1.1 Resilience Metrics

To evaluate the effectiveness of competing strategies, some quantitative measure is needed for capturing the rather conceptual idea of resilience. Several recent research efforts have sought to describe resilience in quantitative terms, but standard metrics have not yet been agreed upon \([8, 9, 10]\).

A popular approach for quantifying resilience uses the multi-phase resilience trapezoid that depicts system behavior during a disaster; see figure 1.1 \([8]\). Consider a disruptive event that starts at time \(t_1\). The *first phase* \((t \in [t_1, t_2])\) of event progress is characterized by a rapid deterioration in system performance from its pre-disturbance levels. In the *second phase* \((t \in [t_2, t_3])\), the system operates in a diminished capacity before restoration measures can be undertaken. The duration of this phase could range from hours to months and delays in system recovery could be exacerbated by factors like challenging terrains restricting crew mobility, lack of automation, inadequate system visibility due to communication outages, and supply chain constraints for long-lead items. More than eight months were needed to
1.1. INFRASTRUCTURE RESILIENCE

restore electricity to most parts of Puerto Rico after Hurricane Maria [11]. During the third phase ($t \in [t_3, t_4]$), restoration activities are carried out. The transitions between system states shown in figure 1.1 are linear only for expository convenience. In practice, these may be non-linear, depending on system characteristics, prevalent operating conditions and the nature of the disaster experienced.

The resilience trapezoid describes - a) how fast and how far the system degrades once a disruptive event strikes, b) how long the system resides in a diminished state, and c) how fast it recovers. A metric that combines these measures to convey system resilience is the area under the curve (AUC) of the trapezoid. Any resilience enhancement strategy seeks to maximize this AUC. Note that in figure 1.1, the $y$-axis represents a generic system function as its performance measure. Different quantities could be used here. In this work, total load served has been used as the performance metric. Maximizing load served also improves traditional grid reliability indices such as SAIDI (System Average Interruption Duration Index), SAIFI (System Average Interruption Frequency Index), and MAIFI (Momentary Average Interruption Frequency Index) [12].

1.1.2 Grid Partitioning Schemes

Resilience enhancement strategies described in this work seek to use grid partitioning schemes for boosting performance in the post-disturbance degraded state (second phase in the resilience trapezoid), thereby also improving the AUC metric. Further, it is anticipated that limiting system degradation following a contingency will also aid in faster recovery. In the past, electric utilities have shown that electrical islands can be sustained during extreme events through operator intervention. For example, Hurricane Gustav (August 2008) caused outages on key transmission lines, and an island was formed around metropolitan New Orleans, Louisiana. This island was sustained for several hours before it could be re-synchronized to the Eastern Interconnect [13]. During the July 2012 blackout in India, the area around metropolitan Kolkata survived by isolating itself from the bulk power grid, and later also aided restoration [14]. Having a strategic isolation scheme in place may assist utilities in being better prepared for such scenarios. This fact is gaining recognition and in June 2021, the Indian Power Ministry issued orders for drawing up islanding plans for major cities [15].

The island creation methods proposed in this dissertation use tools from graph theory, operations research, and optimization. The electrical connectivity among different buses and lines in the grid can be abstracted by a simple directed graph, and hence graph-theoretic and spectral techniques are natural choices for power systems analysis and visualization [16, 17, 18]. Pure topological metrics may not adequately capture the physical laws governing power flow, and therefore the graph edges may need to be weighted by some electrical feature. As reconfiguration schemes involve determining whether certain grid components should be in service, they become combinatorial problems that need to use integer decision variables.
To alleviate emergency conditions in the grid, operators employ various measures such as branch switching, adjusting generator set-points, cancelling scheduled outages, and changing power transfer across major interfaces. Remedial Action Schemes (RAS) or Special Protection Schemes (SPS) are being used to prevent single component failures from causing growing oscillations in a highly stressed grid [19, 20, 21]. However, RAS are usually designed with specific cases in mind, and potential mis-operation of such schemes may cause wide-area outages as observed in the Central American interconnected transmission system [22]. Unlike RAS, the grid splitting strategy outlined in this work is meant to be deployed only during potentially catastrophic events.

Controlled islanding has been used to arrest cascading disturbances by grouping coherent generators together in the same island [23, 24, 25, 26]. Such islanding schemes are usually reactive, and need to be executed swiftly following a disturbance. Some ideas regarding proactively splitting the grid also appear in literature—[27] proposes identifying components most likely to fail due to a weather emergency and isolate them from the rest of the grid; and [28] puts forth a spectral clustering based strategy to determine tightly connected portions within the network. The prior approaches assume that minimizing the power flow disruption will result in the creation of stable islands, and the performance of the intended islands is not investigated. In this work, a proactive islanding strategy considering the power imbalance and operation constraint violations in the intended islands is developed. The number of islands is determined based on the nature of the expected contingency, power balance, and the number of line disconnections needed to realize the islands.

The successful implementation of any islanding scheme needs coordination among multiple utilities and Independent System Operators (ISO). While an ISO may determine the island configuration for a particular contingency, physical power systems assets can only be operated by the respective asset owners. Therefore, communication protocols among the stakeholders need to be standardized for occasions where implementing the proactive islanding strategy will be needed. Stakeholders must also come to an agreement about when the splitting scheme should be triggered. Different ISOs and their utility members already have action plans for responding to different emergency scenarios, and a similar action plan can be drawn up for islanding as well [29, 30]. ISOs like PJM Interconnection conduct system restoration drills at regular intervals, and similar drills for islanding under different operating conditions can prove beneficial in fine-tuning the formulated action plans.

1.2 Data-based Grid Monitoring

In the last two decades, electric utilities have made great strides in deploying advanced meters and sensors for monitoring their networks. These devices stream information to a control center in near-real-time, providing increased wide-area situational awareness. Foremost among such sensors are Phasor Measurements Units (PMUs) that provide high-resolution insight into grid dynamics through precise time-synchronized measurements. Simply put, PMUs are high-precision measurement devices that report synchronized phasor quantities
1.2. DATA-BASED GRID MONITORING

Figure 1.2: PMU deployment in the North American power grid, as of October 2014. Image reproduced from NASPI synchrophasor technology fact sheet [31].

(or synchrophasors) at different points of the grid at a rate of 30/60 frames per second (fps). The measured phasors share a common time reference, such as one provided by the GPS. Synchrophasor measurements have proved invaluable in diagnosing the causes of blackouts, monitoring equipment health, and developing tools for resource planning [32]. Funding provided by the American Recovery and Reinvestment Act of 2009 greatly accelerated the deployment of PMUs in the North American transmission network (see figure 1.2), and since then PMU data has been used for various applications such as improved state estimation, asset health monitoring, event localization, oscillation monitoring, forensic analysis of large disturbances, and model validation [22, 33, 34, 35]. The success of synchrophasor technology-based control and monitoring tools in the transmission network has also motivated efforts for developing PMUs for distribution-network-specific applications [36].

Due to their high reporting rates, PMUs paint a more detailed picture of grid conditions than traditional SCADA systems that provide asynchronous magnitude measurements once every 2-3 seconds. Machine learning (ML) based tools have been proposed in literature to use PMU data for tasks like fault identification and classification [37]. However, there are several impediments in the path of wider adoption of ML-based strategies in utility control rooms. Some key challenges are:
• The measurement data has to flow through several intermediate concentrators and communication channels to reach the control center from the point of measurement and hence can get dropped, delayed, or distorted due to congestion or component failures on its path. Measurement errors could also be introduced by inaccurate calibration of instrument transformers, faulty wiring, and GPS signal loss. Noisy, corrupt, and intermittent data can limit what an ML model learns. Therefore, measurement denoising and imputation schemes need to be developed.

• Another reason restricting the rapid development of data-based grid monitoring strategies is the limited availability of labeled datasets and related system models. As PMU measurements contain sensitive information, open-source information cannot be made available to the wider ML community.

• Due to the stochastic fluctuations in connected loads, PMU data may exhibit small disturbances which pose no threat to the grid and hence are not of interest to utilities. Parsing terabytes of data to find interesting information is also a herculean task.

PMU measurements contain information that can provide advance warnings of impending failures, and events that may have evaded traditional detection methods used in the industry. Unsupervised learning techniques that can extract event signatures without explicit knowledge of the time and location of events present an exciting research frontier. Simulated and synthetically generated data could also be used to augment sparse field observations while training ML models. Efforts focused on curating labeled datasets will prove useful in facilitating the development of future ML tools and providing a benchmark for comparing their performance.

This dissertation attempts to improve PMU data-based grid monitoring by developing algorithms for accurately recovering corrupt measurements and identifying the location of temporary faults that may not be registered by protective relays. The proposed methods are model-agnostic, i.e. they do not need the knowledge of model parameters. Hence, they are not affected by issues like latency in breaker status updates and inaccuracies in line parameter information. Techniques from matrix estimation, deep learning, and computer vision domains have been used here.

1.3 Organization

This dissertation describes various ways to mitigate the impact of disruptive events on power systems. The first part of the text explores how network partitioning can improve grid performance in the aftermath of a disaster. In this context, chapter 2 discusses how proactively splitting the interconnected bulk power transmission system into small self-sustaining islands before a High-Impact Low-Frequency (HILF) event can bound cascading failures within affected areas, thereby preventing wider propagation of disturbances. A flexible island determination scheme using constrained hierarchical spectral clustering is illustrated on the
PSS/E model of the Eastern Interconnect (EI). We also explore how to adapt the island determination scheme according to the prevalent operating conditions, the nature of the threat experienced, and operation constraints. The text in this chapter is largely reproduced from our publication [38].

Chapter 3 develops heuristic methods to determine the sequence of operations needed for splitting the grid into self-adequate islands. The proposed algorithms use line outage distribution factors to approximate changes in real power flow on network branches in response to topology changes. They are computationally efficient, and designed to minimize the loading on transmission lines during the separation process. With the help of case-studies on a modified IEEE 39-bus 10-machine test system, we show that the chosen operation sequences minimize line capacity violations and maintain bus voltages within prescribed limits. The proposed algorithms find sequences that outperform a randomly chosen option, but further analysis will be needed to ascertain how far they are from the optimal solution.

Next, in chapter 4, the idea of network partitioning is extended to the power distribution grid. Due to the lack of adequate switching and control capabilities, dynamic island formation may not be possible in the distribution network in its current form. Hence, the optimal partitioning strategy needs to be formulated at the planning stage itself taking uncertainties in load and generation into account. We develop a methodology for identifying potential self-adequate portions of the existing distribution grid that can be converted to microgrids via economically viable retrofitting. The network partitioning problem is posed as a mixed-integer linear program (MILP). The identified microgrid candidates will help ride out long grid outages, thereby improving system resilience. The material presented in this chapter borrows from our publication [39].

The second part of the dissertation focuses on enhancing grid-monitoring strategies using synchrophasor measurements. Chapter 5 addresses the question of PMU data quality. A model-agnostic procedure for recovering noisy PMU data streams with missing entries is proposed. The measurements are first converted into a Page matrix with non-overlapping data segments placed side by side, and then the signals are reconstructed using low-rank matrix estimation techniques. Two variations of the recovery algorithm are developed— a) an offline method for imputing archived measurements, and b) an online method for predicting incoming data. The proposed measurement recovery scheme is very fast, scalable, and can recover noisy data streams with consecutive correlated erasures across multiple PMU channels. The work described in this chapter is part of a manuscript presently under consideration for publication [40].

Subsequently, in chapter 6, PMU data is used to develop a deep-learning framework for determining the location of temporary faults. Frequently recurring temporary faults in power networks may be indicative of impending failures if left unchecked. As these faults are self-clearing and may have high fault impedance, protective relays may not be able to detect their presence in the system. With the help of the fault location algorithm developed herein, grid operators can be alerted if temporary faults keep reappearing at any network
location. Timely remedial actions can prevent equipment failures and consequent outages. Promising results have been obtained for faults simulated on the 16-machine IEEE 68-bus test system. This chapter largely reiterates findings reported in our publication [41].

Chapter 7 summarizes the contributions of this dissertation, outlines future research directions and concludes this document.
Chapter 2

Proactive Islanding of the Power Grid to Mitigate High-Impact Low-Frequency Events

2.1 Introduction

High-Impact Low-Frequency (HILF) events like coordinated cyber, physical, or blended attacks, extreme solar weather, and high altitude detonation of a nuclear weapon may cause catastrophic and long-lasting damage to the power grid [3, 4, 5, 29]. In the interconnected AC grid, an initial disturbance may cause wide-area outages due to cascading failures [43, 44]. The industry deploys intentional controlled islanding (ICI) as a corrective action to arrest cascading events [23, 24, 26, 45, 46, 47], but these are reactive responses to faults and need to be executed swiftly. This work, in contrast, proposes to proactively partition the grid into self-sustaining islands before the disturbance occurs, if credible intelligence of an imminent threat is available. This idea is illustrated in figure 2.1. Here, the y-axis shows system performance as a function of time. As the figure suggests, despite the prior availability of intelligence, conventional reactive responses are deployed only when an event has occurred. Proactive action may degrade system performance before the event but will help in limiting subsequent damage by arresting cascading events, also aiding recovery. Since actions are initiated before the event, grid operators have the time to coordinate dispatch and switching actions needed to ensure system stability.

The proposed mitigation approach may prove especially useful when only limited threat intelligence is available. For example, consider situations where grid attacks are anticipated but the precise target locations are unknown. Moreover, through timely interventions, some threats may also be averted. Hence, the identified islands should meet the applicable operation safety and reliability criteria, be able to survive for extended durations, and minimize load-shedding. Ad hoc operator actions will be needed for continued service; so the prevailing
CHAPTER 2. PROACTIVE ISLANDING OF THE POWER GRID TO MITIGATE HIGH-IMPACT LOW-FREQUENCY EVENTS

Operating conditions and other concerns like maintaining system awareness and observability need to be addressed as well.

Some ideas about proactively splitting the grid appear in existing literature. In [27], the failure probability of components due to weather contingencies are assessed and those most vulnerable are isolated within an island. This approach, however, is effective only for local contingencies. A network splitting strategy is proposed in [28], but in this work the operational viability of the sub-networks is not checked. In this chapter, we outline a flexible methodology for identifying viable islands within a highly meshed grid and verify their electrical performance using steady-state AC power flow. Our approach addresses prevalent system conditions, is independent of disturbance origins and can be tailored to the nature of contingencies. The proposed scheme is demonstrated using the PSS/E model of the PJM transmission network in the Eastern Interconnection (EI). Due to the vast expanse and geography of the PJM footprint (figure 2.2), and the intrinsic densely meshed nature of its infrastructure, PJM has no intuitive ‘natural’ islanding interfaces. Thus, strategically islanding PJM is a non-trivial task.

2.2 Methodology

The proposed multi-step islanding methodology is described in figure 2.3. The electrical network is represented as a graph and weakly interconnected sub-graphs capable of surviving as islands are identified using a constrained spectral clustering technique. The optimal number of islands is determined based on the nature of the expected contingency, and the network needs to be split accordingly. Each step is detailed next.

2.2.1 Graph Construction

The electrical network is represented by a simple weighted undirected graph \( G = (V, E) \), where \( V \) and \( E \) are the vertex and edge sets respectively. Each PJM transmission zone is a
2.2. Methodology

Figure 2.2: PJM backbone transmission network [42]

Figure 2.3: Proposed islanding methodology
CHAPTER 2. PROACTIVE ISLANDING OF THE POWER GRID TO MITIGATE HIGH-ImpACT
LOW-FREQUENCY EVENTS

graph vertex. If two zones $i$ and $j$ are electrically connected, there is an edge $e_{i,j}$ between the corresponding vertices $v_i$ and $v_j$. The edge weights $w_{i,j}$ are the absolute values of the apparent power exchange between zones i.e. $w_{i,j}$ is the sum of apparent power flows on all tie-lines connecting zones $i$ and $j$. The higher the weight of an edge, the tighter the coupling between vertices. HVDC tie-lines are excluded from the graph as they do not propagate disturbances. All electrical connections to areas outside PJM are mapped to a single vertex $v_x$. The PJM transmission zones have been anonymized.

Representing zones as graph vertices will ensure that all equipment operated by the same transmission operator stays within the same island. This serves the following purposes.

- Equipment within each zone is owned and operated by a single transmission owner. Therefore, operating all utility assets from a single control room eliminates the need to establish communication and coordination procedures among multiple stakeholders.
- Only tie-line power flow measurements are required for graph construction. Usually, tie lines between different control areas are better instrumented, and the requisite data is readily available.

Moreover, using zones as vertices instead of individual buses greatly limits the size and order of the graph. This, in turn, helps in formulating a fast adaptive algorithm for island determination, allowing more time for establishing load-generation balance and coordinating switching actions in the transmission network.

2.2.2 Hierarchical Constrained Spectral Clustering

ICI can essentially be described as NP-hard searching problems on graphs [46]. Spectral clustering, a graph-theoretic technique, has been proposed as an alternative for solving the minimum power flow disruption islanding problem [24, 28, 46]. Here, the central idea is to use the eigenvalues and eigenvectors of Laplacian matrices to find groups (clusters) of vertices that are highly connected with each other but weakly connected to vertices in other clusters. A comprehensive background on the clustering methodology and its applicability in power systems is provided in [48] and [28] respectively. One major drawback of the spectral clustering technique applied to power systems islanding is that it does not consider generator coherency constraints, which is a critical concern for transient stability, especially when islanding the system after a fault. The islanding methodology put forth in our work recommends reinforcing the system before splitting it, and hence a controlled sequence of switching actions, generation re-dispatching, and voltage control can be used to ensure transient stability.

**Graph Laplacian:** A simple weighted undirected graph $G = (V, E)$ can be described with a weighted adjacency matrix $A$ and a degree matrix $D$. Let $N$ be the order of graph $G$ and
2.2. METHODOLOGY

Let \( w_{i,j} \) be the weight of edge \( e_{i,j} \). Then, \( A \in \mathbb{R}^{N \times N} \) is a symmetric matrix such that,

\[
[A]_{i,j} = \begin{cases} 
    w_{i,j}, & \text{if } e_{i,j} \in E \\
    0, & \text{otherwise}
\end{cases} \quad (2.1)
\]

Matrix \( D \) is diagonal with non-negative diagonal entries \( d_i \), where \( d_i \) is the weighted degree of vertex \( v_i \) i.e. the total weight of edges connected to that vertex.

\[
d_i = \sum_{j=1}^{N} w_{i,j}, \quad \forall i \in \{1, 2, \ldots, N\} \quad (2.2)
\]

Laplacian matrices have been used extensively to study graphs. Two main variants are proposed, the unnormalized Laplacian \( L \) and the normalized Laplacian \( L_N \). It is beneficial to use \( L_N \) for clustering purposes since it is scale-independent.

\[
L = D - A \quad (2.3)
\]

\[
L_N = D^{-1/2}LD^{-1/2} \quad (2.4)
\]

**Spectral embedding**: This process refers to representing elements in \( V \) in \( k \)-dimensional euclidean space \( \mathbb{R}^k \) using the first \( k \) eigenvectors of \( L \) or \( L_N \). Here, \( 2 \leq k \ll N \). One key property of graph Laplacians is that they have \( N \) non-negative real eigenvalues \( 0 \leq \lambda_1 \leq \lambda_2 \cdots \leq \lambda_{N-1} \) [48]. The first \( k \) eigenvectors refer to those corresponding to the \( k \) smallest eigenvalues. Ordering the \( k \) eigenvectors as columns gives us a matrix \( X \in \mathbb{R}^{N \times k} \) with rows \( \chi_i, i \in \{1, 2, \ldots, N\} \). Vector \( \chi_i \) gives the coordinates of vertex \( v_i \) in \( \mathbb{R}^k \). Any standard algorithm like k-means may then be used to group these points into clusters. When using \( L_N \), vectors \( \chi_i \) must be normalized to length 1 before clustering [28, 48].

An obvious question here is how to choose the value of \( k \). We use the commonly suggested eigengap criterion. Eigengaps are the difference between two consecutive eigenvalues, \( \gamma_k = \lambda_{k+1} - \lambda_k \). A high value of \( \gamma_k \) suggests that the graph maybe decomposed into at least \( k \) clusters and this will be revealed with spectral embedding in \( k \)-dimension.

