VU Research Portal

A Spectral Representation of Power Systems with Applications to Adaptive Grid Partitioning and Cascading Failure Localization

Zocca, Alessandro; Liang, Chen; Guo, Linqi; Low, Steven H.; Wierman, Adam

document version
Early version, also known as pre-print

Link to publication in VU Research Portal

A Spectral Representation of Power Systems with Applications to Adaptive Grid Partitioning and Cascading Failure Localization

Alessandro Zocca*   Chen Liang, Linqi Guo, Steven H. Low, Adam Wierman [†]

May 12, 2021

Abstract

Transmission line failures in power systems propagate and cascade non-locally. This well-known yet counter-intuitive feature makes it even more challenging to optimally and reliably operate these complex networks. In this work we present a comprehensive framework based on spectral graph theory that fully and rigorously captures how multiple simultaneous line failures propagate, distinguishing between non-cut and cut set outages. Using this spectral representation of power systems, we identify the crucial graph sub-structure that ensures line failure localization – the network bridge-block decomposition. Leveraging this theory, we propose an adaptive network topology reconfiguration paradigm that uses a two-stage algorithm where the first stage aims to identify optimal clusters using the notion of network modularity and the second stage refines the clusters by means of optimal line switching actions. Our proposed methodology is illustrated using extensive numerical examples on standard IEEE networks and we discussed several extensions and variants of the proposed algorithm.

1 Introduction

Electrical power grids are among the most complex and critical networks in modern-day society, reliably bringing power from generators to end users, cities and industries, often very far away geographically from each other. Traditionally, power systems have been designed as one-directional networks, where electric energy travels over high-voltage transmission lines from big conventional and controllable generators to distribution networks and eventually to consumers. This paradigm is changing rapidly in recent years due to many concurrent trends. Firstly, more distributed energy resources are coming online and consumers are slowly becoming prosumers, shifting the conventional one-directional power flow paradigm to bi-directional. There are massive investments in renewable energy sources, whose power generation is geographically correlated, more volatile, and less controllable than traditional sources of generation. Moreover, we are electrifying our transportation systems, which is extremely important for reducing our greenhouse gas emissions. However, electric vehicles have high energy demands and the current grid design and operations will not be able to sustain a high penetration of EVs, especially in view of correlated charging patterns. Managing increasing and correlated loads while having more volatile and less controllable generation may seem to be an impossible task, especially when aiming to keep the same operational and reliability standards.

Given these trends, power grids are becoming increasingly stressed and have less margin for maneuver, making failures more likely and harder to contain, increasing the chance of blackouts. The growing complexity and increasing stochasticity of these networks challenge the classical reliability analyses and strategies. For instance, preventing line failures and mitigating their non-local propagation will become ever harder, having to deal with a broader range of more variable power injection configurations and more bidirectional flows on transmission lines.

To respond to these increasing challenges, power systems infrastructure is becoming more adaptive and responsive. Power systems have been traditionally looked at as a static network, but in fact they are not, since many of the transmission lines can be remotely taken online/offline. Rapid control mechanisms and corrective switching actions are increasingly being used to improve network reliability [125] and reduce operational costs [41, 66], especially since it is possible to quickly and efficiently estimate the current status of the grid using new monitoring devices and data processing strategies [72].

However, new transmission infrastructure can be expensive and its placement will not necessarily increase reliability. Therefore, we need to make optimal use of the existing system and its adaptability. Optimally and dynamically switching transmission lines is an inexpensive and promising option because it uses existing

---

*Department of Mathematics, Vrije Universiteit Amsterdam, 1081HV, Netherlands, `a.zocca@vu.nl`

[†]Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA 91125, {`cliang2,` `lguo, slow, adamw`}`@caltech.edu`

hardware to achieve increased grid robustness. The ability of the operator to adaptively change the topology of the grid depending on the current network configuration offers a great potential. However, even with perfect information about the system, finding the best switching actions in real time is not an easy task. The flows on the lines are fully determined by power flow physics and network topology (cf. Kirchhoff's and Ohm's laws), which means that every switching action causes a global power flow redistribution. In view of the combinatorial nature of this problem and of the large scale of the network, it is clear that a brute force approach will fail.

1.1 Contributions of the paper

This paper aims to tackle the aforementioned challenges in power systems operations by taking full advantage of transmission line flexibility. To accomplish this requires new mathematical tools to first understand and then optimally and reliably operate these increasingly complex systems. In particular, a new understanding of the role of the network topology, especially when it comes to non-local failure propagation, is needed.

To this end, we introduce new spectral graph theory tools to analyze and optimize power systems. In particular, we propose a new spectral representation of power systems that effectively captures complex interactions within power networks, for instance inter-dependencies between infrastructure components and failure propagation events. The key observation underlying this representation is the fact that, under the DC power flow approximation, the linear relation between power injections and line flows can be expressed using the weighted graph Laplacian matrix, where the line susceptances are used as edge weights. The eigenvalues and eigenvectors of this matrix thus contain rich information about the topology of the network as well as on its physical and electrical properties. Starting from this observation, Section 2 illustrates how the graph Laplacian and its pseudo-inverse can be used to characterize islands of the power network and the power balance conditions of each island.

When using conventional representations, the impact of line failures is discontinuous and notoriously difficult to characterize or even approximate, but our spectral representation provides a simple and exact characterization. More specifically, in Section 3 we show how the impact of network topology changes on line flows, quantified using the so-called *distribution factors*, can be exactly described using spectral quantities. The results, most of which already appeared in [57, 58] cover not only the case of single line outage, but also the more general case of the simultaneous outage of multiple lines. The analysis distinguishes two fundamentally different scenarios depending on whether the network remains connected or not, namely *cut set outages* and *non-cut set outages*.

Leveraging the spectral representation and the distribution factors analysis, in Section 4 we identify which graph sub-structures affect power flow redistribution and fully characterize the line failure propagation. More specifically, our analysis reveals that the network line failure propagation patterns can be fully understood by focusing on network block and bridge-block decompositions, which are related, respectively, to the cut vertices and cut edges of the power network.

Our failure localization results suggest the robustness of the network is often improved by reducing the redundancy of the network, which indirectly makes the aforementioned decomposition finer. Using this insight, we then suggest in Section 5 a novel design principle for power networks and propose algorithms that leverage their pre-existing flexibility to increase their robustness against failure propagation. More specifically, we propose a new procedure that, by temporarily switching off transmission lines, can be used to optimally modify the network topology in order to refine the bridge-block decomposition. We accomplish this through solving a novel two-stage optimization problem that adaptively modifies the power network structure, using the current power injection configuration as input.

In Subsection 5.1 we introduce the first step of this procedure, which identifies a target number of clusters in the power network by solving an optimization problem whose objective function is a weighted version of the classical *network modularity* problem. Being an NP-hard problem to solve exactly, we show how spectral clustering methods can be used to quickly and effectively obtain good approximated solutions, i.e., nearly-optimal clusters. The second step of our procedure, presented in Subsection 3.3, identifies line switching actions that transform the identified clusters into bridge-blocks, refining the bridge-block decomposition of the network. The optimal subset of line switches are selected using a combinatorial optimization problem which aims to minimize the congestion of transmission lines of the network while achieving the target network bridge-block decomposition.

This two-step procedure, which we refer to as the one-shot algorithm, is summarized in Subsection 5.3, where we also introduce a faster recursive variant that iteratively refines the bridge-block decomposition by bi-partitioning the biggest bridge-block. The numerical implementation of both algorithms is discussed in Section 6, where we test their performance on a large family of IEEE test networks of heterogeneous size. As revealed by our extensive analysis, almost all these power networks have a trivial bridge-block decomposition consisting of a single giant bridge-block, suggesting that their robustness could be greatly improved by our procedure.

Most of the present paper focuses on reliability issues and, in particular, line failure propagation, but we conclude in Section 7 by highlighting some other promising applications of the network block and bridge-block decompositions. Specifically, we demonstrate how a finer-grained block decomposition of a power network can be leveraged to (a) design a real-time failure localization and mitigation scheme that provably prevents and

localize successive failures; (b) accelerate the standard security constrained OPF by decomposing and decoupling the problem into smaller versions in each block; (c) enable more efficient distributed solvers for AC OPF by transforming a globally coupled constraint into their counterparts for each block; and (d) tractably quantify the market price manipulation power of aggregators.

1.2   Related literature

The deep interplay between the topology of a given power system and power flow physics is for some aspects unique in network theory and a large body of literature has been devoted to the challenge of finding effective representations and approximations for these complex networked systems. These are instrumental not only to understand and analyze the behavior of these networks, but also to develop fast and effective algorithms and optimization strategies. In this subsection we briefly review the existing related literature on this topic.

Our work uses tools from spectral graph theory, an established and vast field in which many good books are available, see e.g. [16, 138] and reference therein. Particularly relevant for our analysis is [114], which focuses on the efficient computation of the pseudo-inverse of the graph Laplacian and explores its intimate connection with effective resistances. Graph spectra-based methods have been extensively used in the context of power systems, in particular in the study of phase angles frequency dynamics [31, 35, 36, 110, 82, 81], but also for power system restoration [113] and to analyze of the geometry of power flows [116, 19]. A spectral characterization for the network bridges, which play a crucial role in our analysis, was already given in [20, 128].

Understanding the underlying structure of given graph using spectral graph methods (e.g., Cheeger's inequalities) is a classical problem that received great attention in both discrete math and computer science literature. A canonical problem is the minimum k-cut problem, which aims to find the best partition in k clusters of a given graph [131]. The same problem has then been rediscovered in other domains, e.g., computer vision [127], with more emphasis on how to quickly find approximated solutions for NP-hard k-way partitioning problems. In this context, several clustering techniques based on spectral methods have been proposed [127, 107], whose properties have later been studied analytically in [140]. At the same time, there was an increasing interest in the physics literature to study large networks arising in various domain (among which the world wide web and epidemiology) and unveil their underlying community structure, see [44] for a review. It is in this context that the concept of network modularity [102, 106] that we use in this paper has been introduced. Besides network modularity, many other techniques developed for complex networks have been applied to power systems, in particular to capture their main topological features and assess their robustness [108, 28].

The approach we propose in this paper crucially exploits the fact that the network topology of power systems can be changed by means of line switching actions. The classical *Optimal Transmission Switching* (OTS) problem leverages the same existing flexibility, but with different goals. This optimization problem was first introduced in [83, 46, 96] as a corrective strategy to alleviate line overloading due to contingencies. Afterwards, transmission switching has been explored in the literature as a control method for various problems such as voltage security [126, 79], line overloads [48, 126], loss and/or cost reduction [6, 42, 124, 61, 41, 65], clearing contingencies [79, 84], improving system security [123], or a combination of these objectives [5, 118, 125]. The interested readers can also read [67] for a comprehensive review of the OTS literature.

Power grids are naturally divided in control regions [94], raising the issue of how to optimally cluster and operate these networks. For this reason, clustering recently received a lot of attention in the power systems literature. A very diverse set of methodologies have been considered, ranging from spectral clustering [135] to various heuristics based on the modularity score [50, 88]. A substantial effort has been devoted to expand and augment classical clustering methods in order to account for specific features of power systems, for instance using the notion of electrical distance [10, 26] or conductances [93, 92]. In the context of cascading failure analysis, clustering and community detection methods have also been used on the abstract interaction graphs [99, 100] rather than on the physical network topologies.

Many ad-hoc clustering algorithms have been developed in the power systems literature particularly in the context of *intentional controlled islanding* (ICI), an extreme security mechanism against cascading failures in which the network is disconnected into several self-sustained "islands" to prevent further contingencies. The goal of the standard ICI problem is to find the optimal partition of the network into islands while including additional constraints to ensure generator coherency and minimum power imbalance. This inevitably makes the clustering problem even harder and nontrivial trade-offs arise, which is the reason why several heuristics and approximations methods have been developed for ICI in the recent literature, see [63, 90] for an overview.

Intentional controlled islanding is a rather extreme response to large-scale cascading failures. In [7] the author propose as an alternative emergency measure beside precisely on the core properties of the bridge-block decomposition. Several other mitigation strategies less drastic than ICI could be adopted, but they often require more detailed cascade models. However, modeling cascading failures mathematically in power systems is rather complex due to the underlying power flow physics. The book [8] gives a comprehensive overview of the various models that have been introduced in the literature to describe cascading failures in power systems as well as the various optimization approaches that have been devised to improve the robustness of these systems. Most

cascading failure models usually consider line or generator failures as trigger events, but correlated stochastic fluctuations of the power injections have also been considered in [101].

