Electrical Power and Energy Systems 63 (2014) 687–694

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

Constrained spectral clustering based controlled islanding

Lei Ding [a,*], Peter Wall [b], Vladimir Terzija [a,b]

[a] Key Laboratory of Power System Intelligent Dispatch and Control, Shandong University, Ministry of Education, Jinan 250061, China
[b] The School of Electrical and Electronic Engineering, The University of Manchester, Manchester M13 9PL, UK

ARTICLE INFO

Article history:
Received 12 August 2013
Received in revised form 31 May 2014
Accepted 3 June 2014

Keywords:
Constrained spectral clustering
Controlled islanding
Graph theory

ABSTRACT

Controlled islanding, which splits the whole power system into islands, is an effective way of limiting blackouts during severe disturbances. Calculating islanding solutions in real time is difficult because of the combinatorial explosion of the solution space occurs for large power system. This paper proposes a computationally efficient controlled islanding algorithm that uses *constrained spectral clustering*. An undirected edge-weighted graph is constructed based on absolute values of active power flow and constraints related to transmission line availability and coherent generator groups are included by modifying the edge weights of the graph and using a subspace projection. Spectral clustering is then applied to the constrained solution subspace to find the islanding solution. To improve the clustering quality, a pre-processing procedure is used to detect and eliminate outliers in the eigenvectors of the graph before clustering. A robust *k-medoids* algorithm, which is less sensitive to outliers than the traditional *k-means* algorithm, is then used for clustering. Simulation results show that the proposed algorithm is computationally efficient when solving a controlled islanding problem in real-time.

Introduction

The blackout of a modern power system is a rare, but potentially catastrophic event, in both social and economic terms, that can be initiated by severe system disturbances. It has been shown that by separating a power system into a group of smaller power systems, or *islands*, during the initiation of a blackout that the blackout could be reduced in size or even completely averted [1].

This separation, or *islanding*, of a power system must be performed intelligently or it may further destabilize the system, potentially intensifying the blackout that it hoped to prevent. The real-time intelligent selection, and creation, of islands is referred to as *Controlled Islanding* and is primarily classed as a constrained combinatorial optimization problem. The objective function of the controlled islanding problem is usually *minimal power imbalance*, or minimizing the sum of absolute values of power flow, *minimal power-flow disruption*, whilst the constraints are limited to coherent generator groups [1,2].

To solve this combinatorial optimization problem, many methods have been proposed in the literature. In [1,2], the *Ordered Binary Decision Diagram* (OBDD) method and the *Breadth First Search* (BFS) and *Depth First Search* (DFS) algorithms are proposed to find the boundary that separates coherent generator groups, with minimal power imbalance. However, to satisfy the computational needs of a real-time application all of these search algorithms must be applied to significantly simplified network models. For example, when using an OBDD-based method the network model must be simplified to contain less than approximately 40 nodes [3]. Simplifying the original network to this extent removes potential solutions from the network model that may have been superior to the solutions that exist in the simplified network model.

The Angle Modulated Particle Swarm Optimization (AMPSO) algorithm is also proposed to solve the controlled islanding problem with minimal power imbalance [4]. In this application of the AMPSO the connectivity of the sub-graphs is neglected to improve computational efficiency; unfortunately, this make the solution produced vulnerable to containing isolated buses. In [5], the slow coherency algorithm is extended to load buses to find the weak connections between coherent areas and an OBDD-based method is then used to search for splitting solutions of these weak connections. This allows a dramatic improvement in computational efficiency because the solution space is limited to a small subset of the original network. However, the identification of weak connections focuses solely on dynamic coupling, and does not reflect the power flow between coherent areas. This focus on dynamic coupling produces results, in simulations, that show that the minimum power imbalance solution tends to be located near these weak connections, but there is currently no theoretical proof that quantifies this tendency.

* Corresponding author. Tel.: +86 531 81696132.
E-mail address: dinglei@sdu.edu.cn (L. Ding).

http://dx.doi.org/10.1016/j.ijepes.2014.06.016
0142-0615/

In an attempt to satisfy the requirement of computational efficiency, some researchers also investigate the heuristic methods [6–8], graph search methods [9,10] and spectral methods [11–14]. Spectral clustering has been widely used to solve combinatorial optimization problems and graph-cut problems. The major advantage of this method is that it offers a deterministic solution within polynomial time [11]. The ability to produce deterministic solutions in polynomial time is of particular benefit when dealing with large power systems that other existing methods cannot cope with due to combinatorial explosion. However, the graph that spectral clustering applied on must be undirected and the direction of power flow is omitted. As a consequence, the objective function of controlled islanding problem is changed to minimal power-flow disruption and can be solved efficiently.

Spectral clustering methods proposed in [13,14] have excellent computational efficiency, but fail to consider the constraints that are related to generator coherency. This neglect of generator behavior means that the stability of the islands produced cannot be guaranteed. In addition, direct application of spectral clustering without constraints often leads to simply separating one node from the rest of the graph. The flaws in these two solution types are clearly unacceptable when attempting to use the controlled islanding of a power system to prevent a blackout [15].

In this paper, a constrained spectral clustering based controlled islanding algorithm (**CSCCI** algorithm) that allows islanding solutions to be found in real-time, without simplifying the original network, will be presented. This algorithm offers high speed performance when splitting large power systems. The algorithm is based on converting the optimization problem into a graph-cut problem by constructing a graph for the objective function of minimal power-flow disruption. Information regarding the coherent generator groups and the availability of transmission lines is introduced into the graph-cut problem as constraints. These constraints are implemented by using a subspace projection and modifying the edge weights of the graph. Spectral clustering can be applied to this graph to find an islanding solution.