**Constrained hierarchical clustering**: Several limitations exist when applying spectral clustering to the controlled islanding problem. First, if \( k > 2 \), an additional k-means step is needed to determine the optimal clusters. This approach has several drawbacks, such as- a) the number of clusters needs to be specified a-priori, and b) clustering results depend on the initial choice of cluster centroids. Second, when projecting graph vertices into \( \mathbb{R}^k \), the edge information is ignored. Therefore, there maybe some points which are close in the euclidean space but do not have a connecting edge in the original graph.

The first limitation is overcome using agglomerative hierarchical clustering, similar to the approach shown in [28]. In this method, at the initial step, every point is considered as an individual cluster. Next, the closest clusters according to some distance metric are merged.
together. This process is repeated until all points are merged into a single cluster. This clustering 'hierarchy' can be encoded into a tree-like structure called dendrogram. By ‘cutting’ the dendrogram at different levels, different numbers of clusters can be obtained. We use the criteria outlined later in section 2.2.3 in conjunction with the dendrogram to decide the number of islands. In this work, the ward distance metric is used to determine the distance between two points.

The second shortcoming is dealt with by imposing a connectivity constraint, i.e. clusters are merged only if there is an edge connecting them in the original graph. The complete process is summarized in algorithm 1.

Algorithm 1 Hierarchical Constrained Spectral Clustering

1: Normalized Laplacian: Compute $L_N$, as per eq. (2.1)-(2.4).
2: Eigenvalue Decomposition: Compute eigenvalues and eigenvectors of $L_N$.
3: Spectral Dimension: Sort the eigenvalues of $L_N$ in ascending order, $0 \leq \lambda_1 \leq \lambda_2 \cdots \leq \lambda_N$. Choose $2 \leq k \ll N$ such that eigengap $\gamma_k$ is high.
4: Spectral k-embedding: Construct matrix $X$ with first $k$ eigenvectors of $L_N$ as its columns. Normalize the columns of $X$ to length 1. The $i$-th row of $X$, $\chi_i$, represents the coordinates of vertex $v_i$ in $\mathbb{R}^k$.
5: Constrained Hierarchical Agglomerative Clustering: Agglomerative hierarchical clustering of points represented by vectors $\chi_i$, $i \in \{1, \ldots N\}$ with additional connectivity constraint.

2.2.3 Optimal number of islands

The dendrogram describes the grouping of zones within the electrical network, and by cutting it at different levels, any $r$ number of islands may be determined. How to choose the value of $r$ is a risk-management decision and must consider several questions.

- What is the nature of the contingency? If only local damage is expected, then a small number of islands is advisable. For a coordinated attack at multiple points of the network, a larger number of islands might be useful.
- Are the islands self-sustainable? Solutions that minimize load generation-imbalance and operational violations like line overloading, undervoltages etc are favored.
- How many switching operations would be required to split the network? Evidently, solutions that require fewer number of line disconnections are easier to realize in practice.

2.2.4 Separation into islands

Island creation involves disconnecting a large number of transmission lines. For instance, disconnecting the PJM footprint from the EI would involve switching 212 transmission lines, which is a herculean endeavor [42]. We propose that the splitting is carried out in two steps:
2.3 Results and Discussions

The proposed islanding methodology is demonstrated using the PSS/E model of the PJM network. Three different cases are studied, representative of summer peak, winter peak and spring light load conditions. In these cases, load within the PJM network are 163.6 GW, 140.9 GW and 81.5 GW respectively. Counter-intuitively, during light load conditions, inter-area tie-line flow is higher than peak load periods. This is because most generators need to be in service to meet the peak demand. However, during light load conditions, expensive generation units may be turned off to keep energy prices low and the cheaper generators may be located away from load centers. Since studying the PJM network in isolation is not representative of actual operating conditions, the PSS/E simulation model includes full representations of neighboring systems in the EI. The model has more than 158,000 buses, 20,437 of which are within PJM. The PJM network has 1617 generator buses, 944 fixed shunts and 2084 variable shunts.

An interesting question to consider is: should some PJM zones remain connected to the rest of the EI when separating into islands? In this work, it is considered that certain zones within PJM stay connected to the EI. The methodology can be easily extended to the alternative case, and would be equivalent to removing vertex X from the network graph.

- **Redispatch generation and load shedding within islands:** This step will establish load-generation balance within the intended islands and further reduce the power flow on the branches that need to be disconnected. Thus, the generators within an island are likely to stay synchronized following the network separation, and operational violations will be minimized.

- **Disconnect transmission lines:** Formulating an exact sequence of actions to split the grid into islands is a complicated problem. In chapter 3, we explore how to formulate an optimal sequence of line disconnection steps to isolate any sub-network within the power grid.
CHAPTER 2. PROACTIVE ISLANDING OF THE POWER GRID TO MITIGATE HIGH-ImpACT
LOW-FREQUENCY EVENTS

2.3.1 Case Studies

Let us first consider the summer peak case example. Figure 2.4a shows the network graph. Edge thicknesses are proportional to their weights. The network outside PJM is mapped onto vertex X. Eigenvalues and eigengaps are computed for $L_N$ (figure 2.4b). Since the eigengap $\gamma_k$ is maximum for $k = 3$, spectral embedding is done in three dimensions (figure 2.4c). Hierarchical clustering with connectivity constraints yields the dendrogram shown in figure 2.5. Grouping into three clusters according to the dendrogram is also shown in figure 2.4c.

If the number of islands is varied from two to nine, maximum load-generation imbalance expected in any island would be as shown in figure 2.6. The number of tie-lines to be disconnected for creating $r$ islands is shown in figure 2.7. We see that splitting the grid into three islands from two does not increase the maximum expected imbalance, but needs 82 additional switching operations. Hence, the number of islands needed must be decided on a
2.3. Results and Discussions

The approximate geographical boundaries of the determined islands are shown in figure 2.8 and 2.9. Using the same methodology, the dendrograms shown in figure 2.10 and 2.11 are obtained for winter peak and spring light load conditions. Approximate island boundaries are shown in figure 2.12-2.15. It is evident that the power flow pattern and hence the island configuration changes with seasonal conditions.

2.3.2 Discussions

The determined islands are geographically contiguous. To quantify the quality of the partition, we define a metric $p$ that is a ratio of the total power flow on tie lines to be disconnected and the total power flow within the islands. Evidently, a smaller value of $p$ indicates that zones within an island are tightly coupled. This ratio metric has been used to compare the performance for different system load conditions.

$$p = \frac{\text{Sum of edge weights connecting clusters}}{\text{Sum of edge weights within clusters}}$$
CHAPTER 2. PROACTIVE ISLANDING OF THE POWER GRID TO MITIGATE HIGH-ImpACT LOW-FREQUENCY EVENTS

Figure 2.12: Two islands for winter peak conditions

Figure 2.13: Three islands for winter peak conditions

Figure 2.14: Two islands for spring light-load conditions

Figure 2.15: Three islands for spring light-load conditions
2.3. RESULTS AND DISCUSSIONS

Table 2.1: Clustering performance metric $p$

<table>
<thead>
<tr>
<th></th>
<th>Summer peak</th>
<th>Winter peak</th>
<th>Spring light load</th>
</tr>
</thead>
<tbody>
<tr>
<td>Two clusters</td>
<td>0.0702</td>
<td>0.0955</td>
<td>0.0958</td>
</tr>
<tr>
<td>Three clusters</td>
<td>0.1081</td>
<td>0.1766</td>
<td>0.2495</td>
</tr>
</tbody>
</table>

Table 2.2: Electrical performance when the network is divided into two islands

<table>
<thead>
<tr>
<th></th>
<th>Summer Peak</th>
<th>Winter Peak</th>
<th>Light Load</th>
</tr>
</thead>
<tbody>
<tr>
<td>Island 1</td>
<td>Island 2</td>
<td>Island 1</td>
<td>Island 2</td>
</tr>
<tr>
<td>Gen. redispatch (MW)</td>
<td>-3341.4</td>
<td>3033.8</td>
<td>1218.9</td>
</tr>
<tr>
<td>Load-shedding (MW)</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Min. bus voltage (p.u.)</td>
<td>0.97</td>
<td>0.96</td>
<td>0.97</td>
</tr>
</tbody>
</table>

Table 2.3: Electrical performance when the network is divided into three islands

<table>
<thead>
<tr>
<th></th>
<th>Summer Peak</th>
<th>Winter Peak</th>
<th>Light Load</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>I2</td>
<td>I3</td>
<td>I1</td>
</tr>
<tr>
<td>Gen. redispatch (MW)</td>
<td>-3341</td>
<td>1545</td>
<td>1489</td>
</tr>
<tr>
<td>Load-shedding (MW)</td>
<td>0</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>Min. bus voltage (p.u.)</td>
<td>0.96</td>
<td>0.95</td>
<td>0.96</td>
</tr>
</tbody>
</table>

Table 2.1 shows the metric $p$ for different seasons (calculated before any generation redispatch or load shedding). It can be seen that splitting the network requires high power flow disruption during light load conditions. This follows our previous discussion that there is high inter-area power flow during light load to minimize energy prices. However, during these conditions, higher generation reserves that may be redispatched to sustain the islands will also be available.

A summary of the electrical performance of the islands (bus voltages and power balance) created for each seasonal case is provided in tables 2.2 and 2.3. These values have been determined using steady state power flow solutions with the PSS/E model. In our simulations, generation has been redispatched and voltage control actions have been performed before splitting the network into islands and solving AC power flow. It is evident from the obtained values that the identified islands can be sustained, albeit with operator intervention. The fact that PJM has adequate installed reserve generation capacity minimizes load-shedding in the islands. Additionally, the exact sequence of operations for grid splitting remains to be determined.
2.4 Summary

PJM frequently implements operating procedures to enhance the reliability of the system under stress. These control actions include procuring synchronous reserves, decreasing the power transfer across major interfaces, cancelling scheduled outages, etc. In the context of PJM’s long-term resilience enhancement goals, this work has developed a methodology to proactively split the grid into self-adequate islands in preparation for HILF events. Strategically islanding the grid has two major benefits: 1) it bounds the extent of potential cascading outages, and 2) it facilitates system restoration after the event. The proposed approach is adaptive, independent of fault location, addresses prevailing network conditions and may be tailored according to the nature of the expected contingency. We use the PSS/E model of the heavily meshed PJM transmission network to validate our islanding methodology and check the electrical performance of the resultant islands. Realizing the islands in practice would need formulation of an exact sequence of switching operations. This is a complex problem that is explored in chapter 3.
Chapter 3

Operation Sequence for Proactive Islanding of the Power Grid: Heuristic Methods

3.1 Introduction

The power grid is a critical infrastructure whose availability is instrumental to the normal functioning of modern society. Extreme events like hurricanes, wildfires and coordinated cyber-physical attacks can cause catastrophic and long-lasting damage to the grid, necessitating the formulation of defensive strategies [3, 4]. Prior works suggest that proactively splitting the transmission network into smaller self-adequate islands could be an effective way to arrest cascading disturbances during a severe event, thereby limiting damages [27, 38]. In [27], the authors assess the failure probability of network components due to a weather contingency and isolate the most vulnerable components within an island. In [38], a generalized method for proactive islanding in anticipation of extreme events is presented. However, tools to determine the optimal sequence of line switching operations required for successful grid separation remain to be explored.

To create multiple islands within a real power network, many transmission lines may need to be disconnected. For instance, in [38] it is shown that about 50 line disconnections are needed to split the PJM network into two parts. Evidently, the number of possible switching sequences is very large and some method for selecting a good sequence is required. An acceptable sequence of operations would maintain load-generation balance in the network, and avoid line overloads and voltage deviations. This sequence determination problem could be formulated as a mathematical optimization, but due to the combinatorial nature of the task, integer decision variables may be needed. Solving a mixed-integer optimization problem for a large transmission network is computationally prohibitive for near-real-time applications. Therefore, fast heuristic methods are preferred. In this context, this work proposes two methods to address the sequence formulation task.

It has previously been shown that transmission line switching can reduce generation cost, network losses, and alleviate line congestion. Both exact and heuristic methods have been employed to address this optimal switching problem [49, 50, 51, 52]. A popular approach involves using line outage distribution factors (LODF) to approximate the changes in real
Chapter 3. Operation Sequence for Proactive Islanding of the Power Grid: Heuristic Methods

Power flows in response to a topology change [51, 52]. LODF-based approximations have also been used for contingency analysis [53]. As LODF calculations depend only on the system topology, this class of methods avoids computationally expensive AC power flow computations. If the network topology changes, then LODFs need to be recomputed. Strategies for speeding up LODF computations have been proposed [54, 55, 56]. This work also utilizes LODF computations to approximate how disconnecting one line changes the loading on the others.

The main contribution of this research is the formulation of fast heuristic methods to determine a sequence of line switching operations needed for separating the power grid into self-adequate islands. We present a forward and backward approach using LODF-based calculations for real power flow redistribution. In the forward-approach, the first step is decided at the start, while in the backward-approach method, the last step is decided first. Numeric tests on a modified IEEE 39-bus test system show that the proposed algorithms are computationally efficient, minimize line overloads, and maintain bus voltages within limits prescribed by ANSI standards [57]. This operation sequence determination problem has not been explored before in existing literature.

3.2 Methodology

As discussed before, proactive islanding may be required to arrest the propagation of cascading disturbances in the power grid after an extreme event. In this section, two heuristic methods for determining the operation sequence needed to create the islands are presented. Standard mathematical notations are used, calligraphic symbols represent sets, lower (upper) case bold letters represent column vectors (matrices). All zero and all one vectors and matrices of appropriate size are denoted by \( \mathbf{0} \) and \( \mathbf{1} \) respectively.

3.2.1 Line Outage Distribution Factor

Network sensitivity factors are widely used in power systems studies to obtain quick estimates of power flow shifts in response to changes in generation or network topology. Closed form expressions for the sensitivity factors are derived using the DC power flow model [58].

Let us consider a power network with \( N + 1 \) buses and \( L \) transmission lines. The set of buses is denoted as \( \mathcal{N} = \{0, 1, \ldots, N\} \), where bus 0 is the slack bus. The set of lines is given by \( \mathcal{L} = \{l_1, l_2, \ldots, l_L\} \). Each line \( l_m \) is associated with an ordered pair of nodes \( (i_m, j_m) \), and it is assumed that real power flow \( f_{lm} \) on line \( l_m \) is directed from node \( i_m \) to node \( j_m \). The injection shift factor \( ISF_{i_m}^i \) is the approximate change in \( f_{lm} \) when 1 MW power is injected at some node \( i \in \mathcal{N} \) and withdrawn at the slack bus. Under typical DC power flow assumptions and lossless conditions, the formula for the ISF matrix is

\[
\text{ISF} = \mathbf{B}_d \mathbf{A} \mathbf{B}^{-1}
\]

Here, \( \mathbf{B}_d \) is the \( (L \times L) \) branch susceptance matrix, \( \mathbf{A} \) is the \( (L \times N) \) reduced incidence
matrix, and $\mathbf{B}$ is the $(N \times N)$ reduced nodal susceptance matrix. Using the ISF matrix, other sensitivity factors like power transfer distribution factor (PTDF) and LODF can be calculated.

PTDFs describe how line flows change when there is a transaction of $\Delta t$ MW from node $i$ to $j$. For line $l_{mn}$, the approximate change in real power flow would be as follows:

$$\Delta f_{lm}^{(i,j,\Delta t)} = PTDF_{lm}^{(i,j)} \times \Delta t \quad (3.1)$$

Now, the PTDFs may be computed in the following manner.

$$PTDF_{lm}^{(i,j)} = ISF_{lm}^i - ISF_{lm}^j \quad (3.2)$$

PTDFs need to be recomputed if the network topology changes. Now, let us consider the impact of an outage on line $l_k$ on $f_{lm}$. With the help of LODF, the fraction of pre-outage real power flow on line $l_k$ (between nodes $(i_k, j_k)$, say) redistributed to the remaining lines can be calculated. It can be shown that [56]:

$$LODF_{l_k}^{l_{mn}} = \frac{PTDF_{lm}^{(i_k,j_k)}}{1 - PTDF_{l_k}^{(i_k,j_k)}} \quad (3.3)$$

Hence, approximate real power flows in a network following an outage on line $l_k$ may be computed as:

$$f^c_S = f^0_S + LODF_{S}^{l_k} \times f^0_k \quad (3.4)$$

Here, the superscripts 0 and c are used to denote pre and post-outage conditions respectively. The real power flow on lines in service are stacked in vector $f_S$ of length $(L - 1)$. Pre-outage real power flow on line $l_k$ is given by $f^0_k$. It must be noted that the values of sensitivity factors depends on the choice of the slack bus. Most power flow solvers used in the industry and academia provide built-in commands for computing PTDF and LODF, once the slack bus is specified.

### 3.2.2 Generalized LODF

With the LODF expression of equation (3.3), approximate redistribution of real power flow can be calculated for a single line outage. Closed form expressions for computing LODF when multiple transmission lines are disconnected have also been derived [56]. It can be shown that

$$LODF_S^O = PTDF_S^O(I - PTDF_O^O)^{-1} \quad (3.5)$$

Here, $O$ denotes the set of lines on outage and $S$ is the set of lines remaining in service. Matrix $I$ is the identity matrix of size $|O| \times |O|$. The PTDF values are calculated for the pre-contingency network topology. Proof of the relationship in equation (3.5) is available in [54, 56]. Once the Generalized LODF (GLODF) values are calculated, post-contingency approximate real power flow distribution can be computed using the following expression.

$$f^c_S = f^0_S + LODF_S^O \times f^0_O \quad (3.6)$$
Algorithm 2 LODF-based forward approach

1: Start
2: $\mathcal{T} \leftarrow$ Initial network topology
3: $k \leftarrow 1$
4: Perform the generation redispatch and load-shedding required to establish load-generation balance in the intended islands. Compute power flows on all transmission lines.
5: Determine the cut-set of lines to be disconnected. Denote it by $\mathcal{L}_{\text{cut}}$. Let $|\mathcal{L}_{\text{cut}}| = n$.
6: Initialize an empty sequence $\{L'_{j}\}_{j=1:n}$ which will store the chosen switching sequence.
7: while $\mathcal{L}_{\text{cut}}$ is non-empty do
8: Calculate LODF matrix for topology $\mathcal{T}$.
9: for $i = (1 : n - k + 1)$ do
10: Compute approximate real power flows when the $i$-th element of $\mathcal{L}_{\text{cut}}$ is disconnected using equation (3.4).
11: Calculate line loadings.
12: $S_i \leftarrow$ sum of loading on the five most heavily loaded lines.
13: $l \leftarrow \mathcal{L}_{\text{cut}}(i)$ for which $S_i$ is minimum.
14: $L'_{k} \leftarrow l$
15: $k \leftarrow k + 1$
16: Disconnect line $l$. $\mathcal{T} \leftarrow$ updated topology.
17: Remove $l$ from $\mathcal{L}_{\text{cut}}$.
18: End
19: Output: Sequence $L'$.

3.2.3 LODF-based Forward-Approach

We propose heuristic methods for formulating the operation sequence needed to separate a transmission network into islands. It is assumed that at the initial stage, generation dispatch and load-shedding needed to establish power balance in the intended islands will be executed. Once the new operating conditions are realized, sequential line disconnections can start. It is further supposed that the lines are to be disconnected one at a time to limit system shock, and there will be sufficient delay between consecutive switching actions so that the system can reach a steady state before the next line disconnected. Dynamic performance has not been considered by the proposed methods, with the assumption that minimizing line congestions will avoid voltage violations, and offer operators adequate opportunity to maintain system stability using power systems stabilizers, voltage regulators, etc. Of course, the heuristic methods could also be used to screen initial candidates, and the optimal operation plan may be chosen after performing dynamic simulation studies. The possibility of component failures during the line switching operations has not been studied herein.