Contingency analysis is a very commonly used tool for reliable operations of a power system that assesses the impact of either generator and transmission line outages [147]. Such an impact can be assessed by solving the AC power flow equations that describe the network after each contingency. Due to the large number of contingencies that must be assessed in order to satisfy $N - k$ security for $k \geq 1$, it is a common practice to first use DC power flow models to quickly screen contingencies and select a much smaller subset that result in voltage or line limit violations for more detailed analysis using AC power flow models. Such a contingency screening uses the *power transfer distribution factors* (PTDFs) and *line outage distribution factors* (LODFs) for a DC power flow model, see [147, Chapter 7] for more details. The main advantage of using these factors is that the impact of generator and transmission outages on the post contingency networks can be analyzed using the common pre-contingency topology across contingency scenarios. LODFs for multi-line outages were first considered in [38] to study the impact of network changes on line currents using the network Laplacian matrix, obtained from the admittance matrix after the linearization of AC power flow equations. However, the refined proof approach using the matrix inversion lemma was introduced later in [2, 132]. Note that the paper [132] allows for more general outages (e.g., generator outages) and proposes a methodology to quickly rank contingencies in security analysis. Probably unaware of the results in [2, 132], some of the formulas for generalized LODF (GLODF) have been re-discovered in [52, 53, 54] using different approaches based on the Power Transfer Distribution Factors (PTDF). More specifically, line outages are emulated through changes in injections on the pre-contingency network by judiciously choosing injections at the endpoints of each outaged/disconnected line using PTDF. Distribution factors for linear flows have been studied extensively in the recent literature [73, 77, 119, 120, 134]. Some recent work [57, 58, 59, 60, 78, 76] rigorously proves that the presence of specific graph sub-structures guarantees that some of these distribution factors are zero, hence showing the potential for localizing outages and avoiding a global propagation by means of optimal network reconfiguration. The localization effects of these graph substructures can be effectively visualized by means of an ad-hoc version of the so-called *influence* (or *interaction*) *graph* [69, 70, 99, 100]. In [60, 87, 56] the authors also explore how carefully designed network substructures synergize well with other control mechanisms that provide congestion management in real time. LODFs are also studied more recently as a tool to quantify network robustness and flow rerouting [134]. While PTDF and LODF determine the sensitivity of power flow solutions to parameter changes, one can also study the sensitivity of optimal power flow solutions to parameter changes; see, e.g., [49, 64].

2 A spectral representation of power systems

A power transmission network can be described as a graph $G = (V, E)$, where $V = \{1, \ldots, n\}$ is the set of vertices (modeling *buses*) and $E \subset V \times V$ is the set of edges (modeling *transmission lines*). Denote by $n = |V|$ the number of vertices and by $m = |E|$ the number of edges of the network G. In this section we describe power network model that we consider throughout the paper and highlight an intimate connection between spectral graph theory and power flow physics. We first review in Subsection 2.1 some key notions from graph theory and some classical network decompositions that are crucial for our analysis. Then, in Subsection 2.2, we introduce the power flow model that we will focus on in this paper in spectral terms and present some immediate results that follow from this representation.

2.1 Network decompositions and substructures

Due to power flow physics, the topology of a power transmission network plays a central role in the way power flows on its lines. In fact, the power flows are intrinsically determined by the network physical structure via Kirchhoff's laws once that the power injections are fixed. This also means that the power flow redistribution after a contingency and possible cascading outages are intimately related to the network structure. It comes as no surprise that to study the robustness of a power network makes uses of advance graph theory notions and algorithms. We now briefly review some useful network decompositions and substructures that will play a crucial role in our analysis in the next sections.

In this subsection we look at a transmission power network in purely topological terms as an undirected and unweighted graph G, thus temporarily ignoring the physical and electrical properties of its transmission lines. The *k-partition* of a network $G = (V, E)$ is a finite collection $\mathcal{P} = \{\mathcal{V}_1, \mathcal{V}_2, \cdots, \mathcal{V}_k\}$ of nonempty and disjoint subsets of V such that $\bigcup_{i=1}^{k} \mathcal{V}_i = V$. Denote by $\Pi_k(G)$ the collection of k-partitions of G and by $\Pi(G) := \bigcup_{i=1}^{|V|} \Pi_i(G)$ the collection of all its partitions. Given a partition $\mathcal{P}$, each edge of the network G is either a *cross edge* if its two endpoints belong to different clusters of $\mathcal{P}$ and *internal edge* otherwise. We denote the collections of cross edges and internal edges as $E_c(\mathcal{P})$ and $E_i(\mathcal{P})$, respectively. There is a natural partial order $\succeq$ on the collection $\Pi(G)$ of partitions, defined as follows: given two partitions $\mathcal{P}^1 = \{\mathcal{V}_1^1, \mathcal{V}_2^1, \cdots, \mathcal{V}_{k_1}^1\}$ and $\mathcal{P}^2 = \{\mathcal{V}_1^2, \mathcal{V}_2^2, \cdots, \mathcal{V}_{k_2}^2\}$, we say that $\mathcal{P}^1$ is *finer* than $\mathcal{P}^2$, denoted as $\mathcal{P}^1 \succeq \mathcal{P}^2$, if each subset in $\mathcal{P}^1$ is contained

in some subset in $\mathcal{P}^2$. More precisely, $\mathcal{P}^1 \succeq \mathcal{P}^2$ if for every $i = 1, 2, \ldots, k_1$, there exists some $j(i) \in \{1, 2, \ldots, k_2\}$ such that $\mathcal{V}_i^1 \subseteq \mathcal{V}_{j(i)}^2$.

Each partition $\mathcal{P}$ induces a *quotient graph* $G/\mathcal{P}$, which is the undirected graph whose vertices are the clusters in $\mathcal{P}$ and two distinct clusters $\mathcal{V}_i, \mathcal{V}_j \in \mathcal{P}$, are adjacent if and only if there exists at least one cross edge in the original graph G whose endpoints belong one to $\mathcal{V}_i$ and one to $\mathcal{V}_j$. A slight variation of the quotient graph is the *reduced graph* $G_\mathcal{P}$, which is defined as the (multi)graph whose vertices are still the clusters in $\mathcal{P}$, but in which we draw an edge between two distinct clusters for every cross edge between them. The reduced graph $G_\mathcal{P}$ can thus have multiple (parallel) edges, and coincides with the quotient graph only when it is simple.

It is well known that any equivalence relation on the set of vertices of the network G induces a partition. We now briefly review three canonical decompositions of an undirected graph that have been introduced in the graph-theory literature (see, e.g., [62, 145]) using specific equivalence relations.

First of all, consider the partition $\mathcal{P}_{\text{path}} = \{\mathcal{I}_1, \ldots, \mathcal{I}_k\}$ in which two vertices are in the same subset if and only if there is a path connecting them. In this case $E_c(\mathcal{P}_{\text{path}}) = \emptyset$ by definition, the quotient and reduce graphs trivially coincides and and have no edges. The subgraphs $G_1, \ldots, G_k$ induced by $\mathcal{P}_{\text{path}}$ are the *connected components* of G, also referred to as *islands* in the power systems literature. A graph is *connected* if it consists of a single connected component, i.e. $|\mathcal{P}_{\text{path}}| = 1$. In general, a power network G may not be connected at all times, either due to operational choice or as a result of severe contingencies).

The notion of graph connectedness allows to introduce two more notions that will be crucial for our network analysis. A subset $\mathcal{E} \subset E$ is a *cut set* of G if the deletion of all the edges in $\mathcal{E}$ disconnects G. If the cut set consists of a single edge, we refer to it as *bridge* or *cut edge*. A *cut vertex* of G is any vertex whose removal (together with its incident edges) increases the number of its connected components.

A second network decomposition can be obtained by looking at the network circuits. Recall that a circuit is a path from a vertex to itself with no repeated edges. We consider the partition $\mathcal{P}_{\text{circuit}}(G)$ such that two vertices are in the same class if and only if there is a circuit in G containing both of them. The subgraphs induced by the subset of nodes in each of the equivalence classes of $\mathcal{P}_{\text{circuit}}(G)$ are precisely the connected components of the graph obtained from G by deleting all the bridges and, for this reason, they are called *bridge-connected components* or *bridge-blocks*. We will refer to the partition $\mathcal{P}_{\text{circuit}}(G)$ as *bridge-block decomposition* of G and denote it by $\mathcal{BB}(G)$.

The next lemma summarizes the intuitive fact that the bridge-block decomposition of a graph becomes finer if some of its edges are removed. This result is the cornerstone of the strategy to improve the network reliability that is presented in Section 5.

Lemma 2.1 (Removing edges makes bridge-block decomposition finer)**.** *For any graph $G = (V, E)$ and subset of edges $\mathcal{E} \subset E$, the bridge-block decomposition of the graph $G^\mathcal{E} := (V, E \setminus \mathcal{E})$ obtained from G by removing the edges in $\mathcal{E}$ is always finer than that of G, i.e.,*

$$\mathcal{BB}(G^\mathcal{E}) \succeq \mathcal{BB}(G).$$

Proof. Consider the partition $\mathcal{BB}(G) = \mathcal{P}_{\text{circuit}}(G)$ and recall that two vertices belong to the same equivalence class (bridge-block) if and only if there exists a circuit that contains both of of them. By removing a subset of edges $\mathcal{E}$ from G, each of the equivalence classes either remains unchanged or gets partitioned in smaller equivalence classes. This readily implies that for every new bridge-block $B \in \mathcal{BB}(G^\mathcal{E})$ obtained after the edge removal there exists a bridge-block B' of the original network, i.e., $B' \in \mathcal{BB}(G)$, such that $B \subseteq B'$. $\square$

A third network decomposition can be obtained by considering the network cycles, which are the circuits in which the only repeated vertices are the first and last ones. More specifically, we consider the equivalence relation for edges of being contained in a cycle and denote by $\{E_1, E_2, \ldots, E_k\}$ to be the resulting partition of E. Let $R_i = (V_i, E_i)$ be the corresponding subgraph of G, consisting of the edges in E_i and the vertices V_i that are the endpoints of these edges. We refer to each subgraph R_i either as *block* and to $\mathcal{B}(G) = \{V_1, \ldots, V_k\}$ as the *block decomposition* of G. Every block is a maximal biconnected (or 2-connected) subgraph, since the removal of any of its vertices does not disconnect it. A vertex could appear in more than one block and, since its removal disconnects G, this is an equivalent characterization of cut vertices. The block decomposition $\mathcal{B}(G)$ thus is not a partition of the vertex set V, but nonetheless for any pair of vertices there is either no block or exactly one block containing them both. We distinguish two types of blocks, *trivial blocks* and *nontrivial blocks*, depending on whether they consist of a single edge or not. A trivial block consists of an edge that is not contained in any cycle; such an edge must then be a bridge since its removal disconnects the graph, cf. [62, Chapter 3]. Two nontrivial blocks either share a cut vertex or are connected by a bridge, i.e., a trivial block.

We remark that a statement similar to Lemma 2.1 holds also for the edge partition $\{E_1, E_2, \ldots, E_k\}$ corresponding to the block decomposition of a graph G and for the block decomposition $\mathcal{B}(G)$ itself. However, for the strategy to improve network reliability that we present in Section 5, it turns out that $\mathcal{B}(G)$ is less convenient to work with than the bridge-block decomposition $\mathcal{BB}(G)$ since the block decomposition $\mathcal{B}(G)$ is not a proper vertex partition (recall that the cut vertices always appear in two or more blocks).

Both the block and bridge-block decompositions of a graph G are unique. We can informally say that the block decomposition is finer than the bridge-block decomposition, since each bridge corresponds to a trivial block, while every bridge-block consists of several non-trivial blocks connected by cut vertices. Such a nested structure of the block and bridge-block decompositions is illustrated for a small graph in Fig. 1. Another visualization of these network decompositions for an actual power network can be found later in Section 4, see Fig. 3.

A graph naturally decomposes into a tree/forest of blocks called its *block-cut tree/forest*, depending on whether the original graph is connected or not. More specifically, in such a block-cut tree/forest there is a vertex for each block and for each cut vertex, and there is an edge between any block and cut vertex that belongs to that block. Similarly, also the bridge-blocks and bridges of G have a natural tree/forest structure, called *bridge-block tree/forest*, depending if G is connected or not. Such a tree/forest is the graph with a vertex for each bridge-block and with an edge for every bridge. We refer the reader to [62, Chapter 3] for further details.
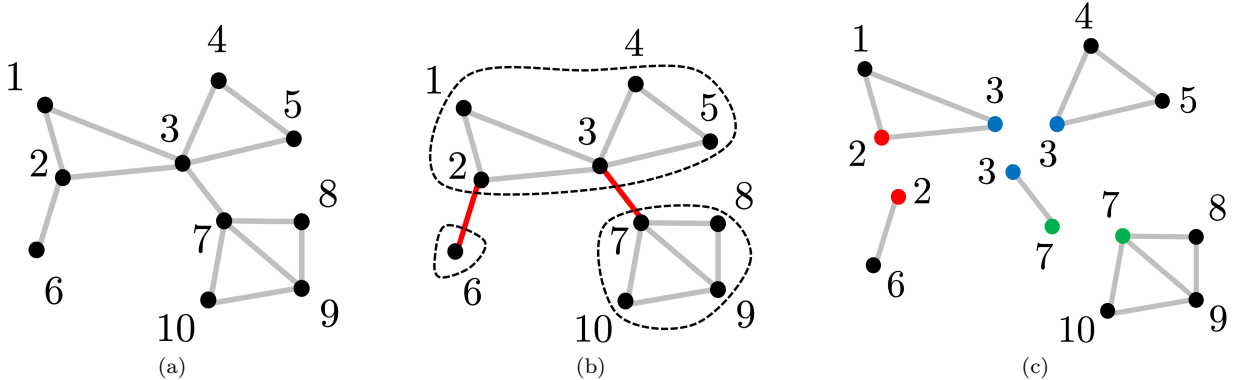

Figure 1: An undirected graph G in (a) with its bridge-block decomposition in (b) and with its block decomposition in (c). Edges $(2,6)$ and $(3,7)$ are bridges. Vertices $2, 3, 7$ are the cut vertices of G and appear in multiple blocks.

