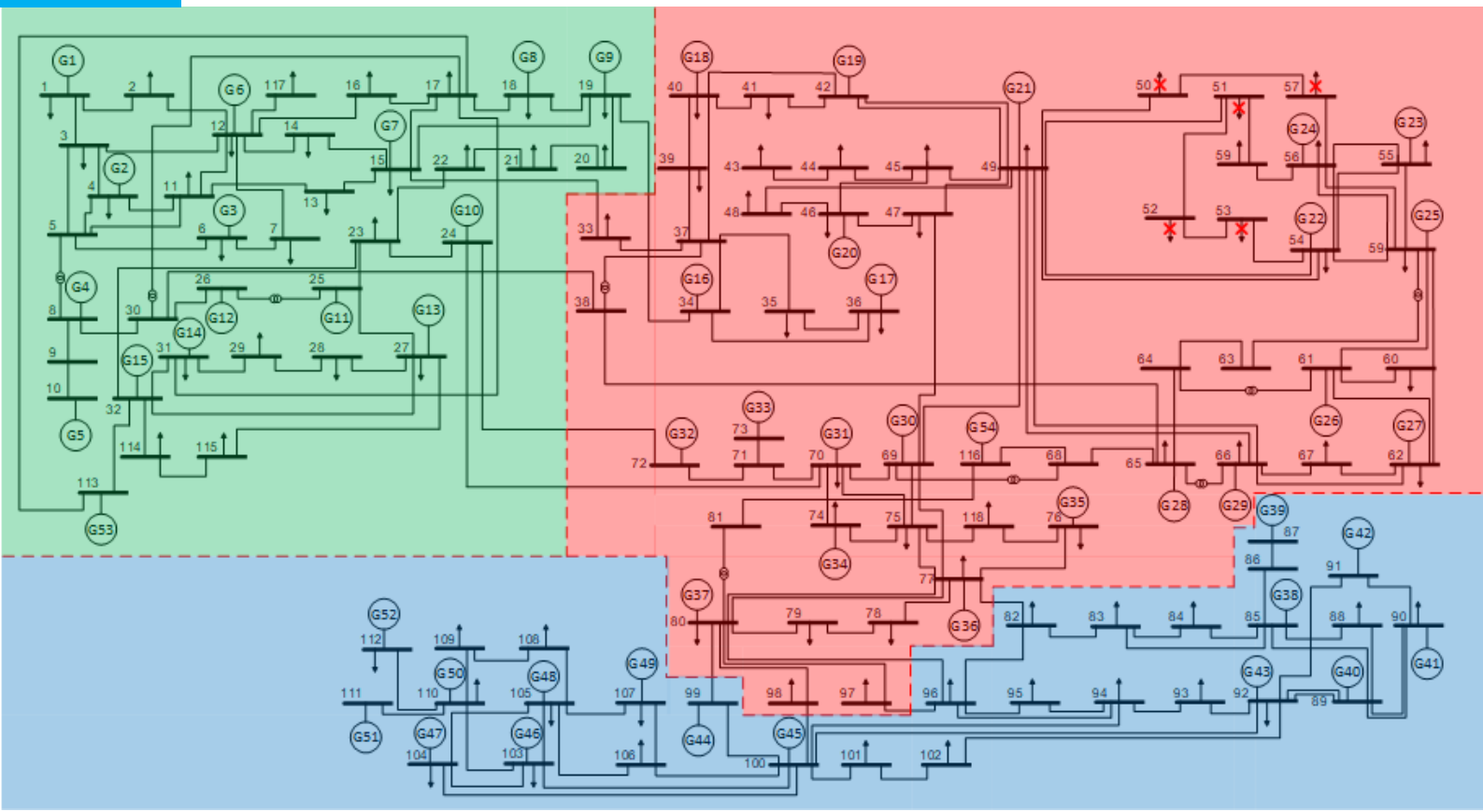
Development of an adaptive intentional controlled islanding scheme and post-islanding corrective control actions

Master of Science thesis

Charalampos Alexandros Karagiannis Kaltsikis

Delft University of Technology



TU Delft Delft University of Technology

Challenge the future

Development of an adaptive intentional controlled islanding scheme and post-islanding corrective control actions

Master of Science Thesis

by

Charalampos Alexandros Karagiannis Kaltsikis

in partial fulfilment of the requirements for the degree of
Master of Science
in Electrical Engineering
in Electrical Sustainable Energy
Department of Intelligent Electrical Power Grids (IEPG)
Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS)
Delft University of Technology
To be defended publicly on 11 October 2017 at 3:00 PM

Supervisors:	Dr.ir. Marjan Popov
	Phd candidate Ilya Tyuryukanov
Thesis Committee:	Prof.ir. Mart van der Meijden
	Dr.ir. Marjan Popov
	Dr.ir. Mohamad Ghaffarian Niasar

Abstract

Power systems have undergone major changes the past few decades, leading to an increase in the frequency of occurrence and severity of blackouts. Intentional controlled islanding constitutes the final resort to rescue the system from a blackout by separating the network into islands in a controlled manner. Various post-separation control action such as generation rejection and load shedding have to be considered in order to restore the active power balance in the created islands.

The transient stability of the constructed islands can be enhanced by identifying and grouping together the coherent generators of the power system. This can be achieved by representing the electrical network as an undirected graph and performing constrained graph partitioning with respect to the generator nodes.

If the demand is higher than the active power produced, the frequency of the formed island starts declining. To deal with this problem and keep the frequency within acceptable limits, it often becomes necessary to curtail a proportion of the load. An advanced underfrequency load shedding scheme can dynamically adjust the shedding steps based on accurate power deficiency estimation and thus minimize the total load shed.

In this thesis, the coherent generator groups are identified for several power systems. Then constrained graph partitioning is used to split these networks into islands, while forcing the coherent generators to remain in the same island. For this step, three different algorithms performing constrained clustering are implemented in Matlab and compared for two different study cases. Moreover, an advanced underfrequency load shedding scheme is applied on various constructed islands by utilizing PowerFactory and Python. Its objective is to minimize the load shedding amount, while keeping the frequency within the predefined limits. The amount of load that needs to be curtailed is estimated based on the frequency gradient of the generators in the island with an excess in power demand.

Key words: Intentional Controlled Islanding, Constrained Graph Clustering, Coherency, Underfrequency Load Shedding, Power Imbalance Estimation, Matlab, PowerFactory, Python

Acknowledgements

First of all, I would like to express my heartfelt gratitude to my supervisor Ilya Tyuryukanov who helped me by providing both useful literature and valuable advice whenever difficulties arose. His constant guidance and academic knowledge greatly aided throughout this thesis. I am compelled to admit that I have been extremely lucky to have him as my thesis supervisor.

I would like to thank Dr.ir. Marjan Popov of the Intelligent Electrical Power Grid (IEPG) group. He gave me the opportunity to work on this very interesting research topic in the field of power systems. The master courses he teaches are very important for my professional knowledge and skills.

I would also like to take this opportunity and thank the committee members Prof.ir. Mart van der Meijden and Dr.ir. Mohamad Ghaffarian Niasar for taking the time to read and evaluate my work.

Furthermore, a special thanks goes to all my colleagues and friends who supported me and helped me throughout my studies, here in TU Delft. I am really grateful to my friends for the great times we had together, for expanding my horizons and teaching me invaluable life lessons. You are all magnificent and I am really lucky that you are part of my life. I have shared with you some of the best experiences of my life. This journey would certainly not have been possible without you. I cannot thank you enough for your friendship and support over these years.

Finally, I want to thank my family for their encouragement and support all these years and the trust in my decisions they always demonstrate.

Charalampos Alexandros Karagiannis Kaltsikis
Delft, October 2017

List of figures

List of tables

List of symbols

Scalars

Name	Unit	Description
N	1	Number of buses of the power system
n	1	Number of machines of the power system
f_n	Hz	Nominal system frequency
ω_n	rad/s	Nominal system frequency
E'	p.u.	Transient emf of synchronous machine
x'	p.u.	Transient reactance of synchronous machine
δ	rad	Angle of transient emf
H	s	Inertia constant
P_m	p.u.	Input mechanical power
P_e	p.u.	Output electrical power
λ	1	Eigenvalue
k	1	Number of clusters
n_s	1	Number of slow modes
a	1	Constraint satisfaction threshold of FCSC
b	1	Constraint threshold of FCSC
P_{cut}	MW	Total power cut
ω_s	rad/s	Synchronous rotational speed
ω	rad/s	Rotational speed
P_{Li}	p.u.	Active power of i^{th} load
P_{L0i}	p.u.	Pre-fault active power of the i^{th} load
V_{Li}	p.u.	Voltage of the i^{th} load
V_{L0i}	p.u.	Pre-fault voltage of the i^{th} load
Q_{Li}	p.u.	Reactive power of the i^{th} load
Q_{L0i}	p.u.	Pre-fault reactive power of the i^{th} load
α_i	1	Voltage dependency on active power of i^{th} load
β_i	1	Voltage dependency on reactive power of i^{th} load
P_{def}	p.u.	Active power deficiency
f_{COI}	Hz	Center of inertia frequency
H_{sys}	s	Equivalent apparent inertia of the system
S_{sys}	MVA	Equivalent apparent power of the system
P_{L0}	MW	Total pre-fault load power
N_L	1	Number of loads in the system
$P_{shed,i}$	1	Shedding amount of the i^{th} load shedding step
Δ_i	1	Change in the ROCOF before the i^{th} load shedding step
$P'_{shed,i}$	1	Adjusted shedding amount of the i^{th} load shedding step
$f_{th,i}$	Hz	Frequency threshold of the i^{th} load shedding step
t_{delay}	s	Time delay
t_{est}	s	Estimation time

Vectors

$\boldsymbol{v}$	Eigenvector
$\boldsymbol{u}$	Cluster indicator vector

Sets

Name	Description
$\mathcal{E}$	Set of edges
$\mathcal{V}$	Set of vertices
C	Subset of the graph

Matrices

Name	Description
$\mathbf{M}$	Diagonal machine inertia matrix
$\mathbf{K}$	Synchronizing torque coefficient matrix
$\mathbf{Y}$	Admittance matrix
$\mathbf{Y}_{\mathrm{red}}$	Reduced admittance matrix
$\mathbf{Q}$	Constraint matrix
$\mathbf{L}$	Laplacian matrix
$\mathbf{D}$	Degree matrix
$\mathbf{W}$	Adjacency matrix
$\mathbf{I}$	Identity matrix

Other symbols

G	Graph
vol	Volume of cluster
∂	Boundary of cluster
φ	Expansion of cluster
N_{cut}	Normalized cut

List of acronyms

UFLS	Underfrequency Load Shedding
PMU	Phasor Measurement Unit
FCSC	Flexible Constrained Spectral Clustering
TCRBGC	Tight Continuous Relaxation of Balanced Graph Cuts
TSO	Transmission System Operator
ROCOF	Rate of change of frequency
COI	Centre of inertia

Table of contents

Chapter 1 Introduction

In this chapter the background of the thesis is introduced. The problem that justifies the present research is explained along with relevant previous work on this topic. Moreover, the objective and the contribution of this thesis are defined. The scope of the work is also provided. Finally, the structure of the thesis is presented.

1.1 Problem definition

In recent years, major modifications have occurred in the electric power systems related to the growth of installed capacities, massive integration of sustainable energy sources and deregulation of the electrical industry. As a result of these changes, more severe and more frequent blackouts have taken place. The precise reasons behind electrical power system blackouts are complex and vary depending on the situation. However, most of wide area blackouts studied in the literature such as [1] and [2] were a consequence of stressed network operation and cascading failures, following the sequence depicted in Figure 1-1.

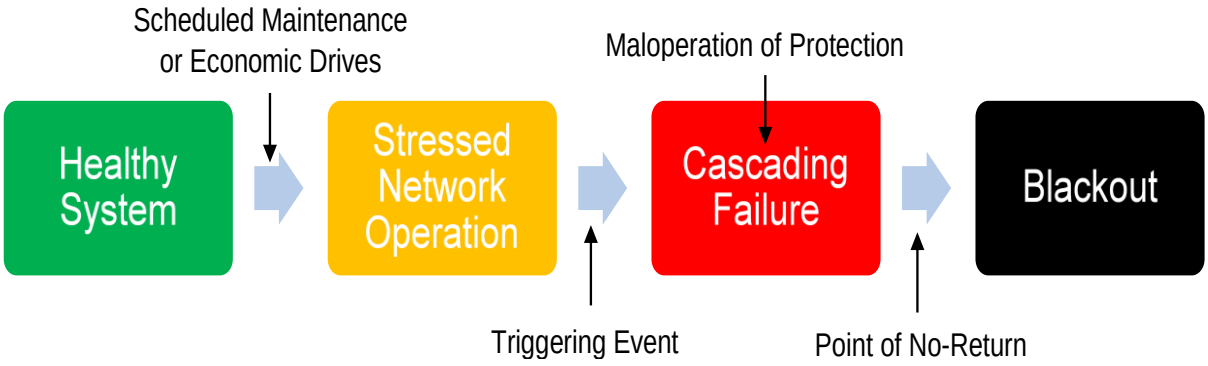


Figure 1-1: Generalized blackout pattern (adapted from [3])

After the protection schemes are triggered by severe disturbances a frequent result is the formation of two or more islands. Islands created in this manner may be unsustainable, as their generators could go out of synchronisation. Furthermore, the generators might lack the production to balance the demand and some of their components can be overloaded leading to even more protective relay actions. If this situation is not controlled rapidly, it can lead to a cascading effect resulting in the collapse of the electrical network into several unsustainable islands and ultimately in large power outages [4].

Intentional controlled islanding constitutes the last-resort measure against blackout following severe disturbances. Instead of letting the transmission network to collapse by itself, controlled power system separation deals with this problem, by splitting the network into islands, each consisting of buses and generators with similar properties, also ensuring that the loads can be supplied by the generators of each island. If the islands formed are properly designed, their resynchronization with the power grid will also be easier [4].

When the islands are created, there is always either a surplus or a deficit of active power [5]. Therefore, several post-separation control actions may be considered in order to repair the constructed islands and restore the balance. Such actions usually include generation rejection in case the production is greater than the demand and load shedding when the demand is higher than the island's active power generation [4]. The present thesis focuses on

the second alternative, which is used to prevent the frequency of the island from further decreasing due to the lack of active power production.

1.2 Previous work

The concept of partitioning the power grid into smaller islands for preventing wide area blackouts is a relatively new research topic. In the past few years, many researchers adopted the approach of representing the electrical network as a graph and built upon it. In [6] a two-step spectral clustering islanding algorithm is introduced. Firstly, generator nodes are grouped with normalized spectral clustering to produce the coherent groups of generators. Secondly, constrained spectral clustering is implemented to minimize the power flow disruption and satisfy the coherency constraints. In both the steps of the algorithm the coherent groups of generators and the final islanding solutions are obtained through the use of the k-medoids algorithm. In [7] hierarchical spectral clustering is applied in order to create the islands, as this algorithm takes into account the interconnection structure of the network and more precisely the distance between the buses.

Underfrequency load shedding (UFLS) constitutes an automatic procedure to disconnect a certain amount of loads. There are several types of UFLS protection schemes: traditional, semi-adaptive and adaptive [8]. The authors of [5] concentrate on the frequency gradient and the main factors affecting it, as indicators to detect the active power deficit in the grid. The goal of [8] is to minimize the amount shed by utilizing the frequency gradient along with as much primary frequency control as possible. Furthermore, a new approach to predict the frequency a few second in advance in order to compute the amount of load to be shed is explained in [9]. Finally, a method is proposed in [10] that combines UFLS with under voltage load shedding, using both voltage and frequency data collected by Phasor Measurement Units (PMU). That way the reactive power is taken into account along with the active power in the load shedding strategy.

1.3 Thesis objective and contribution

The purpose of the present thesis is to implement several methods that could be used as building blocks for the design of an adaptive intentional controlled islanding scheme. Firstly, focus is given into enhancing the transient stability of the created islands. Post-islanding corrective control actions are also considered, because at the moment the island is formed it is impossible to avoid an active power imbalance. Thus, this thesis will be split in two major parts:

- Implementation of constrained graph partitioning based on generator coherency constraints.
- Load shedding, on various islands, if necessary.

As mentioned in the previous section, both these topics have been studied in the literature. This thesis aims to contribute in the following areas. As far as constrained clustering is concerned, the extent to which it is possible to compute coherency constrained cutsets in power systems utilizing semi-supervised learning algorithms is studied. Most of the existing work focuses on relatively small networks (up to 300 buses), while in this thesis networks of up to 2869 buses are considered. Moreover, variations of constrained spectral clustering are used in the literature in order to calculate coherency constrained cutsets. However, in this thesis two alternatives not yet considered for electrical networks are implemented. Their results are compared and the one that satisfies the constraints better is identified.

Besides constrained graph clustering, UFLS related aspects are discussed. The innovative method for estimating the active power deficiency in an island based on the frequency gradient of the generators [5] is implemented and its feasibility is assessed. The parameters affecting the amount of load that needs to be curtailed are identified. What is more, the performance of a load shedding scheme for minimizing that amount is evaluated for various amounts of power deficiency.

1.4 Scope of work

In order to achieve the objective stated in the previous section, the following scope of work had been defined for the two topics studied:

The first part of the thesis implements constrained graph partitioning in various electrical networks based on the coherency assignment of the generators. For this step Matlab R2017a is used. Matpower toolbox is utilized, since it provides power flow data about several power networks and contains multiple power flow solvers. Also, PST toolbox is used to produce the electromechanical model of each system. Based on it the coherent groups of generators are identified and the constraints arise. By combining these two toolboxes, the aim is to identify coherency-constraint minimal cutsets in the network, with the goal of increasing the constraint satisfaction as much as possible. In order to achieve that, three different algorithms performing constrained graph partitioning are implemented and tested in Matlab.

Secondly, having obtained various islanding solutions, underfrequency load shedding is applied to the islands where the demand is higher than the production. This step is implemented utilizing DIgSILENT PowerFactory 2017 SP1 and Python 3.4. The initial load shedding algorithm was scripted by Meng Zhang during his thesis at TU Delft [3]. The goal here is to adjust the already existing scheme to operate in one RMS simulation and minimize the amount of load being curtailed, as this could result in significant economic benefits. After obtaining the results from PowerFactory as csv documents, Matlab is also employed for visualization purposes.

1.5 Thesis structure

Chapter 2 focuses on constrained graph partitioning. Firstly, the derivation of the coherency constraints is explained and for better comprehension the results of the coherency analysis are visualized on a small power system. Furthermore, the different algorithms used for constrained graph partitioning are described and the different quality indicators are demonstrated in order to measure the quality of the partitions obtained. Moreover, two different case studies are carried out and the slow-coherency cutsets of the algorithms are presented and compared. Finally, based on these results, conclusions are drawn on the suitability of the evaluated algorithms for partitioning smaller and larger power systems.

Chapter 3 deals with the topic of underfrequency load shedding. In the beginning, the issues arising when the network operates with frequency lower than its nominal are presented. Then, the method to estimate the active power deficiency is explained, along with the way to distribute the estimated amount on several adjustable steps. Moreover, the simulation environment and the test system employed to perform load shedding are briefly discussed. The necessary assumptions for the implementation of this UFLS scheme and its goal are also addressed. What is more, the results for various power deficiency scenarios are visualized and based on them, the performance of the proposed scheme is evaluated. Finally, the conclusions are presented and the applicability of such a scheme with the available infrastructure is discussed.

In Chapter 4, the accomplishment of the objective of the present thesis, set in this chapter, is addressed and the concluding observations from the research carried out are presented. The possibilities for future work, as a continuation to the present development, are also discussed.

Chapter 2 Constrained graph partitioning

Constrained graph partitioning aims to enforce specific vertices to belong to pre-assigned clusters. In power networks, such constraints can arise by identifying the coherent groups of generators. In order to enhance the transient stability of the islands formed, the generators belonging into one coherent group have to be put together in the same partition. Therefore, the algorithms discussed in this chapter are going to take into consideration the coherent groups and try to assign coherent generators into the same cluster.

2.1 Generator coherency

Generator coherency of four different power systems ranging from 118 buses up to 2869 buses is considered. The coherent generators of each electrical network are identified for different numbers of islands and thus the coherency constraints arise. Variations in power flow usually do not affect much the coherency constraints, therefore the original power flow solution of each network is used.

Producing meaningful constraints is of outmost importance, since apart from the coherency of the generators, also the topology of the grid has to be taken into account. In other words, there should be sufficient physical connections in the network to put any two coherent generators into the same island.

2.1.1 Power system electromechanical model

Several frequency components (modes) can be detected in power system oscillations. Coherent generators in the same area tend to swing together at relatively higher frequencies. On the other hand, generators in the same area often oscillate together against generators in neighbouring areas. This is called inter-area oscillation and the corresponding frequency is called inter-area mode. This phenomenon originates from the coherent areas being weakly coupled and forms the basis of the slow coherency analysis [11].

In order to acquire the natural frequencies of oscillations the classical electromechanical model is used with loads modelled as constant impedances [12]. Consider an N bus, n machines power system with nominal frequency f_n in Hz and $\omega_n=2\pi f_n$ in rad/s. Each machine i is modelled as a transient voltage E'_i behind its transient reactance x'_i. The motion of the machine rotor angle δ can be expressed as:

$$m_i\ddot{\delta}_i = P_{mi} - P_{ei} = P_{mi} - \frac{E_i V_j \sin(\delta_i - \vartheta_j)}{x'_i} \tag{2-1}$$

$$m_i = \left(\frac{2H_i}{\omega_n}\right) \tag{2-2}$$

$$V_j = \sqrt{V_{jre}^2 + V_{jim}^2} \tag{2-3}$$

$$\vartheta_j = \tan^{-1}\left(\frac{V_{jim}}{V_{jre}}\right) \tag{2-4}$$

Where:

H_i inertia constant of the i^{th} machine

P_{mi} is the input mechanical power of the i^{th} machine

P_{ei} is the output electrical power of the i^{th} machine
V_{jre} real part of the bus voltage phasor at bus j (terminal bus of machine i)
V_{jim} imaginary part of the bus voltage phasor at bus j (terminal bus of machine i)

After several mathematical steps and simplifications, explained in [11], the linearized electromechanical model reduced to the machine internal nodes can be defined as:

$$\mathbf{M}\Delta\ddot{\delta} = \mathbf{K}\Delta\delta \tag{2-5}$$

$$\mathbf{M} = diag(m_i) \tag{2-6}$$

$$K_{ij} = E'_i E'_j \left(B_{ij}\cos(\delta_i - \delta_j) - G_{ij}\sin(\delta_i - \delta_j)\right), \qquad i \neq j \tag{2-7}$$

$$K_{ii} = -\sum_{j=1, j\neq i}^{n} K_{ij} \tag{2-8}$$

with $\mathbf{M}$ being the diagonal machine inertia matrix and $\mathbf{K}$ the synchronizing torque coefficient matrix. The entries of $\mathbf{K}$ represent the torque that keeps the machines synchronized and stably connected. The terms B_{ij} and G_{ij} correspond to the real and imaginary part of the equivalent admittance between the machines i and j respectively. In general, $B_{ij} << G_{ij}$. These terms are obtained from the reduced $n \times n$ admittance matrix $\mathbf{Y_{red}}$ of the network, after performing Kron reduction to the original admittance matrix $\mathbf{Y}$, so that only the internal machine nodes remain [11]. After deriving the machine inertia matrix and the synchronizing torque coefficient matrix, the state matrix of the electromechanical model is obtained through the multiplication $\mathbf{M^{-1}K}$. A function was built to derive the $\mathbf{M^{-1}K}$ matrix according to (2-6), (2-7) and (2-8).[1]

2.1.2 Determining the number of clusters

Based on the state matrix of the electromechanical model, the eigenvalues λ and eigenvectors v are computed and sorted from minimum to maximum. Then, the natural frequencies of oscillations, representing the slow modes of the system, are derived based on the equation:

$$natural\ frequency = \sqrt{\lambda} \tag{2-9}$$

From them, the first 10 are kept, shown in Figure 2-1 for the four studied power systems. It is worth noticing that the first one is always zero.

Next, the gaps between the natural frequencies are computed. The bigger this gap is, the closer are the eigenvectors of the ideal and the perturbed case. This is justified through the perturbation theory, where in the ideal case of k disconnected clusters the eigenvalue zero has multiplicity k and after that there exists a gap to the $(k+1)^{th}$ eigenvalue $\lambda_{k+1} > 0$. Therefore, it is important to choose the number of clusters k such that all natural frequencies from 1 to k are very small, while the natural frequency $k+1$ is relatively large. This heuristic method works very well, when in the data set there are well defined gaps [13].