If $n$ transmission lines need to be disconnected to realize the intended islands, then the number of possible switching sequences becomes $n!$, and any brute-force method would have
3.3. Numeric Results

To check \( n \times n! \) power flow cases for possible constraint violations. Needless to say, this would be computationally prohibitive. Using LODFs to approximate changes in real power flows due to changes in grid topology, we formulate a greedy approach that, at each stage, chooses to disconnect the line that produces the least loading on the five highest loaded lines in the post-disconnection topology. The loading on the heaviest lines is minimized so that other parts of the network can be better utilized.

We start with the topology where all branches in the cut-set are connected and disconnect one line at a time. This forward-approach method is summarized in algorithm 2. Of course, if the computation budget allows, accurate line loadings could be computed using AC power flow. The number of power flow cases to be solved would be \( \frac{n(n+1)}{2} - 1 \), which is significantly lower than \( n! \).

3.2.4 GLODF-based Backward Approach

It may be argued that it is critical to ensure that there are no overloads or voltage violations during the later steps of the switching sequence when the system is already weakened. Considering this, we propose a heuristic backward-approach method, as outlined in algorithm 3. Here, we start with the topology where \((n - 1)\) branches in the cut-set have been disconnected. The number of such possible topologies is \(n\), and approximate power flow redistributions are computed for each combination of \((n - 1)\) branch disconnections from the cut-set. The candidate which minimizes loading on the five most heavily loaded branches is selected for the \(n-\)th switching step. Next, the selection process is repeated for topologies with \((n - 2)\) line disconnections and so on and so forth. Again, if computation budget allows, exact line loadings could be calculated by solving AC power flow cases.

3.3 Numeric Results

We illustrate the performance of the proposed sequence determination methods with the help of case-studies on the modified IEEE 39-bus 10-machine test system shown in figure 3.1 [59]. All computations are performed on a 3.6 GHz Intel Core i7-4790 CPU with 16 GB RAM. MATPOWER is used for analyzing the electrical performance of the test system [60].

3.3.1 Experiment Setup

The network in figure 3.1 is to be split into two islands; disconnecting five transmission lines on the boundary. These lines are: \( \mathcal{L}_{\text{cut}} := \{(4,14),(5,6),(5,8),(9,39),(14,15)\} \). Island 1 (shown in color in figure 3.1) has 780 MW load, and 1371 MW generation capacity. Island 2 has 4690 MW load, and 6031 MW generation capacity. Bus 31 and 39 are assigned as the slack buses in islands 1 and 2 respectively. Before network splitting, losses in the network are minimized by redispetching generation (solving AC-OPF with identical cost parameteres for all generators). The load and generation at network buses are shown in table 3.1. As
### Table 3.1: Load and Gen. for the IEEE-39 Bus Test System

<table>
<thead>
<tr>
<th>Bus</th>
<th>$P_{\text{load}}$ (MW)</th>
<th>$Q_{\text{load}}$ (MVar)</th>
<th>$P_{\text{gen}}$ (MW)</th>
<th>$Q_{\text{gen}}$ (MVar)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>97.6</td>
<td>44.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>289.8</td>
<td>2.16</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>300</td>
<td>110.4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>233.8</td>
<td>84.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>522</td>
<td>176.6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>6.5</td>
<td>-66.6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>8.53</td>
<td>88.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>320</td>
<td>153.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>329</td>
<td>32.3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>158</td>
<td>30.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>680</td>
<td>103.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>21</td>
<td>274</td>
<td>115.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>23</td>
<td>247.5</td>
<td>84.6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>24</td>
<td>308.6</td>
<td>-92.2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>25</td>
<td>224</td>
<td>47.2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>26</td>
<td>139</td>
<td>17.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>27</td>
<td>281</td>
<td>75.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>28</td>
<td>206</td>
<td>27.6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>29</td>
<td>283.5</td>
<td>26.9</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>0</td>
<td>452.6</td>
<td>140</td>
</tr>
<tr>
<td>31</td>
<td>9.2</td>
<td>4.6</td>
<td>456.37</td>
<td>274.53</td>
</tr>
<tr>
<td>32</td>
<td>0</td>
<td>0</td>
<td>391.7</td>
<td>214.75</td>
</tr>
<tr>
<td>33</td>
<td>0</td>
<td>0</td>
<td>652</td>
<td>210.2</td>
</tr>
<tr>
<td>34</td>
<td>0</td>
<td>0</td>
<td>508</td>
<td>167</td>
</tr>
<tr>
<td>35</td>
<td>0</td>
<td>0</td>
<td>687</td>
<td>287.43</td>
</tr>
<tr>
<td>36</td>
<td>0</td>
<td>0</td>
<td>580</td>
<td>95.36</td>
</tr>
<tr>
<td>37</td>
<td>0</td>
<td>0</td>
<td>551.78</td>
<td>5.25</td>
</tr>
<tr>
<td>38</td>
<td>0</td>
<td>0</td>
<td>844.36</td>
<td>78.37</td>
</tr>
<tr>
<td>39</td>
<td>1104</td>
<td>250.0</td>
<td>950.84</td>
<td>-60.71</td>
</tr>
</tbody>
</table>
3.3. NUMERIC RESULTS

Algorithm 3 GLODF-based backward-approach

1: Start
2: $k \leftarrow 1$
3: Perform the generation redispatch and load-shedding required to establish load-generation balance in the intended islands. Compute power flows on all transmission lines.
4: Determine the cut-set of lines to be disconnected. Denote it by $L_{\text{cut}}$. Let $|L_{\text{cut}}| = n$.
5: Initialize an empty sequence $\{L'_{j}\}_{j=1:n}$ which will store the chosen switching sequence.
6: $\mathcal{T} \leftarrow$ Network topology with all lines in $L_{\text{cut}}$ disconnected.
7: while $L_{\text{cut}}$ is non-empty do
8:   for $i = (1 : n - k + 1)$ do
9:      $\mathcal{O} \leftarrow L_{\text{cut}} \setminus L_{\text{cut}}(i)$.
10:     In topology $\mathcal{T}$, connect $L_{\text{cut}}(i)$ and compute GLODF for outages on elements of $\mathcal{O}$ using equation (3.5).
11:     Compute approximate real power flow distributions using equation (3.6).
12:     $S_i \leftarrow$ sum of loading on the five most highly loaded lines.
13:     $l \leftarrow L_{\text{cut}}(i)$ for which $S_i$ is minimum.
14:     $L'_{n-k+1} \leftarrow l$
15:     $k \leftarrow k + 1$
16:     Connect line $l$. $\mathcal{T} \leftarrow$ updated topology.
17:     Remove $l$ from $L_{\text{cut}}$.
18:   End
19: Output: Sequence $L'$.

the defensive islanding approach is expected to be deployed only when extreme events are expected, the grid may face emergency operating conditions.

The bus voltages and line loadings in the redispatched network before and after islanding are shown in figure 3.2 and figure 3.3 respectively. Some bus voltages in island 1 are slightly higher than 1.05 p.u. which can be remedied by voltage regulators. Line (3, 4) is loaded to capacity as after islanding, the entire load at bus 4 is being supplied by this line. To alleviate the overload at line (3, 4), some load at bus 4 may be shed.

3.3.2 Operation Sequence

The switching sequence for network separation is determined using algorithms 2 and 3, and the electric performance of the topologies at each step are checked using steady-state AC power flow. Buses 31 and 39 are both assigned as slack buses with equal participation. The operation sequences obtained are compared to a randomly generated sequence of disconnections, as listed in table 3.2. It can be seen that the last step determined by both the forward and backward approach-based methods is the same.
CHAPTER 3. OPERATION SEQUENCE FOR PROACTIVE ISLANDING OF THE POWER GRID: HEURISTIC METHODS

Figure 3.1: Modified IEEE 39-bus test case

Table 3.2: Operation Sequences Determined by Various Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Line Switching Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Sequence</td>
<td>(4, 14), (14, 15), (5, 6), (5, 8), (9, 39)</td>
</tr>
<tr>
<td>Forward-Approach</td>
<td>(9, 39), (14, 15), (5, 6), (5, 8), (4, 14)</td>
</tr>
<tr>
<td>Backward-Approach</td>
<td>(14, 15), (5, 8), (5, 6), (9, 39), (4, 14)</td>
</tr>
</tbody>
</table>

The voltages computed at non-generator buses in the network for each step in the switching sequences are shown in figure 3.4-3.6. Note that voltages at buses 19 and 22 were slightly higher than 1.05 p.u. in the base case itself. It can be seen that in the forward-approach method, the voltage at bus 9 is higher than 1.05 p.u. during steps 1 and 2. For the backward-approach method, no overvoltage at bus 9.

No line was overloaded in both the switching sequences found by the heuristic methods. Average loading on the five highest-loaded lines at each step was similar for both the algorithms. In the random sequence, line (3, 4) was overloaded at steps 2, 3, and 4; and line (6, 7) was overloaded at steps 3 and 4. The number of constraint violations observed for the different switching sequences are summarized in table 3.3.

Hence, we see that the approaches proposed in this work show similar performance and can split the network avoiding line overloads and maintaining voltages within limits. For the power flow-based backward-approach method, fewer overvoltages are recorded. Both
3.3. **Numeric Results**

Figure 3.2: Bus voltages before and after separation into islands

Figure 3.3: Line loading before and after separation into islands

Figure 3.4: Load bus voltages following a random sequence of operations. The black dotted lines correspond to 0.95 and 1.05 p.u.
CHAPTER 3. OPERATION SEQUENCE FOR PROACTIVE ISLANDING OF THE POWER GRID: HEURISTIC METHODS

Figure 3.5: Load bus voltages following the operation sequence chosen by algorithm 2

Figure 3.6: Load bus voltages following the operation sequence chosen by algorithm 3

Table 3.3: Violations Observed for Various Switching Sequences

<table>
<thead>
<tr>
<th>Method</th>
<th>No. of Overloads</th>
<th>No. of Overvoltages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Sequence</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>Forward-Approach</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>Backward-Approach</td>
<td>0</td>
<td>8</td>
</tr>
</tbody>
</table>
methods perform better than a randomly selected sequence of operations.

3.4 Summary

This work presents two heuristic methods for determining the sequence of transmission line switching operations needed to separate the power grid into smaller self-adequate islands. It is shown that the proposed methods are computationally efficient, minimize line capacity violations, maintain bus voltages within prescribed limits, and outperform a random operation sequence for case-studies on the IEEE 39-bus test system. Of course, the approach outlined herein is not without its limitations, and further work is needed to rigorously assess the effectiveness of the proposed algorithms.

Tests on diverse and larger systems will be needed to check the scalability of the proposed approach. The possibility of component failures during line switchings should also be incorporated into the operation plan. Moreover, this work has only considered the steady-state performance of the network and dynamic performance has not been investigated. It is assumed that minimizing operation constraint violations will allow operators to maintain system stability with ad hoc control actions. This work additionally considers that the generation dispatch and load-shedding required to establish power-balance in the intended islands is executed before any lines are disconnected. Establishing power balance may be time-consuming due to ramping constraints on generators, and an operation plan that accounts for redispatch and switching actions simultaneously at each step can be useful.
4.1 Introduction

In recent years, the adoption of renewable energy based distributed energy resources (DERs) has increased due to the recognition of their economic and environmental benefits. A primary advantage of DERs is their ability to sustain local loads if the main grid is lost, possibly due to natural disasters. DERs and loads may be clustered together to form microgrids which supply essential loads and aid service restoration during and after outages, thereby boosting grid resilience [61]. According to the IEEE 1547.4-2011 standard, microgrids: 1) have DERs and load; 2) can operate in both grid-connected and islanded modes; and 3) are intentionally planned [62].

Historically, industrial customers have used diesel generators (DGs) to ensure emergency power supply. However, other DER options such as solar generators can also provide efficient, onsite energy [63]. Similarly, combined heat and power plants (CHPs) can alleviate grid stress and reduce the energy cost at industrial sites [64]. As CHPs are ‘always-on’ units primarily run to serve industrial needs, they can be quickly deployed to provide grid-support during stressed conditions. The energy storage capacity of modern electric vehicles (EVs) may also be exploited to enhance grid resilience [65]. Reference [66] suggests that EVs can be encouraged to interact with the grid through incentivized charging schemes offered by public parking lots. Such parking lots often have a large number of EVs parked at a time. Thus, under a suitable policy framework, utilities may tap into their stored energy during emergencies. A promising direction for using electric buses as mobile energy storage units is explored in [67]. As larger EVs like buses and trucks have very high storage capacity, they can be proactively dispatched as back-up generators to aid areas facing natural disasters. Hence, fixed and mobile DERs provide multiple avenues to boost grid resilience, and effective
strategies should be designed to utilize them.

Note that isolated DERs cannot supply local loads if they lack adequate protection and control capabilities. When DERs are inadvertently separated from the grid (a condition called *unintentional islanding*), electric utilities cannot regulate the voltage and frequency in the unplanned islands. Voltage and/or frequency excursions outside prescribed limits may severely damage system loads and also pose safety concerns. Hence, several measures to detect and avoid unintentional islanding have been developed [68]. On the other hand, the generation capacity of DERs may be exploited to mitigate outages if they can be designed to operate safely in islanded conditions. Therefore, utilities are interested in identifying potential candidates within existing active distribution networks that can be converted to microgrids via economically viable retrofitting. Microgrids need to be equipped with adequate switching, protection and control mechanisms prior to any expected outage. Thus, optimally splitting a network into microgrids constitutes a pertinent planning problem [69, 70, 71, 72, 73, 74, 75].

This optimal distribution network partitioning (ODNP) task seeks to identify potential *self-adequate* sub-networks that can survive the loss of the main grid as islands. Both exact [69, 70, 71] and heuristic [72, 73, 74, 75] methods have been proposed in literature to address the ODNP problem. Self-adequacy in the objective function has been surrogated by either expected load-generation imbalance within microgrids [73, 74, 75], or expected power flow on boundary lines [69, 70, 71, 72]. Some works additionally suggest dynamic identification of boundary lines in response to faults [76, 77]. However, since distribution networks are only served by a limited number of switching devices at present, implementing these suggestions may be difficult in practice. A method for determining self-sufficient islands in transmission networks is described in [38], but cannot be directly extended to ODNP without including distribution system specific constraints.

Distribution networks are usually operated in a radial fashion for protection coordination, and this radiality needs to be maintained in microgrids as well. In [71, 73, 74, 75], ODNP is demonstrated on an already radial feeder and radiality conditions are not explicitly enforced. This approach ignores the presence of normally open switches that allow network reconfiguration, thereby under-utilizing grid flexibility. Radiality is considered in [70], but another restrictive condition is imposed- each microgrid is assigned exactly one DER. This single DER constraint is also present in [76]. In this approach, the number of partitions are predetermined, leading to sub-optimal solutions. A radiality constraint without specifying the number of microgrids was recently presented in [69], and the formulation in the current work builds upon this approach.

A critical aspect that has been overlooked in the existing microgrid planning literature is the requirement of grid-forming generators in viable islands. The IEEE 1547.4-2011 standard mandates that an island should have at least one generator that provides voltage and frequency support during a system disturbance, or has black-start capabilities [62]. An exhaustive path search based method for checking connectivity to black-start generators has been proposed in [78]. Another multiple commodity flow based approach outlined in [79]
separately checks nodes for their connectivity to black-start nodes. Both these approaches become computationally prohibitive for large meshed networks. The ODNP formulation put forth in the present work guarantees that all nodes in each microgrid will be connected to at least one grid-forming generator. The formulation is somewhat similar to the single commodity flow model of [79] but uses fewer constraints and shows faster performance (empirically observed to be 10 to 20% faster).

ODNP is further complicated by the fact that power demands and renewable energy generation are stochastic in nature. In [71, 73, 74, 75, 77], the uncertainty in load and generation is addressed by constructing typical daily profiles, over which optimization is performed. However, the quality of solution obtained is not evaluated. The present work formulates a chance-constrained ODNP (cc-ODNP) to identify optimal microgrids in the planning stage. This is computationally challenging as the underlying deterministic optimization problem is combinatorial, and is formulated as a non-convex mixed integer linear program (MILP). To ensure computational tractability, a sampling and integer programming based strategy has been used to solve an approximation of the cc-ODNP, and the quality of the solution obtained is rigorously evaluated using statistical tools [80].

The main contributions of this work are stated next.

- First, a deterministic ODNP problem for identifying optimal microgrids given real-time load-generation values is formulated. This formulation comprehensively addresses operational constraints, including: a) maintaining network radiality, and b) ensuring every microgrid includes a grid-forming generator, without any pre-assignment. The requirement of grid-forming generators has not been addressed in the existing work on microgrid planning.

- Second, the ODNP formulation is extended to a probabilistic setup, and solved using a computationally tractable sample average approximation (SAA) method. Quality of an obtained solution is evaluated considering: a) an upper bound on the probability that the candidate microgrids are supply-deficient, and b) a lower bound on the objective value for the probabilistic optimization problem.

- Third, the ODNP formulation is evaluated through extensive numerical tests on a modified IEEE 37-bus feeder. It is shown that the SAA approach is able to efficiently utilize network flexibility, and outperforms a robust clustering based method in terms of objective cost.

4.2 Power systems resilience

Resilience may be defined as the ability of a system to prepare for, respond to and recover from natural and man-made disasters. Recent natural disasters have exposed vulnerabilities in the power systems infrastructure, necessitating strategies to enhance its resilience. However, to evaluate the effectiveness of competing strategies, some quantitative measure for capturing the rather conceptual idea of resilience is required. Several recent research efforts have sought to describe resilience in quantitative terms and model the inter-dependencies
A popular approach to quantify resilience builds around the multi-phase resilience trapezoid that depicts system behavior during a disaster; see figure 4.1. Consider a disruptive event that starts at time $t_e$. This first phase ($t \in [t_e, t_d]$) of event progress is characterized by an ongoing deterioration in system performance from its pre-disturbance levels. In the second phase ($t \in [t_d, t_r]$), the system operates in a diminished capacity for some time before restoration steps can be initiated. During the third phase ($t \in [t_r, t_n]$), actions are taken to restore the system to its pre-event performance levels. The transitions between system states in figure 4.1 are linear only for expository convenience. In practice, these may be non-linear, depending on system characteristics, prevalent conditions and the nature of the disaster.

Note that in figure 4.1, the $y$-axis represents a generic system function as its performance measure. Different quantities have been used to convey system performance, for example, reference [9] considers the power supplied to critical loads weighted by their criticality. The formulation presented herein considers the total load served by microgrids once the main grid is lost. Maximizing microgrid service will in turn reduce load interruptions and boost traditional grid reliability measures such as SAIDI (System Average Interruption Duration Index), SAIFI (System Average Interruption Frequency Index) and MAIFI (Momentary Average Interruption Frequency Index) [12].

The resilience trapezoid describes - a) the rate and extent of performance drop when a disruptive event strikes, b) how long the system resides in a degraded state, and c) how fast it recovers to the pre-disturbance state. A metric that combines these measures and quantifies the resilience of a system is the area under the curve (AUC) of the trapezoid. Any resilience enhancement strategy attempts to maximize this AUC. Figure 4.1 shows how ODNP proposes to increase the AUC of the resilience trapezoid. As shown by the dotted
line, if a distribution network is not equipped with microgrids, a loss of the grid may result in a drastic decrease in load served despite installed DER capacity. By optimally planning microgrids with existing DERs, the system behavior can be changed to the solid line shown in figure 4.1. The proposed ODNP formulation maximizes the load served during the degraded state of second phase. Thus, the shaded portion shown in the second phase of the resilience trapezoid is evidently maximized, thereby increasing the AUC. Further, by improving system performance in the degraded state, one can anticipate a faster recovery (from $t'_n$ to $t_n$ in figure 4.1), further increasing the AUC. Of course, the recovery time will depend on multiple factors such as time and nature of disturbance, pre-event system operating state etc. Nevertheless, the ODNP formulation provides a general methodology for enhancing distribution network resilience, that can be adapted to incorporate different DER technologies.

4.3 Preliminaries

In this section, some mathematical preliminaries are revisited before expounding on the problem formulation. Standard mathematical notations are used, calligraphic symbols represent sets, lower (upper) case bold letters represent column vectors (matrices). All zero and all one vectors and matrices of appropriate size are denoted by $\mathbf{0}$ and $\mathbf{1}$ respectively.