The problems of finding the blocks and bridge-blocks of a given graph are well understood: sequential algorithms that run in time $O(n+m)$ were given in [71, 136] and logarithmic-time parallel algorithms are given in [3, 137]. Later in Section 3.1 we also give a precise spectral characterization for bridges (cf. Corollary 3.3).

Another complementary powerful tool to study the topology of power networks is using its spanning trees, which will play a crucial role in the study of distribution factors in Section 3. A spanning tree of G is a subgraph that is a tree with a minimum possible number of edges that includes all of the vertices of G. Let $T_{\mathcal{E}}$ be the collection of all spanning trees of G consisting only of edges in $\mathcal{E} \subseteq E$ and $T = T_E$ the set of all spanning trees of G. We further denote by $T\backslash\mathcal{E} := T_{E\backslash\mathcal{E}}$ the collection of all spanning trees of G consisting of edges in $E\backslash\mathcal{E}$. For any pair of disjoint subsets of vertices $V_1, V_2 \subset V$, we denote by $T(V_1, V_2)$ the collection of spanning forests of G consisting of exactly two trees necessarily disjoint, one containing the vertices in V_1 and the other one containing those in V_2. Note that $T_{\mathcal{E}}$ can be empty if the subset $\mathcal{E}$ is too small and that $T(V_1, V_2)$ is empty by definition whenever V_1 and V_2 are not disjoint.

2.2 Power flow model and its spectral representation

In this subsection, we present the classical DC power flow model and a few key spectral properties that are intertwined with the power flow dynamics.

In order to accurately model a power system, it is important to capture the physical properties of its transmission lines. When necessary, we thus look at $G = (V, E)$ as weighted graph, in which each edge $\ell = (i, j) \in E$ has a weight $b_\ell = b_{ij} = b_{ji} > 0$ that models the *susceptance* of the corresponding line. To stress the difference with its unweighted counterpart, we will refer to the weighted graph $G = (V, E, \boldsymbol{b})$ as *power network* and to its edges as lines.

It is crucial for our analysis to capture the "electrical properties" of specific parts or components of the power network, in particular of its spanning trees. For this reason we introduce a nonnegative weight for any subset $\mathcal{E} \subseteq E$ of lines by defining a function $\beta : 2^E \to \mathbb{R}_+$ as follows:

$$\beta(\mathcal{E}) := \prod_{\ell \in \mathcal{E}} b_\ell, \qquad \mathcal{E} \subseteq E.$$

In other words, $\beta(\mathcal{E})$ is the product of the susceptances of all the lines in the subset $\mathcal{E}$. For consistency we set $\beta(\mathcal{E}) = 0$ if $\mathcal{E} = \emptyset$ and $E \neq \emptyset$, but $\beta(\mathcal{E}) = 1$ when the power network G consists of a single vertex, i.e., $\mathcal{E} = E = \emptyset$.

Let $B := \mathrm{diag}(b_1, \ldots, b_m) \in \mathbb{R}^{m \times m}$ be the diagonal matrix with the line susceptances (i.e., the edge weights) $b_1, \ldots, b_m$ as entries, assuming a fixed order has been chosen for the edge set E. Aiming to describe line flows, it

is necessary to introduce an (arbitrary) orientation for the edges, which is captured by the vertex-edge incidence matrix $C \in \{-1, 0, +1\}^{n \times m}$ defined as

$$C_{i\ell} = \begin{cases} 1 & \text{if vertex } i \text{ is the source of } \ell, \\ -1 & \text{if vertex } i \text{ is the target of } \ell, \\ 0 & \text{otherwise.} \end{cases}$$

The spectral representation of power systems relies on the *weighted Laplacian matrix* of the power network G, that is the symmetric $n \times n$ matrix L defined as $L := CBC^T$ or, equivalently in terms of the susceptances, as

$$L_{i,j} := \begin{cases} \sum_{k \neq i} b_{ik} & \text{if } i = j, \\ -b_{ij} & \text{if } i \neq j. \end{cases} \tag{1}$$

Denote by $\lambda_1, \lambda_2, \dots, \lambda_n$ its eigenvalues and by $\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_n$ the corresponding eigenvectors, which we take to be of unit norm and pairwise orthogonal. The next proposition summarizes some crucial properties of the weighted Laplacian matrix L and its *Moore-Penrose pseudo-inverse* $L^\dagger$. The proof can be found in the appendix and we refer the reader to [114, 138] for further spectral properties of graphs.

Proposition 2.2 (Laplacian matrix L and its pseudo-inverse)**.** *The following properties hold for the weighted Laplacian matrix of a power network $G = (V, E, \boldsymbol{b})$:*

(i) *For every $\boldsymbol{x} \in \mathbb{R}^n$ we have $\boldsymbol{x}^T L \boldsymbol{x} = \sum_{(i,j) \in E} b_{ij}(x_i - x_j)^2 \geq 0$.*

(ii) *L is a symmetric positive semidefinite matrix and, hence, has a real non-negative spectrum, i.e., $0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$.*

(iii) *All the rows (and columns) of L sum to zero and thus the matrix L is singular.*

(iv) *The matrix L has rank $n - k$, where $k \geq 1$ is the number of connected components $\mathcal{I}_1, \dots, \mathcal{I}_k$ of G. The eigenvalue 0 has multiplicity k and $\ker(L) = \ker(L^\dagger) = \text{span}(\mathbf{1}_{\mathcal{I}_1}, \dots, \mathbf{1}_{\mathcal{I}_k})$, where $\mathbf{1}_{\mathcal{I}_j} \in \{0, 1\}^n$ is the vector which is 1 on $\mathcal{I}_j$ and 0 elsewhere. Furthermore, the pseudo-inverse of L can be calculated as*

$$L^\dagger = \sum_{j=k+1}^{n} \frac{1}{\lambda_j} \boldsymbol{v}_j \boldsymbol{v}_j^T = \left(L + \sum_{j=1}^{k} \frac{1}{|\mathcal{I}_j|} \mathbf{1}_{\mathcal{I}_j} \mathbf{1}_{\mathcal{I}_j}^T \right)^{-1} - \sum_{j=1}^{k} \frac{1}{|\mathcal{I}_j|} \mathbf{1}_{\mathcal{I}_j} \mathbf{1}_{\mathcal{I}_j}^T.$$

In particular, in the special case in which G is connected, i.e., $k = 1$, $\ker(L) = \ker(L^\dagger) = \text{span}(\mathbf{1})$, where $\mathbf{1} \in \mathbb{R}^n$ is the vector with all unit entries and

$$L^\dagger = \sum_{j=2}^{n} \frac{1}{\lambda_j} \boldsymbol{v}_j \boldsymbol{v}_j^T = \left(L + \frac{1}{n} \mathbf{1} \mathbf{1}^T \right)^{-1} - \frac{1}{n} \mathbf{1} \mathbf{1}^T.$$

(v) *$L^\dagger$ is a real, symmetric, positive semi-definite matrix with zero row (and hence column) sums and its nonzero eigenvalues are $0 < \lambda_n^{-1} \leq \dots \leq \lambda_{k+1}^{-1}$.*

Denote by $\boldsymbol{p} \in \mathbb{R}^n$ the vector of power injections, where the entry p_i models the power generated (if $p_i > 0$) or consumed (if $p_i < 0$) at the bus corresponding to vertex i. We say that a power injection vector $\boldsymbol{p} \in \mathbb{R}^n$ is *balanced* if $\boldsymbol{p} \in \text{im}(L)$. Since $\text{im}(L^\dagger) = \ker(L^\dagger)^\perp = \text{span}(\mathbf{1}_{\mathcal{I}_1}, \dots, \mathbf{1}_{\mathcal{I}_k})^\perp$, a power injection vector $\boldsymbol{p}$ is balanced if and only if the net power injection is equal to zero in every island $\mathcal{I}_j$ of the network, i.e., $\sum_{i \in \mathcal{I}_j} p_i = 0$.

Transmission power systems are operated using alternating current (AC), but the equations describing the underlying power flow physics are non-linear leading to many computational challenges in solving them [147]. Therefore, linearization techniques are often used to approximate the power flow equations. We now introduce the so-called *DC power flow model*, which is a first-order approximation of the AC equations that is commonly used to model high-voltage transmission system [147, 133, 34].

Given an (oriented) edge $\ell = (i, j) \in E$, we denote interchangeably as f_ℓ or f_{ij} the power flow on that line. We set $f_{ji} = -f_{ij}$, with the convention that a power flow is negative if the power flows in the opposite direction with respect to the edge orientation. The so-called *DC power flow model* is the system of equations

$$\boldsymbol{p} = C\boldsymbol{f}, \tag{2a}$$

$$\boldsymbol{f} = BC^T \boldsymbol{\theta}, \tag{2b}$$

where $\boldsymbol{\theta} \in \mathbb{R}^n$ is the vector of *phase angles* at the network vertices and $\boldsymbol{f} \in \mathbb{R}^m$ is the vector of line power flows. Equation (2a) captures the flow-conservation constraint, while equation (2b) describes Kirchhoff's laws. The DC

model (2) has a unique solution $\boldsymbol{f}$ for each balanced injection vector $\boldsymbol{p}$. Indeed, from (2) we can deduce that $\boldsymbol{p} = CBC^T\boldsymbol{\theta} = L\boldsymbol{\theta}$, and, ultimately, obtain the following spectral representation for the power flows

$$\boldsymbol{f} = BC^T L^\dagger \boldsymbol{p}. \tag{3}$$

The power flows are thus uniquely determined by the injections and physical properties of the network. In view of (3), it is clear that the network operator does not directly control the power routing, which can be only indirectly changed by modifying the power injections or the network topology. This peculiar feature is one of the reasons why it is challenging to study and improve the reliability of power systems and their robustness against failures.

The interactions between components in a power network depend, not only on its topology, but also on the physical properties of its components and the way they are coupled by power flow physics. Such an "electrical structure" can be effectively unveiled and studied using spectral methods, more specifically looking at the so-called *effective resistance*, which is formally defined as follows. The effective resistance $R_{i,j}$ between a pair of vertices i and j of the network G is the non-negative quantity

$$R_{ij} := (\mathbf{e}_i - \mathbf{e}_j)^T L^\dagger (\mathbf{e}_i - \mathbf{e}_j) = L^\dagger_{ii} + L^\dagger_{jj} - 2L^\dagger_{ij}, \tag{4}$$

where $\mathbf{e}_i$ denotes the vector with a 1 in the i-th coordinate and 0's elsewhere. R_{ij} quantifies how "close" vertices i and j are in the power network G and takes small value in the presence of many paths with high susceptances between them. The effective resistance is a proper distance on $V \times V$ and for this reason it is often referred to as *resistance distance* [80]. The *total effective resistance* of a graph G is defined as $R_{\text{tot}}(G) = \frac{1}{2}\sum_{i,j=1}^n R_{ij}$ and it is intimately related to the spectrum of G by the following identity proved by [80]

$$R_{\text{tot}}(G) = n \cdot \text{tr}(L^\dagger) = n \cdot \sum_{j=2}^n \frac{1}{\lambda_j}.$$

The same quantity is also known as *Kirchhoff index* in the special case when all the edges of the network G have unit weights, i.e., $b_\ell = 1$, $\ell = 1, \ldots, m$. The total effective resistance is a key quantity that measures how well connected the network is and for this reason has been extensively studied and rediscovered in various contexts, such as complex network analysis [37] and probability theory [32]. The notion of effective resistances have been first introduced in the context of power networks analysis by [27] and [26] and since then have been extensively used to study their robustness, see e.g., [20, 36, 85, 144, 143], as well as to devise algorithm to improve their reliability, see e.g. [45, 74, 152].

