

Discovering Clusters in Power Networks from Orthogonal Structure of Spectral Embedding

Tyuryukanov, Ilya; Popov, Marjan; van der Meijden, Mart; Terzija, Vladimir

DOI

[10.1109/TPWRS.2018.2854962](https://doi.org/10.1109/TPWRS.2018.2854962)

Publication date

2018

Document Version

Final published version

Published in

IEEE Transactions on Power Systems

Citation (APA)

Tyuryukanov, I., Popov, M., van der Meijden, M., & Terzija, V. (2018). Discovering Clusters in Power Networks from Orthogonal Structure of Spectral Embedding. *IEEE Transactions on Power Systems*, 33(6), 6441-6451. <https://doi.org/10.1109/TPWRS.2018.2854962>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

<https://www.openaccess.nl/en/you-share-we-take-care>

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

Discovering Clusters in Power Networks From Orthogonal Structure of Spectral Embedding

Ilya Tyuryukanov , *Student Member, IEEE*, Marjan Popov , *Senior Member, IEEE*,
Mart A. M. van der Meijden, *Member, IEEE*, and Vladimir Terzija, *Fellow, IEEE*

Abstract—This paper presents an integrated approach to partition similarity graphs, the task that arises in various contexts in power system studies. The approach is based on orthogonal transformation of row-normalized eigenvectors obtained from spectral clustering to closely fit the axes of the canonical coordinate system. We select the number of clusters as the number of eigenvectors that allows the best alignment with the canonical coordinate axes, which is a more informative approach than the popular spectral eigengap heuristic. We show a link between the two relevant methods from the literature and on their basis construct a robust and time-efficient algorithm for eigenvector alignment. Furthermore, a graph partitioning algorithm based on the use of aligned eigenvector columns is proposed, and its efficiency is evaluated by comparison with three other methods. Finally, the proposed integrated approach is applied to the adaptive reconfiguration of secondary voltage control helping to achieve demonstrable improvements in control performance.

Index Terms—Power network partitioning, spectral clustering, number of clusters, adaptive network zone division.

I. INTRODUCTION

WITH massive deployment of renewable generation, the increase of uncertainties and reduction of security margins are expected to become the major obstacles for the safe operation of modern electric power systems. It is widely accepted that coping with these challenges requires new approaches for power system protection, operation, planning and control.

Partitioning (clustering, zoning) of electric power networks is a concept that appears particularly frequently in many advanced control and protection techniques. Cotilla-Sanchez *et al.* [1] mention a large number of existing applications

in planning and operations of power systems that require zone definitions and propose a partitioning method that helps to reduce transaction leakage between zones. Additionally, the high computational burden of many methods used in planning and control of large-scale power systems motivates the identification of weakly-interacting areas for the purpose of power network reduction [2]. Due to the local nature of voltage deviations in AC power systems, zones and areas are widely used in SVC [3], [4] and some other applications related to voltage-var control [5]. In Ding *et al.* [6], [7], network partitioning is used for intentional controlled islanding both to determine the coherent generators and split the network. In general, decoupled control of partitioned electric power networks appears to be a promising strategy for dealing with the anticipated complexity of future power grids [8].

Spectral clustering is an important approach that is extensively used for partitioning of electric power networks. Due to its strong theoretic foundations [9], spectral clustering can be useful for a variety of power system studies [2], [8], [10], [11]. However, the high computational efficiency makes it especially suitable for the applications requiring a time-constrained solution in response to the changes in the network. Among such applications, adaptive SVC [4], [10] has attracted a growing attention during the recent years due to the increasingly dynamic and interconnected structure of modern power grids. The use of adaptive SVC has been reported in [4] for the rapidly developing electric power grids of China. The approach in [4] uses the concept of “Var control space” to select the number and location of voltage-regulated *pilot buses* through the subdivision of the whole network into voltage control zones. As pointed out in [10], this type of approach is more viable for the real-time operation as opposed to the direct search for pilot nodes [12], [13].

The main contribution of this paper is a spectral clustering based approach for the selection of the number of clusters and high quality partitioning of similarity graphs arising in various contexts in power systems. Our methodology combines and extends the ideas from [14], [15] that showed the benefits of applying specially computed orthogonal transformations to the eigenvectors of the normalized graph adjacency matrix. The paper suggests to choose the number of clusters for power network partitioning as the number of graph matrix eigenvectors that allows the closest alignment with the canonical axes and proposes a robust and time-efficient algorithm to recover the aligning orthogonal transformation. We also design an algorithm based on transformed eigenvectors that partitions the underlying power

Manuscript received September 12, 2017; revised February 5, 2018 and June 25, 2018; accepted June 30, 2018. Date of publication July 11, 2018; date of current version October 18, 2018. This work was supported by the Dutch Scientific Council NWO-STW under the project 408-13-025 within the program of Uncertainty Reduction of Smart Energy Systems (URSES) in collaboration with TenneT TSO and by the Dutch National Metrology Institute, van Swinderen Laboratory. Paper no. TPWRS-01401-2017. (*Corresponding author: Ilya Tyuryukanov.*)

I. Tyuryukanov and M. Popov are with the Department of Electrical Engineering and Computer Science, Delft University of Technology, Delft 2628CD, The Netherlands (e-mail: ilya.tyuryukanov@ieee.org; m.popov@ieee.org).

M. A. M. van der Meijden is with the TenneT TSO B.V., Utrechtseweg 310, Arnhem 6812AR, The Netherlands, and also with the Department of Electrical Engineering and Computer Science, Delft University of Technology, Delft 2628CD, The Netherlands (e-mail: mart.vander.meijden@tennet.eu).

V. Terzija is with the School of Electrical and Electronic Engineering, the University of Manchester, Manchester M13 9PL, U.K. (e-mail: terzija@ieee.org).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TPWRS.2018.2854962

network into a set of well-separated clusters. This algorithm shows a high computational efficiency and partitioning quality, while being able to ensure the connectedness of the resulting clusters. In addition, clusters can be biased to contain more than a certain amount of nodes. The fast running time of the proposed clustering framework makes it potentially useful in assisting the real-time decision making in power systems. We illustrate this point on the example of adaptive zone division (AZD) for SVC, an application that may greatly benefit from the ability of our framework to determine the optimum number of zones.

The rest of the paper is organized as follows. Section II introduces the essential preliminaries for this work. Section III outlines the use of orthogonal linear transformations with spectral clustering. Section IV describes the proposed algorithm for robust eigenvector alignment. Section V introduces the used clustering quality metrics. Section VI details the proposed k -way partitioning algorithm. Section VII demonstrates its superior partitioning quality. Section VIII illustrates how the proposed clustering framework can be used in SVC for the task of AZD. Finally, the conclusions are drawn in Section IX.

II. BASIC CONCEPTS

A. Mathematical Notation

Let a power network consisting of n nodes be modeled as an edge-weighted undirected graph G with the associated weighted adjacency matrix $\mathbf{A} = [a_{ij}]$. The edge weights a_{ij} should correspond to a quantity that represents the closeness or similarity between two nodes. The nodes of G are referred to by the indices of the corresponding rows and columns in the adjacency matrix. Sizes of matrices are denoted by subscripts (e.g., $\mathbf{M}_{n \times k}$ for a matrix with n rows and k columns) or introduced when the matrix is defined (e.g., $\mathbf{M} \in \mathbb{R}^{n \times k}$). A submatrix is defined by the indices of the participating rows and columns (e.g., $\mathbf{M}[1, \dots, n; 1, \dots, 3]$ is formed by the first n rows and three columns of $\mathbf{M}$).

Following Luxburg [9], we define $\mathbf{1}_C$ as the indicator vector of the nodes belonging to the cluster or connected component C . That is, $\mathbf{1}_C = [f_1, \dots, f_n]^T$ and $f_i = 1$ if node i belongs to C and $f_i = 0$ otherwise. To denote an all-ones matrix, $\mathbf{1}$ is used, and an all-zeros matrix is denoted by $\mathbf{0}$. The identity matrix of size k is denoted as $\mathbf{I}_k$. A diagonal matrix formed from a vector argument is denoted as $\text{diag}(\cdot)$.

Final solutions of optimization algorithms are marked with an asterisk (e.g., $\mathbf{R}^*$). Partition indicator matrices such as discretized eigenvector matrices are marked with a tilde (e.g., $\widetilde{\mathbf{M}}$). Lower and upper limits of a range of numbers are marked with underbars and overbars respectively (e.g., $\underline{k}$ and $\bar{k}$).