4.3.1 Graph Theory

A graph $\mathcal{G} := (\mathcal{V}, \mathcal{E})$ consists of a vertex set $\mathcal{V}$ and an edge set $\mathcal{E}$, where an edge is an unordered pair of distinct vertices of $\mathcal{G}$. Edge $e_{ij} \in \mathcal{E}$ is denoted by its incident vertices $(i, j)$, such that $i, j \in \mathcal{V}$. If $e_{ij} \in \mathcal{E}$, then vertices $i$ and $j$ are adjacent. Two edges are adjacent if they have a common vertex. A subgraph of $\mathcal{G}$ is a graph $\mathcal{H} := (\mathcal{X}, \mathcal{Y})$ such that $\mathcal{X} \subseteq \mathcal{V}$ and $\mathcal{Y} \subseteq \mathcal{E}$. If $\mathcal{X} = \mathcal{V}$, then $\mathcal{H}$ is a spanning subgraph of $\mathcal{G}$. $\mathcal{H}$ is an induced subgraph of $\mathcal{G}$ if vertices in $\mathcal{X}$ are adjacent in $\mathcal{H}$ if and only if they are adjacent in $\mathcal{G}$.

A path from $i$ to $j$ is a sequence of distinct vertices starting at $i$ and ending at $j$ such that consecutive vertices are adjacent. If there is a path between all pairs of vertices of a graph $\mathcal{G}$, then $\mathcal{G}$ is connected; else $\mathcal{G}$ is disconnected. An induced subgraph of $\mathcal{G}$ that is maximal, subject to being connected, is called a connected component of $\mathcal{G}$. A cycle is a sequence of adjacent edges without repetition that starts and ends at the same node. A graph with no cycles is called acyclic. A connected and acyclic graph is a tree. A spanning tree subgraph of $\mathcal{G}$ is a tree that covers all vertices in $\mathcal{G}$. An acyclic graph with multiple connected components is a forest. A spanning forest subgraph of $\mathcal{G}$ is a forest that covers all vertices in $\mathcal{G}$. Spanning forests may include connected components with a single node. For further reference, a review of graph theory fundamentals is available in [81].
4.3.2 Chance-Constrained Optimization

Stochastic optimization refers to a collection of methods for solving an optimization problem with uncertain parameters. For many real-world applications operating in uncertain environments, ensuring 100% reliability is physically and economically impractical. This difficulty is often dealt with by designing systems that assure a minimum reliability level with high probability. Mathematical models of such reliability-constrained systems involve the use of probabilistic or chance constraints \[80\]. A generic chance-constrained optimization (CCO) problem is of the form:

\[
\begin{align*}
\min_{x \in X} & \quad f(x) \\
\text{s. to} & \quad h(x) \leq 0 \\
& \quad \Pr\{g(x, \xi) \leq 0\} \geq 1 - \varepsilon
\end{align*}
\]

Here, \( x \) is the vector of decision variables, whose feasible region is given by \( X \subset \mathbb{R}^n \). The objective function to be minimized is \( f : \mathbb{R}^n \to \mathbb{R} \). Vector \( \xi \) stacks the uncertain parameters with known probability distribution, and \( \varepsilon \in (0, 1) \) is a tunable risk parameter. Problem \( P_1 \) seeks to find an optimal decision vector \( x^* \) that minimizes \( f(x) \), such that the hard constraints \( C_1 \) are always satisfied, while the chance constraint \( C_2 \) is satisfied with probability at least \( 1 - \varepsilon \).

In power systems literature, CCO has been previously used to address security constrained economic dispatch and unit commitment problems \[82\]. This class of problems is difficult to solve, due to two main reasons:

- Given a candidate solution \( \bar{x} \in X \), accurately computing \( \Pr\{g(\bar{x}, \xi) \leq 0\} \) can be very difficult, making it hard to check if constraint \( C_2 \) is satisfied.

- The feasibility region defined by a chance constraint is usually not convex \[80\]. This makes finding an optimal solution difficult even when the feasibility of \( \bar{x} \) can be checked.

These difficulties may be overcome by considering a sample average approximation (SAA) of the original problem where the true distribution of \( \xi \) is replaced by an empirical distribution with discrete support. The SAA is still a chance-constrained stochastic problem, but with a different distribution for \( \xi \), and may be solved via integer programming \[80\]. This method has been shown to yield good candidate solutions if the sampling is ample and rich. In this work, the SAA approach will be incorporated to solve a probabilistic ODNP and the solution obtained will be further analyzed to verify how well it solves the original optimization problem.

4.4 Problem Formulation

Given a distribution network with DERs, planners would like to optimally construct microgrids, such that the DERs are able to sustain internal loads if supply from the main grid is
lost. Load served is to be maximized. Both load and generation vary with the weather and
assuring self-adequacy for the worst case may lead to very conservative solutions. Hence, a
solution that works well for most operating conditions might be preferable. Thus microgrids
may be designed to be self-adequate with probability at least \((1 - \varepsilon)\) across all possible oper-
ating scenarios, where \(\varepsilon\) is a tunable risk parameter. The value of \(\varepsilon\) may be chosen based on
several factors such as available storage resources. Once optimal microgrids are identified,
they need to be equipped with control capabilities and boundary line switches. It must be
noted that depending on the generation capacity of installed DERs, all load in a network
may not be served by microgrids.

Our mathematical formulation is put forth in three steps. First, a deterministic version of
the problem, d-ODNP is presented where the load served is maximized for a given scenario
of demands and generation. Next, the chance constraints arising from the randomness in
generation and demands are added. Finally, a SAA based algorithm is proposed that can
tractably solve the probabilistic ODNP.

### 4.4.1 Distribution Network Model

A single-phase distribution network may be represented by a connected directed graph \(G_N := (V_N, E_N)\), where vertices denote buses and edges denote lines. The substation node is indexed by 0; and the set of all other nodes is denoted by \(V := V_N \setminus \{0\}\). Each edge \(e_{ij} \in E_N\) is assigned an arbitrary direction from node \(i\) to \(j\). If \(e_{ij} \in E_N\), then \(e_{ji} \notin E_N\). The task at hand considers that the main grid is unavailable, hence partitioning needs to be carried out on \(G := (V, E)\), the induced subgraph of \(G_N\) on vertex set \(V\). In the present setup all lines are considered switchable. Any non-switchable edge coinciding with a microgrid boundary would need to be retrofitted with a switch. Moreover, graph edges include lines with existing normally open and normally closed switches, and hence \(G\) is not necessarily radial.

Each node has an associated demand \((\xi_i^{dp} + j\xi_i^{dq})\) and generation capacity \((\xi_i^{gp} + j\xi_i^{gq})\). The demand and generation capacities are not precisely known at the planning stage and only a probability distribution, possibly empirical, may be available. Let \(v_i\) be the voltage magnitude at bus \(i\) and \((p_i + jq_i)\) be the complex power injection. Bus voltages, demand, generation capacity, and complex power injections are respectively stacked into vectors \(v\), \(\xi^{dp}, \xi^{dq}, \xi^{gp}, \xi^{gq}, p + jq\). All quantities are in per units.

Let us introduce binary variables \(b^n_i\) and \(b^{en}_{ij}\) that respectively dictate if vertex \(i \in V\) and edge \(e_{ij} \in E\) are energized. The end nodes \(i\) and \(j\) of an energized edge \(e_{ij}\) shall be energized, requiring

\[
    b^n_i + b^n_j \geq 2b^{en}_{ij} \quad \forall e_{ij} \in E.
\]

The ANSI standards mandate that voltages at energized buses should be within \(\pm 5\%\) p.u.
of the nominal value \([57]\). Thus,

\[
    0.95 b^n \leq v \leq 1.05 b^n.
\]
4.4. PROBLEM FORMULATION

Let the power flow on line \( e_{ij} \in \mathcal{E} \) be \( P_{ij} + jQ_{ij} \). The line capacity constraints may be formulated as follows.

\[
- b_{ij}^e P_{ij}^{\text{max}} \leq P_{ij} \leq b_{ij}^e P_{ij}^{\text{max}} \quad \forall e_{ij} \in \mathcal{E}
\]

(4.3a)

\[
- b_{ij}^e Q_{ij}^{\text{max}} \leq Q_{ij} \leq b_{ij}^e Q_{ij}^{\text{max}} \quad \forall e_{ij} \in \mathcal{E}
\]

(4.3b)

Flow constraints of the form \( P_{ij}^2 + Q_{ij}^2 \leq S_{ij}^2 \) (where \( S \) denotes apparent power) are not used here to avoid quadratic constraints. A polytopic approximation of this constraint proposed in [83] could also be used.

4.4.2 Power Flow Model

DER units plausible in a low/medium voltage distribution network setup include photovoltaic (PV) generators, diesel generators (DGs) and combined heat and power plants (CHPs). The active power generation capacity of these units maybe stochastic; for instance PV generation is dictated by solar irradiance levels, and CHP generation is affected by local heating demands. Non-utility scale PV generators are usually not dispatchable, while some DER units like CHPs may be dispatchable limited by their stochastic generation capacity. DG units are usually dispatchable and their maximum generation capacity may be fixed based on the machine rating. Loads are assumed to be inelastic.

The power generations by dispatchable units are governed by the following equations:

\[
0 \leq p_i^g \leq b_i^p \xi_i^{g_p} \quad \forall i \in \mathcal{V}_D
\]

(4.4a)

\[
- b_i^p \xi_i^{g_q} \leq q_i^g \leq b_i^q \xi_i^{g_q} \quad \forall i \in \mathcal{V}_D.
\]

(4.4b)

Here, constraints (4.4a)-(4.4b) bound the generator outputs \( p_i^g, q_i^g \) at bus \( i \). Set \( \mathcal{V}_D \) comprises of all nodes with a dispatchable generator. Given the limit on active power, a corresponding limit on reactive power generation can be obtained based on the generator apparent power ratings. Non-dispatchable generators are incorporated into the formulation as shown below.

\[
p_i^q = b_i^p \xi_i^{g_p} \quad \forall i \in \mathcal{V} \setminus \mathcal{V}_D
\]

(4.5a)

\[
q_i^q = b_i^p \xi_i^{g_q} \quad \forall i \in \mathcal{V} \setminus \mathcal{V}_D
\]

(4.5b)

Constraints (4.5a)-(4.5b) state that the power output by an energized non-dispatchable DER is equal to its stochastic generation capacity. For inelastic loads, the constraints for power consumption become:

\[
p_i^d = b_i^p \xi_i^{d_p} \quad \forall i \in \mathcal{V}
\]

(4.6a)

\[
q_i^d = b_i^p \xi_i^{d_q} \quad \forall i \in \mathcal{V}
\]

(4.6b)

Constraints (4.6a)-(4.6b) state that consumption at bus \( i \) is equal to its demand; if energized. Thus the net power injections are:

\[
p_i = p_i^q - p_i^d \quad \forall i \in \mathcal{V}
\]

(4.7a)

\[
q_i = q_i^q - q_i^d \quad \forall i \in \mathcal{V}
\]

(4.7b)
For power flow calculations, the linearized distribution flow (LDF) model proposed in [84] is followed. Despite being an approximation for the full AC power flow model, LDF has been used extensively and shown to perform well in literature [85]. Now, ignoring line losses, the power balance at each node entails:

\[ \sum_{e_{ij} \in E} P_{ij} - \sum_{e_{jk} \in E} P_{jk} = p_j \quad \forall j \in V \]  
\[ \sum_{e_{ij} \in E} Q_{ij} - \sum_{e_{jk} \in E} Q_{jk} = q_j \quad \forall j \in V \]  

(4.8a) \hspace{2cm} (4.8b)

Let \( r_{ij} + jx_{ij} \) be the impedance of line \( e_{ij} \in E \). Then, the relationship between voltages and power injections may be linearized as: \( v_i^2 - v_j^2 = 2(r_{ij}P_{ij} + x_{ij}Q_{ij}) \). Assuming small voltage deviations, the squared terms may be approximated as \( v_i^2 \simeq 2v_i - 1 \). Using these results,

\[ b_{ij}^e(v_i - v_j - r_{ij}P_{ij} - x_{ij}Q_{ij}) = 0, \quad \forall e_{ij} \in E \]  

(4.9)

Here, the indicator \( b_{ij}^e \) is multiplied to enforce the voltage drop relation only for energized lines. For computational ease, bilinear terms like \( b_{ij}^e v_i \) in (4.9) can be handled by McCormick linearization, where the product terms are replaced by their linear convex envelopes to yield a relaxation of the original non-convex feasible set [86]. If at most one of the variables is continuous and the rest are binary, this relaxation is exact. For illustration, let us consider a term \( z = xy \), where \( x \) is binary and \( y \) is a continuous variable bounded in \( y \in [y, \bar{y}] \). Here, \( z = xy \) may be equivalently expressed as four linear inequality constraints.

\[ xy \leq z \leq x\bar{y} \]  
\[ y + (x - 1)\bar{y} \leq z \leq y + (x - 1)y \]  

(4.10a) \hspace{2cm} (4.10b)

Note that putting \( x = 0 \) in equations (4.10a)-(4.10b) yields \( z = 0 \). Similarly, putting \( x = 1 \) yields \( z = y \). All such bilinear terms appearing henceforth in this work will be treated similarly.

4.4.3 Radiality Constraints

Network radiality is essential for distribution system operations. Some prior approaches proposed for enforcing radiality are cycle elimination [78] and virtual commodity flow [69, 87]. The ODNP task needs to identify \( G' = (V', E') \), a spanning subgraph of \( G \), such that every connected component, or simply component, of \( G' \) is a tree, i.e. \( G' \) is a forest. A spanning forest may include isolated nodes, i.e. it may have components with a single vertex and no edges.

As discussed in section 4.1, the radiality formulation outlined in this chapter has the following advantages: a) the optimal number of microgrids and their topologies are determined in a single-shot, and b) DERs need not be pre-assigned to microgrids like in [70, 76]. The radiality constraints will be formulated using the following proposition.
4.4. Problem Formulation

Figure 4.2: Components $\{(A, B, C), \{AB, BC\}\}$, $\{(D, E, F), \{DE, DF\}\}$ and $\{(G), \{}\}$ form a spanning forest. Adding edges $BD$ and $DG$ creates a spanning tree.

**Proposition 1.**\cite{(69)} Given a spanning forest subgraph $F := (\mathcal{V}, \mathcal{E}_F)$ of a connected graph $G := (\mathcal{V}, \mathcal{E})$, there exists at least one spanning tree subgraph of $G$, expressed as $T := (\mathcal{V}, \mathcal{E}_T)$, such that $\mathcal{E}_F \subseteq \mathcal{E}_T \subseteq \mathcal{E}$.

In other words, some edges may be removed from a spanning tree to obtain a spanning forest. This idea is illustrated in figure 4.2. The solid lines show edges in a spanning forest and the addition of dashed edges creates a spanning tree. Hence, the radiality of $G' := (\mathcal{V}, \mathcal{E}')$ holds true if there exists a fictitious spanning tree subgraph $T := (\mathcal{V}, \mathcal{E}_T)$ of $G$ such that $\mathcal{E}_F \subseteq \mathcal{E}_T \subseteq \mathcal{E}$.

Let us first establish a condition to select a spanning tree, and then extract the required spanning forest from it. The base topology of $G$ may be captured by a branch-bus incidence matrix $\mathbf{A}$ of dimension $|\mathcal{E}| \times (|\mathcal{V}| - 1)$, with the following entries:

$$\mathbf{A}_{e_{ij},k} := \begin{cases} 1 & , \ k = i \\ -1 & , \ k = j \\ 0 & , \ \text{otherwise} \end{cases} \quad (4.11)$$

The first column $\mathbf{a}_1$ of $\mathbf{A}$ may be separated as $\mathbf{A} = [\mathbf{a}_1 \mathbf{A}]$. This yields the reduced branch-bus incidence matrix $\mathbf{A}$ of $G$. An efficient model for imposing graph connectivity put forth in \cite{(87)} posits that a graph with vertex set $\mathcal{V}$ and reduced branch-bus incidence matrix $\mathbf{A}$, is connected if and only if there exists a vector $\mathbf{f} \in \mathbb{R}^{|\mathcal{E}|}$, such that $\mathbf{A}^T \mathbf{f} = 1$. For proof, see \cite{(87)}.

For a physical interpretation, consider every vertex in $\mathcal{V} \setminus \{1\}$ injects unit virtual commodity into the network represented by the graph. Then, $\mathbf{f}$ denotes the flow of commodities on the edges. If this flow setup is feasible, then there must be a withdrawal of $|\mathcal{V}| - 1$ units at vertex 1, and every vertex in $\mathcal{V} \setminus \{1\}$ must have a path to reach vertex 1. Vertex 1 may be arbitrarily chosen.

Stating a well-known result from graph theory, a tree with $n$ vertices has exactly $n - 1$ edges.
Hence, the radiality constraints are stated as:

\[ A^T f = 1 \]
\[-(|V| - 1)\theta \leq f \leq (|V| - 1)\theta, \]
\[ \theta \in \{0, 1\}^{|E|} \]
\[ 1^T \theta = |V| - 1 \]
\[ b^e \leq \theta \]

Constraints (4.12a)-(4.12d) ensure that auxiliary binary indicator variables \( \theta \) on the edge-set of the base graph describe a spanning tree. Then, (4.12e) states that the edges selected via the binary variables \( b^e \) are a subset of this spanning tree; and hence form a spanning forest structure as per Proposition 1. As the radiality constraints are thus posed, the number of components in \( G' \) need not be pre-assigned.

### 4.4.4 Connection to grid-forming generators

As stated in Section 4.1, the IEEE 1547 standard mandates that every viable microgrid should have a grid-forming generator that provides voltage and frequency support when isolated from the main grid. In practice, such a unit could be a CHP, a DG, or an inverter-interfaced PV generator with storage.

In the ODNP formulation described in this work, it is guaranteed that every energized bus in the network is connected to a grid-forming generator, or alternatively every microgrid has at least one grid-forming unit. Let \( V_s \) be the set of buses with grid-forming capabilities. Then, the connectivity constraints may be stated as follows:

\[ \sum_{e_{jk} \in E} f'_{jk} - \sum_{e_{ij} \in E} f'_i j = b^e_j, \forall j \in V \setminus V_s \]
\[ -(|V| - |V_s|)b^e \leq f' \leq (|V| - |V_s|)b^e \]

Constraint (4.13a) states that every energized non grid-forming node in \( G \) injects unit virtual commodity into the network. Constraint (4.13b) bounds flows on energized lines and fixes flows on de-energized ones at 0. Here, \( f' \in \mathbb{R}^{|E|} \) is a vector representing virtual line flows. It must be emphasized that \( f' \) is different from \( f \) in (4.12a). Both these vectors are used to impose connectivity conditions, and have no physical significance related to the power flow quantities. Again, this setup is feasible only when all units injected by energized non grid-forming nodes can be withdrawn at grid-forming nodes.

Some grid-forming nodes may not be energized in the optimal topology. This is implicitly considered through the topological constraint in (4.1), that ensures all edges connected to a de-energized node are de-energized as well. Therefore, no path exists from an energized non grid-forming to a de-energized grid-forming node. A potential microgrid candidate may host multiple generators with grid-forming capabilities. In such cases, only one generator shall
serve as the master unit, and suitable power sharing strategies may be required; see [88] and references therein.

4.4.5 Deterministic ODNP

The deterministic ODNP problem (d-ODNP) is solved for one realization of the power generation capacity \((\xi_g^p, \xi_g^q)\), and demands \((\xi_d^p, \xi_d^q)\). The central idea is to sustain maximum load through microgrids if supply from the main grid is lost, thereby minimizing service interruption. Therefore, the objective for d-ODNP becomes maximizing load served. The entire deterministic optimization setup may be mathematically expressed as follows.

\[
\begin{align*}
\min & \quad -1^T p^d \\
\text{s. to} & \quad (4.1) - (4.9), (4.12a) - (4.12e), (4.13a) - (4.13b)
\end{align*}
\]

The relative priority of loads has not been considered in (ODNP-1). However, this cost may be modified by assigning weights to loads in proportion to their criticality.