3   Distribution factors and power flow redistribution

In this section we introduce two families of sensitivity factors of power networks, also known as *distribution factors*. The first class is that of *power transfer distribution factors* (PTDF), which describe how changes in power injections impact line flows. The second type of factors are the so-called *generalized line outage distribution factors* (GLODF), which capture how line removal/outages impact power flows on the surviving lines. As illustrated in the previous section, if the power injections or the network topology change, one can always recompute the new line flows by solving the power flow equations (3). The distribution factors are based on the DC approximation for power flows and provide fast estimates of the new line flows without solving again the AC power flow equations. For this reason, they have been widely used for security and contingency analysis when power flow solutions are computationally expensive, see [147, Chapter 7].

In this paper, we use distribution factors to analyze structural properties of power flow solutions and to design failure localization and mitigation mechanisms to reduce the risk of large-scale blackouts from cascading failures. In this section we unveil the deep connection between the distribution factors and specific substructures of the power network topology, complementing [119, 134]. We henceforth assume the power network to be connected. The rest of the section is structured as follows. We first focus on the impact of power injection changes on line flows in Subsection 3.1, introducing the PTDF factors and matrix and showing how they can be calculated in terms of spectral quantities. We then illustrate the effect of network topology changes on power flows. In our analysis we distinguish two very different cases. In Subsection 3.2 we look at the scenario in which the set of outaged/removed lines does not disconnect the power network, obtaining the close-form expression for the GLODFs. In Subsection 3.3 we consider the more involved case in which the outaged lines disconnect the power network into multiple islands and show how the impact of topology change can be clearly distinguished from that of the balancing rule that needs to be invoked to rebalance the power in each of these newly created islands.

3.1   Power transfer distribution factors (PTDF)

The goal of this subsection is to analyze the effect of power injections change on line flows. We will use $\tilde{\cdot}$ to denote variables after such a change (possibly a contingency, but not exclusively).

Consider two nodes $s, t \in V$ (not necessarily adjacent) and a scalar δ_{st} describing an injection change, which can be either positive or negative. Suppose the injection at node s is increased from p_s to $\tilde{p}_s := p_s + \delta_{st}$ and that at node t is reduced from p_t to $\tilde{p}_t := p_t - \delta_{st}$, while all other injections remain unchanged $\tilde{p}_i = p_i$, $i \neq s, t$, so that

$$\tilde{\boldsymbol{p}} = \boldsymbol{p} + \delta_{st}(\mathbf{e}_s - \mathbf{e}_t)$$

remains balanced, i.e., $\sum_{i \in V} \tilde{p}_i = \sum_{i \in V} p_i = 0$. Note that this is an equivalent condition a power injection for being balanced since we assumed that G is connected and thus $\ker(L) = \mathbf{1}$ (cf. Proposition 2.2).

The *power transfer distribution factor* (PTDF) $D_{\ell,st}$, also called the *generation shift distribution factor*, is defined to be the resulting change $\Delta f_\ell := \tilde{f}_\ell - f_\ell$ in the line flow on line ℓ normalized by δ_{st}:

$$D_{\ell,st} := \frac{\Delta f_\ell}{\delta_{st}} = \frac{\tilde{f}_\ell - f_\ell}{\delta_{st}}.$$

For convenience $D_{\ell,st}$ is defined to be 0 if $i = j$ or $s = t$. The next proposition shows how this factor can be computed from the pseudo-inverse $L^\dagger$ of the graph Laplacian matrix L or, alternatively, in terms of the spanning forests of G.

Proposition 3.1 (PTDF $D_{\ell,st}$). *Given a connected power network $G = (V, E, \boldsymbol{b})$, the following identities hold for every line $\ell = (i, j) \in E$ and any pair of nodes $s, t \in V$:*

$$D_{\ell,st} \;=\; b_\ell \left(L_{is}^\dagger + L_{jt}^\dagger - L_{it}^\dagger - L_{js}^\dagger \right) \;=\; b_\ell \frac{\sum_{\mathcal{E} \in T(\{i,s\},\{j,t\})} \beta(\mathcal{E}) \;-\; \sum_{\mathcal{E} \in T(\{i,t\},\{j,s\})} \beta(\mathcal{E})}{\sum_{\mathcal{E} \in T} \beta(\mathcal{E})}. \tag{5}$$

Recall that $T(\{i, s\}, \{j, t\})$ has been defined in Section 2 as the collection of spanning forests of G consisting of exactly two disjoint trees, one containing nodes i and s and the other one containing nodes j and t (the definition of $T(\{i, t\}, \{j, s\}))$ is analogous). The proof easily follows combining [57, Theorem 4] and the properties of the pseudo-inverse $L^\dagger$ derived in [55]. Besides the precise spectral relation with pseudo-inverse of the graph Laplacian, the previous proposition shows that the PTDFs depend *only on the topology and line susceptances* and are thus independent of the injections $\boldsymbol{p}$. We remark that the insensitivity of the PTDFs to injections is, however, specific to the DC power flow approximation, which yields a linear relation between power injections and line flows.

As noted above, $D_{\ell,st}$ is the change in power flow on line ℓ when a unit of power is injected at node s and withdrawn at node t where nodes s and t need not be adjacent. When they are adjacent, i.e., $\hat{\ell} := (s, t)$ is a line in E, then the PTDFs define a $m \times m$ matrix $D := (D_{\ell\hat{\ell}}, \ell, \hat{\ell} \in E)$, to which we refer as the *PTDF matrix*. The following proposition summarizes some properties of the PTDF matrix that already appeared without proof in [57, Corollary 5]. A detailed proof is thus provided in the Appendix.

Proposition 3.2 (PTDF matrix). *If $G = (V, E, \boldsymbol{b})$ is a connected power network, then the following properties hold for the PTDF matrix $D = (D_{\ell\hat{\ell}}, \ell, \hat{\ell} \in E)$:*

(i) $D \;=\; BC^T L^\dagger C$.

(ii) For every line $\ell \in E$ the corresponding diagonal entry of the PTDF matrix D is given by

$$D_{\ell\ell} = 1 \;-\; \frac{\sum_{\mathcal{E} \in T \setminus \{\ell\}} \beta(\mathcal{E})}{\sum_{\mathcal{E} \in T} \beta(\mathcal{E})}.$$

Hence, in particular, $0 < D_{\ell\ell} \leq 1$.

(iii) For every line $\ell = (i, j) \in E$ the corresponding diagonal entry of the PTDF matrix D is given by

$$D_{\ell\ell} = b_\ell \left(L_{ii}^\dagger + L_{jj}^\dagger - 2L_{ij}^\dagger \right) = b_\ell R_{ij}.$$

Identity (ii) reveals that the more ways to connect every node without going through $\ell = (i, j)$, the smaller $D_{\ell\ell}$ is, i.e., the smaller it is the impact on the power flow on line ℓ when a unit of power is injected and withdrawn at nodes i and j respectively.

Recall that $T \setminus \{\ell\}$ is the collection of all spanning trees of G consisting of edges in $E \setminus \{\ell\}$. Using (4) and the fact that if ℓ is a bridge, then $T \setminus \{\ell\} = \emptyset$, we immediately obtain the following spectral characterization for the bridges of the network.

Corollary 3.3 (Spectral characterization of network bridges). *If $G = (V, E, \boldsymbol{b})$ is a connected power network, the following three statements are equivalent:*

- *Line ℓ is a bridge;*

- $D_{\ell\ell} = 1$;

- $R_{ij} = L_{ii}^{\dagger} + L_{jj}^{\dagger} - 2L_{ij}^{\dagger} = b_{\ell}^{-1}$.

The proof immediately follows from the statements (ii) and (iii) of Proposition 3.2 noticing that line ℓ is a bridge if and only if it belongs to *any* spanning tree of the power network G, or equivalently $T \backslash \{\ell\} = \emptyset$.

Lastly, we present a sufficient condition for having zero PTDF in purely topological terms. Its proof is given in the Appendix and relies on the representation of the PTDF in terms of spanning forests given in (5).

Theorem 3.4 (Simple Cycle Criterion for PTDF). *If there is no simple cycle in a power network $G = (V, E, \boldsymbol{b})$ that contains both lines ℓ and $\hat{\ell}$, then $D_{\ell\hat{\ell}} = 0$.*

3.2 GLODF for non-cut set

Consider a connected power network $G = (V, E, \boldsymbol{b})$ with balanced injections $\boldsymbol{p}$. When a subset of lines $\mathcal{E} \subset E$ trips or is removed from service, the network topology changes and, as a consequence, the power flow redistributes on the *surviving network* $G^{\mathcal{E}} := (V, E \setminus \mathcal{E})$ as prescribed by power flow physics. To fully understand the power flow redistribution and unveil the impact of network topology on the flow changes, we distinguish two scenarios, depending on whether $\mathcal{E}$ is a cut set or not for the power network G, and analyzed them separately.

Even if the disconnection of the subset of lines $\mathcal{E} \subset E$ can be deliberate, for instance due to maintenance or network optimization (as it will be the case in the Section 5), in most cases it is due to a *contingency*, e.g., a line or node (bus) failure. For this reason and for conciseness, we present the next family of distribution factors using the standard power system terminology, referring to the lines in $\mathcal{E}$ as *outaged lines* and distinguishing between *pre-contingency* and *post-contingency line flows*.

Let $\mathcal{E} \subsetneq E$ be any non-cut set of $k := |\mathcal{E}|$ lines that are simultaneously removed from the power network. The resulting surviving network $G^{\mathcal{E}} = (V, E \setminus \mathcal{E})$ is still connected. Assuming that the injections $\boldsymbol{p}$ remain unchanged (and thus balanced), the new power flows can be calculated directly from (3) after having updated the matrices B, C, and L to reflect the structure of the surviving network $G^{\mathcal{E}} = (V, E \setminus \mathcal{E})$. We now review this calculations in more detail and show how it leads to the notion of *generalized line outage distribution factors (GLODFs.* These factors quantify the impact of the removal of each line in $\mathcal{E}$ to each of the survival lines and are thus essential for any power network contingency analysis.

We first introduce some auxiliary notation. Let $\boldsymbol{f}_{\mathcal{E}}$ be the vector of the pre-contingency power flows on the outaged lines in $\mathcal{E}$ and let $\boldsymbol{f}_{-\mathcal{E}}$ and $\tilde{\boldsymbol{f}}_{-\mathcal{E}}$ be the pre and post-contingency power flows respectively on the surviving lines in $-\mathcal{E} := E \backslash \mathcal{E}$. Partition the susceptance matrix B and the incident matrix C into submatrices corresponding to surviving lines in $-\mathcal{E}$ and outaged lines in $\mathcal{E}$:

$$B =: \operatorname{diag}(B_{-\mathcal{E}}, B_{\mathcal{E}}), \qquad C =: [C_{-\mathcal{E}} \;\; C_{\mathcal{E}}]. \tag{6}$$

Assuming that the injections $\boldsymbol{p}$ remain unchanged, we can calculate the post-contingency network flows by solving the DC power flow equations (3) $\tilde{\boldsymbol{f}}_{-\mathcal{E}} = B_{-\mathcal{E}} C_{-\mathcal{E}}^T L_{-\mathcal{E}}^{\dagger} \boldsymbol{p}$, expressed in terms of the Laplacian matrix $L_{-\mathcal{E}} := C_{-\mathcal{E}} B_{-\mathcal{E}} C_{-\mathcal{E}}^T$ of the surviving network.

The main result of this section shows that for a non-cut set $\mathcal{E} \subsetneq E$ the post-contingency flow net changes $\Delta \boldsymbol{f}_{-\mathcal{E}} := \tilde{\boldsymbol{f}}_{-\mathcal{E}} - \boldsymbol{f}_{-\mathcal{E}}$ depend linearly on the pre-contingency power flows $\boldsymbol{f}_{\mathcal{E}}$ on the outaged lines. Their sensitivity to $\boldsymbol{f}_{\mathcal{E}}$ defines a $|E \backslash \mathcal{E}| \times |\mathcal{E}|$ matrix $K^{\mathcal{E}} := (K_{\ell\hat{\ell}}^{\mathcal{E}}, \ell \in E \backslash \mathcal{E}, \hat{\ell} \in \mathcal{E})$, called the *Generalized Line Outage Distribution Factor* (GLODF) matrix, through

$$\Delta \boldsymbol{f}_{-\mathcal{E}} = K^{\mathcal{E}} \boldsymbol{f}_{\mathcal{E}},$$

i.e., $\Delta f_{\ell} = \tilde{f}_{\ell} - f_{\ell} = \sum_{\hat{\ell} \in \mathcal{E}} K_{\ell\hat{\ell}}^{\mathcal{E}} f_{\hat{\ell}}$, for any $\ell \in E \backslash \mathcal{E}$. Like the PTDFs, also the GLODFs depend solely on the network topology and susceptances, as illustrated by the next theorem, which shows how the GLODF matrix $K^{\mathcal{E}}$ can be calculated using spectral quantities.