Based on the aforementioned method, it can clearly be seen from Figure 2-1 that for case2383wp the largest gap is the fourth one, therefore four islands can be created, in order for the $k+1$ natural frequency to be relatively large compared to the first $k=4$ natural frequencies. Following the same logic, in case2869pegase the largest gap is the third one, so three islands may be formed.

The goal of the coherency analysis is to produce meaningful constraints in order to evaluate the constraint satisfaction of the studied algorithms. Therefore, for the cases 1354pegase, 2383wp and 2869pegase, constraints are generated for splitting the power system from two up to ten islands. Since case118 is a relatively small electrical network, up to five islands are considered.

[1] The built in function of the PST toolbox approximated the state matrix of the electromechanical model, but was not accurate enough.

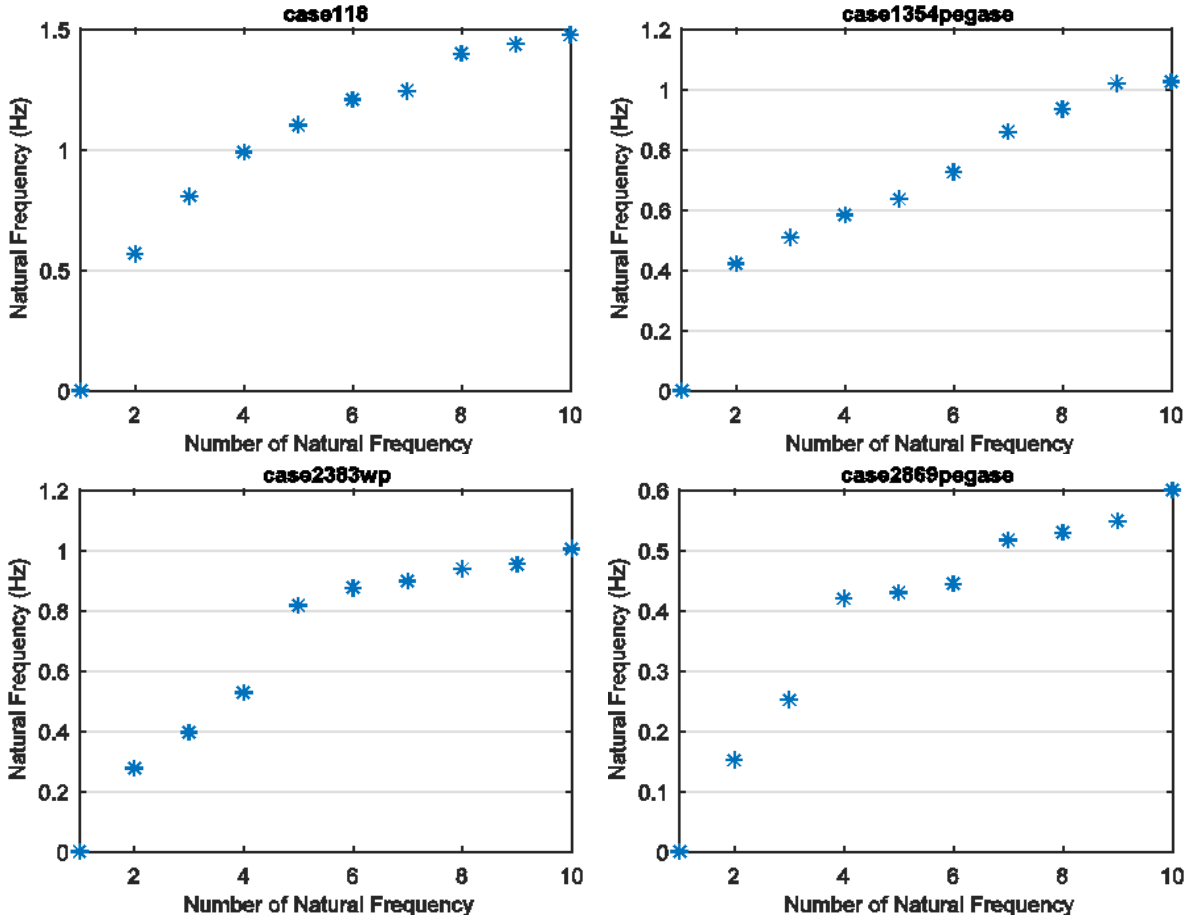


Figure 2-1: Natural frequencies of the networks studied

To sum up, a common heuristic to identify the number of slow electromechanical modes n_s, is to find the largest gap of the natural frequencies. However, in order to acquire more insight on the constraint satisfaction of the studied algorithms, the power grids considered are going to be split into multiple islands.

2.1.3 Identification of coherent generators

Assigning slow coherent generators to the same partition is crucial for maintaining the transient stability of the constructed islands. Transient stability is the ability of the electrical network to remain in synchronism after a large disturbance. Stability depends both on the initial operating state and severity of the disturbance [14]. In the case of intentional controlled islanding, assuring transient stability is more crucial than balancing load and generation. Load-generation balance can be achieved with load shedding, while an unstable island will collapse even if demand and production are balanced [15].

To identify the slow coherent generators, the first n_s eigenvectors are kept, while the first one is dropped, since it is always constant. Then, each eigenvector $\boldsymbol{v}$ is normalized to have length one, based on the equation [7]:

$$\boldsymbol{v} = \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \tag{2-10}$$

Finally, hierarchical clustering is performed on the eigenvector coordinates of the generators, which determines which generators belong in each cluster. Clustering the generator eigenvector coordinates using the k-medoids algorithm instead of hierarchical clustering was also considered, but it was dropped as it resulted in a less consistent grouping,

as explained in the following section. The whole process for identifying the coherent groups of generators of each network, described for the experiments in section 2.1, is depicted in Figure 2-2.

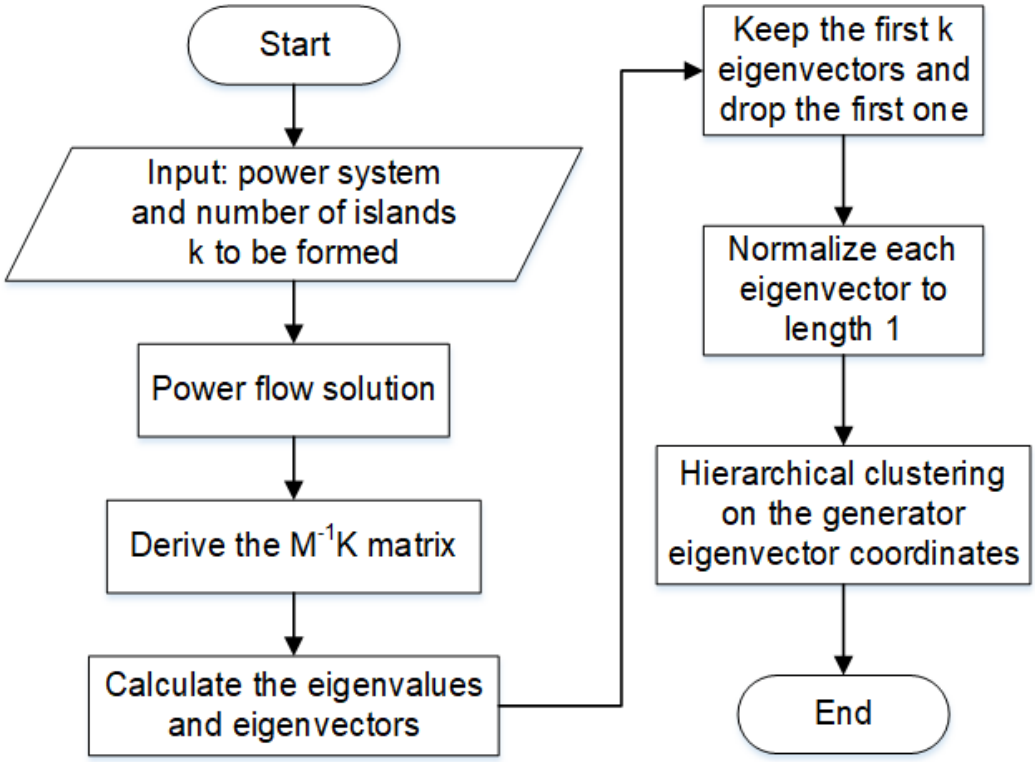


Figure 2-2: Flow chart for the identification of the coherent generators

2.1.4 IEEE 118-Bus test system coherency constraints

For better understanding of the methodology presented, the coherent generator groups identified in the IEEE 118-bus test system are visualized in this section. This electrical network corresponds to the simplified model of the American electric power system in the U.S. Midwest as of December 1962. In this system there are 19 generators, 35 synchronous condensers, 177 lines, 9 transformers, 91 loads and 118 buses [16].

When clustering the generator eigenvector coordinates, initially both hierarchical clustering and k-medoids were considered. In Figure 2-3 and Figure 2-4 the boundary between the two identified generator groups produced by k-medoids and hierarchical clustering respectively is shown with a red dotted line. Clearly, hierarchical clustering produces more consistent constraints than k-medoids, as it splits the islands in a position where a larger gap is detected in the generator eigenvector coordinates. A similar situation was observed when using these two algorithms for identifying more than two coherent generator groups. Thus, hierarchical clustering is more suitable for the production of the constraints, as it has less clustering biases than k-medoids.

When splitting a network into two islands, since the first eigenvector is dropped only one eigenvector remains. Therefore, in the plots the Y axis corresponds to the coordinates of the remaining eigenvector, while the X axis represents the generator numbers. For better visualization, the generator eigenvector coordinates are sorted and therefore their numbers are also shown above the corresponding coordinate.

These two generator groups can be easily identified, due to the existence of a rather large gap between the eigenvector coordinates of generators 37 and 44. In fact, by inspecting

Figure 2-4 more closely another possible group of generators can be identified as there is another relatively large gap between the eigenvector coordinates of generator 19 and 20.

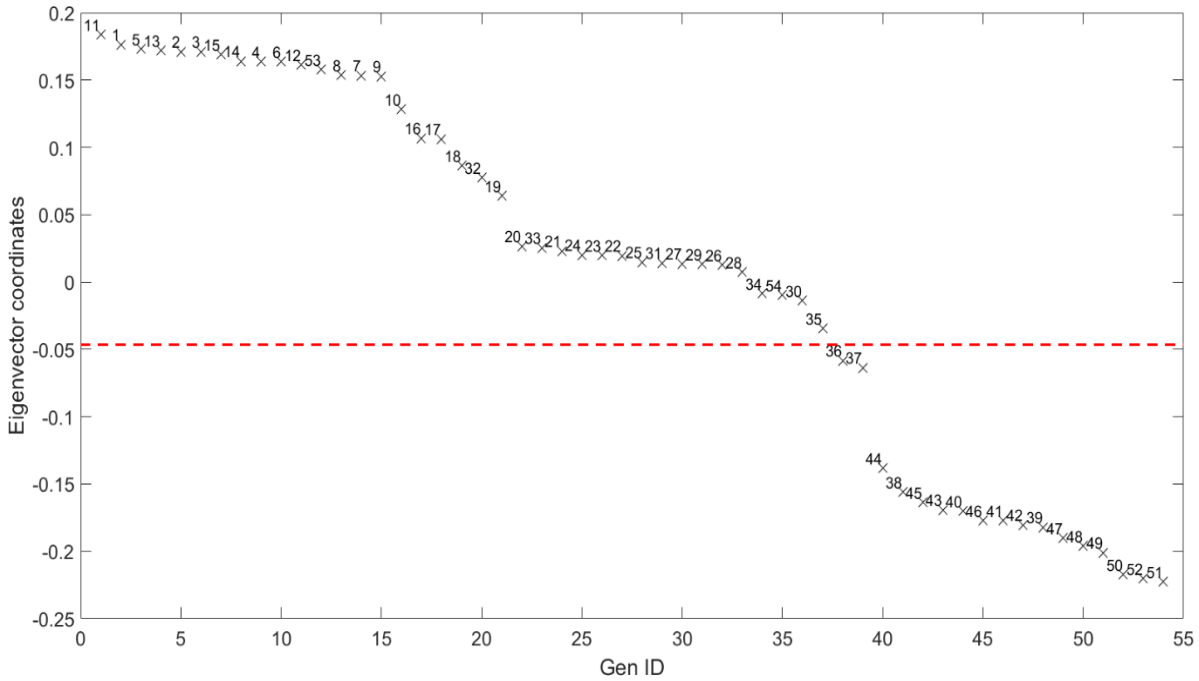


Figure 2-3: Coherent groups when splitting case118 in 2 islands with k-medoids

Since the size this electrical network is rather small, the results shown in Figure 2-4, can also be seen on the single line diagram in Figure 2-5. Splitting this power system in order to satisfy these constraints does not seem very challenging as only few lines must be disconnected, confirming the previous observation that there are indeed two distinct areas in this network.

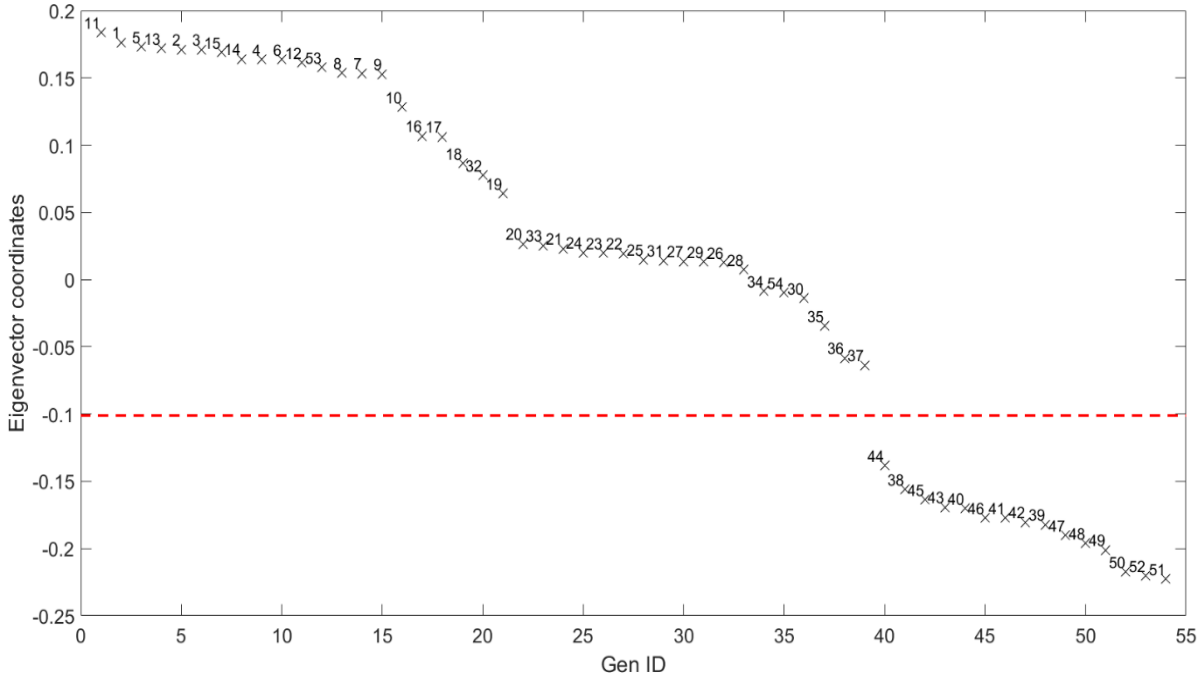


Figure 2-4: Coherent groups when splitting case118 in 2 islands with hierarchical clustering

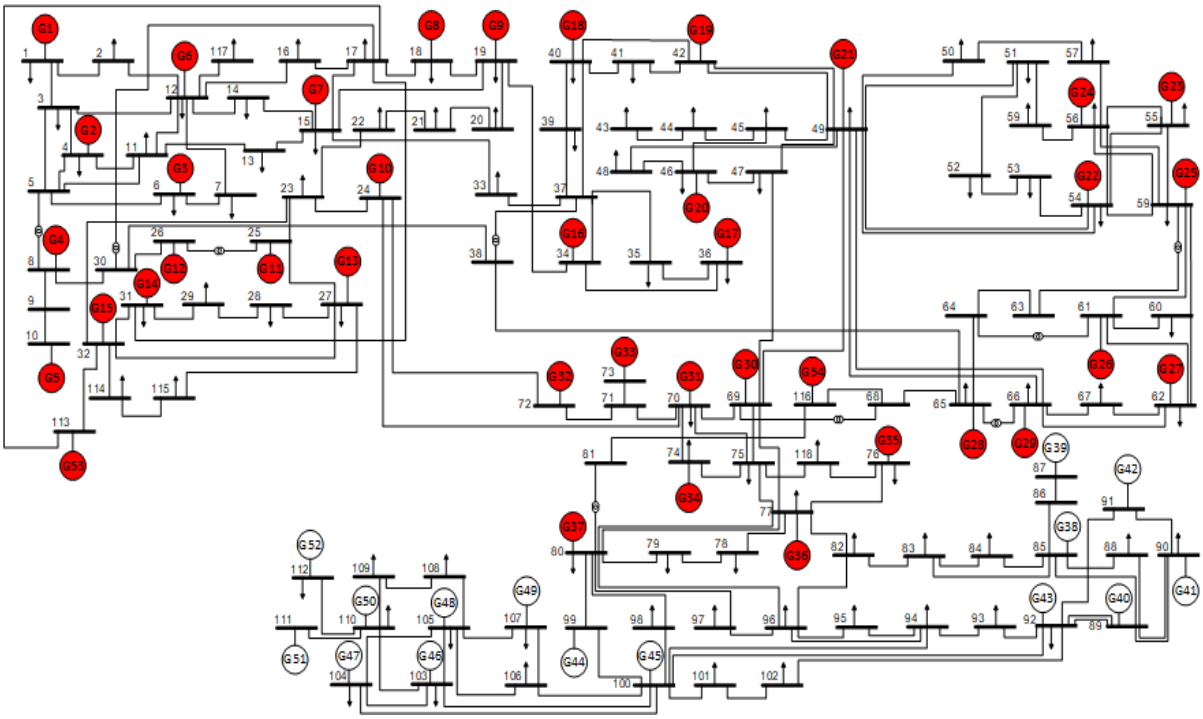


Figure 2-5: Single line diagram of case118 for 2 coherent generator groups

The eigenvector coordinates when splitting this network in three islands are shown in Figure 2-6 and the resulting generator groups are shown on the single line diagram in Figure 2-7. Both these figures confirm the previous observation made through Figure 2-4, that there is also another distinct group due to the relatively large gap between the generators 19 and 20. Through Figure 2-5 and Figure 2-7, it is clear that the generator group on the bottom of this power grid remained unaltered, while the upper group (generators coloured red) is split into two groups. Of course, the grouping has been slightly modified as now two eigenvectors are taken into account instead of just one and as a result the generators 16, 17, 19 and 32 remain in the red group in Figure 2-6, while based on Figure 2-4, they would be parts of a different group.

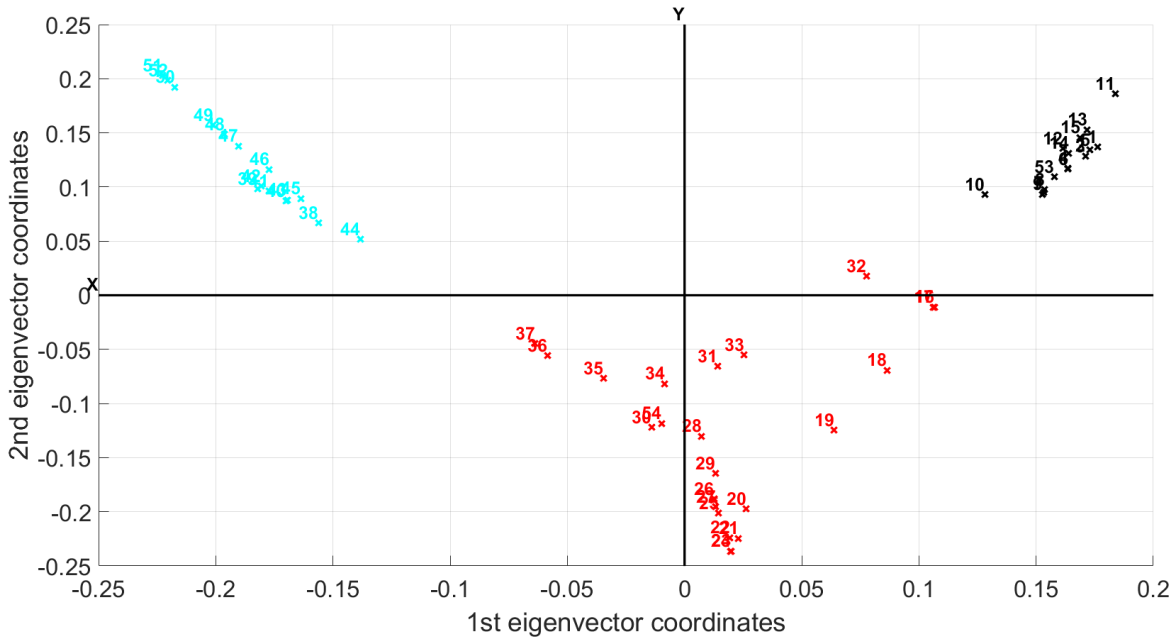


Figure 2-6: Generator eigenvector coordinates when splitting case118 in 3 islands

The coherency of the IEEE 118-bus test system is not further investigated but in the sections 2.5.1.1 and 2.5.2.1 the slow-coherency cutsets for splitting this network from two to five islands are presented.

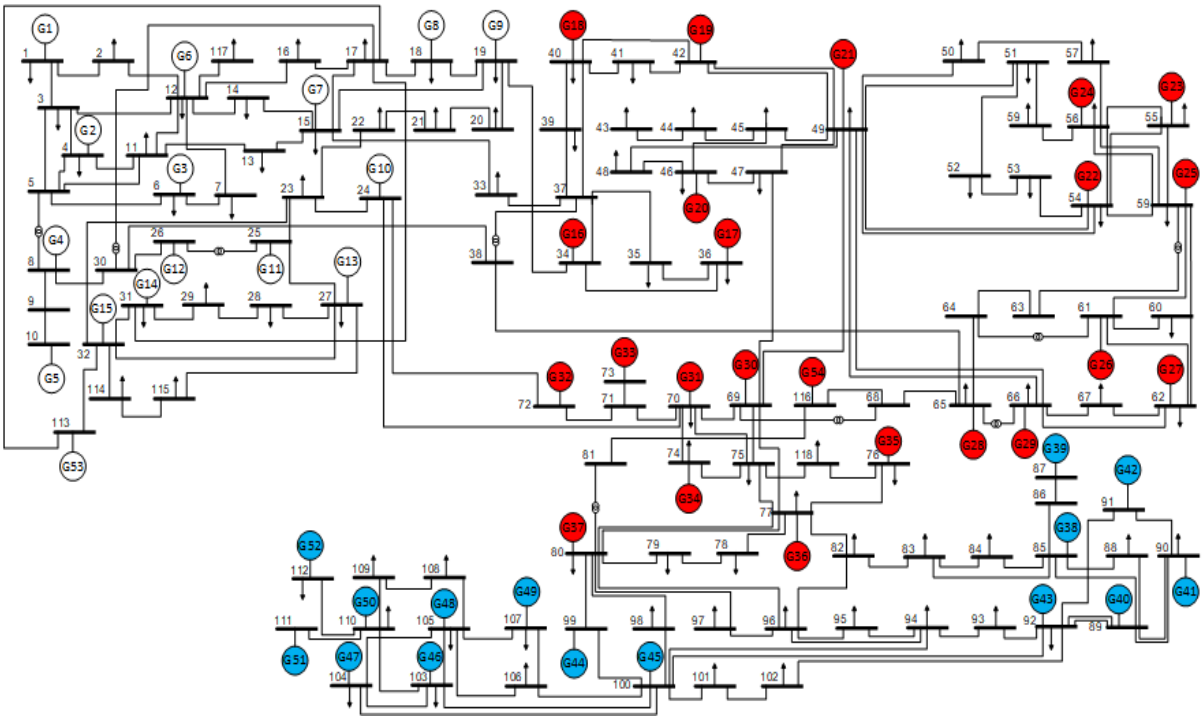


Figure 2-7: Single line diagram of case118 for 3 coherent generator groups

2.2 Simulation environment

As stated in section 1.4, Matpower and PST toolboxes are used. The original power flow solution of the studied electrical network is obtained from the Matpower toolbox. Then the data from this solution (e.g., bus voltages, line power flows, generator production) is stored in a Matlab object, called MatpowerIn. The data from MatpowerIn is then converted to graph data (e.g., adjacency, incidence, Laplacian matrices) and is stored in another Matlab object, named PFgraph. PST toolbox is utilized to acquire the necessary data to build the state matrix of the electromechanical model, as explained in section 2.1.1. Based on this matrix, the coherency constraints for the generators arise and are stored in both MatpowerIn and PFgraph. The PFgraph object constitutes the input for the partitioning methods explained in the following section. The interaction between the toolboxes, objects and partitioning methods is shown in Figure 2-8.