4.4.6 Probabilistic ODNP

The problem (ODNP-1) applies to one realization of the generation-demand scenario variables \((\xi_g^p, \xi_g^q, \xi_d^p, \xi_d^q)\). However, a more realistic goal would be to identify microgrids that are optimal in some sense for a large set of realizations of the generation-demand scenarios. In the latter setup, the decision variables \(\psi_1 := \{b^n, b^e, f, f', \theta\}\) shall remain fixed for all realizations of the uncertainties. The realization dependent variables would be \(\psi_2 := \{p^d, q^d, p^g, q^g, v, p, Q\}\). Collecting all the uncertainties in \(\xi := \{\xi_g^p, \xi_g^q, \xi_d^p, \xi_d^q\}\), the probabilistic ODNP may seek to solve-

\[
\begin{align*}
\min & \quad - \mathbb{E} [1^T p^d] \\
\text{s. to} & \quad \Pr (\exists \psi_2 | (I ((4.1) - (4.9), (4.12a) - (4.13b)) = 1) \geq 1 - \epsilon)
\end{align*}
\]

The probabilistic constraint in (ODNP-2) is very difficult to enforce in practice. However, we will next discuss some reformulations that simplify the setup without loss of generality. First, note that if the power demands at all nodes are zero, then for a feasible \(\psi_1\) satisfying (4.1) and (4.12a)-(4.13b), there always exists a \(\psi_2\) that satisfies all other constraints. Therefore, the probabilistic constraint may be equivalently posed by enforcing all constraints other than (4.6a)-(4.6b) as hard constraints, and putting the probabilistic requirement on (4.6a)-(4.6b). Setting aside (4.6b) for expository convenience, notice that the equality constraints in (4.6a) may be decomposed into the following inequality constraints.

\[
\begin{align*}
p_i^d - b^n_i c_i^{dp} & \leq 0, \quad \forall i \\
-p_i^d + b^n_i c_i^{dp} & \leq 0, \quad \forall i
\end{align*}
\]
Now, (4.14a) can be posed as a hard constraint, leaving (4.14b) as the main chance constraint. To reiterate, the ODNP task seeks to identify potential microgrids within an existing distribution network, such that load served is maximized, and the microgrids are self-adequate with probability at least \((1 - \varepsilon)\). Islanded microgrids are called self-adequate when their internal load can be met by their internal generation. Mathematically, \(\Pr(-p^d_i + b^n_i \xi_i \leq 0, \forall i) \geq 1 - \varepsilon\) (4.15)

The self-adequacy condition of microgrids can hence be posed at the node level since constraints (4.12a)-(4.12e) ensure that the network topology is a spanning forest, and loads in a microgrid can be supplied only by generators within the same microgrid. If needed, one may relax the reliability requirement by modifying (4.15) slightly. For instance, the condition that a microgrid should be able to serve 90\% of its internal load could be written as \(\Pr(-p^d_i + 0.9 \times b^n_i \xi_i \leq 0, \forall i) \geq 1 - \varepsilon\).

### 4.4.7 Sample Average Approximation

Recall from Section 4.3.2 that a chance constraint needs to be satisfied with a probability specified by a risk parameter \(\varepsilon\). The chance-constraint \((C_2)\) may also be rewritten as \(q(x) \leq \varepsilon\), where \(q(x) = \Pr(g(x, \xi) > 0)\). Let \(\xi^1, \xi^2, ..., \xi^N\) be \(N\) independent and identically distributed (iid) samples of the uncertainty vector \(\xi\); then \(\hat{q}_N(x)\), an estimator of \(q(x)\) is equal to the proportion of realizations in the sample where \(g(x, \xi^i) > 0, i = 1, .., N\). This is a sample average approximation of the chance-constrained problem \((P_1)\) for the samples \(\xi^1, \xi^2, ..., \xi^N\):

\[
\min_{x \in X} f(x) \quad \text{s.t.} \quad \hat{q}_N(x) \leq \gamma \quad (P_2)
\]

Here, \(\gamma \in (0, 1)\) and is the risk level for the SAA problem. Assuming that the SAA can be solved, a) if \(\gamma < \varepsilon\), and \(N\) is sufficiently large, SAA is a restriction on the true problem and a feasible solution of SAA is likely to be feasible for the true problem as well, b) if \(\gamma > \varepsilon\), SAA is a relaxation of the true problem and the optimal value of SAA is likely to be a lower bound to the optimum for true problem. It can be shown that for \(\gamma = \varepsilon\), the SAA optimum approaches its true counterpart with probability one as \(N\) approaches infinity [80].

The chance-constrained SAA problem \(P_2\) can be solved using MILP for \(N\) iid samples of \(\xi\) as shown below [80].

\[
\min_{x \in X} f(x) \quad (P_3)
\]

s. to \(h(x) \leq 0\) \quad (4.16a)

\[
g(x, \xi^\alpha) \leq M(1 - z_\alpha), \quad \alpha = 1, 2, \ldots, N \quad (4.16b)
\]

\[
1^T z \geq (1 - \gamma)N \quad (4.16c)
\]

\[
z \in \{0, 1\}^N, \quad \alpha = 1, 2, \ldots, N \quad (4.16d)
\]

Here, \(\alpha\) is used to index samples of \(\xi\); \(z_\alpha\) is a binary variable and \(M\) is a large number such that \(M > \max_{x \in X} g(x, \xi^\alpha)\) for all \(\alpha = 1, 2, \ldots, N\). Vector \(z\) stacks all \(z_\alpha\) values. In
4.4. Problem Formulation

Figure 4.3: Modified IEEE 37-bus benchmark feeder showing the location of DER units and normally open switches.

Constraint (4.16b), if \( z_\alpha = 1 \), then the chance constraint is not violated. If \( z_\alpha = 0 \), then no bound is imposed. The cardinality constraint in (4.16c) bounds the proportion of constraint violations.

For ODNP the hard constraints are given by \( \{(4.1) - (4.4b), (4.6b) - (4.9), (4.12a) - (4.13b), (4.14a) \forall \alpha\} \). Probability of violating the chance-constraint \( \{(4.14b) \forall \alpha\} \) is to be bounded. Equation (4.16b) becomes:

\[
-p_d^\alpha + \text{diag}(b^n) \times \xi^{dp,\alpha} \leq M(1 - z_\alpha) \times 1, \quad \forall \alpha
\]  

(4.17)

Putting everything together, the problem becomes:

\[
\text{min} \quad -\frac{1}{N} \sum_{\alpha=1}^{N} 1^T p_d^\alpha \\
\text{s. to} \quad p_d^\alpha \leq z_\alpha \times \xi^{dp}, \quad \forall \alpha
\]

(4.1) - (4.4b), (4.12a) - (4.13b), (4.14a) - (4.16c) - (4.17)

(ODNP-3)

The objective function in (ODNP-3) is the sample-based estimator of the objective in (ODNP-2) designed to maximize average load served across considered scenarios. Constraint (4.18) fixes bus consumptions at zero when constraint (4.14b) is not satisfied. This motivates the optimal solution for the ODNP to be one that also increases \( 1^T z \), lower bounded by \( (1 - \gamma)N \).

The optimal topology obtained by solving (ODNP-3) is determined by vectors \( b^{n*} \) and \( b^{e*} \). In practice, microgrids can be isolated by opening the boundary lines between energized nodes and deenergized nodes.
4.5 Solution Validation

Consider a candidate solution \( \bar{x} \) found by the SAA approach of (ODNP-3). To adjudge its quality, two aspects need to be analyzed: 

a) Can it be said with some desired confidence that \( \bar{x} \) a feasible solution for the true problem (ODNP-2)?

b) If yes, then how far is \( f(\bar{x}) \) from the optimal value \( f(x^*) \)?

A method for checking an upper bound of \( Pr\{g(\bar{x}, \xi) > 0\} \) and lower bound on \( f(x^*) \) is shown in [80] and references therein.

- **Upper bound on violation probability:**

Consider \( N' \) iid realizations of \( \xi \), such that \( N' >> N \), where \( N \) is the number of \( \xi \) samples used for solving the SAA problem. Here, \( N' \) may be large as the samples will not be used in solving an optimization problem, avoiding computational issues. Let \( q_{N'}(\bar{x}) \) be an estimator of \( q(\bar{x}) \); equal to the proportion of times the event \( 1(g(\bar{x}, \xi^j) > 0) = 1 \) is observed in \( N' \) trials. Estimator \( \hat{q}_{N'}(\bar{x}) \) of \( q(\bar{x}) \) is unbiased, implying \( \mathbb{E}(\hat{q}_{N'}(\bar{x})) = q(\bar{x}) \). Also, for a large \( N' \), its distribution may be approximated by a normal distribution with mean \( q(\bar{x}) \) and variance \( q(\bar{x})(1-q(\bar{x}))/N' \) [80]. This yields an approximate \((1-\beta)\)-confidence upper bound on \( q(\bar{x}) \):

\[
U_{\beta,N'}(\bar{x}) := \hat{q}_{N'}(\bar{x}) + z_\beta \sqrt{\hat{q}_{N'}(\bar{x})(1-\hat{q}_{N'}(\bar{x}))/N'}
\]

(4.19)

Here, \( z_\beta = \Phi^{-1}(1-\beta) \), where \( \Phi \) is the cumulative distribution function for the standard normal distribution, \( \beta \in (0,1) \). We compare \( U_{\beta,N'}(\bar{x}) \) to \( \varepsilon \) to check if \( \bar{x} \) is a feasible solution. If \( U_{\beta,N'}(\bar{x}) \leq \varepsilon \), then \( \bar{x} \) describes a topology where microgrids are very likely to be self-sufficient. A summary of the steps to be followed to derive this upper bound on chance constraint violation probability is provided in algorithm 4. Note that the scenarios used to solve and validate the SAA problem are generated using independent processes.

**Algorithm 4** Derivation of upper bound on chance constraint violation probability

1. Generate \( N' \) iid realizations of \( \xi \), where \( N' >> N \).
2. For the \( N' \) scenarios, count the number of times \( (N'_c) \) the event \( 1(g(\bar{x}, \xi^j) > 0) = 1 \) is observed.
3. Calculate \( \hat{q}_{N'}(\bar{x}) = N'_c/N' \).
4. Substitute \( \hat{q}_{N'}(\bar{x}) \) in equation (4.19) and calculate \( U_{\beta,N'}(\bar{x}) \).

- **Lower bound on optimal value:** A procedure for deriving a lower bound for \( f(x^*) \) is shown in [80]. Let the SAA problem (ODNP-3) be solved for \( N'' \) iid samples of \( \xi \) with risk level \( \gamma \geq 0 \); and denote this problem by \( P_{\gamma N''} \). Let the true problem (ODNP-2) with risk \( \varepsilon \) be denoted as \( P_{\varepsilon} \). Now, the probability that at most \([\gamma N'']\) constraint violations are observed in \( N'' \) trials while solving \( P_{\gamma N''} \), when the true violation probability is \( \varepsilon \), becomes:

\[
\Theta_{N''} := B([\gamma N'']; \varepsilon, N'')
\]

where,
4.6. NUMERICAL RESULTS

\[ B(k; q, N) := \sum_{r=0}^{k-1} \binom{N}{r} q^r (1-q)^{N-r} \]

is the cumulative density function of the binomial distribution. Say, solving \( P_\varepsilon \) yields an objective value \( f(\bar{x}) \). Assuming \( P_\varepsilon \) has an optimal solution \( f(\bar{x}^*) \), \( \Pr\{ f(\bar{x}) \leq f(\bar{x}^*) \} \geq \Theta_N^{\varepsilon} \). This result yields a method for obtaining lower bounds with a specified confidence level \((1 - \beta)\).

Consider two positive integers \( M \) and \( N'' \), such that \( M > N'' \). Generate \( M \) independent sets of \( N'' \) iid samples of \( \xi \), and solve the SAA problem for each of the \( M \) sets to obtain values \( f(\bar{x})_j, j = 1, 2, \ldots, M \). These can be viewed as iid samples of the random variable \( f(\bar{x}) \). Let \( L \) be the largest integer such that \( B(L - 1; \Theta_N', M) \leq \beta \). If the optimal values are arranged in a non-decreasing order \( f(\bar{x})_1 \leq f(\bar{x})_2 \leq \cdots \leq f(\bar{x})_M \), it can be shown that with probability at least \((1 - \beta)\), \( f(\bar{x})_L \) is lower than the true optimal value \( f(\bar{x}^*) \).

Note that \( f(\bar{x})_L > f(\bar{x}^*) \) if and only if more than \( L \) of the observed \( f(\bar{x})_j \) values are greater than \( f(\bar{x}^*) \). Considering event \( f(\bar{x})_j \leq f(\bar{x}^*) \) as a success, \( f(\bar{x})_L > f(\bar{x}^*) \) if and only if there are fewer than \( L \) successes in \( M \) trials, with success probability \( \Theta_N'' \). Probability of fewer than \( L \) successes in \( M \) trials is \( B(L - 1; \Theta_N'', M) \), and the bounding procedure described in this section restricts this probability value to \( \beta \).

The steps for calculating the lower bound on \( f(\bar{x}) \) with a confidence level of \((1 - \beta)\) are summarized in algorithm 5. Further insights on validating the quality of solutions yielded by SAA problems are available in [80, 89] and references therein.

**Algorithm 5** Derivation of lower bound on optimal value

1. Choose two positive integers \( M \) and \( N'' \) such that \( M > N'' \). A guide for selection is provided in [89].
2. Generate \( M \) independent sets of \( N'' \) iid samples of \( \xi \). For each of the \( M \) sets, solve (ODNP-3) to obtain values \( f(\bar{x})_j, j = 1, 2, \ldots, M \).
3. Find the largest integer \( L \) such that \( B(L - 1; \Theta_N', M) \leq \beta \).
4. Arrange the optimal values in a non-decreasing order \( f(\bar{x})_1 \leq f(\bar{x})_2 \leq \cdots \leq f(\bar{x})_M \). The lower bound for \( f((x^*) \) with confidence level \((1 - \beta)\) is \( f(\bar{x})_L \).

### 4.6 Numerical Results

The performance of the proposed methodology is illustrated through computational experiments on a 3.6 GHz Intel Core i7-4790 CPU with 32 GB RAM. Optimization tasks are solved using YALMIP and Gurobi [90, 91].
4.6.1 Experiment Set-up

The ODNP problem is solved for a modified version of the IEEE 37-bus benchmark feeder (figure 4.3), converted to its single-phase equivalent by: a) assigning average three-phase load as bus spot-loads, and b) assigning average three-phase impedances as line impedances. Four normally open switches are added (shown with dotted edges in figure 4.3). Grid-forming generators are placed at nodes 742, 718 and 710. PV generators of equal rated capacity (without grid-forming abilities) are added at nodes 702, 705, 707, 709 and 737. There are 22 buses with non-zero load. Total rated capacity of grid-forming and PV generators are 13% and 29% of the rated system load respectively. Practical feeders of the scale of our benchmark may not host such extensive distributed resources. However, such a model has been intentionally chosen to capture diverse flexibilities and computational concerns that may show up in real but larger networks. A feeder with fewer generators and tie-lines would have fewer load-generation scenarios and would be faster to solve for.

Load-generation scenarios were constructed as described next. Data corresponding to hourly solar generation in California from NREL’s solar power dataset were used to synthesize five annual generation profiles \[92\]. The first 50 generators in the dataset were used; every 10 generators were aggregated to obtain one profile. The normalized profiles were then scaled to match the rated capacity of the generators. It is further assumed that the PV generators are set to work at unity power factor, implying that they do not participate in reactive power support. This is without loss of generality since PV generators with reactive power support may be indicated with non-zero entries in the $\xi^{\text{gr}}$ vector. In a similar manner, hourly load profiles were constructed for residential and commercial buildings in California with data available from OpenEI [93]. The normalized profiles were scaled such that the 75th percentile of load data coincided with the nominal spot load of the corresponding bus. Thus, a total of 8760 scenarios were constructed for a year; denoted as set $S$.

4.6.2 Chance-Constrained ODNP

As stated previously, the original chance-constrained problem and its SAA counterpart become equivalent in limit as the number of scenarios considered $N$ increases. However, a higher $N$ value also increases computation time. This increasing trend is illustrated in figure 4.4, the markers show median time for 10 runs conducted over the same scenario sets. For computational tractability, let the SAA problem be solved on a smaller sample set $S' \subseteq S$; if $S'$ is sufficiently representative of $S$, then the candidate solution obtained will be close to the true solution for the original CCO problem.

Performance of the SAA approach is compared to a clustering based methodology, wherein set $S$ is divided into clusters and the ODNP task is designed to yield a solution that holds for some representative samples drawn from these clusters. Let us call these two Method 1 and Method 2 respectively.

- **Method 1:** Scenarios are sampled from $S$ at random with uniform probability and used to
4.6. Numerical Results

Figure 4.4: Median computation time increases with number of scenarios

Figure 4.5: 2-D visualization of scenario clusters

Table 4.1: Comparison of SAA and clustering based approaches

<table>
<thead>
<tr>
<th></th>
<th>Method 1</th>
<th>Method 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>(</td>
<td>f(x) - f(x^*)</td>
<td>0.00814</td>
</tr>
<tr>
<td>( U_{0.05,1000}(x) )</td>
<td>0.08</td>
<td>0.0025</td>
</tr>
</tbody>
</table>

solve (ODNP-3).

- **Method 2**: Using principal component analysis followed by agglomerative hierarchical clustering, set \( S \) is divided into 10 clusters [94]. The scenario clusters are visualized in figure 4.5. Once the clusters are determined, equal number of samples are randomly drawn from each cluster. Evidently, samples can only be drawn in multiples of 10. The ODNP is solved such that the optimal topology is feasible for all selected samples, i.e. \( \gamma = 0 \).

The performance of the two solution methods is compared in figure 4.6. For SAA, the value of \( \gamma \) used is 0.1. A 95\% confidence lower bound on the objective value is found using the methodology described in Section 4.5. With 50 runs of independently generated sets of 20 scenarios each, and \( \gamma = 0.7 \), this lower bound is determined to be -0.18536. The parameters \( M, N'' \) and \( \gamma \) here were chosen following the recommendations outlined in [89]. The 95\% confidence upper bound on feasibility of the candidate solution \( U_{0.05,1000}(x) \) is estimated using a set of 1000 scenarios. Median computation time for the feasibility checking process was 1.438 seconds.

Evidently, as more scenarios are considered, both average load served and \( U_{0.05,1000}(x) \) decrease. The trends are not strict as additional scenarios can introduce favorable cases with lower cost. Observe that Method 2 yields a more robust solution (low constraint violation probability) in lieu of a higher objective cost. Method 1 achieves an objective value close to the theoretical lower bound while bounding supply-deficiency probability to an acceptable level. Table 4.1 summarizes observations when both methods are run over 100 scenarios.
4.6.3 Choice of risk parameter

The optimal topology is highly dependent on the specified risk parameter. Of course, if a utility has a high risk budget, they may plan to cover a larger amount of loads with microgrids. The risk appetite may be dictated by a number of factors, such as the installed storage capacity and criticality of loads. If the loads to be served are critical (hospitals, police stations etc), then the risk budget goes down. The intuition of higher load served with higher risk tolerance is experimentally verified and shown in figure 4.7. It can also be seen that computation time increases with $\gamma$; possibly because for higher values of $\gamma$, the feasibility space that the solver has to search for an optimal solution to ODNP grows in size.

4.6.4 Number of Normally Open Switches

Any topology determination problem is combinatorial in nature, and hence the search space and solution time increases with the number of graph edges. In the ODNP task, network flexibility may be better utilized to serve more load by leveraging normally open switches. However, addition of extra edges introduces additional binary decision variables, thereby increasing solution time. In figure 4.8, it is shown that as more switches are added to the base radial 37-bus network, ODNP yields higher average load served (i.e. lower objective values), but the computation time grows. These data points were determined by solving the ODNP problem over 50 scenarios sampled with method 1 and using $\gamma = 0.1$. For each of these cases, multiple combinations of normally open switches are possible. However, switches were added one at a time in a random sequence to the base network for illustration.
4.6. Numerical Results

![Graph showing variation in cc-ODNP performance with \( \gamma \) considering 50 scenarios.](image1)

(a) Average load served  
(b) Median time taken for 10 runs

Figure 4.7: Variation in cc-ODNP performance with \( \gamma \) considering 50 scenarios.