To state this result, we first need some additional notation. Partition the PTDF matrix D into submatrices corresponding to non-outaged lines in $-\mathcal{E}$ and outaged lines in $\mathcal{E}$, possibly after permutations of rows and columns. Since $D = BC^T L^{\dagger} C$ from Proposition 3.2(i), these different blocks of D can be written explicitly in terms of the submatrices of B and C introduced in (6):

$$D = \begin{bmatrix} D_{-\mathcal{E}, -\mathcal{E}} & D_{-\mathcal{E}\mathcal{E}} \\ D_{\mathcal{E}, -\mathcal{E}} & D_{\mathcal{E}\mathcal{E}} \end{bmatrix} = \begin{bmatrix} B_{-\mathcal{E}} C_{-\mathcal{E}}^T L^{\dagger} C_{-\mathcal{E}} & B_{-\mathcal{E}} C_{-\mathcal{E}}^T L^{\dagger} C_{\mathcal{E}} \\ B_{\mathcal{E}} C_{\mathcal{E}}^T L^{\dagger} C_{-\mathcal{E}} & B_{\mathcal{E}} C_{\mathcal{E}}^T L^{\dagger} C_{\mathcal{E}} \end{bmatrix}. \tag{7}$$

We now express the GLODF matrix $K^{\mathcal{E}}$ explicitly in terms of the PTDF submatrices introduced in (7).

Theorem 3.5 (GLODF $K^{\mathcal{E}}$ for non-cut set outage). *Let $\mathcal{E} \subsetneq E$ be a non-cut set outage for the power network $G = (V, E, \boldsymbol{b})$. Then, the matrix $I - D_{\mathcal{E}\mathcal{E}} = I - B_{\mathcal{E}} C_{\mathcal{E}}^T L^{\dagger} C_{\mathcal{E}}$ is invertible and the net line flow changes $\Delta \boldsymbol{f}_{-\mathcal{E}}$ are given by*

$$\Delta \boldsymbol{f}_{-\mathcal{E}} = K^{\mathcal{E}} \, \boldsymbol{f}_{\mathcal{E}}, \tag{8}$$

where the GLODF matrix can be calculated as

$$K^{\mathcal{E}} = D_{-\mathcal{E}\mathcal{E}} \, (I - D_{\mathcal{E}\mathcal{E}})^{-1} = B_{-\mathcal{E}} C_{-\mathcal{E}}^T L^{\dagger} C_{\mathcal{E}} \left(I - B_{\mathcal{E}} C_{\mathcal{E}}^T L^{\dagger} C_{\mathcal{E}} \right)^{-1}. \tag{9}$$

The proof is immediate using [57, Theorem 7], after reformulating the results appearing there in terms of the inverse of reduced Laplacian matrix A using the pseudo-inverse $L^{\dagger}$ using the identities proved in [55].

This formula of $K^{\mathcal{E}}$ is derived by considering the pre-contingency network with changes $\Delta \boldsymbol{p}$ in power injections that are judiciously chosen to emulate the effect of simultaneous line outages in E. The reference [2] seems to be the first to introduce the use of matrix inversion lemma to study the impact of network changes on line currents in power systems. This method is also used in [132] to derive the GLODF for ranking contingencies in security analysis. The GLODF has also been derived earlier, e.g., in [38], and re-derived recently in [53, 52], without the simplification of the matrix inversion lemma.

3.2.1 Non-bridge outages

We now briefly consider the special case in which the subset $\mathcal{E}$ of outaged lines is still a non-cut set, but consists of a single line, i.e., $\mathcal{E} = \{\hat{\ell}\}$. Since the singleton $\{\hat{\ell}\}$ is a cut set if and only if $\hat{\ell}$ is a bridge, we are thus focusing on *non-bridge line outages*.

The GLODF matrix $K^{\mathcal{E}}$ introduced earlier rewrites now as a vector $(K_{\ell\hat{\ell}}, \ell \in E \setminus \{\hat{\ell}\})$, where, since there is no ambiguity, we suppressed the superscript $\{\hat{\ell}\}$ for compactness. We recover in this way the so-called *line outage distribution factor* (LODF) $K_{\ell\hat{\ell}}$, which is formally defined to be the change $\Delta f_{\ell} := \tilde{f}_{\ell} - f_{\ell}$ in power flow on surviving line $\ell \neq \hat{\ell}$ when a *single non-bridge* line $\hat{\ell}$ trips, normalized by the pre-contingency line flow $f_{\hat{\ell}}$:

$$K_{\ell\hat{\ell}} := \frac{\Delta f_{\ell}}{f_{\hat{\ell}}}, \qquad \ell \neq \hat{\ell} \in E, \text{ for non-bridge line } \hat{\ell}$$

assuming that the injections $\boldsymbol{p}$ remain unchanged, cf. [147] and [55]. The LODFs for non-bridge outages inherit all the properties of GLODFs for non-cut sets and, in particular, they are also independent of $\boldsymbol{p}$. The next proposition reformulates the results in Theorem 3.5 in the case of a single non-bridge failure, and relates the LODF to the weighted spanning forests of pre-contingency network; see [57, Theorem 6] for the proof of this latter fact.

Proposition 3.6 (LODF $K_{\ell\hat{\ell}}$ for non-bridge outage). *Let $\hat{\ell} = (s, t)$ be a non-bridge outage that does not disconnect the power network $G = (V, E, \boldsymbol{b})$. For any surviving line $\ell = (i, j) \neq \hat{\ell}$ the corresponding LODF is given by*

$$K_{\ell\hat{\ell}} = \frac{D_{\ell\hat{\ell}}}{1 - D_{\hat{\ell}\hat{\ell}}} = \frac{b_{\ell} \left(L_{is}^{\dagger} + L_{jt}^{\dagger} - L_{it}^{\dagger} - L_{js}^{\dagger} \right)}{1 - b_{\hat{\ell}} \left(L_{ss}^{\dagger} + L_{tt}^{\dagger} - 2L_{st}^{\dagger} \right)} = \frac{b_{\ell} \left(R_{it} - R_{is} + R_{js} - R_{jt} \right)}{2(1 - b_{\hat{\ell}} R_{st})} \tag{10}$$

and can be equivalently be expressed in terms of the spanning forests of G as

$$K_{\ell\hat{\ell}} = \frac{b_{\ell}}{\sum_{\mathcal{E} \in T\setminus\{\hat{\ell}\}} \beta(\mathcal{E})} \Big(\sum_{\mathcal{E} \in T(\{i,s\},\{j,t\})} \beta(\mathcal{E}) - \sum_{\mathcal{E} \in T(\{i,t\},\{j,s\})} \beta(\mathcal{E}) \Big). \tag{11}$$

Identity (10) can be equivalently derived using a rank-1 update of the pseudo-inverse $L^{\dagger}$ as done by [128]. Each LODF accounts for all the alternative paths on which the power originally carried by line $\hat{\ell}$ can flow in the new topology and for how many of these the line ℓ belongs to. This dependence is made explicit in (11) in terms of the weighted spanning forests.

Recall that Proposition 3.2 shows that $D_{\hat{\ell}\hat{\ell}} = 1$ if and only if line $\hat{\ell}$ is a bridge, which corroborates the fact that (10) is valid only for non-bridge failures. We remark that, differently from the diagonal entries $(D_{\ell\ell}, \ell \in E)$ of the PTDF matrix, the factor $D_{\ell\hat{\ell}}$ can take any sign and thus so can $K_{\ell\hat{\ell}}$. From (11) we immediately deduce that $K_{\ell\hat{\ell}} > 0$ if ℓ and $\hat{\ell}$ share either the source node ($i = s$) or the terminal node ($j = t$), since in this case $T(\{i,t\},\{j,s\}) = \emptyset$.

We can also express LODF in (10) in matrix form and relate it to the PTDF matrix D defined in Subsection 3.1. Partition the edge set E in two subsets, E^{b} and E^{n}, by distinguishing between bridges and non-bridge lines and set $m_{\mathrm{b}} := |E^{\mathrm{b}}|$ and $m_{\mathrm{n}} := |E^{\mathrm{n}}|$. Consider the $m \times m_{\mathrm{n}}$ matrix $K^{\mathrm{n}} := \left(K_{\ell\hat{\ell}}, \ell \in E, \hat{\ell} \in E^{\mathrm{n}} \right)$ and

$D^{\mathrm{n}} := \big(D_{\ell\hat{\ell}}, \ell \in E, \hat{\ell} \in E^{\mathrm{n}}\big)$ and rewrite them into two submatrices, possibly after permutations of rows and columns:

$$K^{\mathrm{n}} \;=\; \begin{bmatrix} K_{\mathrm{bn}} \\ K_{\mathrm{nn}} \end{bmatrix}, \qquad D^{\mathrm{n}} \;=\; \begin{bmatrix} D_{\mathrm{bn}} \\ D_{\mathrm{nn}} \end{bmatrix}$$

Note we implicitly extended the LODF definition to the case in which $\ell = \hat{\ell} \in E^{\mathrm{n}}$ as follows: the diagonal entries of the submatrix K_{nn} are defined as

$$K^{\mathrm{n}}_{\hat{\ell}\hat{\ell}} := \frac{D_{\hat{\ell}\hat{\ell}}}{1 - D_{\hat{\ell}\hat{\ell}}} \qquad \text{for non-bridge line } \hat{\ell} \in E^{\mathrm{n}}$$

and represents the change in power flow on line $\hat{\ell}$ if $\hat{\ell} = (s,t)$ is *not* outaged but injections p_s and p_t are changed by one unit.

To express the LODF (11) in matrix form in terms of $L^{\dagger}$, partition the matrices B and C into two submatrices corresponding to the non-bridge lines and bridge lines (possibly after permutations of rows and columns), i.e., $B =: \operatorname{diag}(B_{\mathrm{b}}, B_{\mathrm{n}})$ and $C =: [C_{\mathrm{b}} \; C_{\mathrm{n}}]$. Let $\operatorname{diag}(D_{\mathrm{nn}})$ be the $m_{\mathrm{n}} \times m_{\mathrm{n}}$ diagonal matrix whose diagonal entries are $D_{\hat{\ell}\hat{\ell}}$ for non-bridge lines $\hat{\ell}$.

Corollary 3.7. *With the submatrices defined above we have* $D^{\mathrm{n}} \;=\; BC^{T}L^{\dagger}C_{\mathrm{n}}$ *and*

$$K^{\mathrm{n}} \;=\; D^{\mathrm{n}}\left(I - \operatorname{diag}(D_{\mathrm{nn}})\right)^{-1} \;=\; BC^{T}L^{\dagger}C_{\mathrm{n}}\left(I - \operatorname{diag}(B_{\mathrm{n}}C_{\mathrm{n}}^{T}L^{\dagger}C_{\mathrm{n}})\right)^{-1}.$$

Given a non-cut set $\mathcal{E}$ of lines, let $-\mathcal{E} := E \backslash \mathcal{E}$. Consider the submatrix $K_{-\mathcal{E}\mathcal{E}} := \big(K_{\ell\hat{\ell}}, \ell \in -\mathcal{E}, \hat{\ell} \in \mathcal{E}\big)$ of K^{n} and the submatrices $D_{-\mathcal{E}\mathcal{E}} := \big(D_{\ell\hat{\ell}}, \ell \in -\mathcal{E}, \hat{\ell} \in \mathcal{E}\big)$ and $D_{\mathcal{E}\mathcal{E}} := \big(D_{\hat{\ell}\hat{\ell}}, \hat{\ell} \in \mathcal{E}\big)$ of D^{n}. Corollary 3.7 allows us to rewrite the line outage distribution factors in (10) in matrix form as

$$K_{-\mathcal{E}\mathcal{E}} = D_{-\mathcal{E}\mathcal{E}}\left(I - \operatorname{diag}(D_{\mathcal{E}\mathcal{E}})\right)^{-1}. \tag{12}$$

Equation (12), despite looking similar to the formula (9) for the GLODF matrix $K^{\mathcal{E}}$ in Theorem 3.5, is fundamentally different since only the diagonal elements of $D_{\mathcal{E}\mathcal{E}}$ are used, while the full matrix $D_{\mathcal{E}\mathcal{E}}$ appears in (9).

It is important to stress that the LODF submatrix $K_{-\mathcal{E}\mathcal{E}}$ is *not* the GLODF matrix $K^{\mathcal{E}}$. Indeed the matrix $K_{-\mathcal{E}\mathcal{E}}$ is a compact way to write all the LODF factors $K_{\ell\hat{\ell}}$ corresponding to the *individual* line outages $\hat{\ell} \in \mathcal{E}$ and does *not* give the sensitivity of power flows in the post-contingency network to a set $\mathcal{E}$ of *simultaneous* line outages. For this reason, each column of $K_{-\mathcal{E}\mathcal{E}}$ must be interpreted *separately*: column $\hat{\ell}$ gives the power flow changes on each surviving line $\ell \in -\mathcal{E}$ due to the outage of a *single* non-bridge line $\hat{\ell} \in \mathcal{E}$. Nonetheless, combining (9) and (12), we can show that in the case of a non-cut set $\mathcal{E}$, the GLODF matrix $K^{\mathcal{E}}$ and the LODF submatrix $K_{-\mathcal{E}\mathcal{E}}$ are still related as follows:

$$K^{\mathcal{E}} = K_{-\mathcal{E}\mathcal{E}}\left(I - \operatorname{diag}(D_{\mathcal{E}\mathcal{E}})\right)\left(I - D_{\mathcal{E}\mathcal{E}}\right)^{-1}. \tag{13}$$

This expression shows that if $\mathcal{E} = \{\hat{\ell}\}$ is a singleton, then the two matrices coincides, but in general $K^{\mathcal{E}} \neq K_{-\mathcal{E}\mathcal{E}}$.