B. Basics of Spectral Clustering

Given the above definitions for the graph G , the *normalized adjacency matrix* of G can be defined as

$$\mathbf{A}_n = \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \quad (1)$$

where $d_i = \sum_{j=1}^n a_{ij}$ is the weighted degree of node i , $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$ is the diagonal *degree matrix* of the graph G and $\mathbf{D}^{-\frac{1}{2}} = \text{diag}(\frac{1}{\sqrt{d_1}}, \dots, \frac{1}{\sqrt{d_n}})$.

While the normalized adjacency matrix (1) is used in [14]–[16], several authors [8], [9] mention an alternative matrix $\mathbf{L}_n = \mathbf{I} - \mathbf{A}_n$, commonly referred to as the *normalized Laplacian*. Both matrices have the same eigenvectors, and the smallest eigenvalues of $\mathbf{L}_n$ correspond to the largest eigenvalues of $\mathbf{A}_n$ [16]. The choice of $\mathbf{A}_n$ is motivated by numerical considerations: computing several largest eigenpairs of a sparse matrix with iterative eigensolvers has better numerical properties than computing several smallest eigenpairs. Additionally, the symmetry of $\mathbf{A}_n$ and $\mathbf{L}_n$ is beneficial, as eigenvector computations for symmetric matrices are more robust numerically.

The normalized adjacency matrix has the following important properties [9], [15], [16]:

- 1) The eigenvalues of $\mathbf{A}_n$ are real and satisfy the inequality $-1 \leq \lambda_i \leq 1$, $i = 1, \dots, n$;
- 2) 1 is an eigenvalue of $\mathbf{A}_n$, and its multiplicity is equal to the number of connected components of G ;
- 3) The eigenspace of 1 is spanned by the k column vectors $\mathbf{D}^{\frac{1}{2}} \mathbf{1}_{C_i}$, where $C_1, \dots, C_k$ represent the k connected components of G .

The k largest eigenvectors of $\mathbf{A}_n$ can be combined into the matrix $\mathbf{X} \in \mathbb{R}^{n \times k}$. The rows of $\mathbf{X}$ can be seen as the coordinates of the nodes of the original power network in $\mathbb{R}^k$. This representation of the nodes of the original network by the points in the Euclidean space formed by the first k eigenvector coordinates is often called *spectral embedding* [8], [9].

The third property motivates the use of the largest eigenvectors of $\mathbf{A}_n$ for clustering purposes. If G has k connected components, the rows of $\mathbf{X}$ will lie along the axes of the canonical coordinate system in $\mathbb{R}^k$, and the k connected components can be easily retrieved from $\mathbf{X}$. The multiple connected components of G can also be considered as perfectly separated clusters. According to matrix perturbation theory [9], the addition of some low-weight edges between the k perfectly separated clusters only slightly perturbs the k largest eigenvectors from their ideal values. Thus, an observation can be made [9], [15], [16] that the more the first k eigenvectors resemble the ideal structure corresponding to fully separated clusters, the more closely those eigenvectors represent the dominant clustering structure of G .

In practice, the rows of the eigenvector matrix $\mathbf{X}$ are normalized to have length one [14], [16]. Therefore, it is convenient to introduce the matrix $\mathbf{Y} \in \mathbb{R}^{n \times k}$ that is obtained from $\mathbf{X}$ by normalizing the rows of $\mathbf{X}$ to have length one.

$$Y_{ij} = X_{ij} / \left(\sum_{j=1}^k X_{ij}^2 \right)^{1/2} \quad (2)$$

The most common final step of spectral clustering is to assign each row of $\mathbf{Y}$ to a fixed cluster. This procedure is commonly referred to as *discretization*, and its result can be thought of as a conversion of real-valued $\mathbf{Y}$ into a *discrete* matrix $\tilde{\mathbf{Y}} \in \{0, 1\}_{n \times k}$ with the property $\tilde{\mathbf{Y}} \mathbf{1}_{k \times 1} = \mathbf{1}_{n \times 1}$.

III. ORTHOGONAL INVARIANCE OF SPECTRAL CLUSTERING

A. Alignment Cost Minimization

In their seminal work, Yu and Shi [14] have shown that the optimal solution of the continuous relaxation of the NP-complete

normalized cut problem for k clusters can be represented by the columns of the normalized eigenvector matrix $\mathbf{Y} \in \mathbb{R}^{n \times k}$. Moreover, it was shown in [14] that this solution is invariant with respect to orthogonal linear transformations applied to the initial matrix $\mathbf{Y}$. That is, the continuous optima form a subspace characterized as

$$\{\mathbf{Y}\mathbf{R} : \mathbf{R}^T \mathbf{R} = \mathbf{I}_k\} \quad (3)$$

where $\mathbf{R} \in \mathbb{R}^{k \times k}$ is an arbitrary orthogonal matrix.

Furthermore, the authors of [14] propose an algorithm to find an orthogonal transformation $\mathbf{R}^*$ that would facilitate the discovery of a good discrete solution $\tilde{\mathbf{Y}}$ from the initial optimal solution of the continuous relaxation $\mathbf{Y}$. The discretization algorithm is stated as the optimization problem

$$\begin{aligned} \text{minimize} \quad & Q(\tilde{\mathbf{Y}}, \mathbf{R}) = \|\tilde{\mathbf{Y}} - \mathbf{Y}\mathbf{R}\|_F \\ \text{s.t.} \quad & \tilde{\mathbf{Y}} \in \{0, 1\}^{n \times k}, \quad \tilde{\mathbf{Y}} \mathbf{1}_{k \times 1} = \mathbf{1}_{n \times 1}, \\ & \mathbf{R}^T \mathbf{R} = \mathbf{I}_k \end{aligned} \quad (4)$$

where $\|\cdot\|_F$ is the matrix Frobenius norm: $\|\mathbf{M}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^k M_{ij}^2}$.

The problem (4) has two unknowns: the discrete solution $\tilde{\mathbf{Y}}$ and the orthogonal matrix $\mathbf{R}$ that brings $\mathbf{Y}$ closest to $\tilde{\mathbf{Y}}$. As there is no direct method to solve (4) simultaneously for $\tilde{\mathbf{Y}}$ and $\mathbf{R}$, an iterative procedure was proposed in [14]. If $\mathbf{R}$ is given in (4), $\tilde{\mathbf{Y}}$ is determined by non-maximum suppression of $\mathbf{Y}\mathbf{R}$, that is by setting the maximum entry of each row of $\mathbf{Y}\mathbf{R}$ to 1 and the remaining entries to zero. If $\tilde{\mathbf{Y}}$ is given, $\mathbf{R}$ is determined by *singular value decomposition* (SVD) of $\tilde{\mathbf{Y}}^T \mathbf{Y}$

$$\begin{aligned} \mathbf{R} &= \mathbf{U}\mathbf{V}^T \\ \tilde{\mathbf{Y}}^T \mathbf{Y} &= \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \end{aligned} \quad (5)$$

where $\mathbf{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_k)$ is the diagonal matrix of singular values of $\tilde{\mathbf{Y}}^T \mathbf{Y}$ and $\mathbf{U}$, $\mathbf{V}$ are the matrices of the left and right singular vectors respectively.

The iterative approach in [14] consists of alternating the steps of optimal alignment (5) and non-maximum suppression that rapidly converge to the initialization-dependent local optimum of (4). By analyzing the cost function (4), it is possible to notice that it has the goal of maximizing one entry per matrix row to be close to one, while minimizing the remaining entries, by applying a single orthogonal transformation $\mathbf{R}$ on the input matrix $\mathbf{Y}$. Geometrically this corresponds to the alignment of the initial spectral embedding with the axes of the canonical coordinate system. A cost function that expresses the degree of alignment of spectral embedding with the canonical coordinate system is further called *alignment cost*.

B. Eigenvector based Selection of Number of Clusters

The concept of alignment cost was used by Zelnik-Manor [15] to select the number of eigenvectors that most closely resembles the ideal result of spectral clustering discussed in Section II-B. To enable comparison with (4), the cost function of method [15] is directly given in terms of row-normalized eigenvectors instead of the original formulation [15] in terms of *unnormalized*

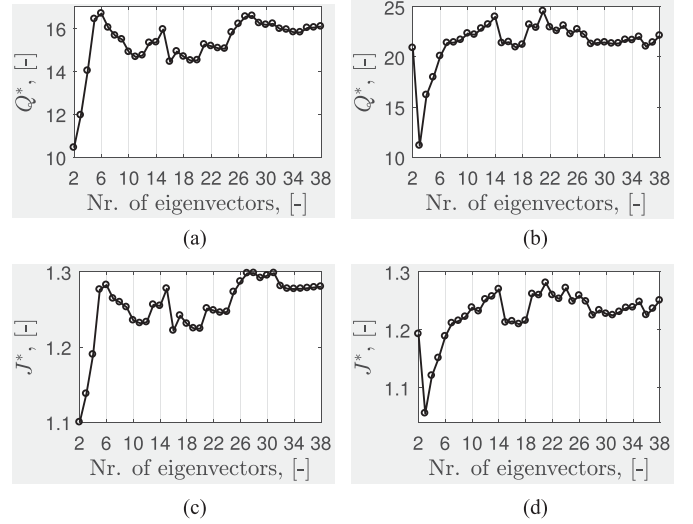


Fig. 1. Minimized alignment costs (4) and (6) for branch admittance graphs of the two networks from MATPOWER [17], [18] with all transformer phase shifts set to zero. (a) Case1354pegase test network. (b) Case2869pegase test network. (c) Case1354pegase test network. (d) Case2869pegase test network.

eigenvectors $\mathbf{X}$:

$$\begin{aligned} \text{minimize} \quad & J(\mathbf{R}) = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^k \frac{[\mathbf{Y}\mathbf{R}]_{ij}^2}{M_i^2} \\ \text{s.t.} \quad & \mathbf{R}^T \mathbf{R} = \mathbf{I}_k \end{aligned} \quad (6)$$

where $M_i = \max_j [\mathbf{Y}\mathbf{R}]_{ij}$.