![Graph showing variation in performance with number of normally open switches. Number of scenarios considered is 50, \( \gamma = 0.1 \).](image2)

(a) Average load served  
(b) Median time taken for 10 runs

Figure 4.8: Variation in performance with number of normally open switches. Number of scenarios considered is 50, \( \gamma = 0.1 \).

4.6.5 Microgrid Topology

Optimal microgrids determined for the base radial network with and without normally open switches are shown in figure 4.9 and 4.10 respectively. Microgrid components are indicated in color while external elements are in grey. When all switches are considered, higher load can be served. When only the base radial network is considered, load served by microgrids is lower, and so is the supply-deficiency violation probability.

The determined microgrids do not violate the self-adequacy and power systems constraints for more than \( \gamma \) fraction of cases, are radial and contain at least one grid-forming generator.
Figure 4.9: Optimal microgrid topology considering all switches, 100 scenarios and $\gamma = 0.1$. Average load served is 0.1935 p.u.. $U_{0.05,1000}(x) = 0.08005$.

Notice that despite hosting a solar generator, bus 707 is not included in any of the microgrids. This may be because there are no possible ways to connect bus 707 to a grid-forming generator without violating one of the prescribed constraints.

4.6.6 Computational Performance

As illustrated in the simulated examples, the time required to solve the stochastic ODNP problem depends on several factors such as network size, number of scenarios considered, number of switches and the choice of risk parameter. Evidently, the inherent combinatorial and stochastic nature of the underlying network splitting problem poses computational challenges. Fortunately, since ODNP is a planning problem, the scalability concern is less severe compared to similar applications pertaining to real-time operation. Nevertheless, to enhance scalability, the linearized distribution flow model [85] is used in this work instead of the full AC power flow model, thus yielding a MILP formulation. Recent advancements in commercial solvers like CPLEX and Gurobi have significantly improved both the speed and scale at which MILPs can be solved. This makes MILP formulations attractive even for operation problems like topology reconfiguration, and well-suited for planning-stage ODNP. Thus, the computational burden for solving ODNP is not prohibitive and can be used to identify potential microgrids in large distribution feeders with high DER penetration.

4.7 Summary

The power grid is critical for maintaining essential sectors like healthcare, transportation and emergency services. This has motivated research towards enhancing grid resilience, and formulating measures to hedge against extreme events. Efficiently planned microgrids can help minimize load interruptions and aid restoration during and after large-scale outages. To
4.7. Summary

Figure 4.10: Optimal microgrid topology for the base radial network, 100 scenarios and $\gamma = 0.1$. Average load served is 0.1483 p.u.. $U_{0.05,1000}(x) = 0.06465$.

This end, this work proposes a chance-constrained optimal network partitioning problem and presents a computationally tractable solution methodology. Specifically, the ODNP problem seeks to enhance resilience by improving grid performance in the post-disaster degraded state.

First, a deterministic version of the ODNP problem is formulated considering practical constraints such as maintaining network radiality and the availability of grid-forming generators. The optimal network topology is determined without pre-fixing the number of partitions, or pre-assigning DERs to microgrids. Moreover, the requirement of grid-forming generators in viable islands had not been addressed in prior microgrid planning literature. Next, the deterministic problem is extended to a probabilistic setup to account for the stochasticity in energy demand and renewable energy generation. The probabilistic optimization problem is solved using sample average approximation. Further, the work discusses how to assess the quality of the obtained SAA solution using rigorous statistical tools.

Case-studies on a modified version of the IEEE 37-bus feeder show that good quality candidate solutions can be found with modest computation cost; network flexibility is well-utilized; and partitioning changes with risk budget. Future work will focus on extending the present planning-stage formulations to multi-phase topologies and near real-time applications.
Chapter 5

A Model-Agnostic Method for PMU Data Recovery Using Optimal Singular Value Thresholding

5.1 Introduction

Phasor Measurement Units (PMUs) allow high-resolution insight into power systems dynamics through precise time-synchronized measurements [32]. In recent years, electric utilities have made great strides towards deploying PMUs in their networks and utilizing the reported measurements for wide-area situational awareness [33]. Phasor measurements are used in both online (state estimation, remedial action schemes) and offline (model validation, contingency analysis, post-event diagnosis) applications [34],[35], [22]. As shown in figure 5.1, to reach the point-of-use from the point-of-measurement, PMU data flows through various communication channels and intermediate data concentrators. Hence, the data is susceptible to channel congestion or component malfunction issues which lead to degraded information quality [95]. Reliable measurements are critical to developing robust control and monitoring algorithms for the power grid, making fast and accurate data recovery critical as well.

Traditional model-based recovery methods are sensitive to the underlying model accuracy, and may need the knowledge of real-time network topology. Some such methods assume the knowledge of transmission line parameters, and are affected by inaccurate information [96]. Another class of model-based methods use linear state estimation (LSE) for PMU data conditioning [35, 97]. LSE-based data recovery is contingent on breaker status information and any error or latency in reporting can adversely impact performance [98]. On the other hand, measurements from PMUs within a network exhibit spatial and temporal correlation that can be utilized to estimate missing samples without explicit knowledge of the power system model. Several recent papers have addressed model-agnostic PMU data recovery [99, 100, 101, 102, 103, 104, 105]. Some of the proposed methods target imputation of data blocks [99, 100, 102], while others are aimed at step-ahead prediction [101, 103, 104, 105]. Further, these recovery approaches broadly employ strategies based on - a) filtering [101], b) low rank matrix completion [102, 103, 104, 105], or c) low rank tensor completion [99, 100].

In [101], the authors propose a kalman filter-based missing data estimation algorithm that predicts the value of an incoming sample using the last three measurements. The accuracy of
5.1. Introduction

Step-down TX Filters
A/D Oscillator Phasor Estimator Substation Aggregator Central Aggregators

PT CT GPS Receiver PMU
Point of Measurement

Archived data Application

Figure 5.1: PMU data flow from point-of-measurement to point-of-use

this method deteriorates when more than three consecutive entries are missing. Moreover, as the algorithm processes one PMU channel at a time, information from other channels or nearby PMUs cannot be leveraged to reconstruct segments of consecutive missing data. Matrix (tensor) estimation methods propose to stack correlated PMU measurement channels together to construct matrices (tensors), whose low-rank property can be exploited to recover corrupt data. Tensor estimation methods are more computationally expensive, limiting their potential for real-time use. Methods to speed up tensor estimation have been explored in [99, 100].

In the matrix estimation (ME) area, various strategies have been put forth for stacking measurement channels. In [102], the channels are stacked rowwise, and an iterative alternating direction method of multipliers is used to fill missing measurements. In [103], the channels are stacked columnwise, and the spatial correlation among PMUs is used to recover missing entries using singular value thresholding. Another variation that uses the temporal correlation among channels is shown in [104]. Since these methods process multiple PMU channels together, they are able to accurately recover missing data sequences on one channel using information from other devices. On the other hand, data prediction accuracy is severely affected by high noise content in any one channel. An online recovery method exploiting the low-rank property of the hankel matrix constructed by overlapping segments of PMU data has been proposed in [105]. As columns in a hankel matrix overlap, this approach faces the following drawbacks- a) repeated entries greatly increase the size of the hankel matrix, thereby increasing computation burden, b) noise in the matrix elements are highly correlated, c) noisy entries are repeated multiple times affecting recovery accuracy.

To overcome the limitations discussed above, this paper proposes a novel technique for PMU data recovery. A sequence of PMU measurements is first transformed into a page matrix [106], and then recovered using a variation of the truncated singular value thresholding algorithm [107]. As the matrix columns consist of non-overlapping data segments, they are smaller in size than hankel matrices and the problem of highly correlated noise in matrix entries is avoided. An optimal hard threshold is used for singular values; thus the matrix rank does not need to be explicitly estimated at every step, resulting in significant computational savings for online prediction. Two variations of the algorithm are proposed, a) an offline
imputation method for archived PMU data, and b) an online one-step ahead prediction method. Performance of the page matrix method is compared to the hankel matrix method using the same estimation technique. Extensive numerical tests show that both methods have similar recovery accuracy, and the page matrix method is computationally much faster.

The proposed algorithms can be applied to both univariate and multivariate time-series (both single-channel and multi-channel cases). Although using measurements from multiple PMUs translates to higher accuracy in data recovery, some use cases might warrant processing single channels. For instance, researchers may have access to limited PMU data due to their sensitive nature. Moreover, computation may be sped up by parallelly processing single measurement channels. This may be useful in cases where the streaming synchrophasor data is of superior quality, and significant data drops are not expected.

Contributions of this work may be summarized as follows. First, we propose a model-agnostic method for recovering PMU measurements from noisy signals with data drops. The proposed methodology is fast, scalable, easy to implement, and poses minimal memory requirements, making it well-suited for real-time use. Second, the methodology is extended into two algorithms - a) an offline method intended for recovering archived data, and b) an online method for predicting the next measurement, aimed at real-time applications. Third, through extensive numerical experiments on simulated and real data, effectiveness of the algorithms is verified. It is shown that the original measurement signals can be reconstructed with high accuracy even in the presence of additive noise and simultaneous data erasures across multiple channels.

5.2 Problem Set-up

In this section, we establish the mathematical set-up for the PMU data recovery problem and show how it relates to low-rank ME. Standard mathematical notations are followed. Calligraphic symbols represent sets, lower case bold letters represent column vectors, and upper case bold letters denote matrices. All zero and all one vectors and matrices of appropriate size are denoted by \( \mathbf{0} \) and \( \mathbf{1} \) respectively.

5.2.1 PMU Data Recovery Using Matrix Estimation

Simply stated, PMUs are sensors deployed at different points of a power network to measure electrical quantities like voltage and current magnitudes, angles, frequency and rate of change of frequency. Measurements are time-synchronized and typically reported at 30 or 60 frames per second (fps). Due to the physical laws that govern power flow, measurements recorded by a PMU and its neighbors are correlated. Moreover, data within a channel is correlated in time. These spatial and temporal correlations may hence be utilized to recover missing and corrupt measurements.

Formally, the data recovery problem can be posed as follows. Consider a discrete-time setting with time instants indexed by \( t \in \mathbb{Z}^+ \). Let us say that for each \( t \in [1, 2, \ldots T] \), PMU \( i \) records
a measurement vector $\mathbf{x}_i(t)$ of length $c$, where $c$ is the number of measurement channels. Measurements may contain observation noise, and it is assumed that $\mathbb{E}[\mathbf{x}_i(t)] = \mathbf{f}_i(t)$, where $\mathbf{f}_i(t)$ are the true values of system states. Although the underlying mean signal $\mathbf{f}_i(t)$ is strongly correlated in time, it is assumed that the per-step noise are independent mean-zero random variables with time-varying variance. Given some $\mathbf{x}_i(t)$, data recovery algorithms may be designed to address two goals: a) imputation (estimate $\mathbf{f}_i(t)$ for $t \in [1, 2, \ldots, T]$), and b) prediction (estimate $\mathbf{f}_i(T + 1)$).

Time-series data recovery is a well-studied problem that appears in different domains like econometrics, geosciences and healthcare. Classical methods for time-series imputation and prediction have employed approaches such as hidden Markov and state-space models [108]. Different deep neural networks (NN) have also been used [109, 110]. Recent work has shown that low-rank matrix estimation methods can provide simple, effective and computationally efficient means for time-series recovery [111, 112]. This class of methods eliminates the training data requirement of NN models, and hence provide a generalized framework suitable for quick deployment.

The objective of ME is to recover a parameter matrix $\mathbf{M}$ from a partially observed signal matrix $\mathbf{X}$ with corrupt entries, where $\mathbb{E}(\mathbf{X}) = \mathbf{M}$. A detailed picture of the state-of-the-art is available from [113, 114] and references therein. A key observation from ME literature is that matrix $\mathbf{M}$ can be reconstructed from partial and noisy observations by considering a low-rank approximation of the observed matrix. ME algorithms are fairly model-agnostic in terms of the structure of $\mathbf{M}$ and the distribution of $\mathbf{X}$ given $\mathbf{M}$. Therefore, PMU measurements can be transformed into matrices and recovered by applying ME methods to the transformed matrix. Truncated singular value decomposition (SVD) based matrix estimation methods are popularly used [107].

### 5.2.2 Matrix Transformation

Several methods have been proposed to transform time-series signals into matrices. A naive method involves simply stacking signals together [102, 103, 104]. Although empirically this approach has shown reasonable effectiveness, it cannot be used if very few measurement channels are available. Of course, this is not a pressing concern for a transmission network with many PMUs. But in some cases, it may be necessary to work with a limited number of measurement signals. For instance, researchers outside electric utilities may only have access to limited PMU data. Moreover, it may be desirable to process data from blocks of few PMUs in a parallel manner to speed up computation during real-time application.

An alternative approach described in [105] uses the hankel matrix transformation. Overlapping segments of PMU data are placed side by side to form a hankel matrix. That such matrices are approximately low-rank has been empirically verified in [105]. As the hankel matrices contain repeated entries, they are large in size and noisy elements appear multiple times. Repetition of noisy entries may reduce data recovery accuracy and the large matrix size increases computation burden.
The limitations of hankel matrices may be overcome by using page matrices [111, 112]. A page matrix is constructed from observation vector $x(t)$ by placing contiguous segments of size $L > 1$ side by side as non-overlapping columns of the resultant matrix [106]. A schematic description of the matrix transformation process is shown in figure 5.2. It can be seen that when a time-series of the same length is transformed, the hankel matrix is much larger than the corresponding page matrix. The low-rank property of page matrices has also been examined in [111]. It is established that, in expectation, for a large class of processes, the generated page matrix is either exactly or approximately low-rank. These processes include linear recurrent functions (LRF) described by $f(t) = \sum_{g=1}^{G} \alpha_g f(t-g)$, for some $G \geq 1$. That power systems quantities such as bus voltages, line currents and frequency follow LRF has been previously concluded in literature. For example, it is posited that steady state power systems measurements at any instant are a linear combination of the last three measurements [115]. This result has also been used to design the kalman filter-based data prediction and smoothing algorithm in [101].

The LRF nature of PMU measurements also helps in formulating a prediction algorithm. Once the low-rank approximation of the transformed page matrix is obtained via some ME technique, the last row of this matrix can be expressed as a linear combination of the other rows. Therefore, future values can be forecast by applying linear regression to the approximate page matrix.

### 5.2.3 Optimal Singular Value Thresholding (OSVT)

Various techniques exist for low-rank ME [113, 114]. In the proposed model-agnostic recovery framework, no information about the rank of the page matrices is available beforehand. Hence, a variation of the truncated SVD method that does not need matrix rank information has been used in this paper [107]. The optimal singular value threshold is chosen based on findings reported in [116]. The main steps in the estimation algorithm are detailed next in algorithm 6. It is assumed that missing data points in the observation matrix are preliminarily filled in by the last available observation.

The hard singular value threshold proposed in [116] is optimal in an asymptotic sense. It
Algorithm 6 Optimal Singular Value Thresholding (OSVT)

1: **Scaling:** Entries of the observation matrix $X$ are scaled to lie in the interval -1 to 1. Let the scaled observation matrix be called $Y$ with individual entries $y_{i,j}$. Mathematically, $y_{i,j} = (x_{i,j} - 0.5(a+b))/0.5(b-a)$, where $a$ and $b$ are the minimum and maximum entries of $X$ respectively.

2: **Singular value decomposition:** Let $Y = \sum_{i=1}^{m} \sigma_i u_i v_i^T$ be the singular value decomposition of $Y$. The singular values are given by $\sigma_i$; and $u_i$ and $v_i$ are the corresponding left and right singular vectors respectively.

3: **Singular value thresholding:** Choose a set $S$ of thresholded singular values such that: $S := \{\sigma_i > \sigma_{th}\}$, where the optimality threshold $\sigma_{th}$ is given by:

$$\sigma_{th} = \sqrt{2(\zeta + 1) + \frac{8\zeta}{(\zeta + 1) + \sqrt{\zeta^2 + 14\zeta + 1}}}$$

Here, $\zeta = m/n$, where $Y$ is a $m \times n$ matrix. Moreover, $m \leq n$. In the case that $m > n$, the estimation algorithm must be applied to $X^T$ to obtain an estimate of $M^T$.

4: **Low rank approximation:** The low rank approximation of matrix $Y$ is given by $\hat{Y} = \sum_{\sigma_i \in S} \sigma_i u_i v_i^T$. The final estimation $\hat{M}$ of the parameter matrix is obtained by scaling back the values of $\hat{Y}$ to the interval $[a, b]$.

is postulated that for large low-rank matrices, when a data singular value $\sigma_i$ is too small, the corresponding singular vectors $u_i$ and $v_i$ are very noisy and the component $\sigma_i u_i v_i^T$ should not be used in approximating matrix $\hat{Y}$ from $Y$. The cutoff for singular values is determined to be $\sigma_{th}$ as described in algorithm 1. The alternative method used in literature for estimating matrix rank involves- a) selecting a threshold for rank approximation error, and b) choosing the lowest rank for which the approximate matrix does not violate the predetermined error threshold. The choice of the approximation error threshold is somewhat arbitrary and has been empirically decided in works like [105]. Using a hard threshold eliminates the need for repeated calculations of rank approximation error at every step, thereby significantly improving computation speed. It is further shown in [116] that the optimally tuned thresholding method outperforms (in terms of mean squared error) classical truncated SVD when signal noise content is low to moderate; and the methods perform roughly similarly when noise content is high.

### 5.2.4 Multivariate Time-Series Recovery

As mentioned before, PMU measurements in a network are correlated, and hence information from different PMUs can be utilized to recover degraded data. Readings from multiple PMU channels can be transformed into a ‘stacked’ page matrix by concatenating individual page matrices columnwise, as shown in figure 5.3. The low-rank property of stacked page matrices for a large class of processes including LRF has been verified in [112]. Let
Chapter 5. A Model-Agnostic Method for PMU Data Recovery Using Optimal Singular Value Thresholding

Figure 5.3: Stacked page matrix for multivariate data recovery

Let us consider the problem of denoising already recorded PMU measurements and imputing missing readings. Assume $n$ number of PMUs, each with $c$ measurement channels. Then the total number of data channels available is $N = n \times c$. The observations are partitioned into $k$ windows of length $T$ each. Note here that for the imputation task, hours of data can be processed at once. Thus, $T$ can be quite large. It is assumed here that $T$ is perfectly divisible by the value of $L$ chosen. The sequential steps to be performed for offline data imputation are listed in algorithm 2.

5.3 Data Recovery Algorithms

This section shows how the individual pieces described in the previous section are put together for recovering degraded PMU data. The algorithms can be applied to both single-channel and multi-channel data using suitable page matrices.

5.3.1 Offline Data Imputation

Let us consider the problem of denoising already recorded PMU measurements and imputing missing readings. Assume $n$ number of PMUs, each with $c$ measurement channels. Then the total number of data channels available is $N = n \times c$. The observations are partitioned into $k$ windows of length $T$ each. Note here that for the imputation task, hours of data can be processed at once. Thus, $T$ can be quite large. It is assumed here that $T$ is perfectly divisible by the value of $L$ chosen. The sequential steps to be performed for offline data imputation are listed in algorithm 2.
5.3. Data Recovery Algorithms

Algorithm 7 Offline PMU data imputation

1: **Initialization:** Set $T \leftarrow$ window length, $N \leftarrow$ number of PMU channels, $k \leftarrow$ number of data windows.
2: For $j = 1 : k$, do
3: **Matrix transformation:** Construct a stacked page matrix using data from $N$ PMU channels in the $j$th measurement window with $T$ observations, as described in section 5.2.4.
4: **Low-rank matrix estimation:** Obtain a low-rank approximation of the stacked page matrix $\hat{X}$ using the OSVT method outlined in algorithm 6.
5: **Recover estimated measurements:** Reshape matrix $\hat{X}$ to recover the estimated measurements.
6: End for

Figure 5.4: Visual description of matrices $G$, $H$, $G'$ and $H'$ for a page matrix using data from a single channel

5.3.2 Online Data Prediction

The online forecast problem pertains to predicting the signal value $f(T + 1)$ given past observations $x(t), t \in [1, 2, \ldots, T]$. This is analogous to performing regression with noisy data. The online forecasting algorithm proposed in this paper—first, denoises and imputes past observations (algorithm 7), and second, uses linear regression to learn the relationship between the last row and remaining rows of the imputed observation matrix. Next, the learned regression parameters are used to predict the next sample from a page matrix shifted by one sample. Of course, the first data window of length $T$ needs to be filled before next-step prediction can proceed. The forecast procedure is described with better clarity in algorithm 8.