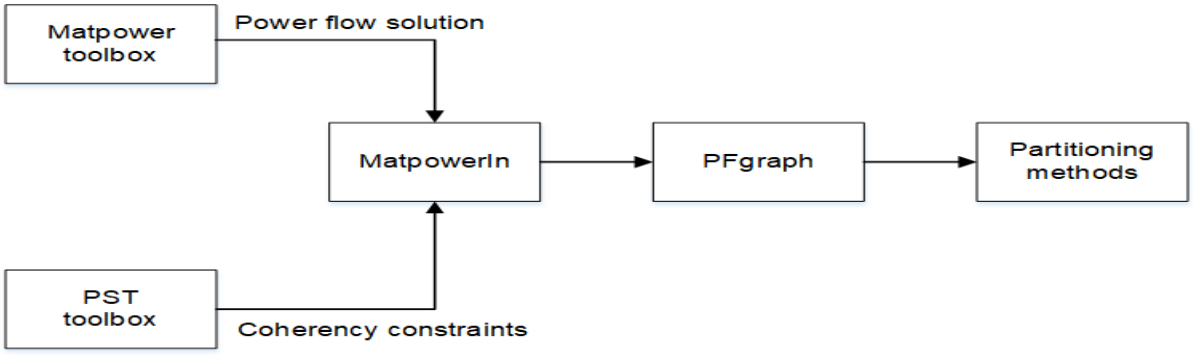


Figure 2-8: Interaction between the toolboxes, objects and partitioning methods

It is worth mentioning that the generator data available in the Matpower toolbox does not contain the dynamic data of the generators (e.g., transient reactances and inertia constants), required to build the $\mathbf{M^{-1}K}$ matrix. Therefore, a database with the dynamic data of several generators was used, which was obtained from [17]. The maximal active power output of the generator bus from the Matpower test case is compared with the nominal active power production of the generators in the generator database. Based on this comparison, the closest generator from the database is identified and its data is adopted.

2.3 Partitioning methods

Three different algorithms are assessed during this thesis. Flexible Constrained Spectral Clustering (FCSC) combines Spectral Clustering and pairwise constraints in a principled and flexible manner, employing a user specified threshold to balance the objectives of constraint satisfaction and cut minimisation [18]. hMetis is developed at the University of Minnesota and is based on multilevel hypergraph partitioning schemes such as those corresponding to very-large-scale integration (VLSI) circuits [19]. Tight Continuous Relaxation of Balanced Graph Cuts (TCRBGC) is developed at Saarland University and approaches the balanced k-cut problem using a tight continuous relaxation [20].

2.3.1 Flexible Constrained Spectral Clustering (FCSC)

2.3.1.1 Operation principle

The method used for constrained spectral clustering in the present thesis is based on a degree of belief in MUST-LINK (ML) and CANNOT-LINK (CL) constraints. Firstly, the $N\,x\,N$ constraint matrix $\mathbf{Q}$ needs to be created and can be encoded as follows [18]:

$$Q_{ij} = Q_{ji} = \begin{cases} +1 & if\ ML(i,j) \\ -1 & if\ CL(i,j) \\ 0 & no\ supervision\ available \end{cases} \tag{2-11}$$

If $Q_{ij}=+1$, then the buses i and j are considered to belong to the same group, while if $Q_{ij}=-1$ the buses i and j are considered to belong in different groups. If $Q_{ij}=0$, there is no information whether the buses i and j are considered to belong in the same or in different groups. This is the case for the load buses, as constraints are considered only for the generators.

A measure that determines how well the constraints are satisfied is defined as [18]:

$$\boldsymbol{u}^T\mathbf{Q}\boldsymbol{u} = \sum_{i=1}^{N}\sum_{j=1}^{N} u_i u_j Q_{ij} \tag{2-12}$$

with $\boldsymbol{u} \in \{-1,+1\}^N$ being the cluster indicator vector. In order to include the degree of belief in $\mathbf{Q}$, values between -1 and +1 are used, both the cluster indicator vector and the constraint matrix are relaxed such that:

$$\boldsymbol{u} \in R^N, \mathbf{Q} \in R^{NxN} \tag{2-13}$$

The magnitude of Q_{ij} indicates how strong the belief is that i and j must be in the same partition if Q_{ij} is positive or in different partitions if Q_{ij} is negative.

The constraint matrix $\mathbf{Q}$ and the Laplacian matrix $\mathbf{L}$ are normalized according to the following equations:

$$\bar{\mathbf{Q}} = \mathbf{D}^{-½}\mathbf{Q}\mathbf{D}^{-½} \tag{2-14}$$

$$\bar{\mathbf{L}} = \mathbf{L}_{\mathrm{sym}} = \mathbf{D}^{-½}\mathbf{L}\mathbf{D}^{-½} \tag{2-15}$$

where $\mathbf{D}$ is a diagonal matrix called the degree matrix with the vertex degrees $d_1,\ldots,d_N$ on its diagonal.

$$d_i = \sum_{j=1}^{N} W_{ij} \quad (2\text{-}16)$$

Also, W_{ij} are the weights of the adjacency matrix, associated with an edge connecting the vertices i and j. The adjacency matrix $\mathbf{W}$ of a graph stores the values of W_{ij}, displaying how each vertex is connected with another vertex. When two vertices are not connected $W_{ij} = 0$. If two vertices are connected, since the adjacency matrix is symmetrical, $W_{ij} = W_{ji}$.

Then, constrained spectral clustering is considered an optimization problem aiming to optimize the objective function [18]:

$$argmin\ \boldsymbol{v}^T \bar{\mathbf{L}} \boldsymbol{v}, s.t. \boldsymbol{v}^T \bar{\mathbf{Q}} \boldsymbol{v} \geq \mathrm{a}, \boldsymbol{v}^T \boldsymbol{v} = vol(G), v \neq \mathbf{D}^{1/2} 1. \quad (2\text{-}17)$$

with $vol(G)$ the volume of the graph G, defined as:

$$vol(G) = \sum_{i \in G} d_i \quad (2\text{-}18)$$

Here $\boldsymbol{v}^T \bar{\mathbf{L}} \boldsymbol{v}$ is the cost of the cut needed to be minimized, a is the constraint satisfaction threshold and $\boldsymbol{v}^T \bar{\mathbf{Q}} \boldsymbol{v} \geq \mathrm{a}$ measures how well the constraints in $\mathbf{Q}$ are satisfied, $\boldsymbol{v}^T \boldsymbol{v} = vol(G)$ is used to normalize v while the final constraint $\boldsymbol{v} \neq \mathbf{D}^{1/2} 1$ excludes the trivial solutions.

The constrained optimization problem in (2-17) can be solved using the Karush-Kun-Tucker theorem [21]. After several mathematical steps, explained in [18], the solution of the equation (2-17) is acquired by solving the generalized eigenvalue problem:

$$\bar{\mathbf{L}} \boldsymbol{v} = \lambda (\bar{\mathbf{Q}} - \frac{b}{vol(G)} \mathbf{I}) \boldsymbol{v} \quad (2\text{-}19)$$

with $\mathbf{I}$ being the identity matrix.

In [22] the problem of constrained spectral clustering, described in [18], is extended from 2-way partitioning into k-way partitioning. In that paper, the maximum value of the constraint threshold b, that guarantees at least $k\text{-}1$ feasible eigenvectors, is computed as:

$$b < \lambda_{k-1} vol(G) \quad (2\text{-}20)$$

After obtaining the eigenvectors $\boldsymbol{v}$ from solving (2-19), the ones related with non-positive eigenvalues are removed and the rest are normalized. Then, the cluster indicator vectors $\boldsymbol{u}$ are obtained by sorting $\boldsymbol{v}$ in order to minimize the cut and by removing the trivial solutions in order to satisfy the constraints of (2-17).

2.3.1.2 Implementation

In algorithm 2 proposed in [22], the value of b is treated as an input, while the equation (2-20) gives an upper limit for its value. However, in the Matlab implementation of this algorithm, written by the author of [18] and [22] available in [23], the value b of is computed as follows:

$$b = \frac{\lambda_{k+1} + \lambda_k}{2} - 10^{-6} \quad (2\text{-}21)$$

This value of b calculated through (2-21) is in most cases significantly smaller than the maximum one computed through (2-20). Therefore, in the approach followed in the present thesis the value found through (2-21) is increased and different values of b are tested to find the one that maximizes the coherency constraints satisfaction. The value of b is increased with a step:

$$step = \frac{\mathrm{b}_{max} - \mathrm{b}_{mim}}{10} \quad (2\text{-}22)$$

where b_{max} is the value obtained through (2-20) and b_{min} is the value obtained through (2-21). Then, 10 values of b are used starting from b_{min}, increasing with step and reaching up to the value b_{max}–$step$. If there are many values of b that minimize the number of coherency constraint

violations, the one with the smallest worst expansion is picked. These performance metrics are explained in section 2.4.

After obtaining the cluster indicator vectors $\boldsymbol{u}$, both k-medoids and hierarchical clustering were considered to obtain the slow-coherency cutset. k-medoids is a centroid based clustering algorithm that finds the centroid and then assigns all points based on their distance from the closest centroid. As a result, it will often result in non-compact groups as is the case in Figure 2-3 and lead to many constraint violations. Hierarchical clustering also led to poor constraint satisfaction. Very often it would start merging load buses with other load buses, forming a cluster that increased in size. Then, such a cluster merged with generator buses of various coherent groups and as a result many generators were not being assigned based on their coherency.

Therefore, an algorithm has been written to produce the slow-coherency cutset of FCSC, after acquiring $\boldsymbol{u}$. In this algorithm a label is assigned to each node. Generator nodes get the coherent group number, while load nodes get zero. In the beginning, the algorithm tries to merge generator nodes with other generator nodes, based on the coherency and the physical connections of the network. If many such merges are possible, the one with the smallest average distance between the two generator nodes is chosen. If no valid coherent merges are present, the load node with the smallest average distance from a generator node is merged with that generator node. In this case the load node gets the same label as the generator node it is merged with. This happens in every iteration until all nodes of the network receive a label that corresponds to the island they belong. The developed FCSC-based clustering method is summarized in **Algorithm 1**.

Algorithm 1: Label-constrained graph clustering

Input: Adjacency matrix $\mathbf{W}$, Constraint matrix $\mathbf{Q}$, Number of islands k
Output: Labels corresponding to the island each node belongs to

```
For each value of b
    Compute eigenvectors v as (2-19)
    From v get the cluster indicator vectors u that satisfy (2-17)
    Build graph on V_nXk-1 as in hierarchical clustering
    Compute distance matrix from W and V
    Create labels based on coherency assignment
    While any label is equal to 0
        If merging existing clusters based on coherency and connections is possible
            Merge clusters
        Else
            Expand cluster by adding the load node with the minimum distance
            The load node added to the cluster gets the label of the cluster
        End
    End
End
```

2.3.2 hMetis

2.3.2.1 Operation principle

hMetis is an extension of Metis [24] for partitioning hypergraphs. Metis is another high quality graph partitioning algorithm developed at the University of Minnesota. A hypergraph is a generalisation of a graph. In this case the set of edges is replaced by a set of hyperedges. An edge can connect only two vertices. On the other hand, a hyperedge allows the connection

of more than two edges. Since power networks are graphs and not hypergraphs, when using hMetis all hyperedges will connect only two vertices, transforming them into normal edges. As a result hMetis will create partitions for a regular graph, instead of a hypergraph.

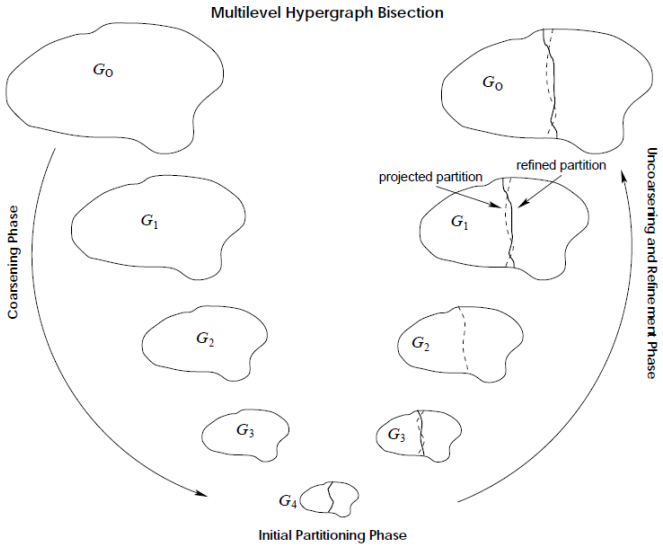


Figure 2-9: The 3 phases used by hMetis to perform graph partitioning [19]

hMetis applies three phases, depicted in Figure 2-9, to perform graph partitioning: graph coarsening, initial partitioning and uncoarsening. During the first phase, the input graph is successively transformed into a sequence of smaller graphs. This takes place until the vertices of the graph have been reduced to a few hundreds. In the initial partitioning phase, a k-way partition is calculated from the coarsest graph, created during the previous phase. Lastly, in the third phase, the clustering of the coarsest graph is projected to the successively larger graphs until the original graph is reached [19].

2.3.2.2 Implementation

The latest stable release of hMetis (1.5.3) is only available for 32bit architectures. To tackle this issue, the latest experimental release (2.0pre1) is used, which is available for 64bit architectures and can be found in [25]. Unfortunately, because this version of hMetis is still under development, there is currently no manual available. Since, the functionalities of this version do not differ significantly from the functionalities of the stable one, the manual of version 1.5.3 was used for guidance. More information on the implementation of hMetis are provided in Appendix A.

Apart from using the command –fixed (Appendix A), in order to perform constrained clustering, another approach was tested, but it was dropped as its results were not as good as those of the command –fixed. Namely, the identified coherent generators of each group formed a hyperedge and were inputted to the hypergraph file of hMetis. As weight on those hyperedges, the sum of all the weights in the adjacency matrix was used. Alternatively, the sum of the weights of the edges connected to the generators of each coherent generator group was tested. When computing the partitioning using the k-way algorithm the performance was poor. When using the recursive bisection algorithm, the performance was slightly worse than that of the command –fixed. In the latter case, the constraints for the hyperedges were usually well satisfied, but when a singleton coherent generator group was identified, it was not possible to create a hyperedge with that 1 generator. Therefore, the implementation of the hyperedges was outperformed by the command –fixed. Finally, both the hyperedges were

inputted to the input file of hMetis, along with the command –fixed and it was found out that the constraint satisfaction was similar to using the command –fixed alone.

2.3.3 Tight Continuous Relaxation of Balanced Graph Cuts (TCRBGC)

2.3.3.1 Operation principle

The balanced k-cut problem can be defined as [20]:

$$\min_{(C_1,\dots,C_k)\epsilon P_k} \sum_{i=1}^{k} \frac{cut(C_i, \bar{C}_i)}{\hat{S}(C_i)} =: BCut(C_1, \dots, C_k) \tag{2-23}$$

with P_k being the set of all k partitions of the vertex set $\mathcal{V}$ and $\hat{S}$ a balancing function with the goal that all sets C_i are not too small. In the case of the normalized cut problem, the goal is to prevent clusters with very small volume:

$$\hat{S}(C) = vol(C) \tag{2-24}$$

While, in the case of the ratio cut problem, the aim is prevent the formation of clusters with very small cardinality:

$$\hat{S}(C) = |C| \tag{2-25}$$

The proposed relaxation of the balanced k-cut problem is rather complicated and therefore it is not presented here, but it is thoroughly explained in [20].

2.3.3.2 Implementation

The main options offered by the implementation of this algorithm in Matlab, available in [26], have to do with the balancing function. In the case of power networks the normalized cut and the ratio cut problem are the most meaningful and relevant ones. However, all the alternatives offered, were explored and it was found out that the ratio cut problem satisfied the constraints better than the other options. Furthermore, a script has been written to change the constraints into a suitable input for this algorithm.

2.4 Performance metrics

The most important quality indicator during constrained graph clustering is the constraint satisfaction. In this thesis the number of generators that are not assigned to the requested island is used to measure how well the constraints are satisfied by each algorithm. Thus, the number of coherency constraint violations is the indicator mostly focused on, that needs to be as low as possible.

Other important performance metrics to take into consideration, especially when the constraint satisfaction is similar, are the total power cut, the overall (maximal) expansion and the normalized cut.

The edge weights of the adjacency matrix $\mathbf{W}$ are the active power flows in their respective branches. The total power cut is the sum of the power cut (in MW), after forming the clusters. In other words, it is the power flow disruption because of the disconnection of certain electrical connections and for k partitions, it can be defined as [13]:

$$P_{cut} = cut(C_1, \dots, C_k) = \frac{1}{2}\sum_{i=1}^{k} \mathbf{W}(C_i, \bar{C}_i) \tag{2-26}$$

with C being a subset of the graph G and $\bar{C}$ its complement.

In addition to this, every created subset C has a volume and a boundary. The volume is defined as in (2-18), while the boundary is the sum of the weights of the edges between the vertices in C and the vertices not in C [7]:

$$\partial(C) = \sum_{i \in C, j \notin C} W_{ij} \tag{2-27}$$

The expansion of C is the ratio between its boundary and volume:

$$\varphi(C) = \frac{\partial(C)}{vol(C)} \tag{2-28}$$

A small expansion means that the vertices inside the island are strongly interconnected among themselves (large $vol(C)$) and weakly connected to the rest of the network (small $\partial(C)$). Thus, the overall expansion is the worst (maximal) expansion of the partitions created.

Moreover, for k clusters, the normalized cut (2-29) measures the overall cut of separating the graph G into k islands [13]:

$$N_{cut}(C_1, \dots, C_k) = \frac{1}{k} \sum_{i=1}^{k} \frac{cut(C_i, \bar{C}_i)}{vol(C_i)} \tag{2-29}$$

Minimizing N_{cut} constitutes the objective function of the normalized spectral clustering.

2.5 Slow-coherency cutset

The algorithms from section 2.3 are going to be evaluated, based on the coherency constraints identified for each electrical network. Two different study cases are carried out. First, in order to place more emphasis on the constraint satisfaction, instead of using the actual adjacency matrix of the power system graph as the input to these algorithms, a binary adjacency matrix is constructed and used. Next the original weighted adjacency matrix of the networks is utilized, so that the active power flows between the buses of each power system are also taken into account.

2.5.1 Binary adjacency matrix

The aim of this study case is to increase the constraint satisfaction as much as possible by utilizing the binary adjacency matrix of the studied electrical networks. The total power cut, the maximal expansion and the normalized cut are all performance metrics associated with the weights of the adjacency matrix. Therefore, these three quality indicators are not relevant for this study case.

2.5.1.1 IEEE 118-Bus test system

In this case all the algorithms achieved perfect constraint satisfaction for the different number of islands requested. It is also worth noticing that when two islands are requested all the algorithms produce the exact same slow-coherency cutset, depicted in Figure 2-10. A total of five lines had to be disconnected. The red dotted line represents the boundary between the islands.

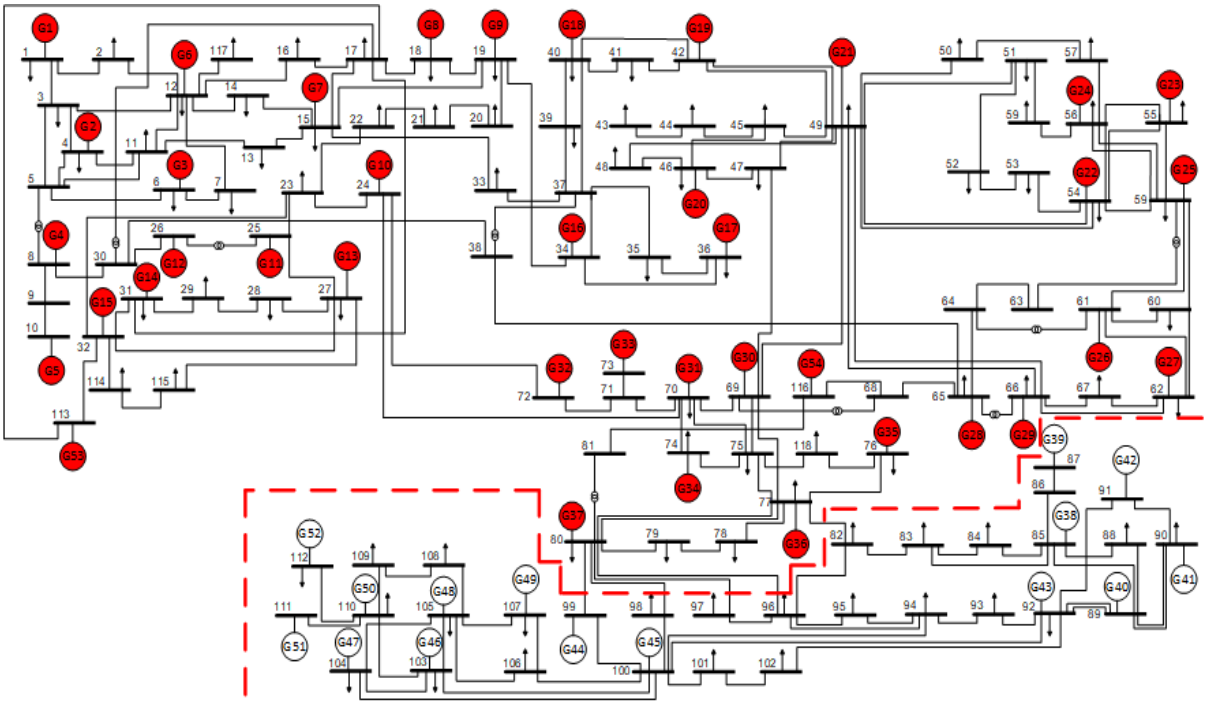


Figure 2-10: Slow-coherency cutset computed for splitting case118 into 2 islands

2.5.1.2 Large power systems

Since the size of the networks considered in this section is relatively large, ranging from 1354 buses to almost 3000 buses, more general conclusions can be drawn about the constraint satisfaction of the evaluated algorithms. Furthermore, for these cases splitting the network in up to 10 islands is considered. Table 2-1 shows the number of buses, branches and generators in the three large power systems considered.

Power system	Number of buses	Number of branches	Number of generators
Case1354pegase	1354	1710	260
Case2383wp	2383	2886	327
Case2869pegase	2869	3968	510

Table 2-1: Number of buses, branches and generators in the large power systems

From Figure 2-11, it is clear that the developed FCSC-based clustering method satisfies the constraints better than the other two algorithms in most cases. It is only slightly outperformed by hMetis in case2383wp for 9 and 10 islands, having one more constraint violation. Only in case2869pegase for 10 islands its results are not good compared to those of hMetis and TCRBGC.

hMetis follows the developed FCSC-based clustering method closely, usually failing to assign correctly only few more generators. TCRBGC has the most constraint violations out of the three algorithms. Still, this number is usually quite low.

Case1354pegase provides also some useful insight on the way the tested algorithms operate. When 5, 6, 7, 8 and 10 islands are requested, the coherency analysis identifies some generators as singleton coherent generator groups and puts them alone in a separate group. hMetis manages to satisfy the constraints in these cases by creating singleton islands, islands consisting of one generator alone and no other buses. On the other hand, TCRBGC tends to create less islands than requested in order to avoid creating islands with very small cardinality. For example, when five islands are requested, the coherency analysis assigns the generators

6, 32, 185 and 219 in one island and the generator 51 in a separate island. TCRBGC creates three islands instead of five and assigns these five generators in the biggest island. On the other hand, the developed FCSC-based clustering method creates a singleton island only when 10 islands are requested. When 5, 6, 7 and 8 islands are requested 10 buses are added to the cluster of the singleton coherent generator group, thus forming an island of 11 buses.