3.3 Islanding and GLODF for cut sets

In this subsection we study the impact of cut set outages, extending the results of the previous subsection to the scenario where a set of lines trip simultaneously disconnecting the network into two or more connected components or islands. We first focus on the case of a general cut set outage and then later, in Subsection 3.3.1, specify our results in the case in which the cut set consist of a single line, i.e., the case of a bridge outage.

Consider a subset of lines $\mathcal{E} \subset E$ that is a cut set for $G = (V, E)$, which means that the surviving network $G^{\mathcal{E}} := (V, E \backslash \mathcal{E})$ consists of two or more islands $\mathcal{I}_1, \ldots, \mathcal{I}_k$. A *power balancing rule* needs to be invoked to rebalance power on each island after the contingency. We assume such operations on the islands are decoupled, so that we can study the power flow redistribution in each of them separately. Therefore, for the purpose of presentation, let us focus on one of these islands, say $\mathcal{I}$.

We can distinguish three types of failed lines and partition $\mathcal{E} = \mathcal{E}_{\mathcal{I}} \cup \mathcal{E}_{\partial\mathcal{I}} \cup \mathcal{E}_{\mathcal{I}^c}$ accordingly as follows:

- $\mathcal{E}_{\mathrm{int}} = \mathcal{E}_{\mathcal{I}} := \{(i,j) \in \mathcal{E} \;:\; i \in \mathcal{I} \text{ and } j \in \mathcal{I}\}$ is the subset of failed *internal lines*, i.e., those both endpoints of which belong to $\mathcal{I}$;

- $\mathcal{E}_{\mathrm{tie}} = \mathcal{E}_{\partial\mathcal{I}} := \{(i,j) \in \mathcal{E} \;:\; i \notin \mathcal{I} \text{ and } j \in \mathcal{I}\}$ is the subset of failed *tie lines*, i.e., those that have exactly one endpoint in $\mathcal{I}$;

- $\mathcal{E}_{\mathcal{I}^c}$ is the subset of failed lines neither endpoints of which belongs to $\mathcal{I}$.

Any outaged line in $\mathcal{E}_{\mathcal{I}^c}$ does not have a direct impact on the island $\mathcal{I}$ and can thus be ignored and thus we can assume without loss of generality that $\mathcal{E} = \mathcal{E}_{\text{tie}} \cup \mathcal{E}_{\text{int}}$.

A main difference between a bridge outage and a cut-set outage is that in the former case $\mathcal{E} = \mathcal{E}_{\text{tie}}$ consists of a single tie line, whereas in the latter case $\mathcal{E}$ may contain internal lines in $\mathcal{E}_{\text{int}}$ as well. Since $\mathcal{I}$ is connected, $\mathcal{E}_{\text{int}}$ is a non-cut set. The power flows on surviving lines in $\mathcal{I}$ are impacted by both internal line outages in $\mathcal{E}_{\text{int}}$ and by tie line outages in $\mathcal{E}_{\text{tie}}$.

Denote by $\boldsymbol{p}_I := (p_i, i \in \mathcal{I})$ the pre-contingency injections in $\mathcal{I}$. The power injections $\boldsymbol{p}_I$ may not be balanced inside the island, as there could be a non-zero total power imbalance

$$b_{\mathcal{I}} := \mathbf{1}^T \boldsymbol{p}_{\mathcal{I}} = \sum_{i \in \mathcal{I}} p_i = - \sum_{\hat{\ell} \in \mathcal{E}_{\text{tie}}} f_{\hat{\ell}}.$$

If $b_{\mathcal{I}} < 0$, then it means that pre-contingecy the island was importing power, while in the opposite case, the island was exporting power. In either case, if $b_{\mathcal{I}} \neq 0$, the network operator has to intervene with a *balancing rule* that, by adjusting injections (generators and/or loads) in some of the nodes, ensures that the power balance is restored in $\mathcal{I}$.

A popular balancing rule, named *proportional control*, is to share the imbalance among a set of participating nodes with a fixed proportion. Specifically a proportional control $\mathbb{G}_{\boldsymbol{\alpha}}$ is defined by a nonnegative vector $\boldsymbol{\alpha} := (\alpha_i, i \in \mathcal{I})$ such that $\sum_{i \in \mathcal{I}} \alpha_i = 1$, with the interpretation that, post contingency, each node $i \in \mathcal{I}$ adjusts its injection by the amount $\Delta p_i := -\alpha_i b_{\mathcal{I}}$. The injection adjustments vector is then

$$\Delta \boldsymbol{p}_{\boldsymbol{\alpha}} := (\Delta p_i, i \in \mathcal{I}) = -b_{\mathcal{I}} \sum_{i \in \mathcal{I}} \alpha_i \boldsymbol{e}_i, \tag{14}$$

where $\boldsymbol{e}_i$ is the standard unit vector of size $n_{\mathcal{I}}$. We refer to each node $i \in \mathcal{I}$ such that $\alpha_i > 0$ as *participating node*. All participating nodes adjust their injections in the same direction, whose sign depends on the sign of $b_{\mathcal{I}}$. The injections are therefore changed from $\boldsymbol{p}$ to a post-contingency vector $\boldsymbol{p} + \Delta \boldsymbol{p}_{\alpha}$. The proportional control rebalances the power on $\mathcal{I}$: the assumption $\sum_{i \in \mathcal{I}} \alpha_i = 1$ guarantees that the post-contingency injections are zero-sum, as

$$\mathbf{1}^T(\boldsymbol{p} + \Delta \boldsymbol{p}_\alpha) = \mathbf{1}^T(\boldsymbol{p} - b_{\mathcal{I}} \sum_{i \in \mathcal{I}} \alpha_i \boldsymbol{e}_i) = b_{\mathcal{I}} - b_{\mathcal{I}} = 0.$$

We now derive the impact of the cut set $\mathcal{E}$ outage on the surviving lines in island $\mathcal{I}$. Let $\boldsymbol{f}$ be the pre-contingecy power flows on the lines in $\mathcal{I}$ and further partition $\boldsymbol{f} = (\boldsymbol{f}_{-\text{int}}, \boldsymbol{f}_{\text{int}})$, where $\boldsymbol{f}_{\text{int}} := (f_{\ell}, \ell \in \mathcal{E}_{\text{int}})$ denote the pre-contingency power flows on the outaged internal lines, $\boldsymbol{f}_{-\text{int}} := (f_{\ell}, \ell \in E_{\mathcal{I}} \backslash \mathcal{E}_{\text{int}})$ those on the surviving lines in $\mathcal{I}$. Let $B, C, L^{\dagger}$ denote the susceptance matrix, the incidence matrix and the graph Laplacian pseudo-inverse associated with the island $\mathcal{I}$. We also partition the matrices B, C according to surviving lines and outaged internal lines as $B =: \text{diag}(B_{-\text{int}}, B_{\text{int}})$ and $C =: \begin{bmatrix} C_{-\text{int}}, & C_{\text{int}} \end{bmatrix}$.

When multiple lines are tripped from the grid simultaneously disconnecting the network, the aggregate impact in general is different from the aggregate effect of tripping the lines separately. Nonetheless, as the following result shows, the impact from balancing rule turns out to be separable from the rest.

Theorem 3.8 (GLODF for cut set with proportional control $G_{\boldsymbol{\alpha}}$). *The net change on the surviving lines of the island $\mathcal{I}$ due to a cut set outage $\mathcal{E} = \mathcal{E}_{\text{tie}} \cup \mathcal{E}_{\text{int}}$ is given by*

$$\Delta \boldsymbol{f}_{-\text{int}} = K^{\mathcal{E}_{\text{int}}} \boldsymbol{f}_{\text{int}} + \sum_{\hat{\ell} \in \mathcal{E}_{\text{tie}}} f_{\hat{\ell}} \sum_{k : \alpha_k > 0} \alpha_k \left(D_{-\text{int}, kj(\hat{\ell})} + K^{\mathcal{E}_{\text{int}}} D_{\text{int}, kj(\hat{\ell})} \right), \tag{15}$$

where $j(\hat{\ell})$ is the endpoint of tie line $\hat{\ell}$ that lies in $\mathcal{I}$.

For a proof, the reader can look at [58, Theorem 5] and eq. (12) therein. The RHS of (15) consists of two terms:

$$\Delta \boldsymbol{f}_{-\text{int}} = \underbrace{K^{\mathcal{E}_{\text{int}}} \boldsymbol{f}_{\text{int}}}_{\text{int. line } \mathcal{E}_{\text{int}} \text{ outage}} + \underbrace{\sum_{\hat{\ell} \in \mathcal{E}_{\text{tie}}} f_{\hat{\ell}} \sum_{k : \alpha_k > 0} \alpha_k \left(D_{-\text{int}, kj(\hat{\ell})} + K^{\mathcal{E}_{\text{int}}} D_{\text{int}, kj(\hat{\ell})} \right)}_{\text{tie line } \mathcal{E}_{\text{tie}} \text{ outage}}.$$

The first term on the RHS accounts for the impact of the outage of the non-cut set $\mathcal{E}_{\text{int}}$ of internal lines in $\mathcal{I}$ through the GLODF $K^{\mathcal{E}_{\text{int}}} := B_{\text{int}} C_{-\text{int}}^T L \, C_{\text{int}} (I - B_{\text{int}} C_{\text{int}}^T L^{\dagger} C_{\text{int}})^{-1}$ (cf. Theorem 3.5). The second term captures the impact of the proportional control $\mathbb{G}_{\alpha}$ in response to tie line outages in $\mathcal{E}_{\text{tie}}$, which also accounts for the mimicked power injection change due to the outaged tie lines $\hat{\ell} \in \mathcal{E}_{\text{tie}}$ at their endpoints $j(\hat{\ell})$ in $\mathcal{I}$. We stress that the impact of the proportional control $\mathbb{G}_{\alpha}$ through internal lines in $\mathcal{E}_{\text{int}}$ is scaled by the GLODF $K^{\mathcal{E}_{\text{int}}}$.

Note that if there are no tie line outages, i.e., $\mathcal{E}_{\text{tie}} = \emptyset$, then (15) reduces to Theorem 3.5 for a non-cut set outage. If there are no internal lines $\mathcal{E}_{\text{int}} = \emptyset$, the first term of (15) is zero and the second term simplifies, so that

$$\Delta \boldsymbol{f}_{-\text{int}} = \sum_{\hat{\ell} \in \mathcal{E}_{\text{tie}}} f_{\hat{\ell}} \sum_{k : \alpha_k > 0} \alpha_k \, D_{-\text{int}, kj(\hat{\ell})}.$$

3.3.1 Bridge outage

Consider now the case of a cut set outage, in which the $\mathcal{E}$ consist of a single line. This means that $\mathcal{E}_{\text{int}} = \emptyset$, $\mathcal{E}_{\text{tie}} = \{\hat{\ell}\}$, and $\hat{\ell} = (\hat{i}, \hat{j})$ is a bridge. After the outage of line ℓ, the power network gets disconnected into two islands, one containing node $\hat{i}$ and one containing node $\hat{j}$, which we denote by $\mathcal{I}_{\hat{i}}$ and $\mathcal{I}_{\hat{j}}$ respectively.