5.3.3 Choice of Hyperparameters

Selecting good hyperparameters is essential for achieving high-accuracy data recovery while minimizing computation time. For the data recovery algorithms proposed in this work, two main hyperparameters need to be chosen— a) $L$ or the number of rows in the page matrix, and b) $T$ or length of data window.
CHAPTER 5. A MODEL-AGNOSTIC METHOD FOR PMU DATA RECOVERY USING OPTIMAL SINGULAR VALUE THRESHOLDING

Algorithm 8 Online PMU data prediction

1: **Initialization:** Set \( T \leftarrow \) window length, \( N \leftarrow \) total number of PMU channels, \( j \leftarrow 0 \).
2: **While** PMU data streams are available, **do**:
3: **Matrix formation:** Convert measurement vectors \( x_i(t+j), i \in [1,2,\ldots,N], t \in [1,2,\ldots,T] \) to a stacked page matrix \( X_{T+j} \), say.
4: **Matrix imputation:** Using algorithm 7, denoise and impute \( X_{T+j} \) to obtain \( \hat{X}_{T+j} \).
5: **Learn the linear forecast model:** Partition matrix \( \hat{X}_{T+j} \) into two parts \( G \) and \( H \) such that \( G \) comprises of the first \( L-1 \) rows in \( \hat{X}_{T+j} \) and \( H \) contains the last. Linear regression here pertains to estimating the parameter vector \( \beta_j \) for which \( H = G\beta_j + \epsilon \) in the least squares sense.
6: **Forecast the one-step-ahead data:** Construct matrix \( G' \) with the last \( L-1 \) rows of \( \hat{X}_{T+j} \). Estimate \( H' \) from \( G' \) as \( H' = G'\beta_j \) using the value of \( \beta_j \) learnt in step 5. For the univariate prediction case, the last entry of \( H' \) is the prediction of the next measurement \( x(T+j+1) \). For the multivariate case, pertinent entries from \( H' \) need to be extracted. For clarity, a visual description of the matrices \( G, H, G' \) and \( H' \) is provided in figure 5.4.
7: **Update:** Set \( j \leftarrow j+1 \)
8: **End while**

- **Choice of \( L \):** Empirically, it was observed that choosing a \( L \) value in the 5-10 range yielded good results for both the online prediction and offline imputation tasks. Keeping the parameter \( L \) small enables capturing the short-term temporal patterns in PMU data. However, for very noisy data, increasing \( L \) can help in obtaining smoother estimates.

- **Choice of \( T \):** The in-sample imputation and prediction error for the multivariate data recovery method scales as \( 1/\sqrt{NT} \). Therefore, the longer the data window, the better the prediction accuracy. On the other hand, choosing a long data window increases computation time. As computation time is not a prime concern for offline data imputation, a fairly long window can be chosen. In the numerical results section of this paper, data imputation with \( T = 54000 \) (30 minutes of data) has been demonstrated.

Computation time is of greater concern when it comes to online prediction, as every prediction step requires a matrix estimation and linear regression operation. Hence, choosing a shorter time window is beneficial. However, the window length \( T \) should be selected carefully. If \( T \) is too small, prediction accuracy will be impacted by measurement noise. If \( T \) is too large, the data window may contain obsolete modes thereby degrading prediction accuracy. It was empirically observed that a data window of about \( \sim 30-45 \) samples yielded good performance without unduly increasing prediction time.

5.3.4 Scalability

The size of the page matrix will also be determined by the number of PMU channels available. In a real system with hundreds of field PMUs, computation may be sped up by dividing the
PMUs into groups with similar modal signatures and processing the groups in parallel. Grouping together PMUs geographically close to each other will also enhance visibility into ‘local’ dynamics which might have been obscured by aggregating signals over a wide-area grid. Some strategies for grouping PMU signals to ensure low-rankness of the measurement window have been presented in literature [117].

During a disturbance, the system deviates from its predicted behavior, and the low-rank property of the stacked page matrix may not hold true. Therefore, at the onset of a disturbance, online predictions may vary greatly from observed measurements. The difference between the actual observations and algorithm predictions decreases gradually.

5.4 Numerical Results

This section describes the numerical tests performed to validate the performance of the proposed imputation and prediction methodologies. The first set of tests are performed on simulated data, artificially distorted by the injection of additive noise and random data drop. The next set of tests considers real noisy PMU measurements recorded by a U.S. utility. In this case, the true values of the measurements are unknown, but visual inspection reveals that reasonable values are predicted for swathes of missing data. All computations are performed on a PC with 16 GB RAM and 2.6 GHz Intel core i7-9750HF processor.

5.4.1 Simulated Measurements Dataset

Numerical tests were performed on 86.6 seconds of measurements from PMUs installed at ten generator bus terminals of the IEEE 39-bus transmission model [118]. Quasi-steady state operations and three-phase faults were recorded using RTDS power systems simulator and GTNETx2 based PMUs at 60 fps reporting rate. During the length of the simulation, quasi steady-state conditions were simulated by modulating the mechanical torque of generator $G_1$ every 200 ms by a random perturbation within $\pm 1\%$ of the nominal value. The data also shows three disturbances. At 18.33 seconds, a self-clearing three-phase fault is followed by tripping of the faulted line, leading to a topology change. At 55.67 seconds, the tripped line is reconnected, restoring the initial network topology. At 78.13 seconds, another three-phase self-clearing three-phase fault takes place.

In this study, three measurement channels from each PMU were used - positive sequence voltage magnitude, positive sequence voltage angle and frequency. Data from all ten PMUs are shown in figures 5.5, 5.6 and 5.7. PMUs are referred to by the generator terminal they are installed at. For example, the PMU at generator $G_1$ terminal is called PMU $G_1$. Voltage angle at PMU $G_1$ is considered as the reference angle.

Simulated measurements have been used for evaluating the data recovery algorithms as the ‘ground truth’ data is available for comparison. On the other hand, in real PMU measurements, some readings may already be missing or corrupt, and there is no way of exactly knowing what those measurements should have been. Further, as this simulation records
Figure 5.5: PMU measurements: Positive sequence voltage magnitude

Figure 5.6: PMU measurements: Positive sequence voltage angle (unwrapped)

Figure 5.7: PMU measurements: Frequency
network topology changes, we can investigate if varying topologies affect recovery accuracy.

### 5.4.2 Data transformation

The PMU measurements are scaled before being transformed into the stacked page matrix described in section 5.2.4. The scaling process used in this paper is described below:

- **Voltage magnitude**: Measurements were transformed into the per unit (p.u.) system.
- **Voltage angle**: The reference voltage angle is subtracted from individual channel data. In the dataset used, angle readings were already unwrapped. For unwrapping angles in real-time, the strategy outlined in [119] may be followed.
- **Frequency**: Frequency measurements were scaled as follows: $f_{\text{scaled}} = (f_{\text{measured}} - 60) \times 10$.

The scaling method described above empirically showed good results; however other approaches may also be used.

### 5.4.3 Error Metric

Mean absolute percentage error (MAPE) has been used as the error metric to evaluate the accuracy of the proposed algorithms. Mathematically, MAPE for a time-series of length $n$ may be expressed as:

$$\text{MAPE} = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{A_t - \hat{A}_t}{A_t} \right|$$

Here, $A_t$ is the actual time-series value and $\hat{A}_t$ is the corresponding prediction. As it intuitively conveys relative error, MAPE is widely used in regression problems and model evaluation tasks.

### 5.4.4 Offline Data Imputation

As communication channel congestion may impact all PMU channels, data erasures may be correlated. To capture the efficacy of the proposed imputation method under realistic data degradation conditions, the following modes were checked:

- **Data drop**: Simultaneous data drops on all PMU channels in the network were considered. A fraction of timestamps (determined by the chosen data drop rate) were randomly selected and corresponding measurements were dropped from all PMU channels. As the data drop rate is increased, the chance of missing consecutive data segments also increases. Error on all channels of PMU $G_2$ as the data drop rate is varied is shown in figure 5.8. These are median values of observations over 20 runs. It is evident that the measurements can be reconstructed with acceptable accuracy. Time taken for imputation did not vary significantly with data drop rate and median time taken over 100 runs was 0.0184 seconds. In the interest of brevity, only results for PMU $G_2$ has been included in this paper, but similar results were obtained for other PMU channels as well.
CHAPTER 5. A MODEL-AGNOSTIC METHOD FOR PMU DATA RECOVERY USING OPTIMAL SINGULAR VALUE THRESHOLDING

66

Figure 5.8: Error in $G_2$ PMU channels with simultaneous data drop in all PMU channels (median over 20 runs)

Figure 5.9: Error in $G_2$ PMU channels with zero-mean additive noise on all $G_2$ PMU channels (median over 20 runs)

Results for the extreme scenario where readings from all PMUs are missing are shown here. For less extreme cases, i.e. when data from only some of the PMU channels are missing, higher accuracy in signal reconstruction may be expected.

- **Additive noise**: Noise in PMU measurements may arise due to errors in calibration, instrumentation and quantization. Existing studies suggest that the zero-mean gaussian distribution is a suitable model to characterize this noise. The signal-to-noise ratio (SNR) for real transmission-level PMUs is estimated to be around $\sim 45$ dB, while for distribution-level PMUs the SNR is estimated to be lower [120]. Similar noise was injected into PMU $G_2$ channels for the numeric tests in this work.

Zero-mean gaussian noise was added to all PMU $G_2$ channels. The standard deviation of the noise distribution on each channel was given by a percentage of the median of true steady-state data (let us call this percentage the noise rate). Data recovery error with varying noise rates is shown in figure 5.9. Median computation time over 100 runs was 0.0192 seconds.

For both the cases above, the number of rows in the stacked page matrix was 10. Figure 5.10 shows both the corrupt and imputed data when 2% noise was added to the PMU $G_2$
5.4. Numerical Results

Figure 5.10: Imputed measurements with 50% simultaneous missing data and 2% additive noise on all $G_2$ PMU channels

Figure 5.11: Predicted measurements with 50% simultaneous missing data and 2% additive noise on all $G_2$ PMU channels

channels and 50% of the readings were missing. It can be seen that the PMU signals are reconstructed with reasonable accuracy even when consecutive data segments are missing. Further, the data recovery accuracy is robust to topology changes in the power network.

5.4.5 Online Data Prediction

Similar numerical tests were conducted with the same PMU dataset for assessing online data prediction accuracy. Performance of the stacked page-matrix based prediction algorithm has been compared with the hankel-matrix based prediction method put forth in [105]. For better comparison, the same ME technique is used for both algorithms.

• Prediction error: Figures 5.12 and 5.13 show the prediction error for different data drop rates (simultaneous) and additive noise on all $G_2$ PMU channels. The results are median observations over 20 runs. Data drops and noise were introduced in the data in the same manner as discussed in section 5.4.4. It is observed that the prediction error of the page matrix method is similar to/slightly better than the hankel matrix based method. The corrupt and reconstructed signals when 50% data drop and 2% noise is added to all $G_2$ PMU channels is shown in figure 5.11. In the experiments, number of rows used was 5, and window length considered was 25.

• Prediction time: One-step ahead predictions have multiple applications. Missing samples
can be filled in with predicted values. Similarly, irregularities in data may be detected looking at how far measurements stray from their predictions. Now, for any real-time algorithm implementation, computation time is of prime concern. U.S. electric utilities typically use PMUs with reporting rates of 30 or 60 fps. For these PMUs, the time intervals between two consecutive samples is 0.0333 or 0.0167 seconds. Therefore, in order to predict a sample before it arrives at the control center, the prediction time must be much lower.

Our numeric tests showed that the prediction time for the proposed page matrix algorithm was \( \sim 0.001 \) seconds. In comparison, time taken by the hankel matrix based method was \( \sim 0.003 \) seconds. All times are median values recorded over 100 runs. The prediction times did not vary significantly with varying rates of data drop or noise. Thus, we see that the method proposed in this paper is much faster than the hankel-matrix based method, and provides similar/slightly better accuracy.

- **Window length of hankel matrix:** The preceding set of experiments showed that the proposed page matrix-based data recovery method provided accuracy levels similar to the hankel matrix-based method, while speeding up computations. The computational savings are largely due to the smaller size of the page matrix for the same measurement window. A
natural question arises here: how would using a shorter time window for the hankel matrix method affect data recovery accuracy? It is expected that predictions obtained using a smaller number of observations will be noisy and less accurate. This notion is experimentally verified in this work. 

Figure 5.14 shows how the recovery accuracy varies when data is dropped simultaneously from all channels of PMU $G_2$ using the same methodology as section 5.4.4. Three matrix transformations were checked, a) page matrix (dimensions $5 \times 6$), b) hankel matrix with longer window (dimensions $5 \times 26$), and c) hankel matrix with shorter window (dimensions $5 \times 6$). For the first two cases, measurement window length used for predicting the next sample is $T = 30$, while for the third case it becomes $T = 10$. As the page matrix and hankel matrix with shorter window have the same dimensions, time taken to predict the next sample using these matrices is almost the same. However, as evident from figure 5.14, the page matrix method yields higher recovery accuracy. When using a hankel matrix with longer window, the prediction accuracy improves, but computation time increases as well.

- **Verifying the low rank-property:** The low rank properties of both the stacked page and hankel matrices were checked for all the data windows, as shown in figure 5.15. It can be seen that both the page and hankel matrices are generally low-rank, and the matrix ranks increase at the beginning of events. During the first event, the hankel matrix becomes full-rank, and the low-rank property does not hold.
5.4.6 Real PMU measurements dataset

Next, we check how well the proposed algorithms perform with real PMU data from an anonymized U.S. electric utility. The data corresponds to 30 minutes of measurements (54000 samples) from four PMUs of the utility, each of which reports three channels—positive sequence voltage magnitude, positive sequence voltage angle, and frequency. PMU reporting rate is 30 fps. The voltage angle of PMU4 is considered as reference, since it has the least amount of missing entries. Voltage angles have been unwrapped using the algorithm from [119].

The recorded measurements were quite noisy, and had large patches of missing data. The percentage of missing entries, and maximum length of consecutive missing data segments in each channel is shown in figure 5.16. It can be seen that the PMU channels have data missing in the ∼20-40% range. Maximum length of missing data segments is ∼100. The noisy and intermittent PMU measurements are shown in figure 5.17, and the imputed measurements are shown in figure 5.18. Since there is no way to know what the actual measurements should have been, objectively evaluating the imputation algorithm is not possible. However, visual inspection shows that the proposed algorithm is able to impute the measurements very well. Median computation time to impute 54000 measurement samples over 20 runs was 0.109 seconds.

Figure 5.19 shows the imputed measurements when using the hankel matrix method (in algorithm 2, measurements are transformed into a stacked hankel matrix in place of a page matrix). Visually, it appears that the hankel method does not provide better estimates than the page method. Moreover, time taken for imputation using the hankel method was ∼ 1.4 seconds, much higher than the page method.
5.4. Numerical Results

Figure 5.17: Thirty minutes of measurements from four PMUs in an anonymized U.S. electric utility. Data is reported at 30 fps. The vertical lines show data drops.

Figure 5.18: Imputed PMU data using page matrix

Figure 5.19: Imputed PMU data using hankel matrix
5.5 Summary

In this work, a simple model-agnostic data recovery method based on low-rank matrix approximation has been proposed for improving the quality of phasor measurements with additive noise and data drop. The method is applicable for both single and multiple measurement channels; and can deal with simultaneous and consecutive data drop on all channels. Two variations of the recovery algorithm are shown- a) an offline block-processing method for imputing past measurements, and b) an online method for predicting future measurements. The performance of the proposed algorithms have been illustrated through extensive numeric experiments on simulated measurements on the IEEE 39-bus test system. It is seen that the proposed methodology has high accuracy, has low memory requirement and is computationally faster than other methods proposed in existing literature. The performance of the proposed data recovery algorithms is independent of the underlying system model, topology changes, and measurement noise distribution. Tests on real PMU data validate the performance of the data recovery strategy put forth in this work. The fast computation speed and ease of implementation make the algorithms developed in this work suitable for quick deployment.
Chapter 6


6.1 Introduction

In the power transmission network, temporary faults may be caused by momentary line contact with vegetation or animals. Resultant faults may have high fault impedance and get cleared without the action of any protective element, making the localization task especially challenging. Frequently recurring disturbances in close proximity to each other might indicate the presence of system or equipment vulnerabilities, which might lead to catastrophic failures in the future. Therefore, localizing disturbances and rectifying them in time is a critical requirement for maintaining safe and reliable grid operations. Traditional methods for disturbance localization have included impedance measurement and travelling wave based approaches which are sensitive to line parameters or have high sampling rate requirements [121]. The recent large-scale deployment of Phasor Measurement Units (PMU) in the transmission network motivates the exploration of data-driven approaches for localizing transient faults [122, 123, 124].

In recent years, deep learning (DL) based models have been gaining acceptance and popularity in the power systems community due to their capability of learning highly non-linear relationships among variables. In recent literature, DL has been used for diverse applications like event classification [125], dynamic security assessment [126], and load forecasting [127]. One of the main hurdles in the path of wider adoption of DL in the power domain is the limited availability of labeled data. Therefore, unsupervised learning methods, which attempt to learn some structural information embedded in the data itself need to be explored.

This paper proposes a Deep Visual Learning framework (DeVLearn) for locating transient
CHAPTER 6. A DEEP VISUAL LEARNING FRAMEWORK FOR LOCALIZING TEMPORARY FAULTS IN POWER SYSTEMS

Figure 6.1: The DeVLearn framework

Figure 6.2: Procedure for constructing unthresholded RP images from time-series data, reproduced from [131]. On the left panel, we show a simple univariate time series $f(t)$ with 12 samples. The middle panel shows its two dimensional phase space trajectory with delay embedding 1. The dots are system states such that $s_i : (f(i), f(i + 1))$. The right panel shows the unthresholded RP for $f(t)$. It is a $11 \times 11$ matrix, whose $(i, j)$-th entry is the euclidean distance between $s_i$ and $s_j$ in the phase space.

faults, particularly identifying the faulted line. Once the line has been identified, analytical methods, like the one proposed in [122], may be used to pinpoint the exact fault location. Our primary objective is to compute a compressed representation of measurement data in a lower dimensional space by training a deep Variational Auto-Encoder (VAE) [128]. In DeVLearn (Fig. 6.1), this is done by first embedding a time series of length $n$ into a $n \times n$ image using unthresholded Recurrence Plots (RP) [129, 130]. The image is then compressed to a single point in a lower order $k$-dimensional space, also called ‘latent space’. This latent space may then be analyzed to efficiently learn measurement features. The image embedding step enables direct application of DL techniques from the computer vision domain.

Performance of DeVLearn is demonstrated using measurements from two generator buses in the IEEE 68 bus test system network. Temporary three-phase faults with different clearing times and fault impedances are simulated on two different lines. Here, only generator buses have been explicitly chosen as measurement sites since they are more likely to be instrumented with PMUs in reality. As the DeVLearn model is trained, machine responses to faults at different locations separate into well-defined clusters in latent space.