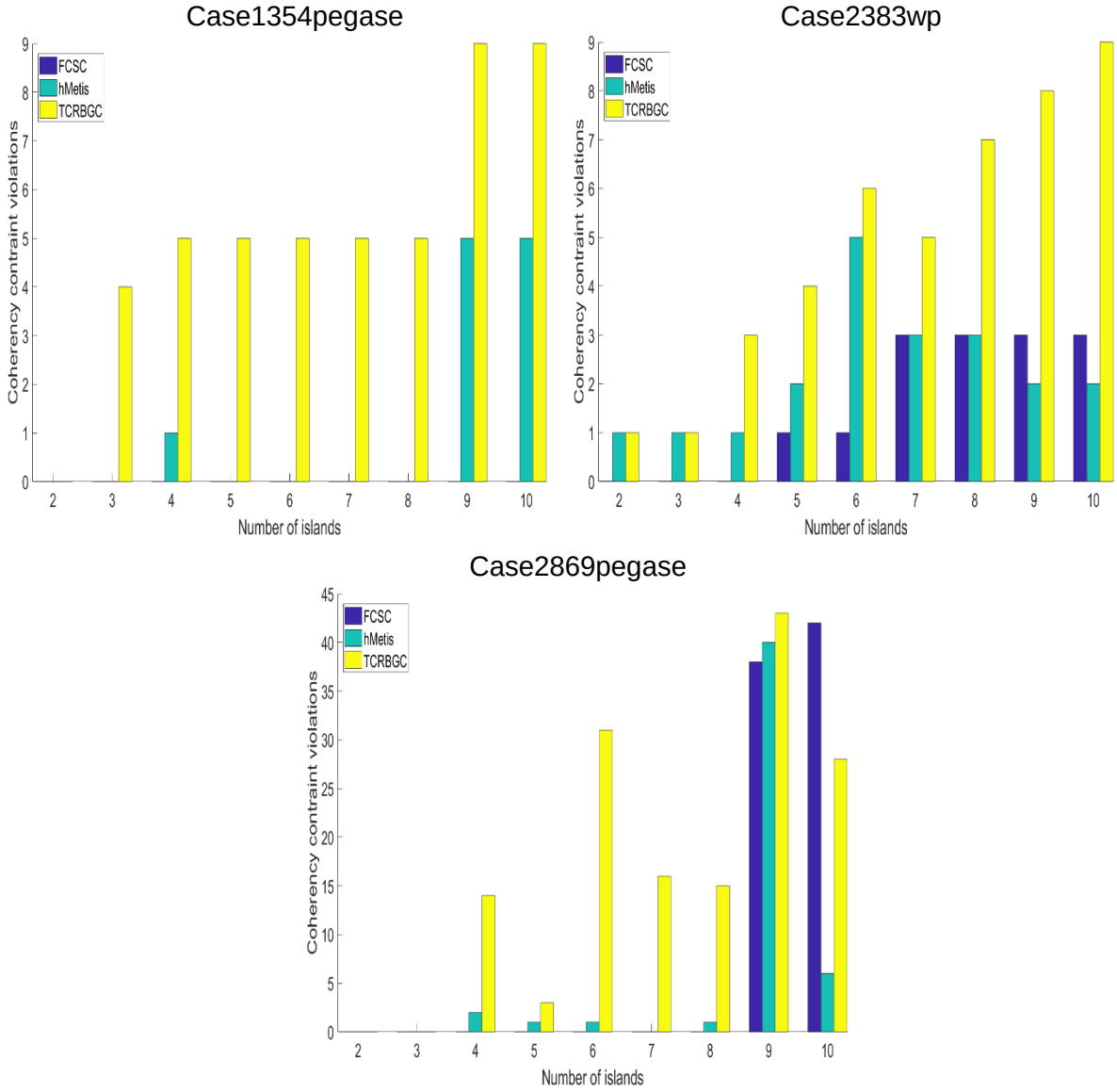


Figure 2-11: Number of coherency constraint violations in large power systems

Finally, it can be seen that in case2869pegase when 9 islands are requested all three algorithms fail to assign correctly about 40 generators. This particular case is worth further investigation as in all the other cases at least one of the three algorithms has a very small number of constraint violations.

2.5.2 Weighted adjacency matrix

In this study case, the weighted adjacency matrix of the studied power systems' active power flow graphs is used to obtain insight on the constraint satisfaction when the actual power flow data is considered.

2.5.2.1 IEEE 118-Bus test system

Again all the algorithms managed to satisfy the constraints for the different numbers of islands requested. Therefore Figure 2-12 shows the rest of the quality indicators for case118. It is observed that the developed FCSC-based clustering method tends to cut slightly more power than hMetis and TCRBGC. Furthermore, it can be seen that all the performance metrics of hMetis and TCRBGC are similar.

When 4 islands are requested, generator 39 on bus 87 is determined as a singleton coherent generator group during the coherency analysis. In order to satisfy this constraint both hMetis and TCRBGC, create a singleton island consisting of that generator and that is why they have a maximal expansion equal to one and such a high value of normalized cut in that particular case. On the other hand, the developed FCSC-based clustering method creates a cluster consisting of generator 39 and bus 86 because bus 86 is found to have very small distance from bus 87.

It is also worth noticing that when two islands are requested the slow-coherency cutset of the developed FCSC-based clustering method, shown in Figure 2-10, was unaffected by adding the actual power flow data. On the other hand, the slow-coherency cutsets of hMetis and TCRBGC have been slightly modified. These two algorithms produced the same slow-coherency cutset.

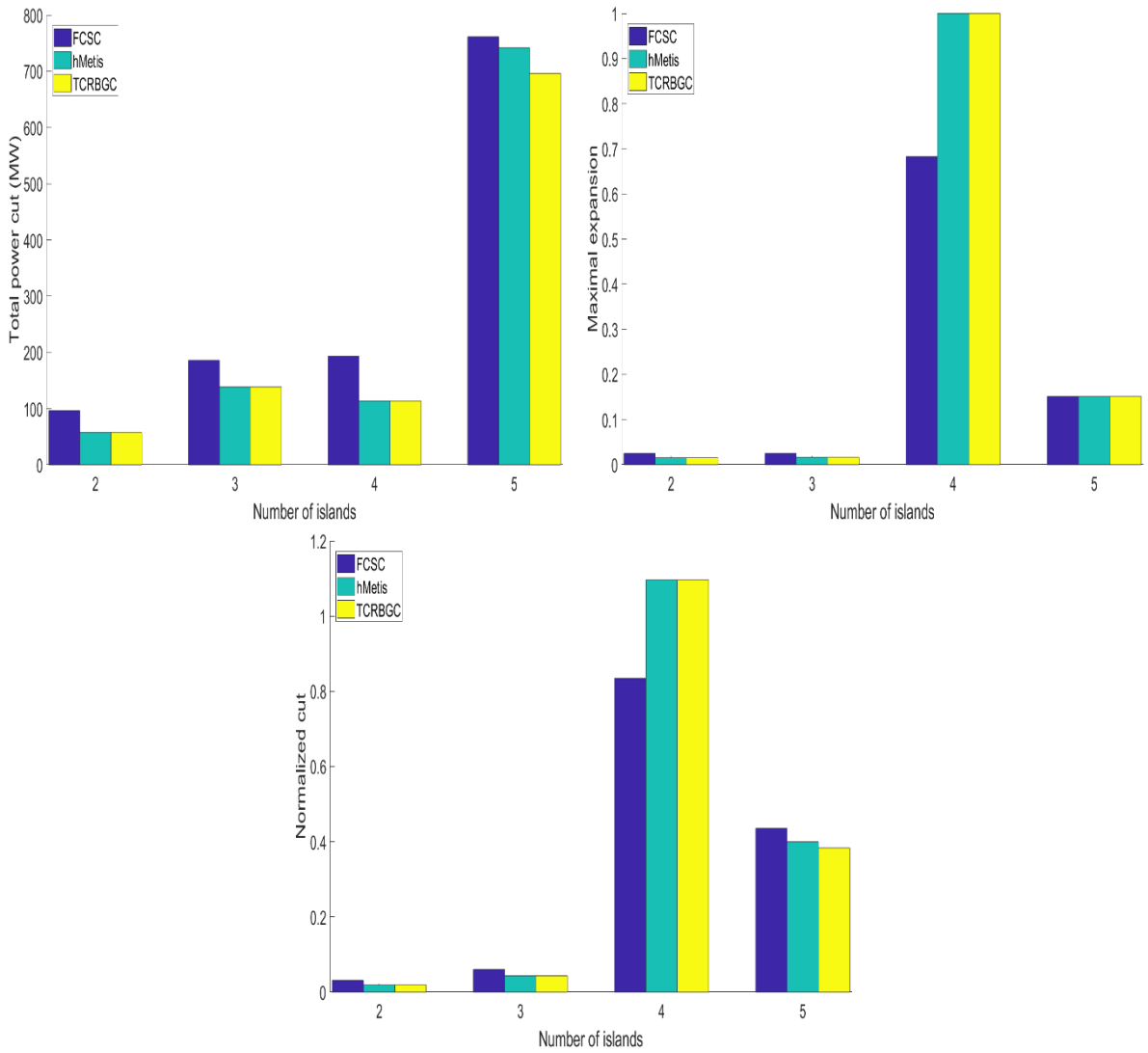


Figure 2-12: Partitioning quality metrics of all algorithms for case118

In Figure 2-13, the coordinates of the cluster indicator vector of each bus computed by the developed FCSC-based clustering method are presented, for two islands created in the case118 Matpower network. By comparing Figure 2-10 with Figure 2-13 it is clear, that the same buses can be detected close to the boundary of the islands. hMetis is a closed source software and TCRBGC is based on a sophisticated relaxation of the balanced k-cut problem, both going straight into their islanding solutions. Therefore, for these two algorithms it was not possible to create similar figures for comparison.

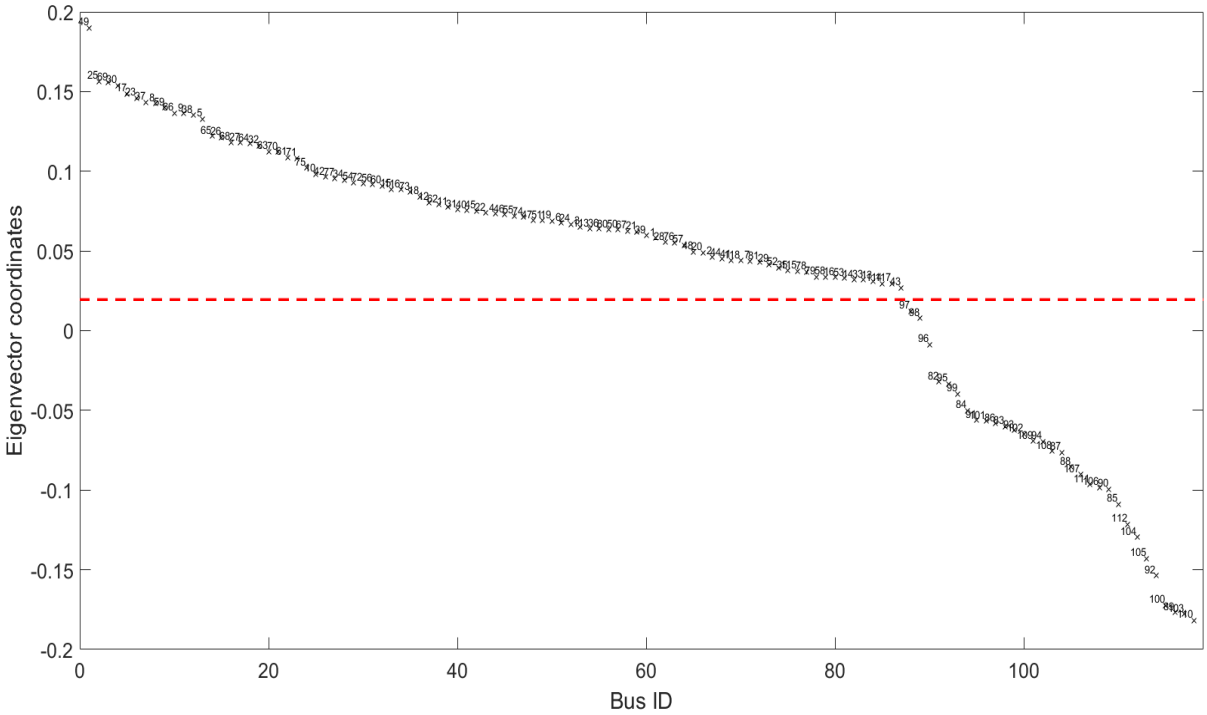


Figure 2-13: Bus eigenvectors computed by FCSC for splitting case118 into 2 islands

2.5.2.2 Large power systems

From Figure 2-14, it is observed that in case1354pegase all the algorithms satisfy the constraints when up to 8 islands are requested. For 9 islands the developed FCSC-based clustering method has no constraint violations, but for 10 islands it is outperformed by hMetis and TCRBGC.

It is worth noticing that by adding the actual power flow data the constraint satisfaction of TCRBGC has actually improved. By comparing these results with the ones from the binary adjacency matrix it can be seen that the bias to very small islands has been reduced. Now both hMetis and TCRBGC create islands with least number of nodes equal to 11, when the requested number of islands ranges from 3 to 8, just like the developed FCSC-based clustering method did in section 2.5.1.2.

The rest of the quality indicators of hMetis and TCRBGC are very close. In Figure 2-14 it is also clear that the developed FCSC-based clustering method tends to cut more power than hMetis and TCRBGC. This occurs possibly because in the function that produces the slow-coherency cutset of the developed FCSC-based clustering method, more emphasis was given into constraint satisfaction as explained in section 2.3.1.2.

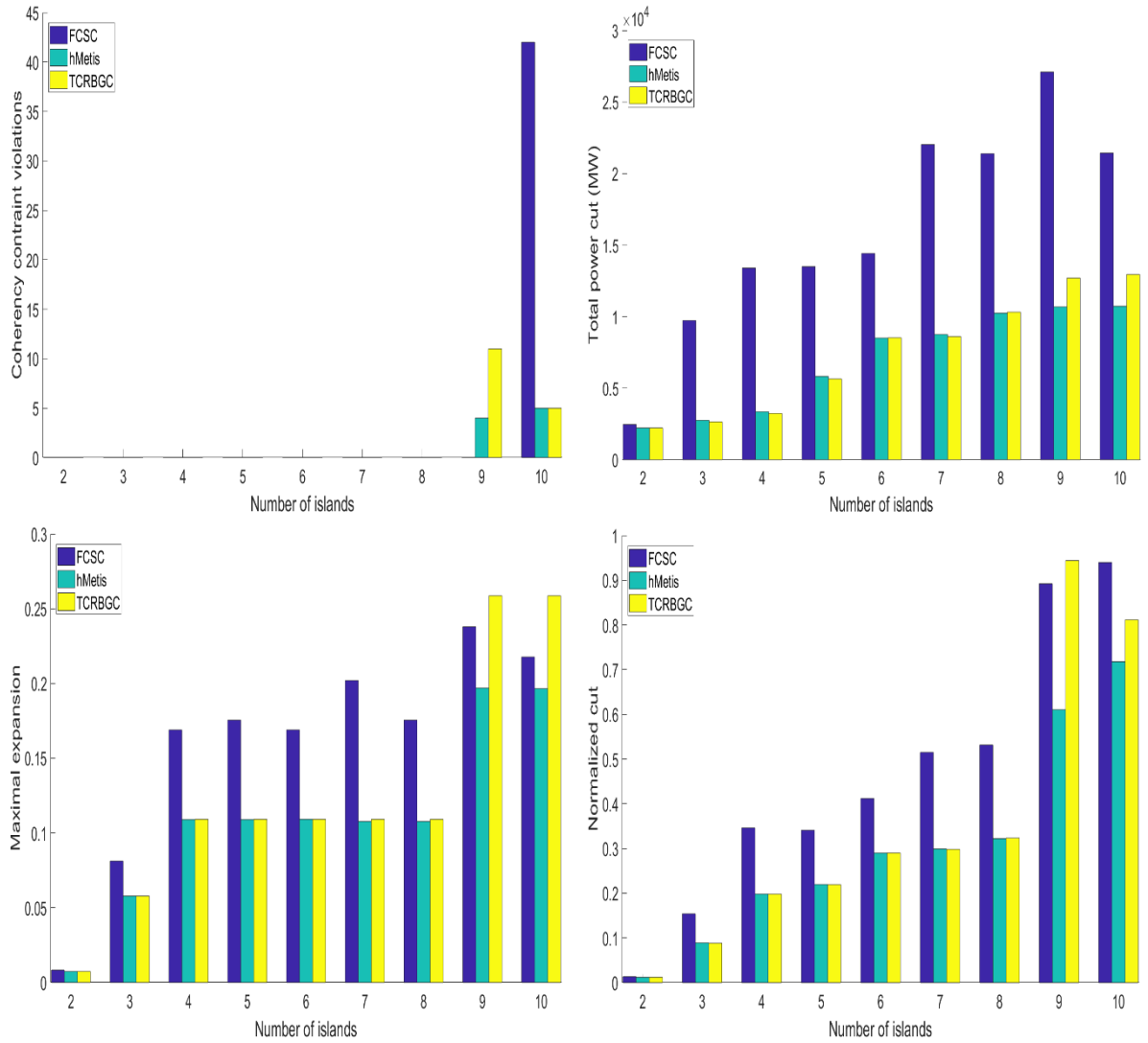


Figure 2-14: Partitioning quality metrics of all algorithms for case1354pegase

In case2383wp, the developed FCSC-based clustering method clearly achieves the best constraint satisfaction out of the three algorithms as seen in Figure 2-15. hMetis and TCRBGC have the same number of constraint violations in most cases. Sometimes, TCRBGC fails to assign correctly one more generator than hMetis. It can also be observed that as the number of the requested islands increases, the constraint satisfaction of all the algorithms tested deteriorates.

The rest of the quality indicators of hMetis and TCRBGC are once more very close. The developed FCSC-based clustering method cuts significantly more power than the other two algorithms possibly because it was designed with a strong bias towards satisfaction of bus grouping constraints. Since the purpose of this thesis is to minimize the number of coherency constraint violations rather than the total amount of power cut, the developed FCSC-based clustering method constitutes the best alternative for case2383wp.

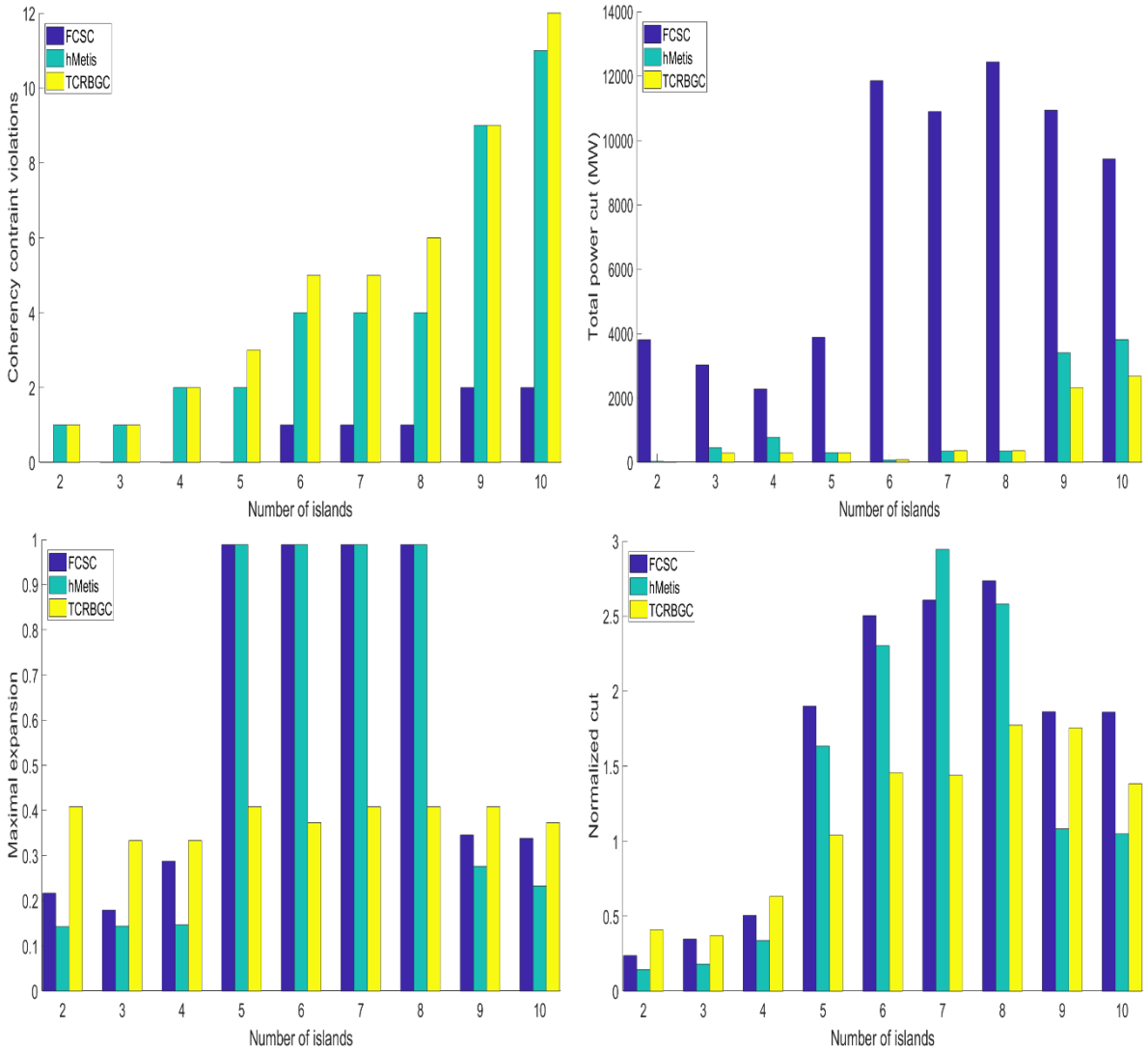


Figure 2-15: Partitioning quality metrics of all algorithms for case2383wp

In case2869pegase, the developed FCSC-based clustering method has again the smallest number of constraint violations when up to 9 islands are requested, as seen in Figure 2-16. When 10 islands are created it is outperformed by both hMetis and TCRBGC, just like in the case of the binary adjacency matrix.

Figure 2-16 also agrees with the observation that the developed FCSC-based clustering method cuts the most power out of the three algorithms. Once more, all the performance indicators of hMetis and TCRBGC are very close, except for the case of 8 islands. In that case the maximal expansion of hMetis is equal to one, meaning that a singleton island is created. As explained earlier TCRBGC is more biased towards equal cardinality and usually avoids creating very small islands.

As in section 2.5.1.2, it is noted that when 9 islands are formed in case2869pegase, all the algorithms have a rather high number of constraint violations. Thus, this case needs further investigation.