The cost function (6) was minimized in [15] by optimizing the orthogonal matrix $\mathbf{R}$ with gradient descent. The lowest feasible minimum for the cost J is equal to one, and it is achieved when rotation $\mathbf{R}^*$ recovers a discrete matrix from $\mathbf{Y}$. According Section II-B, this case corresponds to the best possible outcome of spectral clustering, as every node is perfectly assigned to one of the k clusters. This observation can be used to select the number of eigenvectors k that, after applying the orthogonal transformation $\mathbf{R}^*$, leaves the lowest ambiguity in the cluster assignment of the graph nodes.

IV. ALIGNMENT OF SPECTRAL EMBEDDING WITH THE STANDARD BASIS

A. Selection of Alignment Cost

The idea to use a measure of eigenvector alignment with the canonical axes for the estimation of the number of clusters was initially introduced in [15] for the cost function (6). However, the results in Fig. 1 demonstrate that the minimization of the alignment costs (4) and (6) discovers a very similar pattern.

For the results in Fig. 1, the cost (4) was optimized with the algorithm of Section IV-C. The minimization of (6) has been implemented by using the ideas of Sections IV-B and IV-C and some additional heuristics to multiply initialize the gradient descent optimization. The gradient-based minimization of (6) uses Givens angles as optimization variables. The number of Givens angles for k eigenvectors is equal to $k(k-1)/2$; i.e., the solution space grows quadratically with the increase of k .

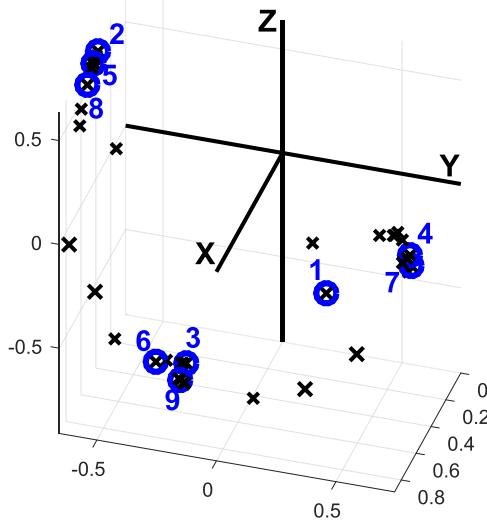


Fig. 2. Selection of starting points by the initialization algorithm for eigenvector alignment. The 3-spectral embedding is computed for the admittance graph in Fig. 4. The circles represent start points, and the numbers represent their selection order.

Therefore, the computation time of the gradient-based eigenvector alignment (6) showed to be noticeably higher, especially as the number of eigenvectors increased. Another issue with the gradient-based minimization is the necessity to choose the learning rate. Thus, the minimization of objective (4) was chosen in this paper to discover the orthogonal structure of spectral k -embeddings.

B. Robust Orthogonal Initialization

Both (4) and (6) are formulated as non-linear non-convex optimization problems. Due to multiple local optima, the achieved final solution generally improves with a good initialization. The authors in [14] use the problem-specific initialization approach from [16], which is a fast greedy algorithm to find a set of k nearly orthogonal rows in the matrix $\mathbf{Y}$. The clustering initialization algorithm [16] is extended here to handle the two important issues:

- 1) Starting the initialization [16] only once may not lead to a good result. Situations are possible, when a set of nearly orthogonal initialization points lies close to a poor local optimum. Therefore, it is desirable to develop a systematic strategy for multiple initializations.
- 2) The rows found by the initialization [16] are generally not perfectly orthogonal to each other, thus the transformation matrix formed by those rows is not strictly orthogonal.

The above issues are resolved in Algorithm 1 which uses at most $\bar{r}$ restarts to robustify the initialization. The restart strategy selects the first vector of the next k -dimensional basis formed from the rows of $\mathbf{Y}$ as the row of $\mathbf{Y}$ that has the minimal cumulative *cosine similarity* to the first vectors of the previously selected bases. As the rows of $\mathbf{Y}$ are normalized by (2), cosine similarity is equivalent to dot product. Rows that

Algorithm 1: Robust Orthogonal Initialization.

Input: $\mathbf{Y}_{n \times k}$, $\bar{r}$, δ
Output: $\mathbf{R}^*$, $\mathbf{Q}^*$

- 1: $\mathcal{S} \leftarrow \{1, \dots, n\}$ // Rows of $\mathbf{Y}$ eligible for basis initialization
- 2: $\mathbf{s} \leftarrow \mathbf{0}_{n \times 1}$ // Cumulative cosine similarity
- 3: **for** $i = 1, \dots, \bar{r}$ **do**
- 4: **if** $\mathcal{S} = \emptyset$ **then break end if**
- 5: $r_1 \leftarrow \operatorname{argmin}_{l \in \mathcal{S}} \mathbf{s}[l]$ // Index of initiating basis row
- 6: $\mathbf{P}[1, \dots, k; 1] \leftarrow \mathbf{Y}[r_1; 1 \dots, k]^T$
- 7: $\mathbf{c} \leftarrow \mathbf{Y}\mathbf{P}[1, \dots, k; 1]$
- 8: $\mathcal{S} \leftarrow \mathcal{S} \setminus \{l \mid \mathbf{c}[l] > \delta\}$
- 9: $\mathbf{s} = \mathbf{s} + \mathbf{c}$
- 10: $\mathbf{c} = \operatorname{abs}(\mathbf{c})$ // Element-wise absolute value
- 11: **for** $j = 2$ to k **do**
- 12: $r_j \leftarrow \operatorname{argmin} \mathbf{c}$ // Most orthogonal to prev. $j-1$ rows
- 13: $\mathbf{P}[1, \dots, k; j] \leftarrow \mathbf{Y}[r_j; 1 \dots, k]^T$
- 14: $\mathbf{c} = \mathbf{c} + \operatorname{abs}(\mathbf{Y}\mathbf{P}[1, \dots, k; j])$
- 15: **end for**
- 16: $\mathbf{R} \leftarrow \operatorname{loewdin}(\mathbf{P})$ //(7)
- 17: $\mathbf{Q} \leftarrow \text{Evaluate (4) as } \mathbf{Q}(\tilde{\mathbf{Y}}, \mathbf{I}) \text{ with } \tilde{\mathbf{Y}} \text{ obtained from } \mathbf{Y}\mathbf{R} \text{ via non-maximum suppression.}$
- 18: Save $\mathbf{Q}$ and $\mathbf{P}$ obtained at each iteration.
- 19: **end for**
- 20: Set $\mathbf{Q}^*$ as the lowest $\mathbf{Q}$ and $\mathbf{R}^*$ as the corresponding $\mathbf{R}$.
- 21: **return** $\mathbf{R}^*$, $\mathbf{Q}^*$

are more similar to any previously selected first basis row than the threshold δ are constrained not to initiate a basis.

The retrieved k rows of $\mathbf{Y}$ (combined into matrix $\mathbf{P}$ in Algorithm 1) may not form an orthonormal basis. For a set of linearly independent vectors, the closest orthonormal basis is given by the SVD-based Loewdin orthogonalization:

$$\begin{aligned} \mathbf{P} &= \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \\ \mathbf{R} &= \mathbf{U}\mathbf{V}^T \end{aligned} \quad (7)$$

The Loewdin orthogonalization (7) is used to transform the retrieved k rows of $\mathbf{Y}$ stored in the columns of $\mathbf{P}$ to a proper orthogonal transformation. Finally, Algorithm 1 evaluates the alignment cost associated with each obtained set of k rows of $\mathbf{Y}$ and returns the best encountered aligning transformation. In addition, all discovered sets of k rows are saved to subsequently provide multiple initializations for the optimization method (4).