If the pre-contingency power flow $f_{\hat{\ell}} = 0$ on the outaged line $\hat{\ell}$, then the operations of these islands, as modeled by the DC power flow equations, are decoupled pre-contingency and therefore remain unchanged post-contingency. We henceforth consider only the case where $f_{\hat{\ell}} \neq 0$, which means that there is a nonzero power imbalance over each of the two islands post contingency. For example, if $f_{\hat{\ell}} > 0$, i.e., the power flows from island $\mathcal{I}_{\hat{i}}$ to island $\mathcal{I}_{\hat{j}}$ pre contingency, then there is a power surplus of size $f_{\hat{\ell}}$ on $\mathcal{I}_{\hat{i}}$ and a shortage of size $f_{\hat{\ell}}$ on $\mathcal{I}_{\hat{j}}$ post contingency. Like before, since the two islands operate independently post contingency, each according to DC power flow equations, we can focus only on one island, say $\mathcal{I}_{\hat{j}}$. Let $\Delta \boldsymbol{f} := (\Delta f_\ell, \ell \in E_{\mathcal{I}_{\hat{j}}})$ be the vector of net power flows changes on all lines in island $\mathcal{I}_{\hat{j}}$. Then, (15) simplifies to

$$\Delta \boldsymbol{f} = f_{\hat{\ell}} \sum_{k \in \mathcal{I}_{\hat{j}}} \alpha_k \left(D_{E_{\mathcal{I}_{\hat{j}}}, k\hat{j}} \right).$$

More explicitly, for any line $\ell = (i, j) \in E_{\mathcal{I}_{\hat{j}}}$ we have

$$\frac{\Delta f_\ell}{f_{\hat{\ell}}} = \sum_{k : \alpha_k > 0} \alpha_k D_{\ell, k\hat{j}} = \sum_{k \in \mathcal{I}_{\hat{j}}} \alpha_k b_\ell \left(L_{ik}^\dagger + L_{j\hat{j}}^\dagger - L_{i\hat{j}}^\dagger - L_{jk}^\dagger \right),$$

where the last equality follows from Theorem 3.1. Therefore the power flow change on line ℓ is the superposition of two factors: (i) injecting $\alpha_k f_{\hat{\ell}}$ at participating nodes $k \in \mathcal{I}_{\hat{j}}$ and (ii) withdrawing $f_{\hat{\ell}}$ at node $\hat{j}$. Using the last identity, we can thus extend the definition of LODF $K_{\ell\hat{\ell}}$ to allow for outages of any bridge $\hat{\ell} = (\hat{i}, \hat{j})$ with nonzero flow $f_{\hat{\ell}} \neq 0$ under the proportional control $\mathbb{G}_{\boldsymbol{\alpha}}$ as follows

$$K_{\ell\hat{\ell}} := \frac{\Delta f_\ell}{f_{\hat{\ell}}} = \sum_{k \,:\, \alpha_k > 0} \alpha_k D_{\ell, k\hat{j}}, \qquad \text{for all } \ell \in E_{\mathcal{I}_{\hat{j}}}. \tag{16}$$

4 Localization results

In this section we show how the block and bridge-block decompositions of the power network G fully captures the power redistribution effects after a contingency and give a very transparent view of the global network robustness against failures. More specifically, we characterize when LODF and GLODF are nonzero in terms of these network structures. The highlight is that an outage line has only a local impact which is confined in the (bridge-)block to which it belongs. We present the results for non-cut set and cut set outages separately in the next two subsections.

4.1 Non-cut set outages

The goal of this subsection is to show that the block decomposition of the network G induces a block-diagonal structure for the PTDF and LODF matrices. The key idea is using the representation of these factors in terms of spanning forest and the intimate relation between the block decomposition and the cycles in the graph G. Lastly, we show that when the considered outage $\mathcal{E}$ is a non-cut set, the GLODF matrix $K^{\mathcal{E}}$ inherits the block-diagonal structure of the PTDF matrix.

The next lemma states the *Simple Cycle Criterion*, which has been proved in [57, Theorem 8]. It provides a sufficient condition for the LODF to be zero expressed purely in topological terms, namely using the cycles of the power network G.

Lemma 4.1 (Simple Cycle Criterion)**.** *Consider a non-bridge line $\hat{\ell} \in E$ and any other line $\ell \in E$. If there is no simple cycle in G that contains both ℓ and $\hat{\ell}$, then the LODF $K_{\ell\hat{\ell}} = 0$.*

Used in combination with the network block decomposition introduced in Section 2.1, this criterion can be used to show that the PTDFs and LODFs are always zero when they refer to two lines that belong to different blocks of the power network G.

Proposition 4.2 (LODF and PTDF characterization for non-bridge single line failure)**.** *Consider a single non-bridge line $\hat{\ell} = (\hat{i}, \hat{j})$ outage. For any surviving line $\ell = (i, j) \neq \hat{\ell}$ the following statements hold:*

(i) *The PTDF $D_{\ell\hat{\ell}} = 0$ if and only if the LODF $K_{\ell\hat{\ell}} = 0$.*

(ii) If ℓ and $\hat\ell$ are in different blocks of G, then $K_{\ell\hat\ell} = 0$ and $D_{\ell\hat\ell} = 0$.

To formally state the next results for the distribution factor matrices when $\mathcal{E}$ is a non-cut set outage, we first need some preliminary definitions to partition outage lines in $\mathcal{E}$ and surviving lines in $-\mathcal{E}$ depending on which block they belong to. More specifically, let $E = E_1 \cup \cdots \cup E_k$ be the block decomposition of (the edge set E of) the network G. We partition accordingly into k disjoint subsets both the outaged line set $\mathcal{E}$ and the surviving lines set $-\mathcal{E}$ as follows:

$$\mathcal{E} = \bigcup_j \mathcal{E}_j \quad \text{with} \quad \mathcal{E}_j := \mathcal{E} \cap E_j, \qquad j = 1, \ldots, k, \tag{17a}$$

$$-\mathcal{E} = \bigcup_j \mathcal{E}_{-j} \quad \text{with} \quad \mathcal{E}_{-j} := -\mathcal{E} \cap E_j = E_j \setminus \mathcal{E}_j, \qquad j = 1, \ldots, k. \tag{17b}$$

Without loss of generality we assume that the lines in $\mathcal{E}$ are indexed so that the outaged lines are ordered depending on which subset $\mathcal{E}_j$ they belong to. Similarly, the surviving lines in $-\mathcal{E}$ are indexed so that the surviving lines are ordered depending on which subset $-\mathcal{E}_j$ they belong to. In view of the partition in (17), the submatrices $B_{-\mathcal{E}}$, $B_{\mathcal{E}}$, $C_{-\mathcal{E}}$ and $C_{\mathcal{E}}$ appearing in (6) can be further decomposed as

$$B = \begin{bmatrix} B_{-\mathcal{E}} & 0 \\ 0 & B_{\mathcal{E}} \end{bmatrix} =: \begin{bmatrix} \text{diag}\,(B_{-j}, j = 1, \ldots, k) & 0 \\ 0 & \text{diag}\,(B_j, j = 1, \ldots, k) \end{bmatrix}, \tag{18}$$

$$C = \begin{bmatrix} C_{-\mathcal{E}} & C_{\mathcal{E}} \end{bmatrix} =: \begin{bmatrix} C_{-1} & \cdots & C_{-k} & C_1 & \cdots & C_k \end{bmatrix}. \tag{19}$$

In particular, C_j and C_{-j} are the incidence matrices for the lines $\mathcal{E}_j$ and $\mathcal{E}_{-j}$, respectively.

Proposition 4.3 (Block diagonal structure of LODF and PTDF submatrices)**.** *Let $\mathcal{E}$ be non-cut set outage for the power network $G = (V, E, \boldsymbol{b})$. Then the two submatrices $D_{-\mathcal{E}\mathcal{E}}$ and $D_{\mathcal{E}\mathcal{E}}$ of the PTDF matrix defined in (7) decompose as follows:*

$$D_{-\mathcal{E}\mathcal{E}} =: \begin{bmatrix} D_{-1} & 0 & \ldots & 0 \\ 0 & D_{-2} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & D_{-k} \end{bmatrix} \quad and \quad D_{\mathcal{E}\mathcal{E}} =: \begin{bmatrix} D_1 & 0 & \ldots & 0 \\ 0 & D_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & D_k \end{bmatrix} \tag{20a}$$

where, for each $j = 1, \ldots, k$, D_{-j} is the $|\mathcal{E}_{-j}| \times |\mathcal{E}_j|$ matrix given by $D_{-j} := B_{-j} C_{-j}^T L^\dagger C_j$, and D_j is the $|\mathcal{E}_j| \times |\mathcal{E}_j|$ matrix given by $D_{-j} := B_j C_j^T L^\dagger C_j$.

Furthermore, also the LODF matrix $K_{-\mathcal{E}\mathcal{E}}$ defined as in (12) has a block-diagonal structure, namely

$$K_{-\mathcal{E}\mathcal{E}} =: \begin{bmatrix} K_1 & 0 & \ldots & 0 \\ 0 & K_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & K_k \end{bmatrix}, \tag{20b}$$

where K_j is the $|\mathcal{E}_{-j}| \times |\mathcal{E}_j|$ matrix given by $K_j := D_{-j} (I - \text{diag}(D_j))^{-1}$, for $j = 1, \ldots, k$.

Proof. Recall from (7) that $D_{-\mathcal{E}\mathcal{E}} = B_{-\mathcal{E}} C_{-\mathcal{E}}^T L^\dagger C_{\mathcal{E}}$ and $D_{\mathcal{E}\mathcal{E}} = B_{\mathcal{E}} C_{\mathcal{E}}^T L^\dagger C_{\mathcal{E}}$. Proposition 4.2 then implies that the PTDF submatrices $D_{-\mathcal{E}\mathcal{E}}$ and $D_{\mathcal{E}\mathcal{E}}$ decompose into a block-diagonal structure corresponding to the blocks of G, obtaining (20a). Since $K_{-\mathcal{E}\mathcal{E}} = D_{-\mathcal{E}\mathcal{E}}(I - \text{diag}(D_{\mathcal{E}\mathcal{E}}))^{-1}$ in view of identity (12), the LODF submatrix $K_{-\mathcal{E}\mathcal{E}}$ has the same block diagonal structure as $D_{-\mathcal{E}\mathcal{E}}$. The invertibility of $I - \text{diag}(D_j) = \text{diag}(1 - D_{\hat\ell\hat\ell}, \ell \in \mathcal{E}_j)$ follows from Proposition 3.2(ii) and Corollary 3.3, since every line $\hat\ell \in \mathcal{E}_j \subset \mathcal{E}$ is not a bridge and thus $D_{\ell\ell} < 1$. $\square$

The block decomposition of the network G is crucial for failure localization also when a non-cut set $\mathcal{E}$ of lines are disconnected simultaneously. More specifically, the next result shows that even in this scenario the impacts of line outages are still contained in their own blocks.

Proposition 4.4 (GLODF characterization for non-cut set outage)**.** *Let $\mathcal{E}$ be non-cut set outage for the power network $G = (V, E, \boldsymbol{b})$. Then, the rectangular $|-\mathcal{E}| \times |\mathcal{E}|$ GLODF matrix $K^{\mathcal{E}}$ has a block-diagonal structure, i.e.,*

$$K^{\mathcal{E}} =: \begin{bmatrix} K_1^{\mathcal{E}} & 0 & \ldots & 0 \\ 0 & K_2^{\mathcal{E}} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & K_k^{\mathcal{E}} \end{bmatrix}, \tag{21a}$$

where each $K_j^{\mathcal{E}}$ is $|\mathcal{E}_{-j}| \times |\mathcal{E}_j|$ and involves lines only in block E_j of G, given by:

$$K_j^{\mathcal{E}} := B_{-j} C_{-j}^T L^\dagger C_j \left(I - B_j C_j^T L^\dagger C_j \right)^{-1}, \qquad j = 1, \dots, k, \tag{21b}$$

where the submatrices B_j, B_{-j}, C_j and C_{-j} have been obtained by partitioning the matrices B and C according to the block structures of both $-\mathcal{E}, \mathcal{E}$. Equivalently, we have

$$K_j^{\mathcal{E}} = D_{-j}(I - D_j)^{-1} = K_j \left(I - \mathrm{diag}(D_j) \right) \left(I - D_j \right)^{-1}, \qquad j = 1, \dots, k. \tag{21c}$$

where $D_j, D_{-j}, K_j, j = 1, \dots, k$, are the submatrices defined in Proposition 4.3. Furthermore, for every $\ell \in -\mathcal{E}$ and $\hat{\ell} \in \mathcal{E}$, if ℓ and $\hat{\ell}$ are in different blocks of G, then the GLODF $K_{\ell\hat{\ell}}^{\mathcal{E}} = 0$.

Proof. The proof of (21) readily follows from Theorem 3.5 after rewriting all the various submatrices resulting from the block decomposition and using Proposition 4.3. The last statement is an immediate consequence of (21). $\qquad \square$

We remark that, differently for the case of a single non-bridge failure in Proposition 4.2, the PTDFs and GLODFs are not always simultaneously zero. Specifically, when multiple lines fail simultaneously, the fact that GLODF $K_{\ell\hat{\ell}}^{\mathcal{E}} = 0$ does not imply that the PTDF $D_{\ell\hat{\ell}} = 0$ or vice versa. Consider the counterexample in Fig. 2 where two dashed lines ℓ_1 and ℓ_2 fail simultaneously, i.e., $\mathcal{E} = \{\ell_1, \ell_2\}$. Assuming all the lines have unit susceptances, the resulting absolute values of PTDF and GLODF are computed as in Table. 1. In particular, $K_{\ell_6,\ell_1}^{\mathcal{E}} = K_{\ell_3,\ell_2}^{\mathcal{E}} = K_{\ell_4,\ell_2}^{\mathcal{E}} = 0$, while the corresponding PTDFs are non-zero. We remark that such counterexamples may happen when the outage set $\mathcal{E}$ is such that the block decomposition of the surviving network has more blocks than the original network (in our example, they are 4 and 1, respectively).
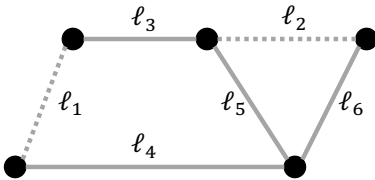

Figure 2: Counterexample with the lines in the non-cut outage set $\mathcal{E} = \{\ell_1, \ell_2\}$ displayed as dashed.