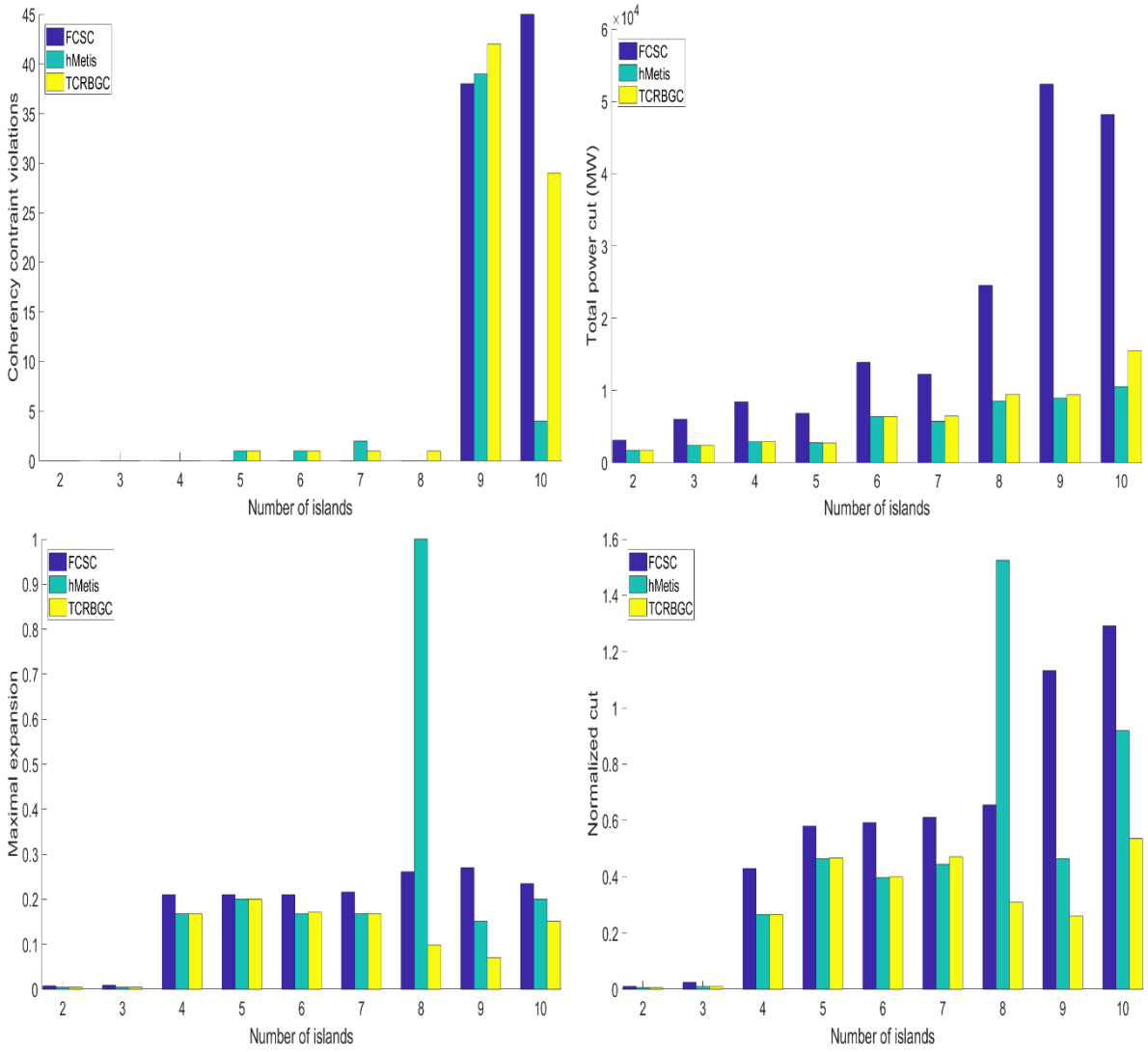


Figure 2-16: Partitioning quality metrics of all algorithms for case2869pegase

2.6 Conclusions

This chapter dealt with the problem of constrained graph partitioning. At first, enhancing the transient stability of the constructed islands was addressed by producing constraints, based on the coherent groups of generators. For illustration purposes, the results of the performed coherency analysis were also visualised in the IEEE 118-bus test system. Next, three different algorithms performing constrained graph clustering are presented along with the way they are implemented in Matlab. Moreover, the quality indicators used for the comparison of these three algorithms were discussed. Finally, the results of the algorithms for two different study cases were presented.

From the two study cases carried out in section 2.5, it is rather challenging to say that one of the evaluated algorithms is the best. All the algorithms have in most cases a quite low number of constraint violations. The developed FCSC-based clustering method tends to outperform the other two algorithms in terms of constraint satisfaction, especially if up to 9 islands are considered. hMetis also constitutes a very good alternative as its constraint satisfaction is slightly worse than that of the developed FCSC-based clustering method. TCRBGC has the most constraint violations out of the three algorithms. Still, the constraint satisfaction of TCRBGC is quite good.

The major difference in the performance of these algorithms can be noticed when the coherency analysis identifies very small groups of generators or singleton coherent generator groups. In these cases, hMetis puts the generators into the requested islands and sometimes creates singleton islands consisting only of that generator. On the other hand, TCRBGC, as explained in section 2.5.1.2, sometimes fails to create the requested number of islands, as it is more biased towards equal cardinality among the islands. In these cases, TCRBGC assigns the singleton coherent groups in a bigger island and therefore creates less islands.

Another observation is that the performance of TCRBGC actually improved when using the actual power flow data. This improvement was noticed when implementing TCRBGC with the ratio cut problem instead of the normalized cut problem. This is probably due to the fact that power flows outside of coherent areas are smaller than inside the areas.

It is also worth mentioning once more that both k-medoids and hierarchical clustering were found to be inadequate for the final step of FCSC. Therefore, another algorithm had to be designed that takes into account the constraints along with the connectivity of the network and the minimum distances between the buses.

Originally, a third study case was also considered. Based on the method of power flow tracing [27] additional constraints arise for the load buses that are supplied mainly by a generator belonging in a specified island. The purpose of this study case was to check whether introducing power flow constraints can improve the power balance in the constructed islands and reduce the amount of power cut and to study the influence of the newly added constraints on the old ones. The results of this study case are not presented, since there were no considerable differences from the results shown in section 2.5.2. The main conclusions drawn were that increasing the number of constraints by adding constraints based on power flow tracing, made the constraint satisfaction slightly worse but slightly improved the total power cut.

Chapter 3 Load shedding

As stated in the introduction, achieving power balance at the instant when the islands are formed is rather challenging. To tackle this issue, post-separation control actions are employed. If the active power produced is lower than the demand, then the frequency of the island starts dropping. To prevent it from declining below the acceptable limits, a percentage of the load of that island needs to be curtailed. The amount of load to be shed must be reliably detected as quickly as possible. Furthermore, this amount has to be minimized, with the frequency remaining above the minimum critical value at all times.

3.1 Under frequency operation

The swing equation of a synchronous machine can be expressed as [28]:

$$\frac{2H}{\omega_s}\frac{d\omega}{dt} = P_m - P_e \tag{3-1}$$

where ω_s is the synchronous rotational speed, ω is the rotational speed, P_m is the input mechanical power and P_e is the output electrical power of the synchronous machine. Through (3-1) it is clear that when the system works with less production (mechanical power) than the demand (electrical power), the electrical frequency drops.

Although all the components of the power system are designed to operate at nominal frequency, they can withstand only small variations. Particularly for costly units such as synchronous generators the magnitude and the duration of the frequency deviation is strictly limited. Table 3-1 shows the maximum time a synchronous machine can operate when subjected to various frequency deviations at full load [3].

Frequency deviation at full load	Maximum operating time
$\Delta f=1\%f_n$	Continuously
$\Delta f=2\%f_n$	100 minutes
$\Delta f=3\%f_n$	10 minutes
$\Delta f=4\%f_n$	1 minute
$\Delta f=5\%f_n$	0.1 minute
$\Delta f=6\%f_n$	1 second

Table 3-1: Operation capability of a steam turbine [3]

Therefore, in order to avoid blade fatigue and faster aging of synchronous machines, deviations from the nominal frequency must be noticed and dealt with rapidly. What is more, a severe disturbance can lead to oscillation of the rotors of the synchronous generators and ultimately to a loss of synchronism, further increasing the difference between the power production and the load.

3.2 Computation of shedding amount

In order to curtail the correct amount of load, a quick and precise estimation of the active power deficiency has to be carried out first. Then it is a common practice among Transmission System Operators (TSOs) not to shed the whole amount instantly, but instead to shed a percentage of the estimated power deficiency at several predetermined frequency thresholds.

3.2.1 Active power deficiency estimation

The frequency and the voltage are the two parameters that can significantly affect the power consumed by the loads and thus they must be taken into account. The implemented algorithm for the power deficiency estimation is based on [5], as it considers the voltage dependency of the loads and the frequency gradient. In order to model the connection between the consumed power and the load voltage, the following common exponential equations are utilized [14]:

$$P_{Li} = P_{L0i}\left(\frac{V_{Li}}{V_{L0i}}\right)^{\alpha_i} \quad (3\text{-}2)$$

$$Q_{Li} = Q_{L0i}\left(\frac{V_{Li}}{V_{L0i}}\right)^{\beta_i} \quad (3\text{-}3)$$

Where:

P_{Li} is the active power of the i^{th} load [p.u.]
P_{L0i} is the pre-fault active power of the i^{th} load [p.u.]
V_{Li} is the voltage of the i^{th} load [p.u.]
V_{L0i} is the pre-fault voltage of the i^{th} load [p.u.]
Q_{Li} is the reactive power of the i^{th} load [p.u.]
Q_{L0i} is the pre-fault reactive power of the i^{th} load [p.u.]
α_i is the voltage dependency on active power of the i^{th} load
β_i is the voltage dependency on reactive power of the i^{th} load

In most composite system loads, the exponential factors α and β typically range from 0.5 to 1.8 and from 1.5 to 6 respectively [14]. When the factor α is equal to zero, the model is regarded as constant power, when it is equal to one the model is regarded as constant current and when it is equal to two the model is regarded as constant impedance [3].

After several mathematical steps explained in [5], the power deficiency for a multi-machine system can be estimated as:

$$P_{def} = A\frac{df_{COI}}{dt} + B \quad (3\text{-}4)$$

$$A = \frac{2H_{sys}S_{sys}}{P_{L0}f_n} \quad (3\text{-}5)$$

$$B = \frac{1}{P_{L0}}\sum_{i=1}^{N_L} P_{L0i}\left[\left(\frac{V_{Li}}{V_{L0i}}\right)^{\alpha_i} - 1\right] \quad (3\text{-}6)$$

$$S_{sys} = \sum_{j=1}^{n} S_j \quad (3\text{-}7)$$

$$H_{sys} = \frac{1}{S_{sys}}\sum_{j=1}^{n} H_j S_j \quad (3\text{-}8)$$

$$f_{COI} = \sum_{j=1}^{n} f_j H_{j,eq} \quad (3\text{-}9)$$

$$H_{j,eq} = \frac{H_j S_j}{\sum_{j=1}^{n} H_j S_j} \quad (3\text{-}10)$$

where:

f_{COI} is the center of inertia frequency [Hz]
H_{sys} is the equivalent apparent inertia of the system [s]
S_{sys} is the equivalent apparent power of the system [MVA]
P_{L0} is the total pre-fault load power [MW]

S_j is the rated apparent power of the j^{th} machine [MVA]
H_j is the inertia constant of the j^{th} machine based on its own apparent power [s]
$H_{j,eq}$ is the equivalent inertia constant of the j^{th} machine based on the system rated capacity [s]
N_L is the number of loads in the system

By utilizing (3-4), the initial load shedding amount is based on the estimated power deficiency rather than on pre-defined fixed values, thus achieving a more precise UFLS operation. Moreover, the initial shedding amount can be dynamically adjusted based on certain external signals, offering the possibility to take into account other relevant parameters, such as the minimization of the load being curtailed or ensuring voltage stability of the network. Therefore, with the aid of (3-4), advanced UFLS schemes are more capable of optimizing the frequency recovery procedure and maintaining stability, when the power system is stressed or undergoes sudden disturbances.

3.2.2 Distribution of shedding amount

Minimizing the shedding amount while keeping the frequency trajectory within limits constitute the main guideline that must be taken into account in order to design an efficient load shedding scheme [5]. To achieve this, it is necessary to utilize as much primary frequency control as possible. Therefore, the initial computed value of P_{def} has to be distributed among different shedding steps. Thus, the number of shedding steps, the amount of load curtailed in each step and the conditions for the activation of each shedding step must be addressed [8]. These aspects can vary for different networks and are usually determined by the responsible TSO.

In [5] and [8], 4 shedding steps are used at the frequency thresholds 49 Hz, 48.8 Hz, 48.4 Hz and 48 Hz respectively, which are also employed by the traditional UFLS scheme of the Slovenian power system. The main purpose of the aforementioned shedding scheme is to prevent the frequency form falling below 47.5 Hz or 5% of its nominal value. In [8] different shedding steps are considered and it is found out that schemes with higher amounts of shedding steps in the beginning, such as $P_{shed,1}=0.5P_{def}$ or $P_{shed,1}=0.35P_{def}$ with progressively decreasing shedding amount, are more suitable when the frequency gradient is high. In these cases the primary frequency control takes more action and as a result less power is cut. On the other hand, schemes with lower amounts of shedding steps in the beginning, up to $P_{shed,1}=0.15P_{def}$, progressively increasing that amount, are better in the cases where the power deficit is small and frequency decline is slow, as the primary frequency control has more time to contribute in the reduction of the active power deficiency.

3.2.3 Shedding step adjustment

In [8], a method for dynamically adjusting the predefined shedding amount of each step by taking also into consideration the primary frequency control is proposed. The instantaneous value of the rate of change of frequency (ROCOF) has an approximately linear relationship with the instantaneous active power deficiency (3-4). Therefore, the effect of the contribution of the spinning reverse can be taken into account by monitoring the ROCOF between two neighbouring shedding steps. When the i^{th} frequency threshold is reached, the change in the ROCOF is first computed, as follows:

$$\Delta_i = \frac{df_{COI,max}}{dt} - \frac{df_{COI,fth(i)}}{dt} \quad (3\text{-}11)$$

It is worth noting that Δ_i can have either a positive or a negative value. Then, the percentage of change in the ROCOF is found:

$$\Delta_i\% = \frac{\Delta_i}{df_{COI,max}} 100 \quad (3\text{-}12)$$

Because of the aforementioned linearity between the ROCOF and the active power deficit (3-4), the upcoming shedding step can be modified:

$$P'_{shed,i} = P_{shed,i} - \Delta_i\% \quad (3\text{-}13)$$

For better comprehension, the shedding step adjustment procedure is explained with the help of Figure 3-1. At t_0 the maximum ROCOF is detected. Then, at t_1 the first frequency threshold is reached $f_{th,1}$. During the time period between t_1 and t_0, the primary frequency control has been activated and has reduced the initial value of the active power deficiency, by an amount approximately equal to Δ_1. Therefore, the required shedding amount of the first shedding step is no longer the predefined $P_{shed,1}$, but it is modified to $P'_{shed,1}$. The same applies for the time period between t_2 and t_1. Because of the contribution of the spinning reverse, the second predetermined shedding step is altered to $P'_{shed,2}$ by subtracting Δ_2 from $P_{shed,2}$. Following this procedure the predefined amount of each shedding step is adjusted to match the instantaneous active power deficiency, when the different frequency thresholds are reached.

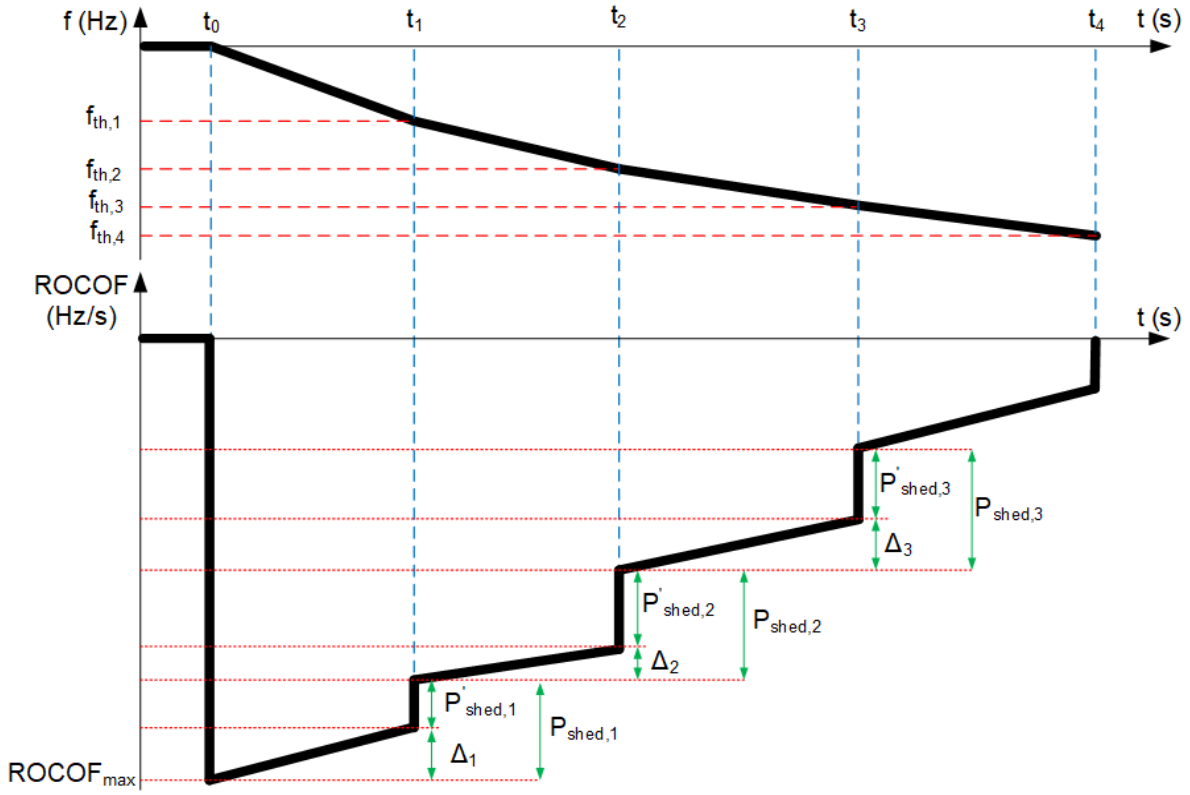


Figure 3-1: Shedding step adjustment scheme

3.3 Simulation environment

As mentioned in section 1.4, the simulations for the load shedding schemes are performed using DIgSILENT PowerFactory. Also, Python is employed to send commands to PowerFactory, create the necessary models and receive results from it as csv documents. Then the results are further processed and visualized with Matlab. The interaction between PowerFactory, Python and Matlab is depicted in Figure 3-2.

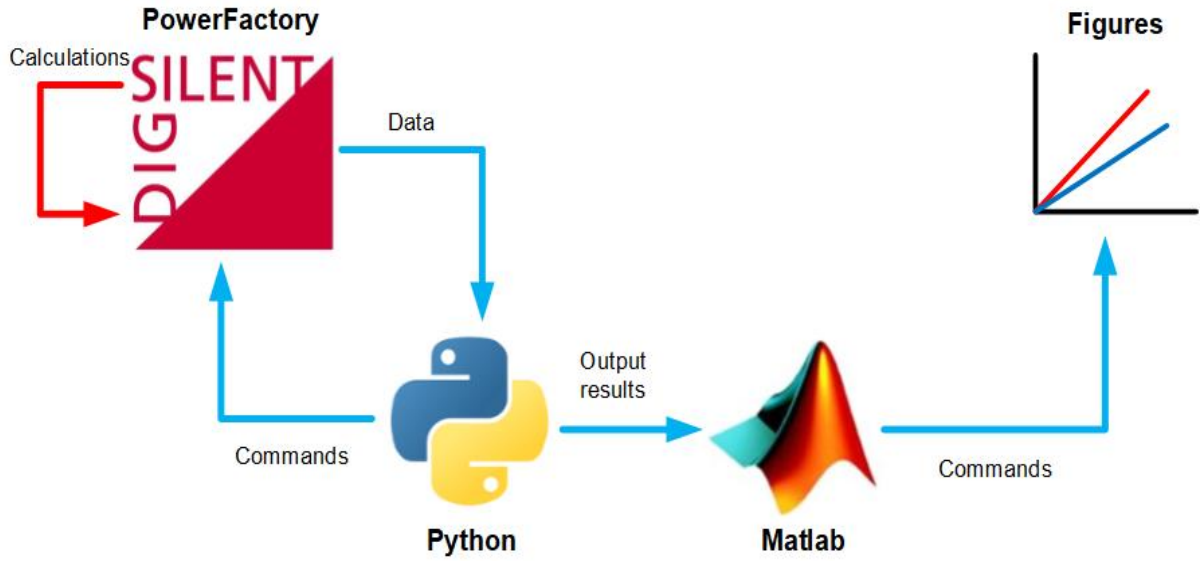


Figure 3-2: Interaction between PowerFavtory, Python and Matlab

3.3.1 Software

DIgSILENT PowerFactory is an established software for power system analysis for applications in generation, transmission, distribution and industrial systems. It offers a wide range of different functionalities, such as load flow calculation, RMS simulation, contingency analysis and modal analysis. For the present thesis the first two functionalities are used in various disturbance scenarios. For the implementation of these scenarios multiple outage events are created.

The selected elements to outage, the performed study cases and the computations to be executed are sent to PowerFactory through Python scripts. After the requested actions have been performed, the relevant data is saved as csv documents and are handled by Matlab for graphical output.

3.3.2 IEEE 39-Bus test system

For the implementation of the adaptive UFLS scheme, various power deficiency scenarios were created using the IEEE 39-bus test system. The single line diagram of this network is displayed in Figure 3-3. This system is available in PowerFactory from version 15.2 and corresponds to the simplified model of the New England system, in the northeast of the USA.

As the name suggests, this network consists of 39 buses, out of which 10 are generators and 19 are loads. There are 34 lines and 12 transformers. The nominal frequency of this network is 60 Hz and its nominal voltage is 345 kV. The generator G1 corresponds to the interconnection with the rest of the transmission system of USA and Canada and is therefore directly connected to the 345kV level, while the rest of the generators require transformers to be connected.

3.4 Methodology

As stated in section 1.4, the original load shedding code was developed by Meng Zhang for his Master thesis at TU Delft [3]. Therefore, most of the assumptions presented in section 3.4.1 are also available in [3]. The original implementations is very shortly described in section 3.4.2 and the necessary modifications required to improve the load shedding scheme are

addressed in section 3.4.3. Finally, the objective of the new load shedding scheme is stated in section 3.4.4.

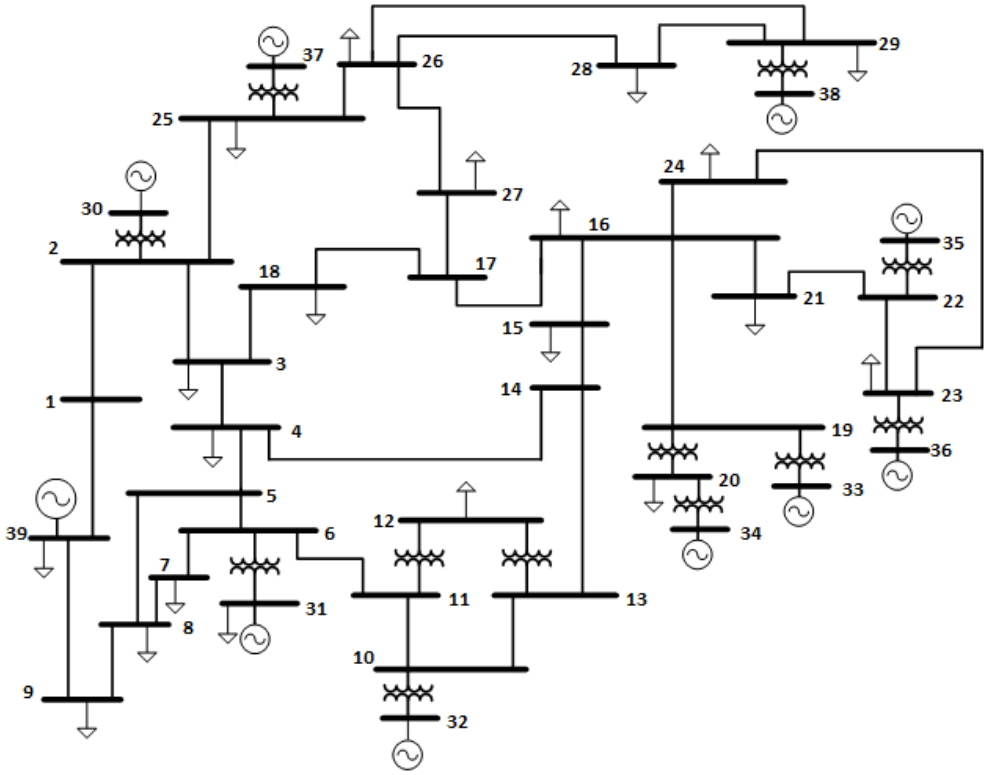


Figure 3-3: IEEE 39-Bus test system

3.4.1 Assumptions

Various assumptions to carry out the implementation are presented below [3].

In order to decrease the influence of transient and sub-transient phenomena, the frequency is measured from the generator rotational speed instead of the terminal voltage. This is a logical assumption, as there exist various technologies able to monitor the generator shaft speed with high accuracy and linearity [29].

The centre of inertia (COI) frequency computed through equation (3-9) is utilized to evaluate the average frequency of the island. The COI corresponds to a generator unit describing the average electromechanical behaviour of the generator units in the formed island. It is worth noting that the theory concerning the COI needs some refinement for power systems with high amounts of generation from renewable energy sources. In our case, there are no renewables in the system, so the theory is suitable and COI can be used. This theory should also work well for electrical networks with high amounts of hydroelectric generation (e.g., the Norwegian power system).