A sample run of the restart strategy is illustrated in Fig. 2. The first starting point is selected at random far from the three dense orthogonal clusters. However, the second starting point is selected in the top dense cluster, and all the following starting points are selected in the dense orthogonal clusters. In other words, the proposed restart strategy selects points representative for the orthogonal structure of spectral embedding. The similarity threshold parameter serves as a “step size” that prevents the subsequent starting points from being too close.

Although the proposed algorithm is based on the clustering *initialization* method [16], the added extensions make it a robust

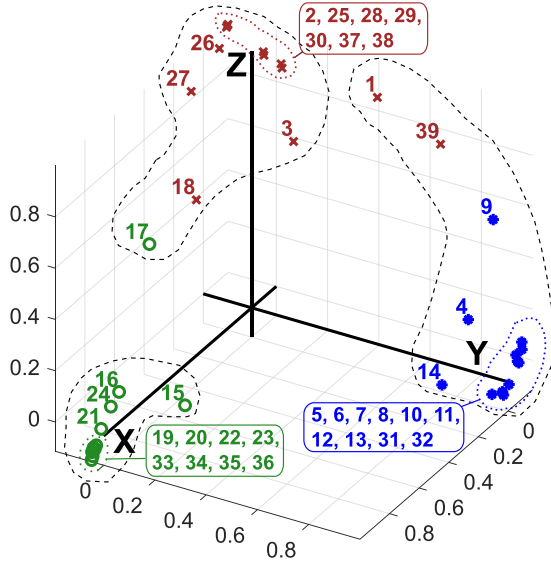


Fig. 3. Aligned 3-spectral embedding for the partitioning study shown in Fig. 4. The different colors and marker shapes represent the result of k -means clustering; the dashed lines show the grouping obtained by our partitioning method.

(i.e., not prone to poor local optima) method to minimize the eigenvector alignment cost. The returned orthogonal transformation often needs only little improvement by the specialized minimization methods (see Section III).

C. Combined Method to Minimize the Alignment Cost

Based on the information given in Sections III and IV-B, we formulate the combined eigenvector alignment algorithm consisting of the following three steps:

- 1) Initialization from previously aligned columns.
- 2) Robust orthogonal initialization.
- 3) Final alignment cost minimization.

The overall philosophy of the proposed three-step algorithm is to apply several efficient methods to sequentially bound the alignment cost and reach a near-global optimum. At first, the alignment cost is reduced by applying the previous orthogonal transformations (accumulated in a matrix) to the next set of row-normalized eigenvectors. This step is mainly included due to its very low computational cost and ability to produce a quick initial bound of the optimization objective. The second step was described in Section IV-B. The third step is the iterative minimization of (4) from multiple initial positions. The multiple restart strategy of Section IV-B supplies initializations of varying quality to the third step, resulting in the overall high-quality optimum. Here it is worth to recall that minimizing (4) is faster than (6), which makes the multiple repeats feasible. Minimizing (6) several times would be significantly more costly, especially for a large number of eigenvectors (above 7–12).

As a graphic illustration of the algorithm's possible outcome, Fig. 3 shows the aligned 3-spectral embedding from which the partitioning result in Fig. 4 has been obtained. Another illustration was already given above in Fig. 1.

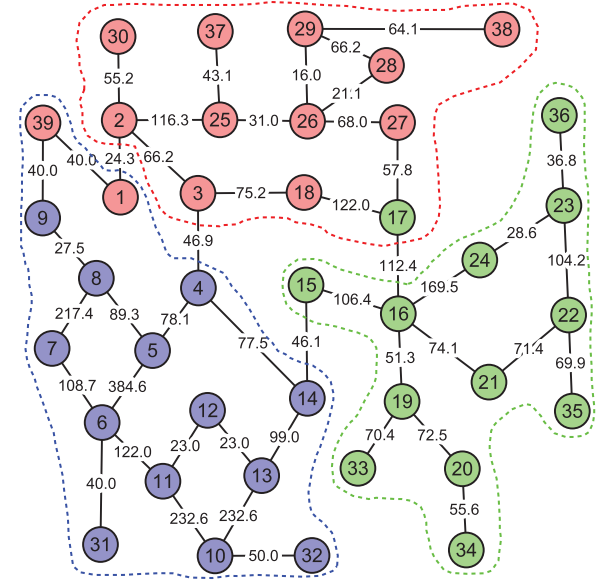


Fig. 4. Branch admittances of the IEEE 39 bus test network, and spectral clustering into three parts with k -means and our methodology. The areas found by k -means are colored differently, and the boundaries of the areas found by our partitioning method are shown with dashed lines.

V. PARTITIONING QUALITY METRICS

For the evaluation of the partitioning algorithm in the next section, we largely adopt the clustering quality evaluation methodology from [8]. First, *cut* and *volume* are introduced for each cluster. The *cut* of cluster C_l represents the total weight of the edges that separate the cluster from the rest of the network, which can be expressed as $cut(C_l, \bar{C}_l) = \sum_{i \in C_l, j \in \bar{C}_l} a_{ij}$. The volume of cluster C_l is the sum of the weighted degrees of its nodes: $vol(C_l) = \sum_{i \in C_l} d_i$. Then the *expansion ratio* (or *expansion*) of cluster C_l [8] is defined as

$$\phi(C_l) = \frac{cut(C_l, \bar{C}_l)}{vol(C_l)} \quad (8)$$

The value of $\phi(C_l)$ can take values from zero to one, with smaller values corresponding to better clusters. The partitioning quality is accessed by the maximal expansion ratio over all clusters [8]

$$\phi_{\max}(C_1, \dots, C_k) = \max_{1 \leq l \leq k} \phi(C_l) \quad (9)$$

Minimizing the cluster expansion (8) promotes a high sum of internal connections (high volume) combined with a low sum of external connections (low cut), which are the desirable properties of a good power network partitioning according to [1]. Asking for a small maximal expansion ratio is reasonable from the power system point of view, because it is usually desirable in practice to avoid any loose clusters. A low value of (9) implies that all clusters are well separated from each other in terms of the similarity relationships determined by the graph adjacency matrix $\mathbf{A}$. Thus (8) and (9) evaluate the solution in terms of the defined power system model itself. At the same time, (9) allows for an efficient optimization via the normalized spectral

clustering [8], while some more complex objective functions can often only be optimized via metaheuristic approaches (e.g., [1]).

The arithmetic mean of expansion ratios of all clusters is known as *normalized cut* [9]. It is widely used to assess the quality of graph partitioning [9], [14] and gives the information about the average quality over all clusters

$$Ncut(C_1, \dots, C_k) = \frac{1}{k} \sum_{l=1}^k \phi(C_l) \quad (10)$$

A good partitioning should also contain no disconnected or too small clusters [1]. In order to account for the latter requirement, the minimal cluster size (as a percentage of the average cluster size) is introduced as a quality indicator.

$$\varepsilon(C_1, \dots, C_k) = \frac{\min_{1 \leq l \leq k} (|C_l|)}{n/k} \cdot 100 \quad (11)$$

We aim to treat cluster sizes separately from the partitioning quality measures (9) and (10). While (9) and (10) should be ideally as low as possible, (11) is only meant to be higher than a certain predefined minimal cluster size.

VI. EIGENVECTOR ALIGNMENT BASED PARTITIONING

Apart from providing good indicators to select the number of clusters, the axes-aligned spectral embedding can also be a valuable input to partition the network. By looking at Fig. 3, it is possible to see that some buses reside in dense *cluster cores*, while others (e.g. 1, 17, 18, 39) have their cluster membership less certain. If the computed aligning orthogonal transformation is denoted as $\mathbf{R}^*$, and $\mathbf{Y}$ is the initial set of row-normalized eigenvectors of the matrix $\mathbf{A}_n$, the axes-aligned eigenvectors $\mathbf{Y}\mathbf{R}^*$ can be referred to as $\mathbf{Y}^*$. With eigenvectors $\mathbf{Y}^*$, a *cluster core* becomes numerically recognizable as the corresponding entries of some column of $\mathbf{Y}^*$ will be close to one. And because the Euclidean norm of each row equals to one, the entries of the remaining columns in the same row will be close to zero.