It is acknowledged that collecting enough field data to train a DL model can be a daunting challenge, and this lack of data availability has been a prime obstacle in the path of widespread usage of learning based methods in power systems applications. However, sys-
tem operators invest significant efforts in maintaining accurate models of their transmission networks [132]. Hence, this work assumes that the training data may be generated using simulation models, and then validated using sparse data recorded from the field. The current work demonstrates that the proposed methodology performs well on simulated fault data.

The main contributions of this work are summarized as follows. 1) We show how the un-thresholded RP method may be used to represent univariate time series data as images that preserve the temporal relationship in the data. RPs can be used for learning latent structures which might not have been very apparent in the original 1D signal. 2) We introduce a DL framework to compute a compressed representation of RP images in low dimensional latent space. We show that the deep VAE is able to separate fault measurements into discernible clusters in this latent space, and hence off-the-shelf classifiers may be used to localize the faults. This dimensionality reduction or feature learning technique is a novel contribution in the power system domain and has immense potential even beyond the fault localization task.

6.2 Methodology

Recent years have seen DL achieve major breakthroughs in computer vision and speech recognition [133, 134]. DL approaches for time series analysis, however, have not made comparable strides so far. Some deep generative models have been proposed for learning underlying structures in one-dimensional time series data [135], but their performance is heavily dependent on hyperparameter tuning. We propose to leverage image processing advancements in the power domain by first converting measurements to unthresholded RP images and then training a DL model to recognize latent structures present in them. The efficiency of using RP-based learning for time series classification (TSC) has been demonstrated in [131]. Here, the authors show that RP-embedding is more efficient for TSC than other benchmark methods as well as other image embedding methods proposed in literature, for example in [136].

6.2.1 Recurrent Plots

Time series data are characterized by distinct behavior like periodicity, trends and cyclicities. Dynamic nonlinear systems exhibit recurrence of states which may be visualized through RPs. First introduced in [129], RPs explore the \(m\)-dimensional phase space trajectory of a system by representing its recurrences in two dimensions. They capture recurrence, i.e. how frequently a system returns to or deviates from its past states. Mathematically, this recurrence may be expressed as below.

\[
R_{i,j} = ||\vec{s}_i - \vec{s}_j||, \quad i, j = 1, 2, \ldots K
\]  

(6.1)

Here, \(\vec{s}_i\) and \(\vec{s}_j\) represent the system states at time instants \(i\) and \(j\) respectively. \(K\) is the number of system states considered. The quantity \((i - j)\) maybe tuned, and is called delay
In the original RP method, the $R$ matrix is binary, i.e. its entries are 1 if the value of $||s^i - s^j||$ is above a pre-determined threshold and 0 otherwise. We do away with the thresholding since unthresholded RPs capture more information. Images so obtained capture patterns which may not be immediately discernible to the naked eye. A detailed procedure for constructing a RP plot of a simple time series is shown in Fig. 6.2.

### 6.2.2 Variational Autoencoder

Learning approaches may be broadly categorized into discriminative and generative models. Discriminative models try to learn boundaries between different classes of labeled data. On the other hand, generative models are trained to reconstruct high-dimensional data from restricted information, thereby forcing them to learn some latent structural features in the data itself. In simple words, these models take a training set of examples drawn from an unknown data generation distribution ($p_{\text{data}}$, say) and returns an estimate of the distribution, ($p_{\text{model}}$, say). Generative models are especially powerful since they do not need labeled data.

An autoencoder (AE) is an unsupervised generative model where a neural network (NN) is trained to generate outputs that replicate its inputs [133]. Particularly, a ‘bottleneck’ in the NN architecture is leveraged to create a lower dimension representation of the inputs in the latent space. AEs comprise of two components- a) an encoder that learns a compressed representation of the input data (dimensionality reduction), and b) a decoder that learns to reconstruct input data from the compressed representation.

A VAE uses variational inference to generate the distribution of latent variables in the lower dimensional space [137]. The distribution of latent variables $z$ for a given input data $x$ follows the posterior distribution $p(z|x)$. Computing $p(z|x)$ in closed form results in an intractable integral. To this end, the variational inference method uses a different distribution $q(z|x, \lambda)$ to approximately infer the computationally intractable distribution $p(z|x)$. This is ensured by training the VAE such that the KL-divergence between the distributions $q(z|x, \lambda)$ and $p(z|x)$ is minimized. It can be shown that the distribution $q(z|x, \lambda)$ which approximates $p(z|x)$ minimizes the expression in (6.2).

$$\arg \min_{q(z|x, \lambda)} -\mathbb{E}_{q(z|x, \lambda)} \log p(x|z) + \mathbb{KL}(q(z|x, \lambda)||p(z))$$ \tag{6.2}$$

The first term in (6.2) is the reconstruction loss or expected negative log-likelihood for input
6.2. METHODOLOGY

Measurements
Downsampled data
Fault Location
Unthresholded RP

Figure 6.4: Pipeline showing DeVLearn operation

Just like an AE, a VAE also consists of an encoder, a decoder and a loss function. The encoder is a NN which generates parameters $\lambda$ for the distribution $q(z|x, \lambda)$. The decoder is another NN trained to reconstruct the input data $x$ from a given latent representation $z$. The loss function is a weighted sum of the reconstruction loss and KL-divergence terms. Choosing a weight corresponding to the reconstruction loss which is significantly higher than the other results in overfitting of the VAE, whereas a higher weight for the KL-divergence term enforces the distribution $q(z|x, \lambda)$ to follow the prior distribution $p(z)$. It is important to note that VAEs are generative models that learn the distribution of the dataset and do not require labeled data.

6.2.3 DeVLearn Framework

The DeVLearn framework puts the components discussed above together to achieve a very powerful latent space representation. The pipeline of the framework is shown in fig. 6.4. The steps involved are reiterated below, for clarity.

Step 1: To reduce computation burden, the time series data (in this case, a measurement
window containing the temporary fault) is first downsampled using Piecewise Aggregate Approximation (PAA). In this paper, the original signal has been downsampled by a factor of five.

**Step 2:** The down-sampled data is converted to corresponding unthresholded RP images following the procedure described in section 6.2.1. The delay embedding is tunable; for this work a delay embedding of 1 was empirically chosen.

**Step 3:** A Convolution Neural Network (CNN) based deep VAE model is trained to learn the latent space distribution of the fault data \[138\]. CNNs have been used due to their excellent ability of extracting features from images. The latent space is considered to be two dimensional. The intention was to keep the latent space dimension as low as possible, so as to lower the computation burden for the downstream classifier. The encoder has two hidden layers while the decoder has a single hidden layer. The structure of the deep VAE is similar to the Keras example available in [139].

As explained in section 6.2.2, the VAE loss function has two elements- reconstruction loss (RL) and KL-divergence term (KLD). The mean squared error (MSE) metric between input data \(x\) and reconstructed data \(\hat{x}\) has been used for the RL term. Since our downstream application desires well-separated clusters in the latent space, we reduce the relative weight for the KLD term. Potential for other downstream applications also exist. For instance, VAEs have been employed to generate synthetic images in literature [133]. Methods of recovering original signals from unthresholded RPs have already been proposed in literature [140]. Therefore, DeVLearn may be modified to generate realistic synthetic PMU data. This is an exciting research direction that the authors intend to pursue in their future work.

**Step 4:** In the latent space learned in step 3, each signal is compressed to a single point in two-dimensional space. The novelty of DeVLearn is in its capability to learn such a latent space representation, where measurements corresponding to different fault locations are automatically separated into disentangled clusters, even when the DL model has no explicit knowledge of the data labels. Any off-the-shelf computationally efficient classifier like Support Vector Machines (SVM) can now be trained with fault location labels and used to determine the location of an unseen fault.

### 6.3 Results and Discussions

This section describes the computational experiments performed to assess the performance of the proposed DeVLearn platform. All power systems simulations are carried out using PSS/E.
6.3. **Results and Discussions**

**6.3.1 Experimental Setup**

In this paper, transient faults on two transmission lines A and B in the standard IEEE-68 bus power system have been analyzed. To this end, 1000 three-phase faults were simulated for each line and the voltage magnitudes at two generator buses (Generators 1 and 4) were recorded. Location of the generator buses and faulted lines is shown in Fig. 6.5.

The fault impedance for each event is randomly sampled from a uniform distribution between 0 and 1000 Ohms. Similarly, the fault duration is assumed to be uniformly distributed over 10 to 20 cycles of power systems frequency. The simulated events are split into training and testing datasets. Training and testing datasets have 1800 and 200 events respectively. DeVLearn is trained using the GPU hardware acceleration option available on Google Co-laboratory. Training for a batch size of 100 took around 550 µs for a single epoch.

It must be mentioned here that detecting presence of temporary faults has not been considered in the scope of DeVLearn. Multiple methodologies have been proposed to detect temporary disturbances in literature and may be used for this purpose [141].

**6.3.2 Recurrent Plots for Faults**

In order to better understand how generator response to fault events translate to RP images, let us look at the RPs for voltage magnitude measurements at generator buses 1 and 4 for two faults at lines A and B. Fig. 6.6 and 6.7 respectively show the time series measurements (downsampled by a factor of 5) at buses 1 and 4, while Fig. 6.8 shows the corresponding RP images. In these images, time progresses in a diagonal manner, from the upper left corner to the lower right. Unthresholded RPs preserve temporal ordering in the original time series data. It is evident that RP images for the different events may be distinguished, even by the naked eye. Preliminary exploration revealed that images for similar events indeed

Figure 6.6: Voltage at Gen. 1 for faults at lines A and B

Figure 6.7: Voltage at Gen. 4 for faults at lines A and B

Figure 6.8: Unthresholded RP images for measurements shown in Fig. 6.6 and Fig. 6.7.
6.3. Results and Discussions

Figure 6.9: Evolution of latent attribute distribution in the compressed two dimensional space over different training epochs. Figs. 6.9a-6.9c shows the evolution of the latent space when the DeVLearn framework is trained with voltage magnitude measurements at generator bus 1. The distribution of latent space attributes for measurements at generator bus 4 is shown in Figs. 6.9d-6.9f. It can be seen that as training epoch progress, measurements corresponding to faults at different locations separate into discernible clusters in the latent space.
look similar in shape (not necessarily magnitude), even for high impedance faults, where the voltage deviation at generator buses is not very high. The objective now is to teach DeVLearn to recognize the RP images and associate them with the events they correspond to.

6.3.3 Training the VAE

The deep VAE component of DeVLearn is trained using 1800 instances of $120 \times 120$ grayscale images generated from voltage magnitude measurements. Each training epoch uses a batch size of 100 data points. A separate DeVLearn framework was trained for each of the generators, but the VAE architecture and weights of loss function terms were not altered. This was done to check if satisfactory results could be obtained without relying on significant hyperparameter tuning.

The encoder projects the training set onto a compressed two dimensional latent space, whose evolution with training epochs is shown in Fig. 6.9. It can be clearly seen that data from different faults start separating out into clusters as training progresses, and significant separation is achieved at 500 epochs from both the models. Therefore, an estimate of fault location may be obtained using only local voltage magnitude measurements. It must be emphasized that the fault location labels are not utilized by the VAE to learn the distribution of the latent variables.

The data window length considered for experimentation in the present work was fixed. However, further experimentation with varying lengths can provide further insights into optimal window length requirements.

6.3.4 Determining Fault Location

We check the performance of a simple SVM-based classifier with linear kernel on the latent space learnt by the DeVLearn framework after 1000 training epochs. The resultant decision boundary is shown in Fig. 6.10. It is evident that in the latent space, fault data for two lines are almost linearly separable. With a linear SVM classifier, we obtain a training accuracy of 99.33% and 99.72% for generators 1 and 4 respectively. Testing accuracy for both generators is 99.5%. Although a classifier with a non-linear kernel (Radial Basis Function or RBF kernel, for instance) would have achieved higher accuracy, the intention was to show that satisfactory accuracy levels may be obtained even with simple linear kernels.

6.4 Summary

This paper provides a proof of concept that image embedding aided deep learning may be utilized to determine the location of temporary faults in power systems with a reasonably high accuracy. The capability of the proposed framework DeVLearn in learning useful information from unlabeled univariate time series data in the context of distinguishing faults at different
6.4. Summary

(a) Classifying measurements from Gen. bus 1: training data
(b) Classifying measurements from Gen. bus 1: testing data
(c) Classifying measurements from Gen. bus 4: training data
(d) Classifying measurements from Gen. bus 4: testing data

Figure 6.10: Using a SVM classifier with linear kernel to classify the latent space learned by DeVLearn. Red and blue dots indicate data points corresponding to faults at lines A and B respectively.

locations has been demonstrated. This is lucrative, keeping in mind the limited availability of labeled data in the power domain which has hindered the large-scale adoption of learning based tools in the industry thus far.

Of course, more tests with different systems, faults at different lines, network topologies and operating conditions are required to place higher confidence in DeVLearn, and this is a direction that the authors are pursuing. The idea is to train the DeVLearn framework with fault data obtained from simulation models; and then validate its performance with the help of actual PMU data recorded in the field. Further potential exists in extending the proposed methodology beyond the fault localization application, for example, in event classification and synthetic PMU data generation. Research scope also exists in extending image embedding strategies to multivariate time series data, which maybe leveraged to capture information from multiple measurement channels in a PMU.
Chapter 7

Conclusion

The power grid is a critical national infrastructure, and its availability is integral to the normal functioning of modern society. In this dissertation, novel strategies for limiting the impact of extreme events on the grid are developed. Broadly, the proposed approaches address two questions:

• How to best improve grid performance in the aftermath of a disruptive event using existing resources?
• How to enhance grid monitoring using data-driven strategies?

In this chapter, we summarize how the present work develops answers to the above questions, and outline future research directions.

7.1 Contributions

The first part of the dissertation (chapters 2 - 4) develops network partitioning strategies for improving grid performance after an extreme event. Improved system availability at this phase will alleviate the impact of disasters before grid restoration activities can be completed.

• Chapter 2 illustrates a constrained hierarchical spectral clustering methodology to proactively split the AC interconnected bulk power grid into smaller self-sustaining islands before HILF events. The proposed strategy accounts for prevailing operating conditions, the nature of the expected contingency, and the ease of control room operation. The island identification strategy is evaluated under different seasonal load-generation conditions on the heavily meshed transmission grid operated by PJM Interconnection in the eastern US. The operational viability of the identified islands is verified using steady-state AC power flow on the PSS/E model of the PJM transmission network. This work provides a proof of concept that proactive islanding can mitigate extreme events by arresting the propagation of cascading disturbances. This forms the first step towards formulating a standard operating procedure for defending the power grid against catastrophic events.

• Chapter 3 describes two fast heuristic methods to determine an operation sequence that can be followed to isolate chosen sub-networks from the rest of the grid. LODF values are used to compute the approximate redistribution in network real power flows.
Such proactive islanding may be necessary to mitigate extreme events, as discussed in chapter 2. Case-studies on a modified IEEE 39-bus 10-machine test system show that the operation sequences chosen by the proposed methods can avoid line overloads, maintain bus voltages within prescribed limits, and outperform a randomly chosen sequence of operations. Such operation sequence determination tasks have not been adequately explored in existing literature.

- Chapter 4 proposes a methodology for identifying potential microgrids in the existing distribution network. The aim is to identify sub-networks that can be converted to IEEE 1547 standard compliant microgrids by economically viable retrofitting. Microgrids improve grid resilience by maintaining supply to essential loads and aiding recovery efforts during and after outages in the bulk power grid. In the arena of optimal microgrid planning, this work makes the following contributions:
  
  - **First**, we formulate a deterministic optimal distribution network partitioning problem for identifying optimal microgrid candidates in a network. The formulation is posed as a MILP, and comprehensively addresses operation constraints such as maintaining a) voltage and power flow limits, b) network radiality, and c) availability of grid-forming generators in each microgrid. The formulation is without any pre-assignment and determines the ideal network topology in a single stage. The grid-forming generator availability constraint had not been considered in the prior microgrid planning literature.
  
  - **Second**, to account for uncertainties in demand and generation, the ODNP formulation is extended to a probabilistic setup, and a computationally tractable SAA technique is used to solve it. The quality of the obtained solution is evaluated through statistical tools.
  
  - **Third**, the probabilistic ODNP formulation is illustrated through extensive numeric tests on a modified IEEE 37-bus feeder. It is shown that the SAA approach efficiently utilizes network flexibility, and outperforms a robust clustering-based method in terms of objective cost.

The second part of the dissertation (chapters 5 and 6) suggest ways to enhance data-driven grid monitoring. Model-agnostic algorithms are developed for recovering degraded PMU data, and a deep learning methodology is formulated to identify the location of temporary faults in the transmission network.

- Chapter 5 presents a data recovery method for noisy PMU data streams with missing entries. The measurements are transformed into a stacked page matrix, whose low-rank property is utilized to reconstruct the corrupt signals. Two variations of the recovery algorithm are demonstrated - a) an offline method for reconstructing past measurements, and b) an online method for predicting future measurements. Information within a PMU channel (temporal correlation) as well as from different PMU
CHAPTER 7. CONCLUSION

channels in a network (spatial correlation) are utilized to recover the degraded data. The proposed method is faster than other methods in literature, poses minimal memory requirement, and needs no explicit knowledge of the underlying system model or measurement noise distribution. The performance of the recovery algorithms is illustrated using simulated measurements from the IEEE 39-bus test system as well as real measurements from an anonymized U.S. electric utility. Extensive numeric tests show that the original signals can be accurately recovered in the presence of additive noise, consecutive data drop as well as simultaneous data erasures across multiple PMU channels.

• In chapter 6, a novel image embedding aided deep learning framework is proposed for faulted line location using PMU measurements at generator buses. Frequently recurring transient faults in a transmission network may be indicative of impending failures. Hence, determining their location is a critical task. PMU data provides high-resolution insight into grid dynamics, and can be leveraged to identify precursors to faults. Inspired by breakthroughs in computer vision, we represent measurements (one-dimensional time series data) as two-dimensional unthresholded Recurrent Plot (RP) images. These RP images preserve the temporal relationships present in the original time series and are used to train a deep Variational Auto-Encoder (VAE). The VAE learns the distribution of latent features in the images. Our results show that for faults on two distinct lines in the IEEE 68-bus network, the DL framework can project PMU measurements into a two-dimensional space such that data for faults at different locations separate into well-defined clusters. This compressed representation may then be used with off-the-shelf classifiers for determining fault location. The efficacy of the proposed method is demonstrated using local voltage magnitude measurements at two generator buses.

7.2 Future Work

In this dissertation, optimization and learning techniques have been developed for enhancing power grid resilience and data-driven monitoring. Some future research directions are outlined below:

• Splitting an interconnected power grid into multiple islands requires the coordinated switching of multiple lines and generator dispatch actions. The formulation of an exact sequence of steps poses an interesting research question that we have tried to answer in this dissertation. However, the heuristic approaches presented in this work do not consider dynamic system performance or the possibility of component failures during line switchings. The present work must be extended to address these limitations. Moreover, our work makes an assumption that generation dispatch and load-shedding required to achieve power balance in the intended islands will be executed before line switching begins. Establishing power balance may be a time-taking process due to
the ramping constraints of generators, and hence an operation plan that considers line switching and generator set-point control actions in tandem at each step may be useful. Further, standardized action plans will need to be formulated to ensure coordination among multiple stakeholders before any splitting action of the bulk power grid can be executed.

- The potential to use heavy-duty electric vehicles as mobile black-start generators is promising for microgrids in the distribution network. Multiple DER-load clusters could function as microgrids sharing generators with black-start (grid-forming) capabilities. Efficient deployment of EV generator fleets can help maintain service to a larger area following an outage.

- By providing high-resolution precise time-synchronized measurements at different points in the power network, PMUs offer an accurate description of the state of the grid. In this work, we have demonstrated how tools from the computer vision domain can be applied for identifying the location of temporary faults in the transmission network. The proposed framework can be further developed for other applications such as generating synthetic fault data. Moreover, strategies for fusing multiple data streams to create a wide-area image of grid conditions must be pursued.
Bibliography


[57] American national standard for electric power systems and equipment—voltage ratings (60 Hertz), 2016.


[62] IEEE guide for design, operation, and integration of distributed resource island systems with electric power systems.


In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of Virginia Tech’s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink. If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.