Although the inertia constants of generators may vary according to the operation status and the kinetic energy stored in the turbine, they are considered time-invariant for this implementation. This is justified by the fact that the time required from the detection of the frequency decrease to the protection actions performed by UFLS relays is usually around 0.3 seconds [10]. A time delay $t_{delay}=0.3$ s is also used in the UFLS scheme, in order to make it as close to the real operating conditions as possible. The synchronous turbines have a relatively large time constant, therefore the inertia can be considered time-invariant during that emergency control process.

For the aforementioned reason, load characteristics can also be considered time-invariant. Although active and reactive power of induction machines usually depend on the terminal

frequency, the active power imbalance estimation is based on the ROCOF after the disturbance. Of course there is a small time delay in the measurement and control system, but the frequency decline in a measurement interval can be considered negligible and therefore the load frequency dependence is not taken into account in the present thesis.

The active α and reactive β power dependency on voltage of the loads vary within certain ranges mentioned in section 3.2. Through equation (3-4), it is obvious that β can be neglected for the active power deficiency estimation, therefore it is assigned a constant value equal to two for the present thesis. In order to approximate the real variations of load dynamics the constant values zero, one and two are assigned to α to the load models. Then, the active power deficit is estimated and the amount of load shed is computed, considering that all the loads in the system have the same α.

To validate the performance of the proposed UFLS scheme, the assumption is made that all the loads are equally and continuously reducible. Of course, under real conditions some loads are of higher importance than others and must not be curtailed. Therefore, before using the proposed scheme, load priority must be taken into account and the necessary adjustments have to be implemented.

The active power deficit is estimated after 0.2 seconds following the disturbance, since the ROCOF has settled and only small variations are observed. As mentioned earlier, the time from the detection of the frequency decline to the protection actions of the UFLS relays is usually around 0.3 seconds, therefore an estimation time $t_{est}=0.2$ s is justified. In section 3.2.3, the shedding step adjustment procedure is explained and the maximum value of the ROCOF is utilized. For this implementation, the value of ROCOF at $t_{est}=0.2$ s is used instead, as exactly predicting the maximum value of ROCOF is rather challenging. In actual power systems the time of the disturbance is not known. Most protection devices would measure the ROCOF and would make an estimation when the current value of the ROCOF would not differ much from the previous two or three values, therefore waiting for it to settle.

3.4.2 Original implementation

Python is used to input the outage events to PowerFactory, allowing to approximate the randomness and unpredictability of disturbances in electrical networks. After the outage events are defined, the necessary study cases are created and then the 4 sub-processes presented in Appendix B are required before executing the final RMS simulation and shedding the estimated load. Finally, the data are exported to a csv file and read from python to produce the desired plots [3].

This implementation curtails quite accurately the percentage of load that must be shed. However, the main drawback of this implementation is that two separate RMS simulations are required. During the first one (sub-process 3) the active power deficiency is estimated and the results from it are inputted to sub-process 4 to create the necessary models and assign their parameters. Then, the final RMS simulation is executed in which the computed amount of load gets disconnected. When a disturbance occurs in an actual power system, the imbalance has to be estimated online. Therefore the goal is to model the online power imbalance estimation in a way as it should realistically occur in power systems.

3.4.3 New implementation

In the new implementation, it is possible to adjust many of the simulation constants through Python (e.g., the number of shedding steps, the frequency thresholds, the percentage of load curtailed in each step, the time delay in the relay along with the outage events). As explained in section 3.4.2, the sub-process 3 from the original implementation had to be eliminated as

the active power deficit is aimed to be estimated online. The necessary modifications were carried out to sub-process 4. Sub-processes 1 and 2 had minor alterations, but their function is similar to the original ones, so they will be described very briefly in the following paragraph. The original sub-processes are shown in Appendix B.

In actual power systems, island detection is performed by the control centre using the data obtained through SCADA [10]. In the PowerFactory implementation, this is performed by sub-process 1. Two switch manipulation steps are employed, between which the power system model is programmatically separated, and the islands are detected through the Python script. The topological data gathered from sub-process 1 constitutes the input to the following sub-process that collects the pre-fault data of the electrical network. To achieve that, a load flow calculation is performed during the second sub-process, and the active and reactive powers of generators and loads are computed for the power system steady state before the disturbance. Finally, this data is used by the third sub-process, explained in Algorithm 2. Here it is worth mentioning that during sub-process 3, various modifications were necessary to the adaptive UFLS Relay, in order for it to function properly without requiring the data from a separate RMS simulation. The new implementation of the adaptive UFLS Relay is presented in Appendix C.

Algorithm 2: Sub-process 3 (Create models, assign data and produce results)

Input: Topological data (sub-process 1), Pre-fault data (sub-process 2)
Output: Results in csv format

1 Create block frame for the islands
2 Create common models
3 Create composite models
4 Link models with their block definitions
5 Assign parameters to models
6 RMS-simulation
7 Estimate the active power deficiency at $t_{est}=0.2$ s
8 Perform load shedding
9 Export results to csv file

Figure 3-4 constitutes an example adaptive UFLS block diagram created for an island consisting of 3 generators and 3 loads. The first output of the generators is the rotational speed that is used as the input to the *COI* block, where the COI frequency and the ROCOF are calculated. The second output of the generators is the turbine power used as input to the *Island prod* block that computes the power generated in the island. The outputs of the *COI* block are passed to the UFLS relay, where the value of the active power deficiency is estimated at $t_{est}=0.2$ s. To estimate it based on equation (3-4) the term B that takes into account the voltage dependency on active power of the loads needs to be considered. The term B is computed through equation (3-6) in the *B* block. The output of the *UFLS relay* block is the same scaling factor for all the loads, a number between zero and one indicating how much all the loads have to be reduced. Inside the gain blocks the multiplication of the scaling factor with the initial active and reactive power of the loads occurs and the results are inputted to the loads. The first output of each load is its instantaneous voltage and the second output is its instantaneous active power consumption. Both these parameters are fed into the *B* block that calculates the term B of the equation (3-4) and sends it to the *UFLS Relay* block. Inside the *B* block also the instantaneous island consumption is found and is used as input to the *Actual imb* block. That block takes as input also the instantaneous production of the island and therefore the actual instantaneous power deficiency is computed there, in order to make a comparison with the estimated instantaneous power deficiency in the *UFLS Relay* block.

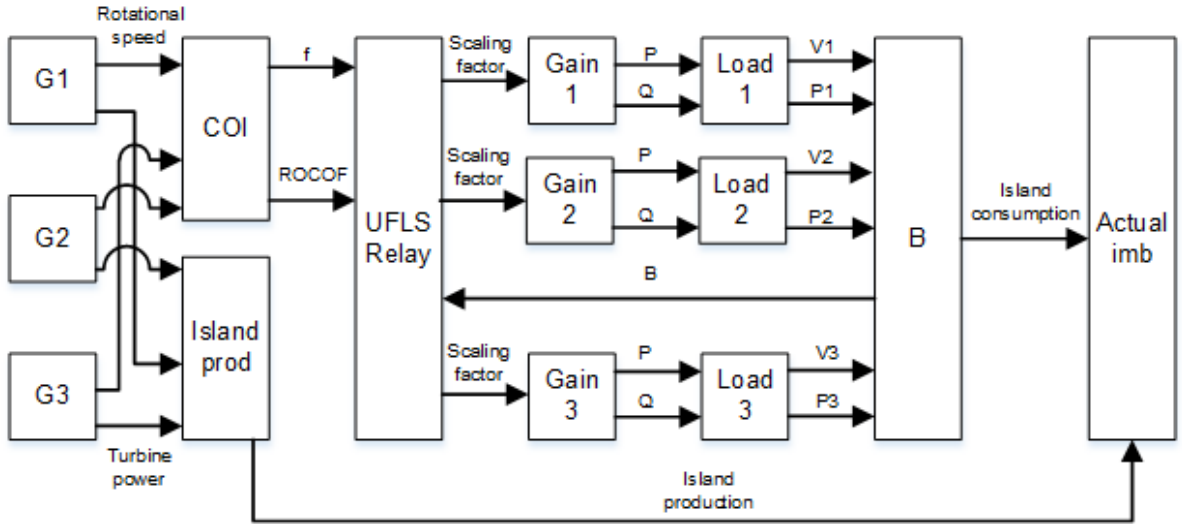


Figure 3-4: Block diagram of the UFLS scheme for an islands consisting of 3 generators and 3 loads

3.4.4 Implementation objective

The objective of the newly developed load shedding scheme is to minimize the amount of load being curtailed, while keeping the frequency within predetermined limits. Therefore, the same number of shedding steps and the same frequency limits as the ones used in the Slovenian power system [5], explained in section 3.2.2, are employed but they are adapted to a system with a nominal frequency of 60 Hz. It is worth noting that these limits are not universal and ideally should be adapted by the corresponding TSO, based on the distinctive features of the electrical network.

The new load shedding scheme firstly estimates quickly and accurately the active power deficiency and is able to track it throughout the duration of the disturbance. Then, the frequency must not decrease below 95% of the nominal system frequency, but the closer it gets to that value the more successful the scheme is, as less load is curtailed and more primary frequency control is utilized. Two different load shedding distributions are evaluated, the first one cuts a high amount of the estimated power imbalance in the beginning and afterwards cuts smaller amounts, while the second one works the other way around. Table 3-2 summarizes the load shedding distributions tested, the frequency thresholds used and the minimum allowed frequency for a power system with nominal frequency of 50 Hz and 60 Hz.

f_n (Hz)	Frequency thresholds (Hz)				f_{min} (Hz)
50	49	48.8	48.4	48	47.5
60	58.8	58.6	58.1	57.6	57
Distribution	$P_{shed,1}$	$P_{shed,2}$	$P_{shed,3}$	$P_{shed,4}$	
1	50%	20%	20%	10%	
2	10%	20%	20%	50%	

Table 3-2: Load shedding distributions and frequency thresholds used

3.5 Results

In this section the results of the implemented load shedding scheme are presented and explained. Various disturbances are considered that could lead to various active power deficits in the IEEE 39-Bus test system. First, the outage of two large generators is examined, resulting in a very large active power deficiency. Second, some predefined lines are disconnected

resulting in the formation of islands in the original grid and the scheme is implemented on the island with a surplus in demand. For the islanding cases a small and a larger active power deficiency are considered.

In the following figures (Figure 3-6, Figure 3-7, Figure 3-8, Figure 3-10 and Figure 3-12) the two aforementioned distributions are evaluated. First, distributions 1 and 2 correspond to the distributions presented in Table 3-2, without the shedding step adjustment procedure, explained in section 3.2.3. Then, distributions 3 and 4 correspond to the same distributions, when the step adjustments are implemented. The scheme that curtails the least amount of load is identified.

3.5.1 Event 1: Outage of large generators

During this event, two large generators are disconnected at the same time, the generator 8 that corresponds to bus 37 and the generator 9 that corresponds to bus 38, as shown in Figure 3-5. As a result, when this event takes place an active power deficit of 1326 MW is detected.

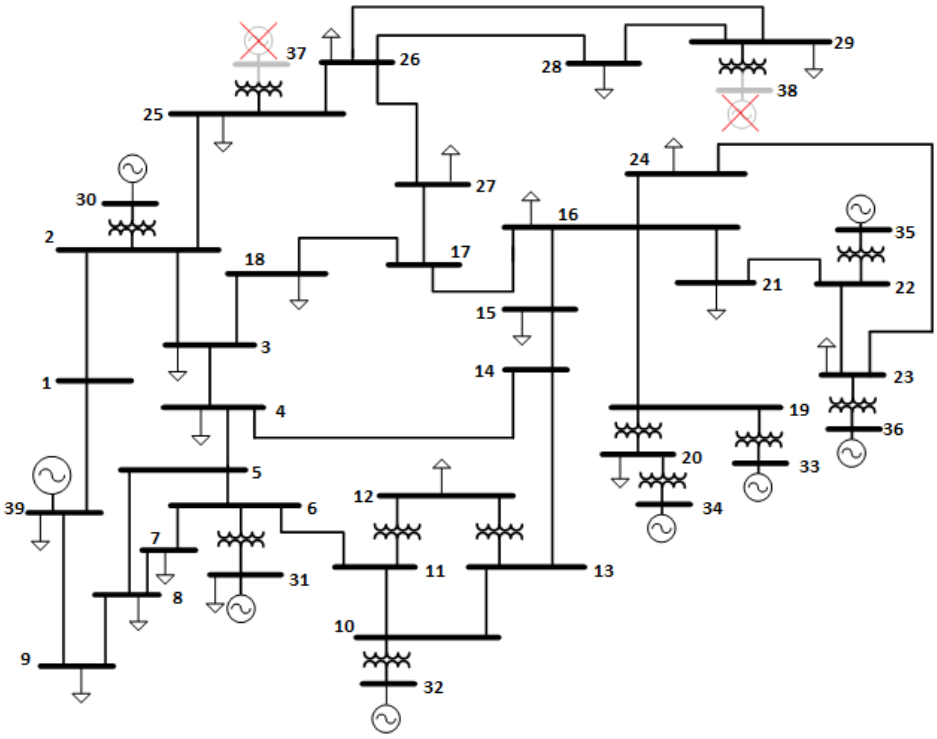


Figure 3-5: Outage of generators 8 and 9

Figure 3-6 shows the reached minimum frequency and the total load shed by each distribution for $\alpha=2$. In this case distribution 2 produces quite poor results as it has the lowest minimum frequency and curtails the highest amount of load, even more than the power deficit detected when this event occurred. Distribution 4 on the other hand, reaches the same frequency, but sheds the least load out of all the distributions. Thus, it could be considered the best alternative, when trying to minimize the load shed. Furthermore, the frequency is kept above the minimum limit at all times. A safer alternative is also distribution 1 as it sheds about 40 MW more and keeps the frequency at 58.5 Hz.

The total load curtailed and the reached minimum frequency of each distribution when $\alpha=1$ are presented in Figure 3-7. Once more distribution 2 performs poorly. However, this time the best alternative seems to be distribution 3, as it curtails the least load and does not allow the frequency to drop below 58.2 Hz.

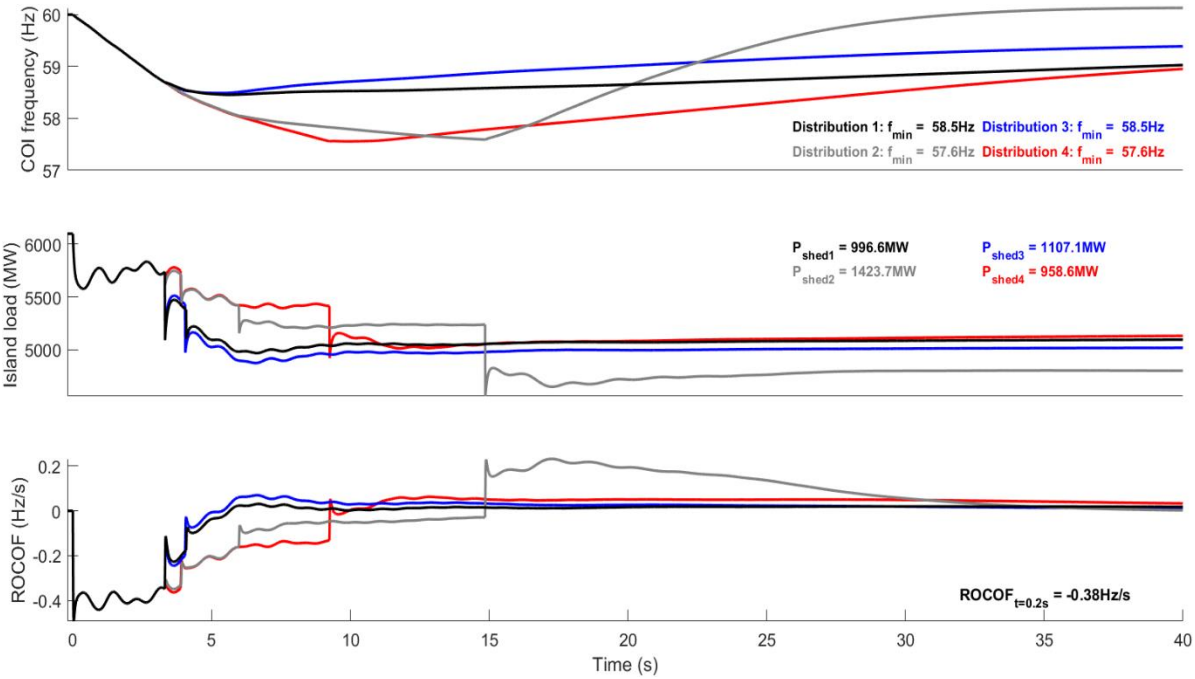


Figure 3-6: COI frequency, island load and ROCOF during event 1 for α=2

Through Figure 3-8, it can be seen that when $\alpha=0$, again distribution 3 curtails the least load, while keeping the frequency within the acceptable limits. Distribution 1, which is the same as 3 without the shedding step adjustments, also produces similar results but is more on the safer side.

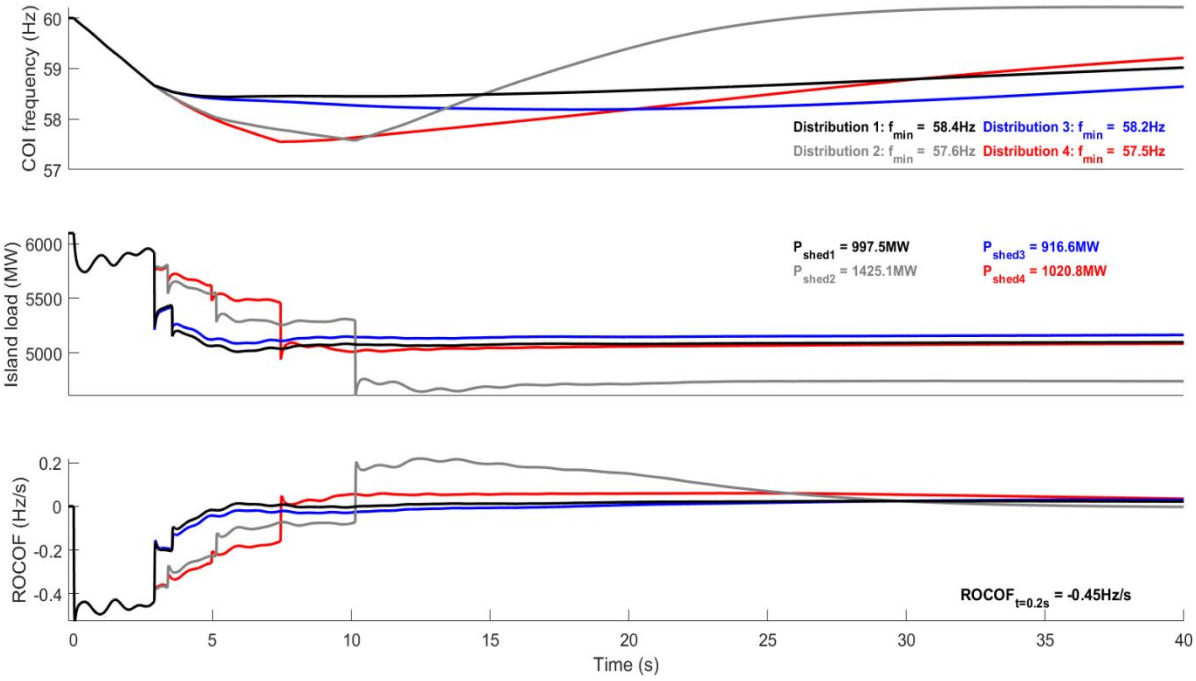


Figure 3-7: COI frequency, island load and ROCOF during event 1 for α=1

The reason why different distributions are better as the value α changes is explained by the observation made in section 3.2.2. There it is observed that distributions with higher amounts of shedding steps in the beginning, like distributions 1 and 3, are more suitable when the frequency gradient is high, while distributions with lower shedding amounts in the beginning, like distributions 2 and 4, are better in the cases where the initial active power

deficit is small and frequency decline is slow. As the value of α increases, the power deficit remains the same, but the value of ROCOF decreases, shown in Figure 3-6, Figure 3-7 and Figure 3-8 for the value of ROCOF at t_{est}. Therefore, when $\alpha=2$ the ROCOF is low and distribution 4 is more suitable, while when $\alpha=1$ and $\alpha=0$ the ROCOF is higher and distribution 3 is better. The value of ROCOF is higher when $\alpha=0$ because as seen in Figure 3-8 the island load remains unaltered after the disturbance occurs and only decreases when load shedding takes place. On the contrary, when $\alpha=1$, the island load decreases after the disturbance, as seen on Figure 3-7, and therefore the active power deficit is reduced, leading to smaller ROCOF. This decrease in the island load is justified because all the loads depend on their respective voltages and their voltages drop after the event. For $\alpha=2$ the island load drops even more as the dependence on the voltages is higher and thus the active power deficit is even smaller leading also to lower ROCOF.

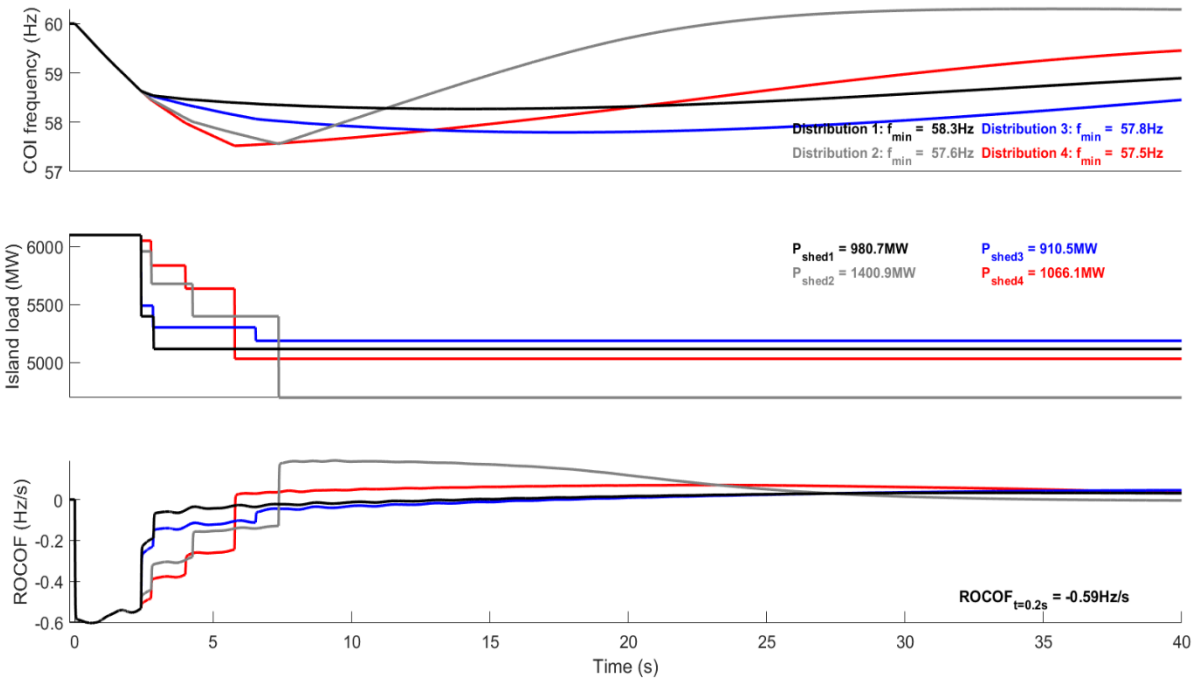


Figure 3-8: COI frequency, island load and ROCOF during event 1 for α=0

For all the values of α, it is found out that the distributions with the shedding step adjustments (3 and 4) outperform the ones without step adjustments (1 and 2) in terms of minimizing the shedding amount. In fact, distribution 2 is found out to perform very poorly in all the cases. Distribution 1 constitutes a good safer alternative for all the values of α to the solution minimizing the shedding amount, as it curtails a little more load, but it keeps the frequency about 1 Hz higher.

In the following events, the results for the two extreme cases of α ($\alpha=0$ and $a=2$) are going to be presented next to each other for easier comparison. It was found out that usually when $\alpha=1$, the more suitable distribution is the same as when $\alpha=0$, just like it was observed in this section.

3.5.2 Event 2: Islanding scenario (small power deficiency)

In this event, the lines 15-16 and 16-17 are tripped simultaneously, resulting in the formation of the two islands shown in Figure 3-9. The grey island has a surplus in active power and thus is not further considered. The resulting initial active power deficiency in the black island is 519 MW.