The cluster core estimation process is formulated as Algorithm 2. First, each eigenvector is sorted to reveal which of its rows have a large magnitude. Then the cluster core is initiated with the original row indices of the first $n_{\min}$ entries of the sorted eigenvector, where $n_{\min}$ is obtained from the requirement (11). The next nodes are added to the core in the decreasing order of the corresponding eigenvector entries until the predefined eigenvector threshold γ is reached. To consider each eigenvector independently, this threshold value should be above $\sqrt{2}/2$. The value $\sqrt{2}/2$ ensures that no two (or more) *row-normalized* eigenvectors can simultaneously assign the same row to their clusters. We typically set the initial value of γ to be $\sqrt{3}/2$, which guarantees the other eigenvector entries for the same node not to exceed 0.5. The expansion of the cluster core is updated after adding each next node, and the final cluster core is selected as the set of nodes with the smallest achieved expansion. In the majority of cases, the minimal expansion ratio corresponds to a cluster core having a single connected component. If there are multiple

Algorithm 2: Cluster Cores from Axes-Aligned Eigenvectors.

Input: $\mathbf{Y}_{n \times k}^*$, $\mathbf{A}$, γ , $n_{\min}$
Output: $\mathcal{CC}$ //The k cluster cores

```

1: for  $j = 1$  to  $k$  do
2:    $\mathbf{y} \leftarrow \mathbf{Y}^*[1, \dots, n; j]$ 
3:    $\mathbf{ord} \leftarrow$  Descending order of entries in  $\mathbf{y}$ 
4:   if  $\max(\mathbf{y}) - 0.1 < \gamma$  then
5:      $\gamma \leftarrow \max(\max(\mathbf{y}) - 0.1, \sqrt{2}/2)$ 
6:   end if
7:    $\mathbf{core} \leftarrow \mathbf{ord}[1, \dots, n_{\min}]$ 
8:    $\mathbf{phi}[1, \dots, n_{\min}] \leftarrow \phi(\mathbf{A}, \mathbf{core})$  //(8)
9:    $i \leftarrow n_{\min} + 1$ 
10:  while  $\mathbf{y}[\mathbf{ord}[i]] \geq \gamma$  do
11:     $\mathbf{core} \leftarrow \mathbf{ord}[1, \dots, i]$ 
12:     $\mathbf{phi}[i] \leftarrow \phi(\mathbf{A}, \mathbf{core})$  //(8)
13:     $i \leftarrow i + 1$ 
14:  end while
15:   $i^* \leftarrow \operatorname{argmin} \mathbf{phi}$ 
16:   $\mathcal{CC}_j \leftarrow \mathbf{core}[1, \dots, i^*]$ 
17: end for
18: return  $\mathcal{CC}$ 

```

connected components, the next smallest expansion with the index higher than $n_{\min}$ is accepted, and the connectivity is checked for the corresponding group of nodes. In the worst case, the largest connected component can be taken as the core. However, such situations are not common in practice and mostly occur when the eigenvector alignment cost is high. An example of the cluster core estimation approach is shown in Fig. 5 for the partitioning of the admittance graph of the IEEE 39 test network (see Fig. 4).

After all cluster cores have been estimated, they are improved one-by-one in the decreasing order of their expansions:

- 1) Rerun the cluster core estimation Algorithm 2 with the eigenvector threshold γ only slightly above of $\sqrt{2}/2$.
- 2) Merge each cluster core to a single *core node* and find the minimum isolating s-t cut from the current core node to the remaining core nodes. A new fictitious sink node should be created and connected to the remaining core nodes with edges of an infinitely large weight. Then the isolating cut is computed as the minimum s-t cut between the current core node and the fictitious sink node [19]. Increase the current cluster core by the nodes that reside on its side of the cut.

The final improvement is chosen as one that reduces the expansion ratio most. The goal of this procedure is to decrease the objective (9) by greedily attempting to reduce the expansion ratios of the least fit cluster cores. The use of minimum s-t cuts (i.e., solutions to the max-flow/min-cut problem [20]) is motivated by their ability to rapidly find the globally optimal cut between two nodes (or two sets of nodes) in the graph. The classical drawback of minimum graph cuts to return highly unequally-sized bisections [9] is circumvented here by looking for the minimum cut that separates a whole cluster core (merged

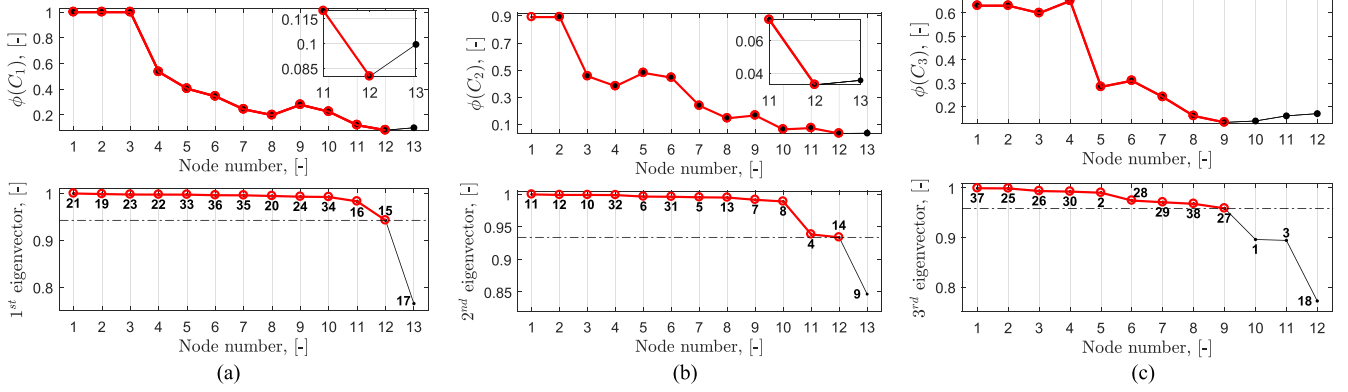


Fig. 5. Cluster core estimation from the axes-aligned spectral embedding in Fig. 3 using Algorithm 2. Each eigenvector was sorted in descending order, and the starting size of cluster core was set at 2. (a) 1st eigenvector. (b) 2nd eigenvector. (c) 3rd eigenvector.

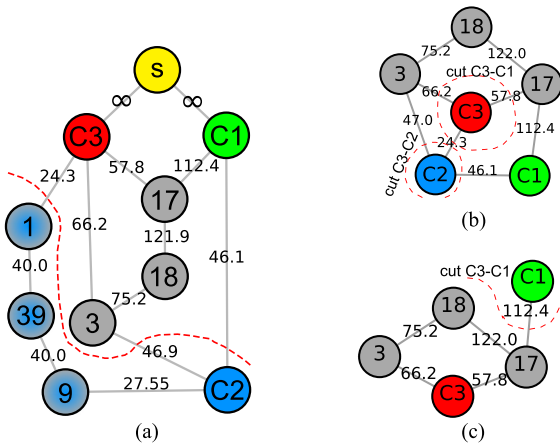


Fig. 6. Use of minimum s-t cuts for cluster core improvement and final recursive bisection. (a) Improvement of cluster core C2. (b) Final bisection (step 1). (c) Final bisection (step 2).

into a node) from the remaining core nodes. An example of use of minimum s-t cuts for cluster core improvement is shown in Fig. 6(a).

After the refinement stage, the updated cluster cores are once again collapsed into single nodes. The reduced network should consist of k core nodes and all the remaining nodes that were not assigned to the cluster cores, and it is partitioned via *recursive bisection*. At every step of recursive bisection, candidate minimum s-t cuts [20] are computed between an arbitrary core node and the remaining ones, and the lowest of these cuts is retained. This process iterates until all core nodes become separated from each other with all the remaining nodes being assigned to a cluster. The resulting partitioning is guaranteed to be connected, as cluster cores were constrained to be connected, and minimal s-t cuts always separate the input graph into two connected parts. For example, Fig. 6(b) shows the two candidate minimum s-t cuts with cluster core C3 being the source and cluster cores C1 and C2 being the two targets. As the value of cut C3-C2 is lower, it is retained, and the final partitioning is obtained by computing the s-t cut C3-C1 in the residual network resulting after the removal of node C2, as shown in Fig. 6(c).

VII. PARTITIONING OF BRANCH ADMITTANCE GRAPHS

To evaluate the proposed graph partitioning method, we have tested it on the branch admittance graphs of the two networks from the MATPOWER toolbox [17], [18] for which the alignment cost plots were presented in Section IV. No modifications (e.g., reduction of leaf nodes [1], [10]) were performed on the networks, except fixing the control angles of the few available phase shifting transformers (PSTs) at zero degrees to preserve the symmetry of the graph adjacency matrix. However, a PST with a non-zero phase shift can be represented in DC power flow by an equivalent (symmetric) admittance (see [21]), thus potentially allowing a broader extension of our experiments to power networks with PSTs.

The partitioning algorithm of Section VI was compared with the k -means clustering of spectral embedding (SKM) [1], [9], multilevel kernel k -means software Graclus [22] and hierarchical spectral clustering (HSC) [8]. The chosen hierarchical clustering linkage criterion of the HSC method was average linkage, as it was producing consistently better results. As Graclus and SKM do not generally guarantee connected partitions, the algorithm from [23] has been used to ensure the cluster connectedness in these two cases.