The content of this paper is arranged as follows. Section 'Controlled islanding and spectral clustering' introduces the mathematics that defines the controlled islanding problem and the basic theory behind spectral clustering. Section 'Constrained spectral clustering' details the execution of the proposed **CSCCI** algorithm. Section 'Simulation' demonstrates the performance of the **CSCCI** algorithm when it is applied to the IEEE 39-bus and 118-bus test systems. Section 'Conclusion' gives some concluding remarks.

Controlled islanding and spectral clustering

In this Section, the controlled islanding problem is defined as a constrained combinatorial optimization problem and a graph-cut problem. A possible method for solving this optimization problem, spectral clustering, is then introduced.

Controlled islanding

Many factors must be considered when attempting to form stable islands e.g. load-generation balance, generator coherency, availability of transmission lines, thermal limits, transient stability [1,2]. The controlled islanding problem is, in mathematical terms, a multi-objective and multi-constraint problem which is very complicated to solve. It is very likely that a solution satisfying all of these objectives and constraints does not exist. So, stable islanding cannot usually be achieved solely through controlled islanding, coordinated load shedding and other corrective measures must also be included. A practical method for solving this problem involves considering only a small sub-set of factors, such as load-generation balance and generator coherency, to produce a set of feasible candidate islanding solutions that can then have a more thorough treatment applied to them [1–5]. Reducing the number of constraints applied to the controlled islanding problem reduces the complexity of the problem, which is beneficial when dealing with large power systems.

The controlled islanding problem can be represented as a constrained combinatorial optimization, with the objective function given in (1). This objective function minimizes the sum of the absolute values of the active power flow between islands, also called *power-flow disruption*. The constraints applied when satisfying this objective function deal with coherent generator groups and transmission line availability.

$$\min_{\mathbf{V_1},\mathbf{V}_2\subset\mathbf{V}}\left(\sum_{i\in\mathbf{V_1},j\in\mathbf{V_2}}|P_{ij}|\right)\quad\text{with the condition}$$
$$\mathbf{V_{G1}}\subset\mathbf{V_1},\mathbf{V}_{G2}\subset\mathbf{V_2},\mathbf{E_s}\neq\mathbf{0}\quad\text{and}\quad\mathbf{E_s}\subset\mathbf{E}\tag{1}$$

where $|P_{ij}|$ denotes the absolute value of the active power flow on the transmission line between node i and j.

Eq. (1) is expressed based on the graph-model $\mathbf{G}(\mathbf{V},\mathbf{E},\mathbf{W})$ of an n-bus power network. In this graph-model, the node set $\mathbf{V}=\{v_1,\ldots,v_n\}$ and the edge set **E**, with elements e_{ij} (i, j = 1,..., n), denote the buses and transmission lines, respectively. The matrix W is a set of edge weights. $\mathbf{E_s}$ is an edge subset that contains all the transmission lines that cannot be removed.

The graph **G** can be partitioned into two sub-graphs $\mathbf{G}_1(\mathbf{V}_1,\mathbf{E}_1,\mathbf{W}_1)$ and $\mathbf{G}_2(\mathbf{V}_2,\mathbf{E}_2,\mathbf{W}_2)$. $\mathbf{V_1}$ and $\mathbf{V_2}$ are two disjoint subsets representing the buses in islands 1 and 2, while $\mathbf{V_{G1}}$ and $\mathbf{V_{G2}}$ are the corresponding subsets of coherent generators within each island.

Spectral clustering

In graph theory, solving the *graph-cut* problem consists of partitioning the graph **G** into two sub-graphs $\mathbf{G}_1$ and $\mathbf{G}_2$ by removing the edges connecting $\mathbf{G}_1$ and $\mathbf{G}_2$. The set of edges to be removed is called the *cutset* and the sum of the weights of the cutset is called *cut*, which is defined as:

$$cut(\mathbf{V_1},\mathbf{V_2})=\sum_{i\in\mathbf{V_1},j\in\mathbf{V_2}}w_{ij}\tag{2}$$

Now, the controlled islanding problem (1) is converted into the problem of finding the *cutset* that bisects a graph with minimum *cut*. The *un-normalized spectral clustering* method can be used to solve this problem.

Un-normalized Spectral Clustering clusters the nodes into two subsets based on the *Laplacian Matrix* **L**, which is defined for a graph **G** as [11]:

$$\mathbf{L}=\mathbf{D}-\mathbf{W}\tag{3}$$

where **D** is a diagonal *degree matrix* that contains the diagonal elements D_i which are equal to the total weight of the edges connected to node i. Defined in this way, the edge weight matrix **W** and the Laplacian Matrix **L** are both symmetric for any undirected graph.

The *un-normalized spectral clustering algorithm*, for the case of bisection, can then be described using the following steps [11]:

(1) Compute the first two eigenvectors ϑ_1, ϑ_2 of the Laplacian matrix **L**.
(2) Let $\mathbf{J}\in\mathbf{R}^{n\times 2}$ be the matrix containing the vectors ϑ_1, ϑ_2 as columns. Let $y_i\in\mathbf{R}^2$ be the vector corresponding to the i-th row of **J**.
(3) Cluster the nodes $y_i\in\mathbf{R}^2$ into clusters c_1, c_2 using a clustering algorithm, such as the *k-means* algorithm.