From Figure 3-10 it is clear that when $\alpha=0$ the distribution that sheds the less load is the third one, while at the same time keeping the frequency above 58 Hz. Of course since the active power deficiency in this case is rather small, the differences between the amounts shed by each distribution are very small. When $\alpha=2$, since the power deficit and the ROCOF are both quite small no shedding amount is required. This is found out when adjusting the shedding steps and thus both distributions 3 and 4 curtail no load, allowing the primary frequency control to take all the action and recover the frequency.

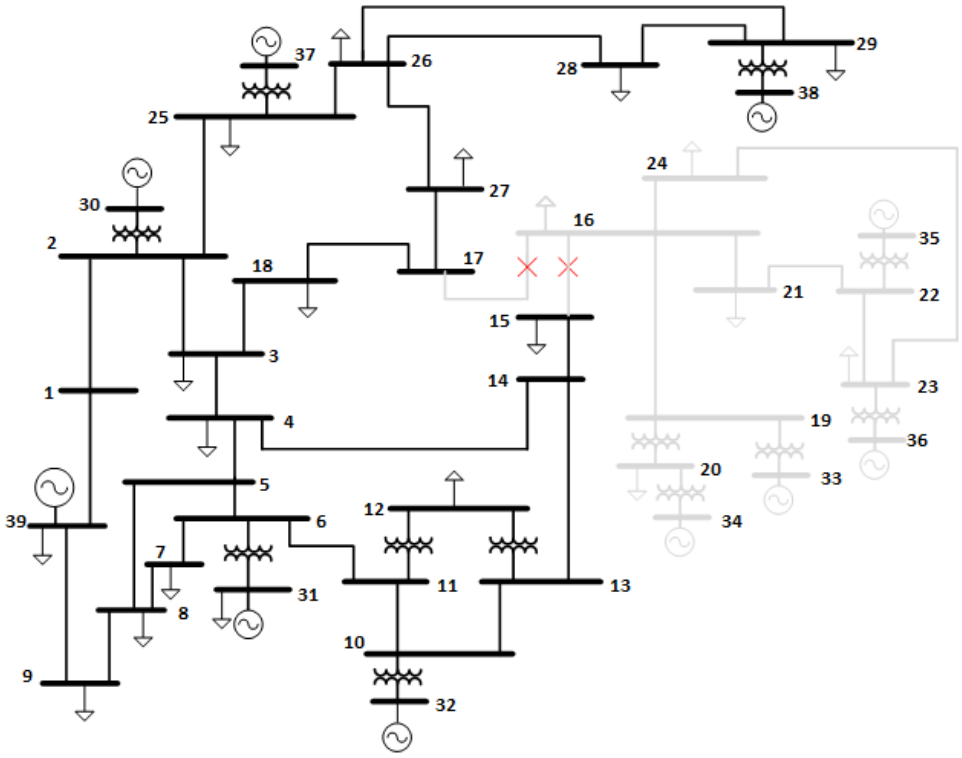


Figure 3-9: Outage of lines 15-16 and 16-17

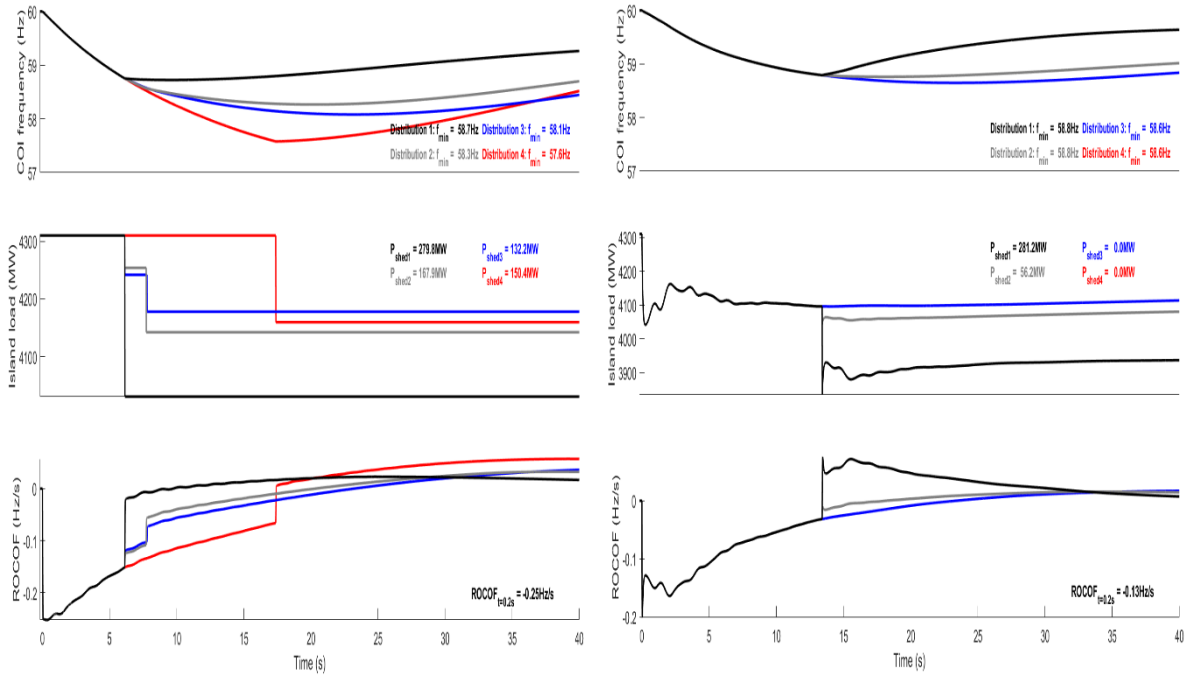


Figure 3-10: COI frequency, island load and ROCOF during event 2 for α=0 (left) and α=2 (right)

Again for both the values of α, the distributions that shed the less amount are the ones with the adjusted steps, as they are able to adapt the amount being curtailed to the value required the time the different frequency thresholds are reached. Moreover, they prove to be reliable as they keep the frequency within the allowed limits.

3.5.3 Event 3: Islanding scenario (larger power deficiency)

During this event, two islands are formed because of the simultaneous tripping of the lines 21-22 and 23-24, as seen in Figure 3-11. In the grey island power production is higher than the demand and therefore no load shedding is needed. The moment the islands are created, an active power deficiency of 919 MW is detected in the black island.

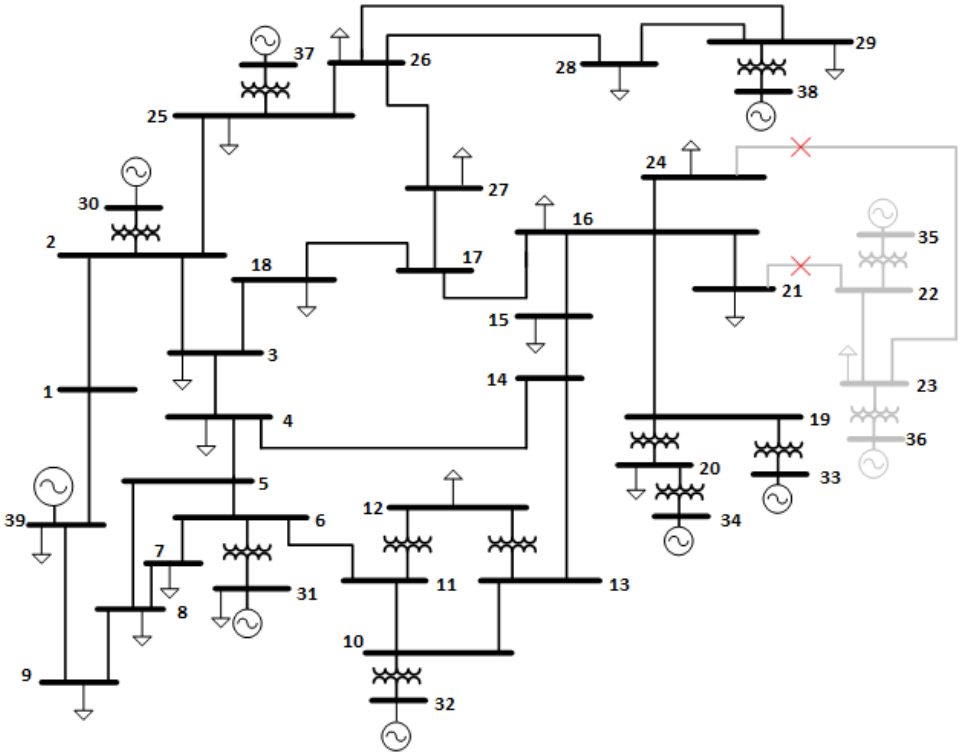


Figure 3-11: Outage of lines 21-22 and 23-24

For both the values of α, it can be seen through Figure 3-12, that the best distribution is the second one, as it curtails the less load and does not allow the frequency to drop below 57.7 Hz. In this case both distributions with shedding step adjustments shed more load than the second distribution. However, they still verify the fact that for higher values of ROCOF ($\alpha=0$) distribution 3 is more suitable than 4, while for lower values of ROCOF ($\alpha=2$), distribution 4 outperforms distribution 3 in terms of minimizing the shedding amount.

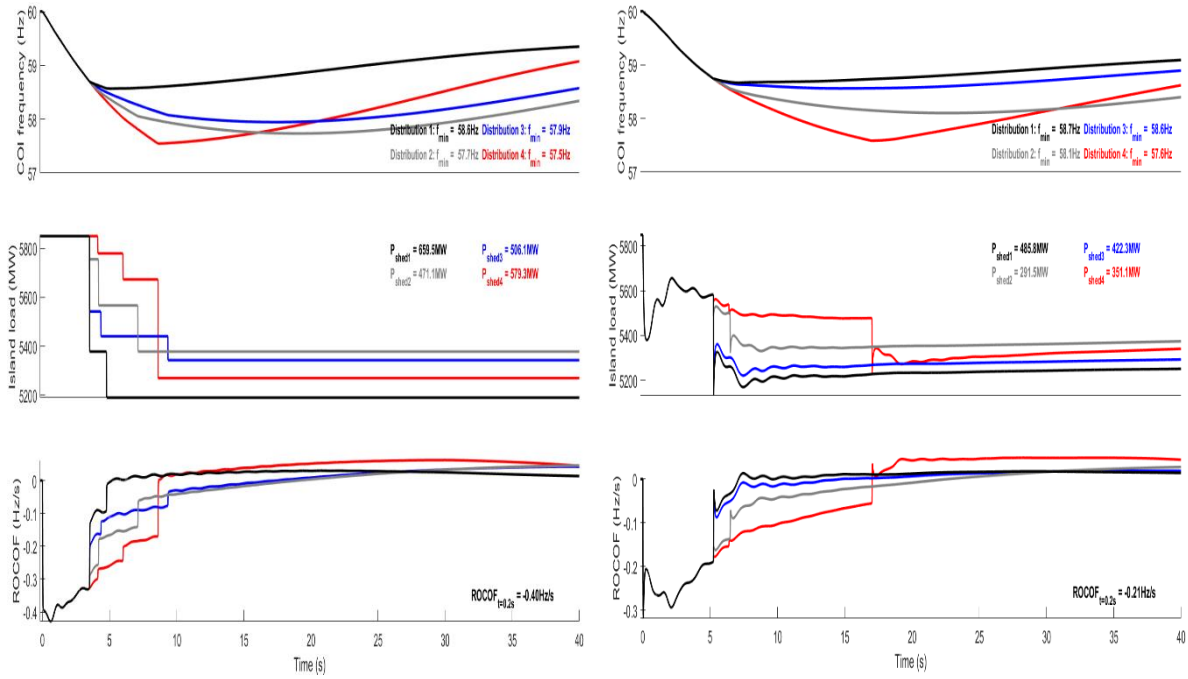


Figure 3-12: COI frequency, island load and ROCOF during event 3 for α=0 (left) and α=2 (right)

3.6 Conclusions

In this chapter, an adaptive UFLS scheme is proposed that minimizes the load being curtailed based on [5] and [8]. Two different load shedding distributions are compared, with and without adjustments on their load shedding steps. All the distributions managed to keep the frequency within the predefined limits for all the disturbances considered.

The distributions with the shedding step adjustments proved to shed less load when small power deficits were detected, even not shedding any load and letting the primary frequency control recover the frequency. Furthermore during the outage of two large generators, a big active power deficit occurred and the distributions with shedding step adjustments again curtailed less load. During the second islanding case they were not as good as the second distribution, but their results were still quite close.

From the results it is observed that as the value of α increases, the ROCOF decreases. Higher values of ROCOF require distributions with higher shedding amounts in the beginning, while lower shedding steps in the beginning are more suitable when the ROCOF is smaller. The severity of the disturbance can also affect the choice of the distribution. Therefore, claiming that one distribution is the best is rather challenging. The distributions with the shedding step adjustments proved to be quite consistent under all the disturbances. On the other hand, the second distribution performed very poorly during Event 1 but was found to be the best alternative during Event 3. Thus, it could be claimed that the distributions with shedding step adjustments are better in terms of consistency of results.

The applicability of the presented scheme with the available infrastructure is also crucial. A lot of the parameters required are easily accessible, such as the nominal frequency of the electrical network or the inertia constants and the rated apparent power of the generators. Some others, such as the consumption and the voltage of each load, prior to the disturbance and after the event, can be measured with the aid of PMUs. Thus, fast and reliable communication links are required. The number of generators and loads of each island can be determined as long as the boundaries of the resulting island are recognized. Therefore, again quick and reliable data transmission between the underfrequency relays and the SCADA system is necessary [5]. Knowledge of the voltage dependency on active power of the loads

in the island formed is also needed, as it was found out to influence the ROCOF and as a result the more suitable load shedding distribution. However, an error in the determination of α does not affect significantly the proposed UFLS scheme reliability, but it could result in higher amounts of load being curtailed.

Finally, it is worth noting that when creating a load shedding scheme the distinct features of each power system must be considered. The frequency thresholds and the shedding steps may be modified accordingly. They also depend on the strategy of the TSO, as shedding more load constitutes a safer approach not allowing the frequency to drop a lot but is more costly, while shedding less load is more risky but could reduce considerably the economic losses of the TSO during a large disturbance.

Chapter 4 Conclusions and future work

In this chapter the accomplishment of the main objective of the present thesis is addressed. Then, the main observations from the research done during this thesis are summarized. Finally, based on the conclusions drawn from the present development, possible recommendations for future research on the topics studied arise.

4.1 Conclusions

For the development of the coherency cutset determination scheme all the evaluated algorithms seem to work well in most cases, which is a sign of their great potential. However, they fail in some cases and this may take some investigation. This may be both due to inconsistent constraints/synthetic generator data or due to problems with the algorithms themselves (e.g., hMetis uses recursive bisection which may accept some bisections that make the later groups separations impossible).

It was observed that using k-medoids, a centroid based clustering algorithm that finds the centroid and then tries to assign all points based on their distance from the closest centroid, will often result in non-compact groups. Therefore is not optimal for obtaining the coherency constraints and the slow-coherency cutset of FCSC. Hierarchical clustering is suitable for producing the coherency constraints, but it also leads to a poor constraint satisfaction when applied as the final step of FCSC. Thus, an algorithm has been developed to produce the slow-coherency cutset of FCSC that considers the coherency constraints, the physical connections of the network and the distances between the buses.

Out of the three evaluated algorithms, the developed FCSC-based clustering method seems to be the best alternative as it managed to satisfy the coherency constraints very well. However, it is worth pointing out that the developed FCSC-based clustering method also cuts the most power out of the three studied algorithms, possibly because it was designed with a strong bias towards satisfaction of bus grouping constraints.

hMetis and TCRBGC also have a low number of coherency constraint violations thus constituting viable alternatives to the developed FCSC-based clustering method. From the different slow-coherency cutsets produced, it was found out that TCRBGC usually avoids creating very small islands, as it is more biased towards equal cluster size. As a result, it may sometimes prioritize the avoidance of very small clusters over the constraint satisfaction. On the other hand, since it is possible to increase the imbalance of the resulting partitions in hMetis, hMetis will prioritize the minimization of the number of coherency constraint violations and in order to achieve that will often return singleton islands.

As far as the post-islanding control actions are concerned, an UFLS scheme was successfully implemented to recover the frequency to its nominal value in the islands suffering from an excess in power demand. This UFLS scheme is based on an innovative method of estimating the active power deficiency using the frequency gradient of the generators. The implemented scheme is also able to minimize the amount of load being curtailed, therefore reducing considerably the economic losses of the TSO during a large disturbance.

From the various load shedding distributions evaluated, it is observed that the ones with the shedding step adjustments produce the most consistent results. The second distribution without any step adjustments was found to be the best alternative during the second islanding scenario, but it was also found to perform quite poorly during the outage of large generators.

On the other hand, both distributions with the shedding step adjustments always constitute very good alternatives when it comes to minimizing the amount of load being curtailed and keeping the frequency within the predefined limits.

What is more, it was observed that the ROCOF decreases as the voltage dependency on active power of the loads increases. This happens because the voltage of the loads decreases when a disturbance occurs. Accordingly high values of α lead to a considerable drop in the power consumed by the loads and to a smaller active power deficit. Therefore, high values of α correspond to slower changes in frequency. For these cases, distributions with lower shedding steps in the beginning tend to cut less power, while preventing the frequency from decreasing too far. On the other hand, lower values of α lead to higher values of ROCOF, as the power consumed by the loads does not drop as much. For these cases, distributions with higher shedding amounts in the beginning are more suitable.

Finally, it is recommended that before applying the proposed UFLS scheme, adaptations are made by the TSO to consider the distinct features of the power system. Therefore, the frequency thresholds and the shedding steps must be modified according to the electrical network and the strategy of the TSO, safer and costlier or riskier and more economic.

4.2 Future work

Despite the work carried out in the present thesis, some assumptions and simplifications have been made, which have to be approached with caution as they could be subject of considerable changes in the highly complex environment of an actual electrical network. Therefore, in order to strengthen the developed method, certain possibilities for continuation of the present implementation are considered.

First and foremost, graph partitioning coherency constraints were based on steady state network conditions, as more focus was placed into achieving good constraint satisfaction and evaluating the performance of the three algorithms tested. The coherency constraints are affected during transients and it is very likely that the coherent generator groups will change when a disturbance takes place. Therefore, a study case for different contingencies could be carried out to evaluate the new coherency constraints and their satisfaction.

Although the results of hMetis were found to be the very good, there can still be improved. As stated in section 2.3.2.2, the version of hMetis used during the present thesis is an experimental release. Thus, when the stable version becomes available, it needs to be tested, as it could potentially lead to further improvement of the constraint satisfaction.

Outlier detection algorithms could also be considered for detecting possible singleton coherent generator groups during coherency analysis that could lead to the formation of singleton islands, when the slow-coherency cutset is produced. Then, these generators could be reassigned to bigger generator groups close to them or shut down, in order to avoid creating islands with very small inertia.

As stated in the first chapter, the goal of this thesis is to implement several methods that could be used as building blocks for the design of an adaptive intentional controlled islanding scheme. Therefore, other methods could be considered such as generation rejection, when the production is higher than the demand.

As already mentioned, the implemented adaptive load shedding scheme is based on the COI theory that works well for electrical networks with conventional and hydroelectric power plants. However, as the share of renewables in most power systems keeps increasing, the COI theory may need some refinement and thus the proposed UFLS scheme will have to be adjusted accordingly. This can be a challenge as the renewable energy sources do not contribute to the oscillations or the inertia of the system [8].

Furthermore, the proposed UFLS scheme could be modified to take into account the reactive power of the resulting islands. There is always a possibility of forming an island, where a small amount of active power deficit is detected and therefore no load shedding is deemed necessary, while there could be a large deficit in reactive power, which could go undetected and lead to severe voltage drops. The proposed UFLS scheme could still operate if the voltage dependency on active power of the loads is high, as this would also lead in a drop in the active power of the loads. However, if the value of α is rather low and a large reactive power deficit is present, the voltage instability risk increases considerably.

Finally, the developed UFLS scheme could be tested in larger power systems. The large electrical networks available in Matlab through the Matpower toolbox are not yet modelled in PowerFactory. When larger power systems become available in PowerFactory, more general conclusions can be drawn about the implemented UFLS scheme.

A hMetis

hMetis supports as input unweighted hypergraphs, hypergraphs with weights on hyperedges, hypergraphs with weights on vertices and hypergraphs with both weights on hyperedges and weights on vertices. For power networks, mainly the first two options are of importance. Thus, a function was written to adjust the content of the adjacency and the incidence matrix of a graph into a suitable file for input to hMetis, shown in Figure A-1. The input file is an unweighted hypergraph, in case the adjacency matrix of the network contains only 0 and 1, otherwise it is a hypergraph with weights on its hyperedges. In both the cases the input file has $|E|$+1 lines, with $|E|$ being the number of hyperedges. During the first case the first line has two numbers, the number of hyperedges and the number of vertices. Then, from the second line, each line corresponds to one of the hyperedges. It displays which vertices are included by the hyperedge i-1, with i being the number of the line in the input file. In the second case, the one used to acquire the results, the first line has three numbers, the number of hyperedges, the number of vertices and 1, meaning that the input file has weights on its hyperedges. After the second line, the i^{th} line corresponds to the hyperedge i-1. It displays the weight of the hyperedge (first number) and the vertices are included by the hyperedge (rest of the numbers) [30].

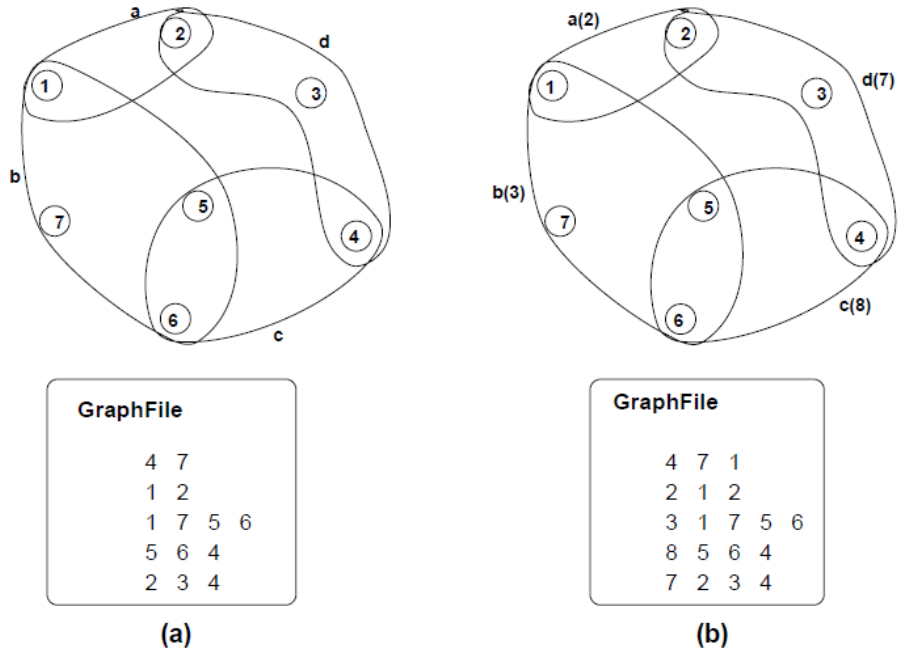


Figure A-1: Format of the input file: a. Unweighted hypergraph, b. Hypergraph with weights on hyperedges [29]

The output of hMetis is a file with N lines with a single number per line. Each line i corresponds to the vertex i and contains the partition it belongs to. Note that the partitions range from 0 to k-1, instead of 1 to k.

In order to specify the vertices that are going to be pre-assigned in certain partitions a file with N lines is used, using the command –fixed. Each line specifies that the i^{th} vertex will be pre-assigned to a specified partition. In this file, the numbers of partitions starts from 0, so if k groups of coherent generators are identified, the generators in the first group will be assigned to partition 0, the generators in the second group will be assigned to partition 1 and the generators in the k^{th} group will be assigned to partition k-1. For the buses without generators there is no information, meaning that they can be put in any partition. Thus, the vertices

associated with these buses receive a value of -1, meaning that hMetis has to judge where they will be assigned [30].

It is worth noting that hMetis offers both the option of direct k-way partitioning and recursive bisection. Both of these options were tested and it was found out that recursive bisection works very well in terms of constraint satisfaction, while the performance of the direct k-way partitioning was rather poor. The rest of the parameters and different options offered by hMetis, as well as the ones used to produce the partitions, can be found in Table A-1.