The maximal expansion ratio (9) as well as normalized cut (10) are normally increasing in magnitude with the growing number of clusters. To compare the partitioning performance for different numbers of clusters on the same scale, we choose to show the ratios of the results by other methods to the result of our method (denoted by the $\dagger$ superscript). In addition, the logarithmic y-axis is often used to make the data on the plots more separable.

For the purpose of comparison, the minimal cluster size was first set to $\lceil 0.03n/k \rceil$ with $k = 2, \dots, 38$ being the requested number of clusters. This small value was chosen as neither of k -means, Graclus and HSC allows the specification of the minimal cluster size, and the authors are not aware of a partitioning method (except the proposed one) that can support this requirement. The maximal number of clusters $\bar{k}$ was set to a relatively high value of 38 in order to demonstrate that the proposed method generally shows good performance both for few and many clusters.

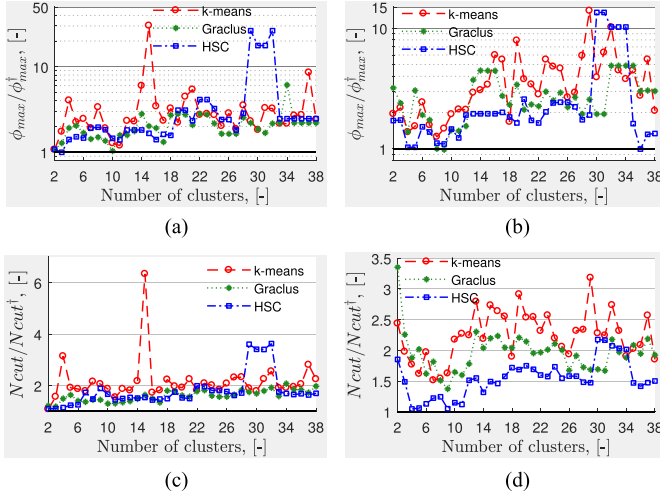


Fig. 7. Maximal expansion ratio and normalized cut for the branch admittance graphs of the two test networks from the MATPOWER toolbox [17], [18]. (a) Case1354pegase test network. (b) Case2869pegase test network. (c) Case1354pegase test network. (d) Case2869pegase test network.

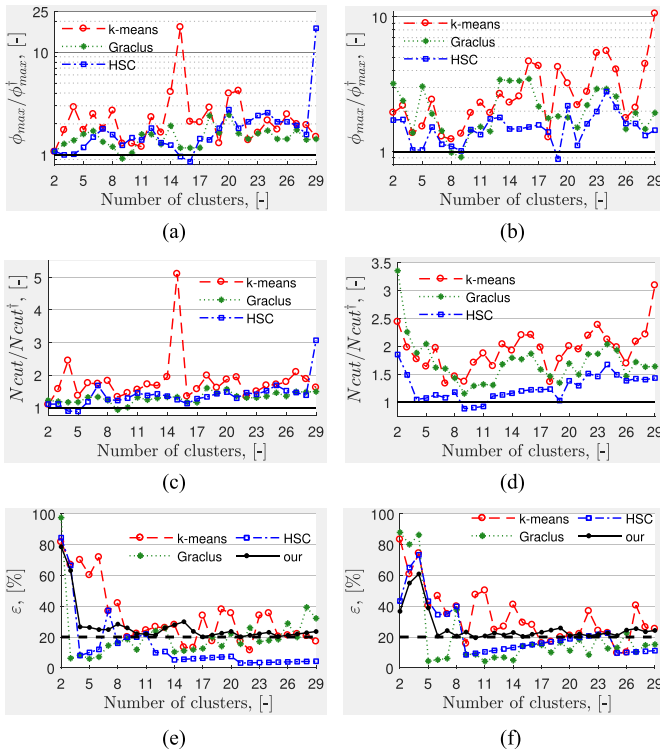


Fig. 8. Partitioning of the two test networks from the MATPOWER toolbox [17], [18] with the minimal cluster size constraint of 20% of the average cluster size. (a) Case1354pegase test network. (b) Case2869pegase test network. (c) Case1354pegase test network. (d) Case2869pegase test network. (e) Case1354pegase test network. (f) Case2869pegase test network.

For the case of unconstrained cluster sizes, Fig. 7 demonstrates that our network partitioning based on the orthogonal structure of spectral embedding outperforms the HSC method [8] in the majority of the cases, and often by a large margin. At the same time, the HSC method with average

TABLE I
TOTAL PARTITIONING TIME FOR 38 CLUSTERS

Method	SKM	HSC	Graclus	our
case1354pegase	0.69 s	0.86 s	0.15 s	3.55 s
case2869pegase	1.93 s	3.55 s	0.26 s	5.62 s

linkage can be considered as an efficient partitioning algorithm, as it usually performs noticeably better than Graclus and SKM.

Fig. 8 demonstrates the test results for the case when all clusters are required to be not smaller than 20% of the average cluster size n/k . The information provided by the aligned spectral embedding about the approximate locations and sizes of both small and large clusters in the network allows to neglect the eigenvectors describing presumably small clusters. As the result, the cluster size constraint is satisfied in all cases (see Figs. 8(e)–8(f)). Satisfying the cluster size constraint has the associated cost in terms of partitioning quality: the HSC method now shows better ϕ_{max} and N_{cut} more often, as it only aims to find compact clusters without imposing additional constraints on their size, as seen from Figs. 8(e)–8(f).

The computational time of the four algorithms for the largest tested number of clusters is shown in Table I. These results were obtained on MATLAB R2017a (64-bit) on a PC with an Intel Xeon E5 3.70 GHz CPU (single core computation) using a Linux virtual machine with 2 Gb of RAM. The computational time of our partitioning method includes the eigenvector computation time, eigenvector alignment time, time to estimate and refine cluster cores and time of final recursive bisection. As it can be seen, the run time of the proposed partitioning method is slower than the HSC run time for the smaller 1354 bus test network, but this relationship improves as the network size increases. In addition, our MATLAB code has some space for efficiency improvement.

VIII. ADAPTIVE ZONE DIVISION FOR SVC

This section aims to illustrate the pilot bus selection for SVC with the proposed clustering methods using the IEEE 39 bus test system [24] as an example. The adopted approach is first to divide the power network into a number of control zones and then to select the pilot nodes in each zone [3], [4].

A. SVC Algorithm and Objective Function

The general purpose of SVC is to maintain the voltage profile of the transmission network by controlling the voltage of several pilot buses to their reference values computed by a higher-level optimization program. The regulation of pilot bus voltages is achieved by updating the terminal voltage set points of generators participating in SVC.

To simulate the outcomes of regulating various sets of pilot buses, the coordinated secondary voltage control (CSVC) formulation [13] has been implemented. Its objective constitutes a trade-off between the regulation of pilot buses' voltages and

balancing of control generators' reactive loading levels, with a higher priority given to the first objective.

The SVC performance is estimated as follows:

$$V_{RMSE} = \sqrt{\frac{1}{|\mathcal{L}|} \sum_{i \in \mathcal{L}} (V_i - V_{i,0})^2} \quad (12)$$

where $\mathcal{L}$ is the set of load buses, V_i is the voltage magnitude at bus i after the CSVC algorithm [13] has converged, $V_{i,0}$ is the pre-disturbance reference voltage magnitude at bus i , and V_{RMSE} is the voltage root mean squared error (RMSE).

B. Zoning Methodology

The proposed zoning method is based on the concept of “Var Control Space” described in [4]. For a network consisting of g reactive power sources participating in SVC and l load nodes, let $S_{ij} = \frac{\partial V_{L,i}}{\partial V_{G,j}}$ be the sensitivity of the i th load node's voltage to the j th control generator's terminal voltage. The terms S_{ij} can be derived from the linearized power flow relationships:

$$\begin{bmatrix} \mathbf{B}_{GG} & \mathbf{B}_{GL} \\ \mathbf{B}_{LG} & \mathbf{B}_{LL} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{V}_G \\ \Delta \mathbf{V}_L \end{bmatrix} = \begin{bmatrix} \Delta \mathbf{Q}_G \\ \Delta \mathbf{Q}_L \end{bmatrix} \quad (13)$$

where $\Delta \mathbf{V}_G$ and $\Delta \mathbf{V}_L$ are vectors of voltage magnitude changes at generator and load buses respectively, $\Delta \mathbf{Q}_G$ and $\Delta \mathbf{Q}_L$ are vectors of changes in reactive power injections at generator and load buses respectively, and $\mathbf{B}_{GG}$, $\mathbf{B}_{GL}$, $\mathbf{B}_{LG}$, $\mathbf{B}_{LL}$ are sensitivity matrices that can be obtained in several ways from the power flow Jacobian (see the Appendix of [12]).