Parameters	
filename	Stores the hypergraph to be partitioned.
nparts	The number of partitions to split the hypergraph.
Options used	
-rtype=string	Specifies the scheme to be used for refinement. The default value was fast (Fast FM-based refinement) and was changed to slow (Slow FM-based refinement), since it led to better results.
-ufactor=float	Specifies the imbalance factor. For -ptype=rb specifies the maximum difference between each successive bisection (a value of 5 leads to a 45-55 split at each bisection). This value was consecutively increasing from 1 until 49 with a step of 1. The value that minimizes the number of coherency constraint violations is picked. If there are many values that minimize the number of coherency constraint violations, the one with the smallest maximal expansion is picked.
-nruns=int	Specifies the number of different bisections to be computed at each level. The final bisection corresponds to the one that has the smallest cut. This parameter was increased to 20.
-nvcycles=int	Specifies the number of solutions to be further refined using V-cycle refinement. If the supplied number is X, then the best X bisections are further refined. The default value is 1 and was changed to 20, so that all the bisections computed are also refined.
-cmaxnet=int	Specifies the size of the largest net to be considered during coarsening. Any nets larger than that are ignored when determining which cells to merge together. The default value is 50 and was changed to 150, as it resulted in better quality partitions.
-rmaxnet=int	Specifies the size of the largest net to be considered during refinement. Any nets larger than that are ignored during the multilevel partitioning phase, and are dealt once the partitioning has been computed. The default value is 50 and was changed to 150, as it resulted in better quality partitions.
-kwayrefine	Instructs hmetis to perform a final k-way refinement once the partitioning has been computed using recursive bisection (applies only when -ptype=rb).
-fixed=string	Instructs hmetis to read the file specified as the argument of this parameter for specifying the groups of cells to be placed in the same partition.
Options not changed	
-ptype=string	Specifies the scheme to be used for computing the k-way partitioning. The default option was used (rb), since the quality of the partitions produced using kway was significantly worse.
-ctype=string	Specifies the scheme to be used for coarsening. The default option was used (h12 if nruns>=20).

-otype=string	Specifies the objective function to use for k-way partitioning (applies only when -ptype=kway).
-reconst	Instructs hmetis to create partial nets within each partition representing the nets that were cut during the bisection. This option produced worse quality partitions and therefore it was abandoned.
-seed=int	Selects the seed of the random number generator.
-dbglvl=int	Specifies the type of progress/debugging information that will be printed.
-help	Displays the command-line options along with a description.

Table A-1: Parameters and options offered by hMetis

B Original adaptive UFLS implementation

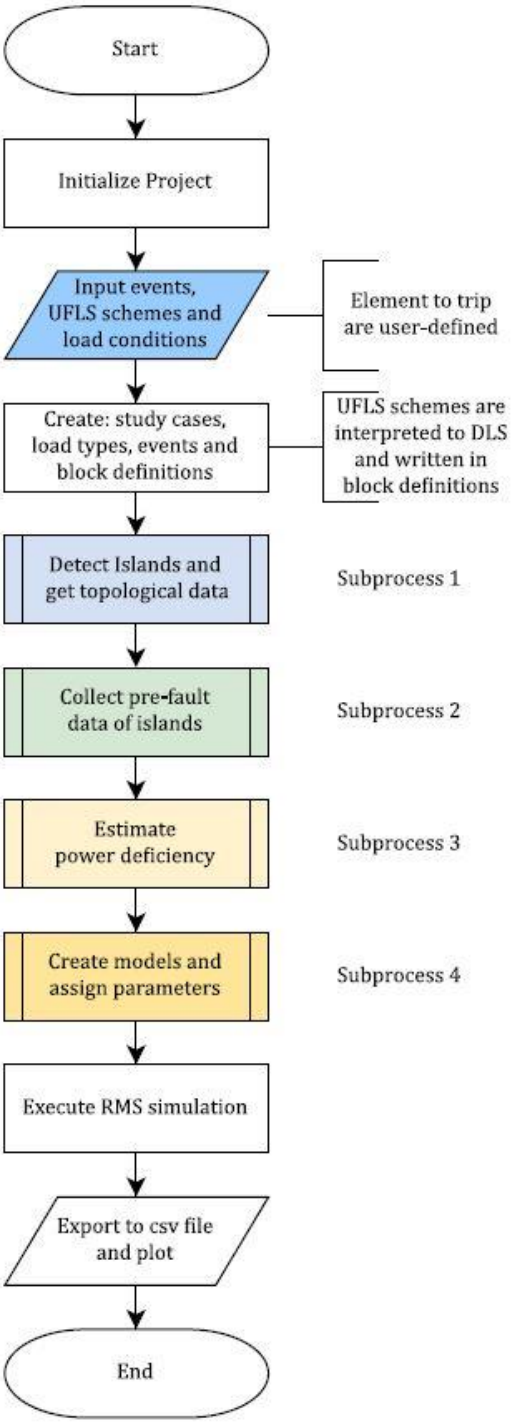


Figure B-1: Original implementation process of UFLS scheme [3]

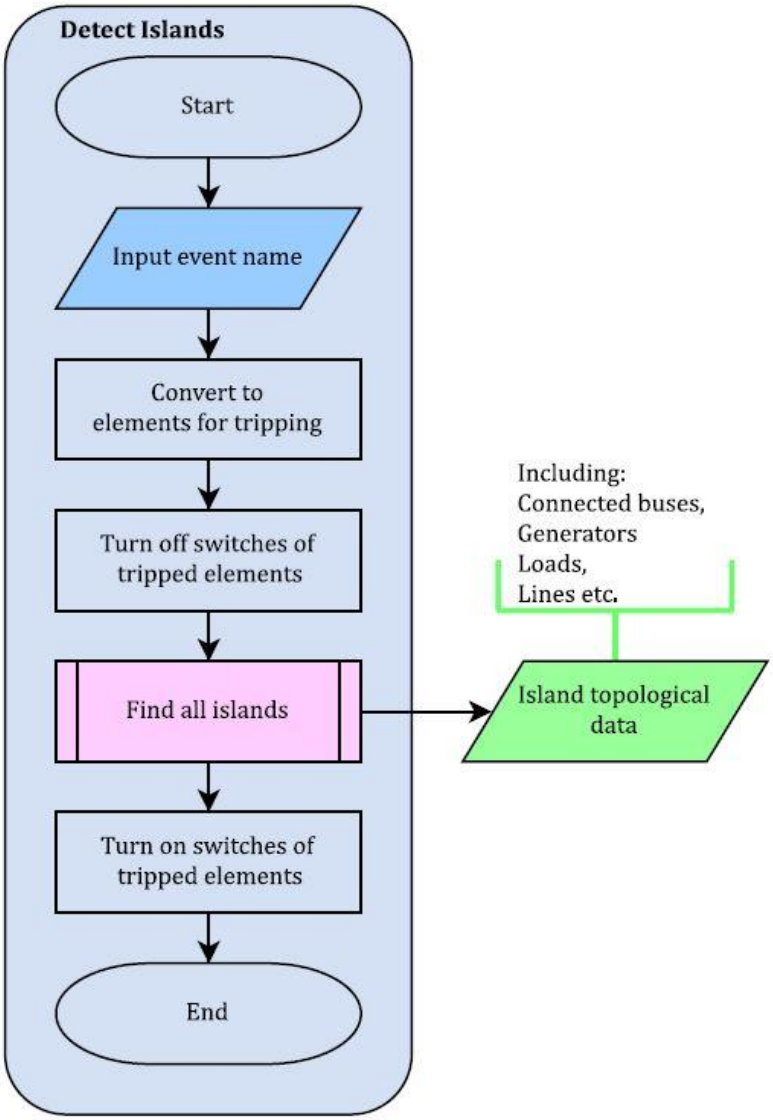


Figure B-2: Sub-process 1: Detect islands and extract topological data [3]

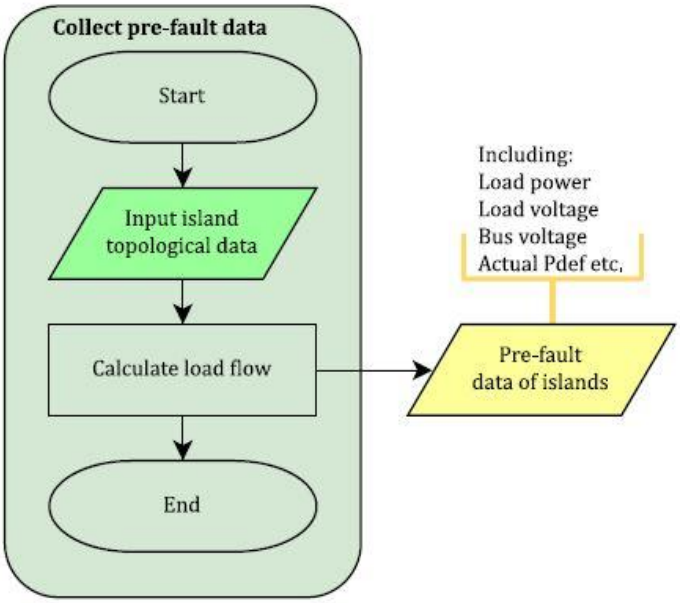


Figure B-3: Sub-process 2: Collect pre-fault data [3]

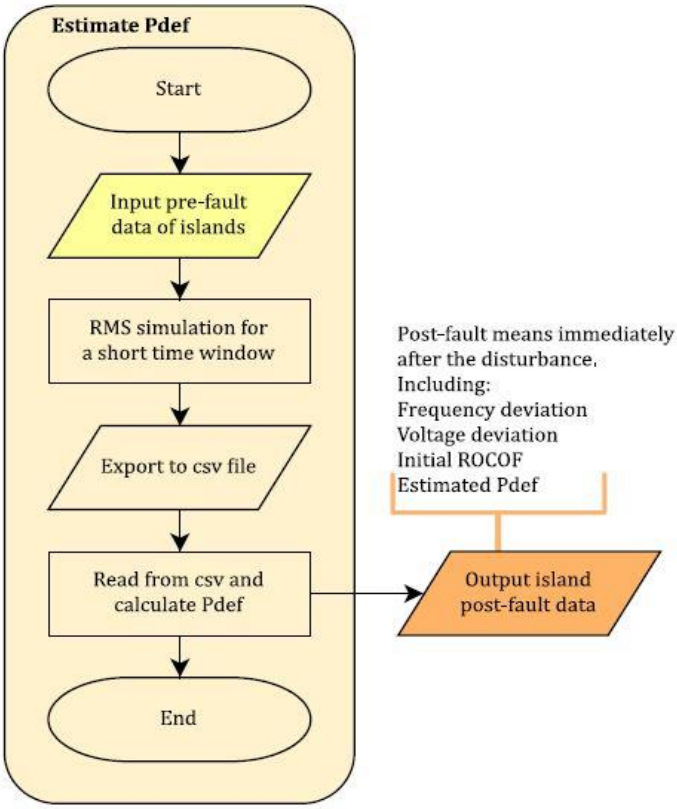


Figure B-4: Sub-process 3: Estimate power deficiency [3]

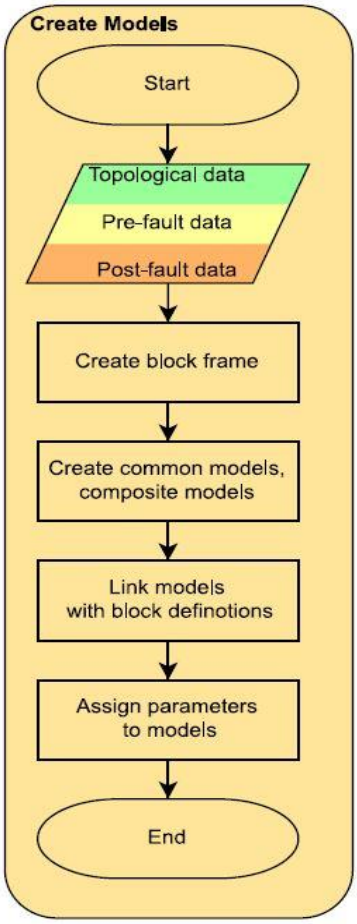


Figure B-5: Sub-process 4: Create models and assign data [3]

C Adaptive UFLS relay in PowerFactory

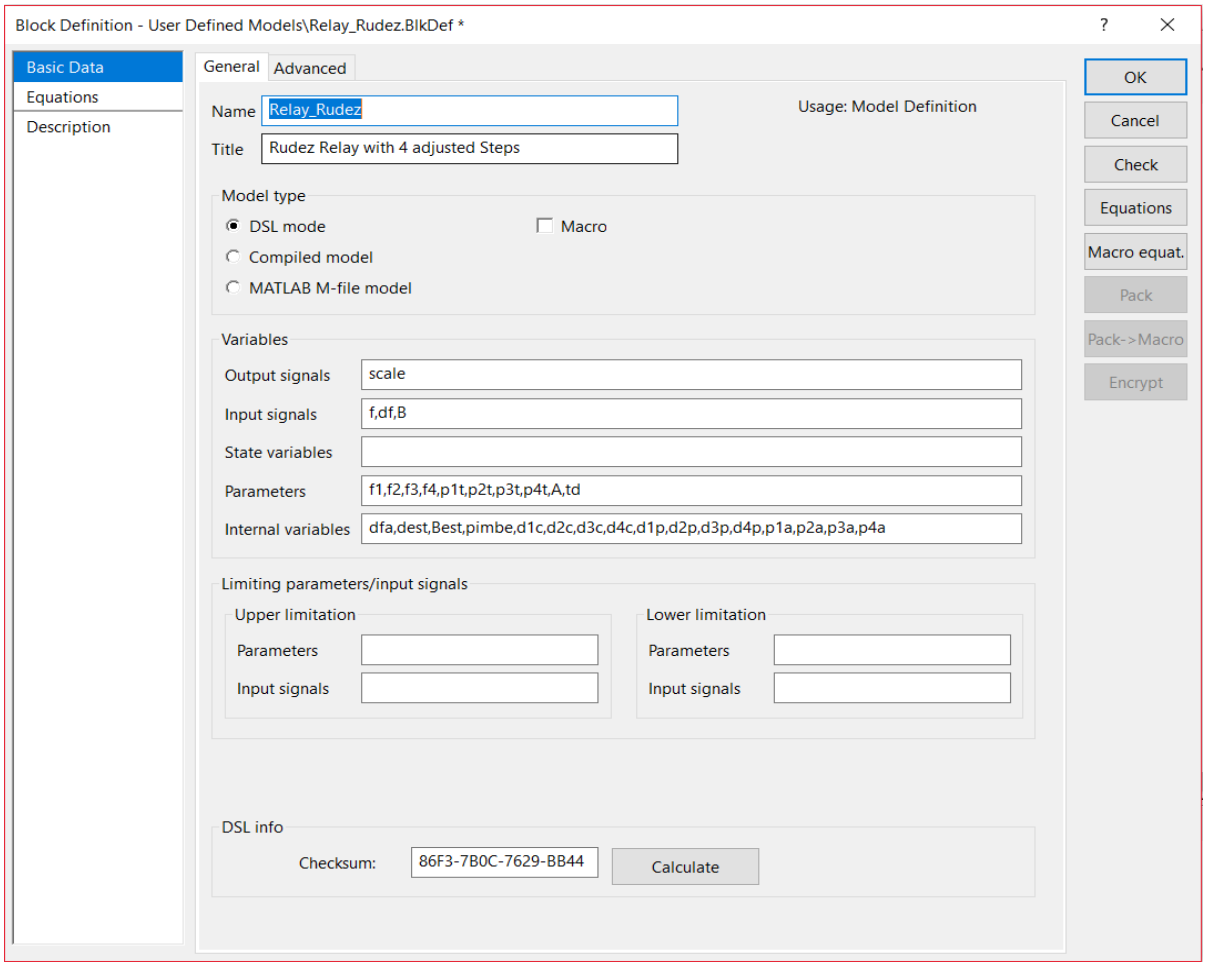


Figure C-1: Screenshot of the adaptive UFLS relay on PowerFactory

The output signal corresponds to a number between 0 and 1 showing how much all the loads have to be reduced, while the input signals correspond to the frequency, the ROCOF and the term B computed through equation (3-6).

In the “Equation” page, the mechanics of the UFLS relay with step adjustments are written in DSL as shown below:

```
dfa=select(time()>=0,df,0)! To avoid the initial error
dest=aflipflop(dfa,time()>=0.2,0)
Best=aflipflop(B,time()>=0.2,0)
pimbe=(dest*A+Best)/100
! d1c,d2c,d3c d4c are changes of df/dt [Hz/s] between steps
d1c=aflipflop(dfa-dest*1,f<=f1,0)*flipflop(f<=f1,0)
d2c=aflipflop(dfa-dest*(1-p1t),f<=f2,0)*flipflop(f<=f2,0)
d3c=aflipflop(dfa-dest*(1-p1t-p2t),f<=f3,0)*flipflop(f<=f3,0)
d4c=aflipflop(dfa-dest*(1-p1t-p2t-p3t),f<=f4,0)*flipflop(f<=f4,0)
! d1p,d2p,d3p d4p are fraction of change w.r.t. estimated df/dt
d1p=d1c/(0-dest)
d2p=d2c/(0-dest)
d3p=d3c/(0-dest)
```

```
d4p=d4c/(0-dest)
! Adjusted load tripping amount
p1a=max((p1t-d1p),0)*flipflop(f<=f1,0)
p2a=max((p2t-d2p),0)*flipflop(f<=f2,0)
p3a=max((p3t-d3p),0)*flipflop(f<=f3,0)
p4a=max((p4t-d4p),0)*flipflop(f<=f4,0)
scale=delay(lim_const(1+pimbe*(lim_const(p1a,0,1)+lim_const(p2a,0,1)+lim_const(p3a,0,1)+lim_const(p4a,0,1)),
0,1),td)
inc(f)=60
inc(df)=0
inc(scale)=1
```

Notes:
The comments are in green and the standard functions of PowerFactory are shown in blue.
The explanation of the PowerFactory standard functions follows based on the manual.

Standard function: select(booleanexpr, x, y)
Returns x if booleanexpr is true, else returns y.

Standard function: time()
Returns the current simulation time.

Standard function: aflipflop(x, boolset, boolreset)
'Analog' flipflop function, that returns the old or the current value of x.
Returns the old value of x if boolset=1 and boolreset=0.
Else returns the current value of x.

Standard function: flipflop(boolset, boolreset)
Logical flipflop function, that returns the internal state (0 or 1).
Changes from 0 to 1 if boolset=1 and boolreset=0.
Changes from 1 to 0 if boolset=0 and boolreset=1.

Standard function: max(x,y)
Returns the larger value between x and y.

Standard function: delay(x, Tdelay)
Delay function, that stores the value x(Tnow) and returns the value x(Tnow-Tdelay). Tdelay must be given in seconds and must be a time independent constant. If it is smaller than the integration step size, the latter is used.

Standard function: lim_const(x, min, max)
Min and max have to be time-independent constants. Usage of is encouraged lim_const against lim for performance reasons.
Returns min if x<min
Returns min if x>max
Else returns x

Standard function: inc(x)
Initialize the value of the signal x.

Reference list

[1] B. Zeng, S. Ouyang, J. Zhang, H. Shi, G. Wu and M. Zeng, "An analysis of previous blackouts in the world: Lessons for China's power industry," *Renewable and Sustainable Energy Reviews 42,* pp. 1151-1163, 2015.

[2] P. Hines, J. Apt and S. Talukdar, "Large blackouts in North America: Historical trends and policy implications," *Energy Policy 37,* pp. 5249-5259, 2009.

[3] M. Zhang, "Modeling and Verification of advanced Under-Frequency Load Shedding Schemes," Delft, 2016.

[4] K. Shun, K. Hur and P. Zhang, "A New Unified Scheme for Controlled Power System Separation Using Synchronized Phasor Measurements," *IEEE Transactions on Power Systems,* vol. 26, pp. 1544-1554, 2011.

[5] U. Rudez and R. Mihalic, "Analysis of Underfrequency Load Shedding Using a Frequency Gradient," *IEEE Transactions on Power Delivery,* vol. 26, no. 2, pp. 565-575, 2011.

[6] L. Ding, F. M. Gonzalez-Longatt, P. Wall and V. Terzija, "Two-Step Spectral Clustering Controlled Islanding Algorithm," *IEEE Transactions on Power Systems,* vol. 28, no. 1, pp. 75-84, 2013.

[7] R. J. Sánchez-García, M. Fennelly, S. Norris, N. Wright, G. Niblo, J. Brodzki and J. W. Bialek, "Hierarchical Spectral Clustering of Power Grids," *IEEE Trans. Power Systems,* vol. 29, no. 5, pp. 2229-2237, 2014.

[8] U. Rudez and R. Mihalic, "Monitoring the First Frequency Derivative to Improve Adaptive Underfrequency Load Shedding," *IEEE Transactions on Power Delivery,* vol. 26, no. 2, pp. 839-846, 2011.

[9] U. Rudez and R. Mihalic, "WAMS-Based Underfrequency Load Shedding With Short-Term Frequency Prediction," *IEEE Transactions on Power Delivery,* vol. 31, no. 4, pp. 1912-1920, 2016.

[10] J. Tang, J. Liu, F. Ponci and A. Monti, "Adaptive Load Shedding Based on Combined Frequency and Voltage Stability Assessment Using Synchrophasor Measurements," *IEEE Transactions on Power Systems,* vol. 28, no. 2, pp. 2035-2047, 2013.

[11] J. H. Chow, Power System Coherency and Model Reduction, Springer, 2013.

[12] P. W. Sauer and M. A. Pai, Power System Dynamics and Stability, Stipes Publishing, 2007.

[13] U. von Luxborg, "A tutorial on spectral Clustering," *Stat Comput,* vol. 17, no. 4, pp. 395-416, 2007.

[14] P. Kundur, Power System Stability and Control, New York: McGraw-Hill, 1994.

[15] A. Esmaeilian and M. Kezunovic, "Controlled Islanding to Prevent Cascade Outages Using Constrained Spectral k-Embedded Clustering," *Power Systems Computation Conference (PSCC),* pp. 1-6, 2016.

[16] (ICSEG), Illinois Center for a Smarter Electric gr, "http://icseg.iti.illinois.edu/ieee-118-bus-system," [Online].

[17] P. M. Anderson and A. A. Fouad, Power System Control and Stability, Wiley.

[18] X. Wang and I. Davidson, "Flexible constrained spectral clustering," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, Washington DC, USA, 2010.

[19] N. Selvakkumaran and G. Karypis, "Multi-Objective Hypergraph Partitioning Algorithms for Cut and Maximum Subdomain Degree Minimization," *IEEE Transactions on Computer Aided Design,* vol. 25, no. 3, pp. 504-517, 2006.

[20] S. S. Rangapuram, P. K. Mudrakarta and M. Hein, "Tight Continuous Relaxation of the Balanced k-Cut Problem," *Advances in Neural Information Processing Systems (NIPS),* 2014.

[21] H. Kuhn and A. Tucker, "Nonlinear Programming," *ACM SIGMAP Bulletin,* pp. 6-18, 1982.

[22] X. Wang, B. Qian and I. Davidson, "On constrained spectral clustering and its applications," *Data Min. Knowl. Discov.,* vol. 28, no. 1, pp. 1-30, 2014.

[23] X. Wang, "https://github.com/gnaixgnaw/CSP," [Online].

[24] G. Karypis and V. Kumar, "A fast and Highly Quality Multilevel Scheme for Partitioning Irregular Graphs," *SIAM Journal on Scientific Computing,* vol. 20, no. 1, pp. 359-392, 1999.

[25] Karypis Lab, "http://glaros.dtc.umn.edu/gkhome/metis/hmetis/download," University of Minnesota. [Online].

[26] Machine Learning Group, "http://www.ml.uni-saarland.de/code.htm," Department of Mathematics and Computer Science - Saarland University. [Online].

[27] J. Bialek, "Tracing the flow of electricity," *IEEE Proceedings on Generation, Transmission and Distribution,* vol. 143, no. 4, pp. 313-320, 1996.

[28] J. J. Grainger and W. D. Steveson, Power System Analysis, New York: McGraw-Hill, 1994.

[29] L. Li, X. Wang, H. Hu and X. Yang, "Speed Condition Monitoring of Rotating Machinery," in *9th International Conference on Sensing Technology (ICST)*, Auckland, 2015.

[30] G. Karypis and V. Kumar, hMETIS* A Hypergraph Partitioning Package, Minneapolis: University of Minnesota, 1998.