The second of the sensitivity equations (13) results in:

$$\Delta \mathbf{V}_L = \mathbf{B}_{LL}^{-1} \Delta \mathbf{Q}_L - \mathbf{B}_{LL}^{-1} \mathbf{B}_{LG} \Delta \mathbf{V}_G \quad (14)$$

which demonstrates that the sensitivity matrix between generator and load voltages is given by $-\mathbf{B}_{LL}^{-1} \mathbf{B}_{LG}$.

Given the generator-load voltage sensitivities, the closeness of loads to reactive power sources can be described by a *bipartite graph* with the following adjacency matrix:

$$\mathbf{S} = \begin{bmatrix} \mathbf{0} & [S_{ij}] \\ [S_{ij}]^T & \mathbf{0} \end{bmatrix} \quad (15)$$

The graph model (15) serves as input to the clustering techniques described in the previous sections to estimate the most suitable number of zones and to perform the actual network partitioning. Unlike the original method [4], model (15) does not use the logarithmic transformation to obtain distances from sensitivities, but instead directly uses the sensitivities S_{ij} as the input for spectral clustering. In addition, the estimated number of zones does not depend on the choice of distance function used by hierarchical clustering as in [3], [4]. Finally, unlike the original “Var Control Space” approach, model (15) uses sensitivities of voltages at load buses to terminal voltages of control generators. However, the original sensitivities from [4] can be used as well.

C. Pilot Bus Selection

In this paper a two-step pilot bus selection process is used. First, the method used in [10] is applied to select the initial set of pilot buses. At the next step, the pilot buses are changed

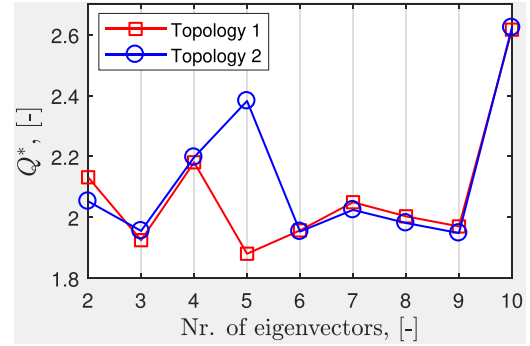


Fig. 9. Spectral clustering alignment costs for SVC zoning.

one at a time and the resulting SVC performance is observed. Similarly to the global search method [13], this process is executed a predefined number of times, but unlike [13] the pilot bus changes are constrained to their control zones, which significantly reduces the search space.

D. Results

Test Network: The IEEE 39 bus test system [24] consists of 29 load buses, nine generator buses and one equivalent generator representing the interconnection to an external power system. The generators at buses 30–38 are assumed to participate in SVC, while the equivalent generator is assumed to only maintain the voltage at its terminal bus 39.

Base Case: To better reflect the bi-objective nature of CSVC, the nominal operating condition of the IEEE 39 bus test network was obtained by running the MATPOWER [17] AC optimal power flow (OPF) with the generation costs favoring the equal reactive loading levels of all control generators. This step was also necessary to respect the reactive power limits of each generator after applying load disturbances to the network.

Adaptive Pilot Bus Selection: The benefits of adaptive pilot bus selection are illustrated by simulating the CSVC strategy [13] with various sets of pilot buses for the two topological states of the IEEE 39 bus test network:

- 1) All elements are in service.
- 2) Line 6–11 is switched off.

For each of the two operating conditions, the adjacency matrix was constructed as (15), and the alignment costs were computed with the method of Section IV for the number of zones ranging from 2 to 10. The results of this process are shown in Fig. 9 with the optimal number of zones being five for the first topological state and six for the second one.

The two resulting zone divisions are shown in Fig. 10. As it can be seen, the disconnection of line 6–11 suggests the splitting of the initial Zone 5 into Zone 5A and Zone 5B. Although the zone border remains the same, the decrease of internal connectedness of Zone 5 effectively leads to its splitting into two clusters, which is well reflected on the alignment cost plot of Fig. 9. Using the pilot bus selection process of Section VIII-C, the set of pilot buses for the first topological state is {3, 28, 16, 20, 5}, and for the second state it is {3, 28, 16, 20, 5, 12}.

Performance evaluation: Similarly to [4], [13], the single disturbance to test the CSVC performance is the positive increase

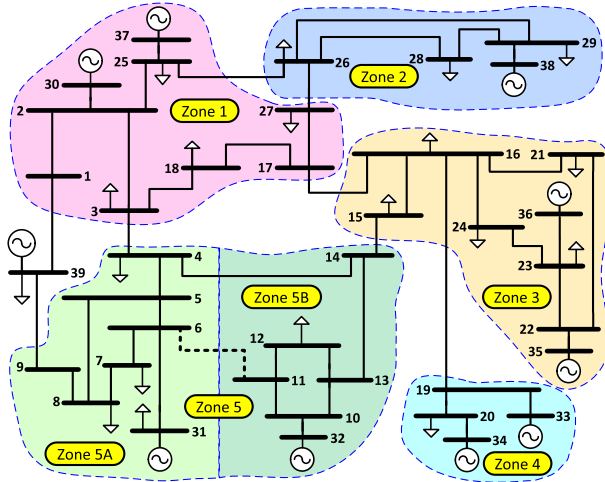


Fig. 10. Results of adaptive zone division.

TABLE II
VOLTAGE RMSE UNDER ADAPTIVE ZONE DIVISION

State \ Pilots	None	{3, 28, 16, 20, 5}	{3, 28, 16, 20, 5, 12}
Topology 1	0.0129 p.u.	0.00292 p.u.	0.00292 p.u.
Topology 2	0.0164 p.u.	0.00594 p.u.	0.00378 p.u.

TABLE III
VOLTAGE RMSE WITH OTHER SETS OF PILOT BUSES

State \ Pilots	{1, 3, 6, 24, 28}	{1, 3, 5, 12, 24, 28}
Topology 1	0.00494 p.u.	0.00510 p.u.
Topology 2	0.00664 p.u.	0.00537 p.u.

of the reactive power demand of all loads by 25%, resulting in the total system reactive load increase from 1408.9 MVar to 1807.2 MVar. To ensure the convergence of all pilot bus voltages close to their reference values, the balancing term between the two CSVC objectives has been lowered by the factor of 10, while the other control parameters are as in [13].

For the given test event, the voltage RMSE (12) is summarized in Table II for the two topological states mentioned above and their corresponding sets of pilot buses. The voltage control performance in absence of SVC (i.e., with no pilot buses) is also provided as a reference. As it can be seen, the CSVC performance is similar with the two sets of pilot buses for the nominal network topology, but once line 6–11 is switched off, the additional pilot bus 12 starts to create a noticeable difference in the system-wide performance indicator (12).

Other sets of pilot buses: As a reference for comparison, the CSVC algorithm [13] was simulated with the sets of pilot buses originally mentioned in [4] for the same situation of 25% positive reactive load increase at each load bus. The results of this case study are given in Table III. To comply with the study in [4], we have added the equivalent generator at bus 39 to the set of control generators. For our study we have empirically found that fixing the voltage at bus 39 instead of choosing a nearby pilot bus yields lower voltage deviations. The results in

Table III demonstrate that the pilot buses obtained with the help of our methods show a markedly lower voltage RMSE (12).

As it can be noticed, the results in Table III basically describe the situation with no pilot bus in Zone 4. The addition of bus 20 to the sets of pilot buses from [4] and Table III has resulted in the following performance improvements: the selection {1, 3, 6, 20, 24, 28} for the Topology 1 has dropped the objective (12) to 0.00332 p.u., and the selection {1, 3, 5, 12, 20, 24, 28} has resulted in the voltage RMS error of 0.00393 p.u. for the Topology 2. These observations further confirm the meaningfulness of AZD based on the approach presented in this paper.

IX. CONCLUSION

This paper has demonstrated the potential of orthogonal transformation of eigenvectors obtained from spectral clustering to the analysis of power system graphs. The first part of the paper illustrated the use of spectral clustering combined with orthogonal linear transformations to the practically important task of the selection of the number of clusters. Based on the ideas from [14], we have proposed a time-efficient combined algorithm to recover the orthogonal transformation that closely fits spectral embedding to the canonical coordinate system. To justify the minimization of the cost function [14], it has been compared with another cost function that was originally proposed in [15] for the selection of the number of clusters. The comparison results confirmed the similar shape of the two objectives for the varying number of clusters.

Further we proposed an efficient k -way spectral partitioning algorithm that uses axes-aligned spectral embedding to estimate the best set of k clusters in the network. This algorithm often achieves good results even if the spectral embedding does not show a distinct orthogonal structure, which may be explained by the great value of the global information about the locations of good clusters in the network that is contained in the axes-aligned spectral embedding. The possibility to approximately estimate cluster sizes allows to introduce a constraint on the minimal number of nodes per cluster. This feature represents a convenient way to avoid small clusters.

The test results on partitioning of branch admittance graphs of two large-scale power networks have demonstrated that the proposed k -way partitioning algorithm compares favorably with the existing methods both in terms of partitioning quality and solution time. To further illustrate how the proposed clustering framework can be applied to various power system problems, an example illustrating its use for the task of AZD for SVC has been devised.

The results of this paper motivate further research on cluster discovery from orthogonal structure of spectral embedding. In particular, performance metrics other than expansion ratio could be optimized for each cluster, and the overall methodology could be adapted to a wider range of applications.

ACKNOWLEDGMENT

The authors would like to acknowledge the valuable feedback received from the anonymous reviewers that has significantly helped to improve the contents of this paper.

REFERENCES

- [1] E. Cotilla-Sanchez, P. D. H. Hines, C. Barrows, S. Blumsack, and M. Patel, "Multi-attribute partitioning of power networks based on electrical distance," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4979–4987, Nov. 2013.
- [2] V. Quintana and N. Müller, "Partitioning of power networks and applications to security control," *Proc. Inst. Elect. Eng. C*, vol. 138, no. 6, pp. 535–545, Nov. 1991.
- [3] P. Lagonotte, J. Sabonnadiere, J.-Y. Leost, and J.-P. Paul, "Structural analysis of the electrical system: Application to secondary voltage control in France," *IEEE Trans. Power Syst.*, vol. 4, no. 2, pp. 479–486, May 1989.
- [4] H. Sun, Q. Guo, B. Zhang, W. Wu, and B. Wang, "An adaptive zone-division-based automatic voltage control system with applications in China," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 1816–1828, May 2013.
- [5] J. Zhong, E. Nobile, A. Bose, and K. Bhattacharya, "Localized reactive power markets using the concept of voltage control areas," *IEEE Trans. Power Syst.*, vol. 19, no. 3, pp. 1555–1561, Aug. 2004.
- [6] L. Ding, F. M. Gonzalez-Longatt, P. Wall, and V. Terzija, "Two-step spectral clustering controlled islanding algorithm," *IEEE Trans. Power Syst.*, vol. 28, no. 1, pp. 75–84, Feb. 2013.
- [7] L. Ding, Z. Ma, P. Wall, and V. Terzija, "Graph spectra based controlled islanding for low inertia power systems," *IEEE Trans. Power Del.*, vol. 32, no. 1, pp. 302–309, Feb. 2017.
- [8] R. J. Sanchez-Garcia, M. Fennelly, S. Norris, N. Wright, G. Niblo, J. Brodzki, and J. W. Bialek, "Hierarchical spectral clustering of power grids," *IEEE Trans. Power Syst.*, vol. 29, no. 5, pp. 2229–2237, Sep. 2014.
- [9] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.
- [10] V. Alimisis and P. C. Taylor, "Zoning evaluation for improved coordinated automatic voltage control," *IEEE Trans. Power Syst.*, vol. 30, no. 5, pp. 2736–2746, Sep. 2015.
- [11] J. Quirós-Tortós, R. Sánchez-García, J. Brodzki, J. Bialek, and V. Terzija, "Constrained spectral clustering-based methodology for intentional controlled islanding of large-scale power systems," *IET Gener., Transmiss. Distrib.*, vol. 9, no. 1, pp. 31–42, Jan. 2015.
- [12] A. Conejo, T. Gomez, and J. de la Fuente, "Pilot-bus selection for secondary voltage control," *Eur. Trans. Electr. Power Eng.*, vol. 3, no. 5, pp. 359–366, 1993.
- [13] A. Conejo and M. Aguilar, "Secondary voltage control: Nonlinear selection of pilot buses, design of an optimal control law, and simulation results," *IEE Proc. Gen., Transm., Distrib.*, vol. 145, no. 1, pp. 77–81, Jan. 1998.
- [14] S. Yu and J. Shi, "Multiclass spectral clustering," in *Proc. 9th Int. Conf. Comput. Vis.*, 2003, pp. 313–319.
- [15] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Proc. Adv. Neural Inform. Process. Syst.*, 2004, pp. 1601–1608.
- [16] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. Adv. Neural Inform. Process. Syst.*, 2001, pp. 849–856.
- [17] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "Matpower: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011.
- [18] S. Fliscounakis, P. Panciatici, F. Capitanescu, and L. Wehenkel, "Contingency ranking with respect to overloads in very large power systems taking into account uncertainty, preventive, and corrective actions," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4909–4917, Nov. 2013.
- [19] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis, "The complexity of multiterminal cuts," *SIAM J. Comput.*, vol. 23, no. 4, pp. 864–894, 1994.
- [20] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver, *Combinatorial Optimization*. New York, NY, USA: Wiley, 1997.
- [21] J. Verboomen, "Optimisation of transmission systems by use of phase shifting transformers," Ph.D. dissertation, Elect. Eng., Math., Comput. Sci., Delft University of Technology, Delft, The Netherlands, 2008.
- [22] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors: A multilevel approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 11, pp. 1944–1957, Nov. 2007.
- [23] I. Tyuryukanov, J. Quirós-Tortós, M. Naglic, M. Popov, M. van der Meijden, and V. Terzija, "A post-processing methodology for robust spectral embedded clustering of power networks," in *Proc. 17th IEEE Int. Conf. Smart Technol.*, 2017, pp. 1–5.
- [24] M. A. Pai, *Energy Function Analysis for Power System Stability*. Boston, MA, USA: Kluwer, 1989.



Ilya Tyuryukanov (S'16) received the B.S.E.E. (Hons.) degree from Moscow Power Engineering Institute, Technical University, Moscow, Russia, and the M.Sc. (Hons.) degree from RWTH Aachen University, Aachen, Germany. He is currently working toward the Ph.D. degree from Delft University of Technology, Delft, The Netherlands.

His research interests include machine learning and optimization techniques applied to power systems, in particular, wide-area control and protection.



Marjan Popov (M'95–SM'03) received the Dipl.-Ing. degree from the University of Skopje, Macedonia, in 1993, and the Ph.D. degree in electrical power engineering from Delft University of Technology, Delft, The Netherlands, in 2002. In 1997,

He was an Academic Visitor with the ARC Research Group, University of Liverpool, Liverpool, U.K., where he was involved in modeling of SF6 circuit breakers. His current research interests include future power systems, large-scale power system transients, intelligent protection for future power systems,

and wide-area monitoring and protection. He is a member of CIGRE. He has actively participated in WG C4.502 and WG A2/C4.39. He was a recipient of the IEEE PES Prize Paper Award and the IEEE Switchgear Committee Award in 2011. He is an Associate Editor of the International Journal of Electric Power and Energy Systems.



Mart A. M. van der Meijden (M'10) received the M.Sc. (*cum laude*) degree in electrical engineering from the Eindhoven University of Technology, the Netherlands, in 1981.

He is Part-Time Full Professor with the Department of Electrical Sustainable Energy of the Faculty of Electrical Engineering, Mathematics and Computers Science, Delft University of Technology, Delft, The Netherlands, since 2011. His chair and research focus is on Large Scale Sustainable Power Systems.

He has more than 30 years of working experience in the field of process automation and the transmission and the distribution of gas, district heating and electricity. He is leading research programs on intelligent electrical power grids, reliable and large scale integration of renewable (wind, solar) energy sources in the European electrical power systems and advanced grid concepts. Since 2003 he is working with TenneT TSO, Europe's first cross-border grid operator for electricity. He is Manager R&D/Innovation and was responsible for the development of the TenneT long-term vision on the electrical transmission system. He is a member of ENTSO-E/RDIC and CIGRE, and he has joined and chaired different national and international expert groups.



Vladimir Terzija (M'95–SM'00–F'16) received the Dipl.-Ing., M.Sc., and Ph.D. degrees in electrical engineering from the University of Belgrade, Serbia, in 1988, 1993, and 1997, respectively.

He is the EPSRC Chair Professor in Power System Engineering with the School of Electrical and Electronic Engineering, The University of Manchester, U.K. His current research interests include smart grid applications; wide-area monitoring, protection, and control; switchgear and transient processes; and ICT, data analytics and digital signal processing applications in power systems.

Prof. Terzija is Editor in Chief of the *International Journal of Electrical Power and Energy Systems*, Alexander von Humboldt Fellow, as well as a DAAD and Taishan Scholar. He is the National Thousand Talents Distinguished Professor at Shandong University, China.