$\hat{\ell}$ \ ℓ	ℓ_3	ℓ_4	ℓ_5	ℓ_6
ℓ_1	3/11, 1	3/11, 1	2/11, 1	1/11, 0
ℓ_2	1/11, 0	1/11, 0	3/11, 1	4/11, 1

Table 1: The distribution factors for all the lines reported as pairs (PTDF $D_{\ell,\hat{\ell}}$, GLODF $K_{\ell,\hat{\ell}}^{\mathcal{E}}$).

Nonetheless, the following proposition lists sufficient conditions for both implications to hold.

Proposition 4.5 (Zero GLODF and PTDF characterization). *Consider a non-cut set outage $\mathcal{E}$ and a surviving line $\ell = (i, j) \notin \mathcal{E}$. The following statements hold:*

(i) If the PTDF $D_{\ell\hat{\ell}} = 0$ holds for any set of susceptances B, then the GLODF $K_{\ell\hat{\ell}}^{\mathcal{E}} = 0$.

(ii) If the GLODF $K_{\ell\hat{\ell}}^{\mathcal{E}} = 0$, then either the PTDF $D_{\ell\hat{\ell}} = 0$ or every cycle in pre-contingency network G that containing ℓ and $\hat{\ell}$ pass through at least one outage line different from $\hat{\ell}$.

4.1.1 Visualizing LODFs using influence graphs

The analysis of the LODFs corroborates a well-known fact in power systems, namely how after a particular line outage the next line to fail may be very distant topologically and geographically, as illustrated by [30]. Classical contagion model in which failures spread locally from a node to its immediate neighbors are thus not appropriate to describe cascading failures on power networks (also due to the fact that line failures are an order of magnitude more likely than node outages).

A better way to capture the interdependencies between (single) line failures in power networks is by means of a specific type of "dual graph", usually referred to as *(failure) influence graph* [69, 70] or *interaction graph* [99, 100]. There are many possible ways to define such a graph, but the high-level idea is that an influence graph $\mathcal{L}(G)$ is derived from the original power network G as follows: the vertex set of $\mathcal{L}(G)$ is in one-to-one correspondence with the set E of the lines of the power network G and the pair of vertices corresponding to the lines $\ell, \hat{\ell}$ are connected in $\mathcal{L}(G)$ if and only if the failure of line $\hat{\ell}$ has a severe impact and/or causes and/or is correlated the failure of line ℓ. The edges (or the edge weights) of the influence graph $\mathcal{L}(G)$ can thus be inferred by historical data or by simulating cascades. In this paper, we consider a version of the influence graph defined using LODFs very similar to the one in [94], which offers a powerful way to visualize the potential risk of failure propagation. More specifically, the vertices corresponding to the lines $\ell, \hat{\ell}$ are connected in $\mathcal{L}(G)$ if and only if LODF is larger than a desired threshold, i.e., $|K_{\ell\hat{\ell}}| > K_{\min}$.
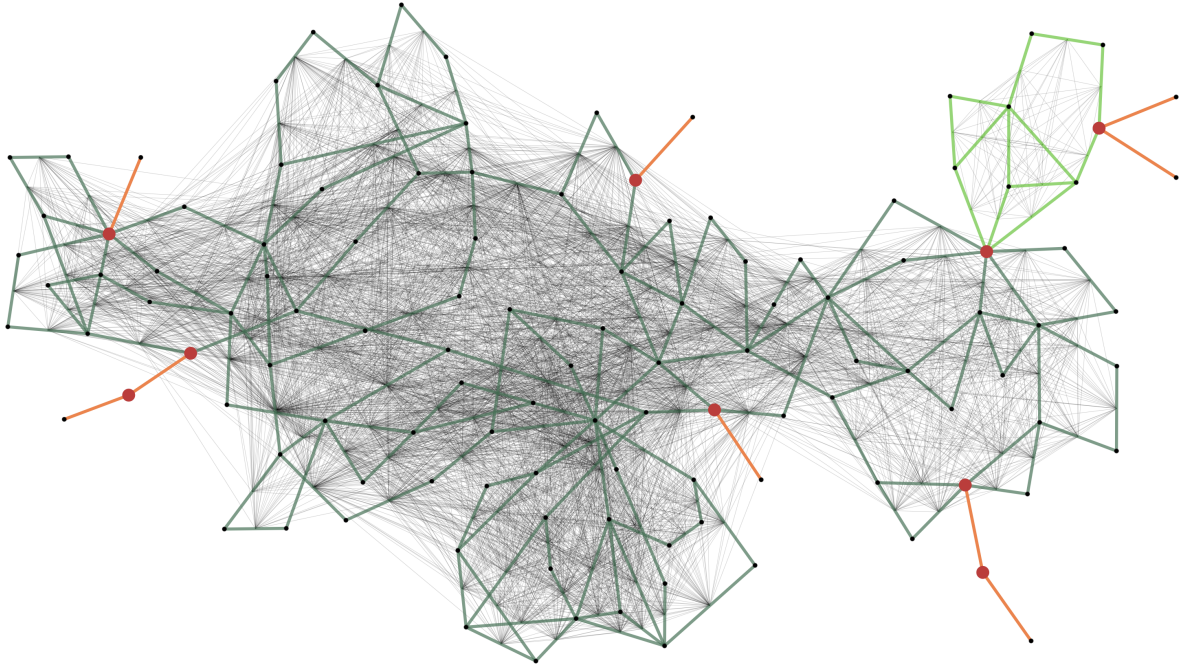
(a) The influence graph (in gray) obtained using the threshold $K_{\min} = 0.005$. Due to the presence of a block, the influence graph has two clearly visible disconnected components, since all the corresponding LODFs are zero.
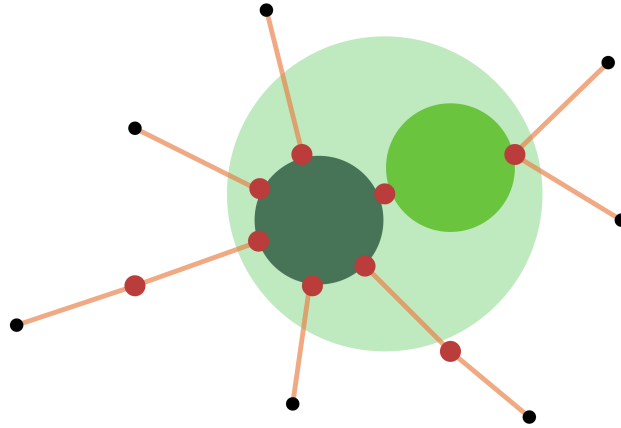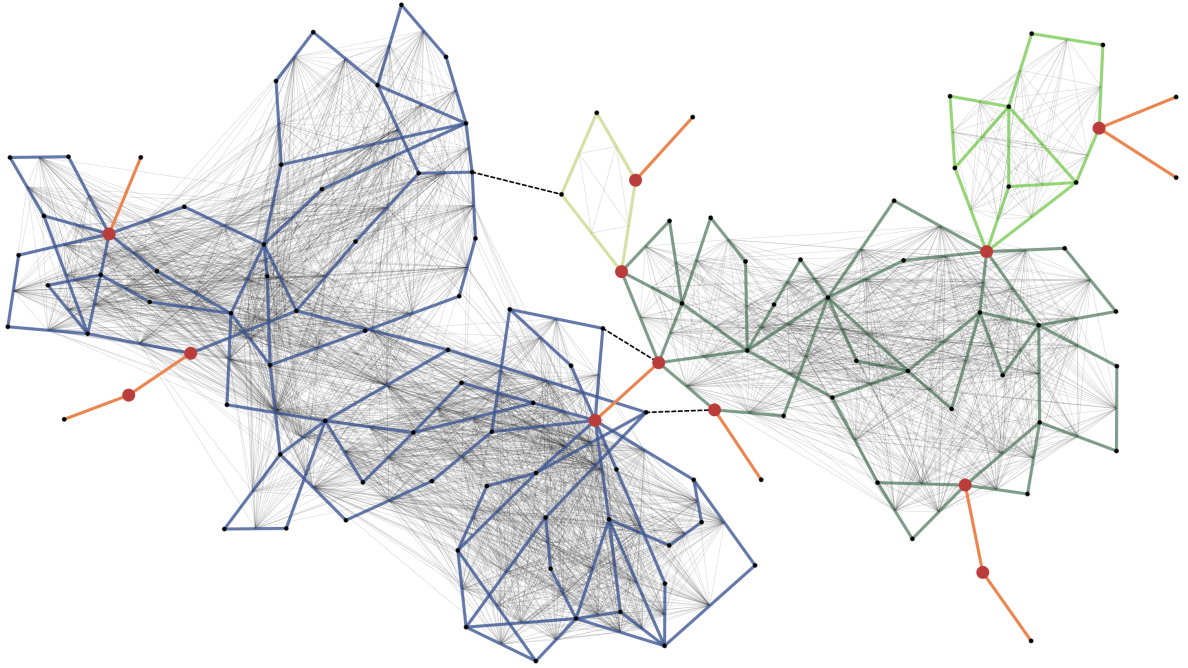

(b) Schematic representation of the block and bridge-block decompositions of the original IEEE 118-node test network. The nontrivial bridge-block is further decomposed by a cut vertex into two blocks (in two different shades of green)
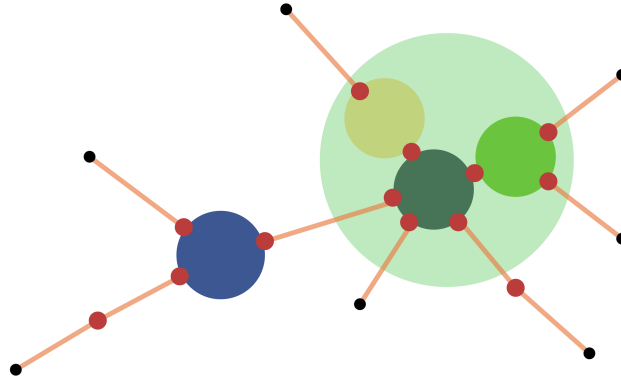
Figure 3: Original IEEE 118-node test network. It has 9 bridges (in orange), one nontrivial bridge-block (in green), and 9 trivial bridge-blocks. Furthermore, it has 9 cut vertices (in red), one of which decomposes the nontrivial bridge-block into 2 blocks (in different shades of green).

As illustrated for the IEEE 118-node test network in Fig. 3(a), the influence graph of power network tend to be very dense and connecting many lines that are topologically far away, corroborating the non-local propagation of line failures. Fig. 3(b) shows the block and bridge-block decomposition of the same network. This network has a trivial bridge-block decomposition, consisting of a single nontrivial bridge-block and 9 trivial ones. This bridge-block consists of two nontrivial blocks, separated by a cut vertex. As predicted by the theory (cf. Proposition 4.2), the influence graph is not connected, since all the LODFs corresponding to pairs of lines in the two blocks are identically zero.

By removing three lines from IEEE 118-node test network, we can make both the block and bridge-block decompositions of this network finer, as illustrated in Fig. 4. The influence graph of the pruned network is now less dense and have more disconnected components, showing a higher localization capability. Fig. 4(b) shows the finer block and bridge-block decomposition of the pruned network. It has two nontrivial bridge-blocks, one of which further decomposes into four nontrivial blocks, separated by cut vertices.

(a) The influence graph (in gray) of the pruned IEEE 118-node test network obtained using the threshold $K_{\min} = 0.005$. Due to the presence of four blocks (in blue and three shades of green), the influence graph has four clearly visible disconnected components, since all the corresponding LODFs are zero.

(b) Schematic representation of the block and bridge-block decomposition of the pruned IEEE 118-node test network. The green bridge-block further is further decomposed by cut vertices into three blocks (in different shades of green).

Figure 4: The network obtained from the IEEE 118-node test network removing three lines (shown as dashed black edges). It has 10 bridges (in orange), two nontrivial bridge-block (in blue and green), and 9 trivial bridge-blocks. Furthermore, it has 13 cut vertices (in red), two of which decomposes the nontrivial green bridge-block into 3 blocks (in three different shades of green).

4.2   Cut set outages: Power rebalance and redistribution

We now discuss the impact of cut set outages. As mentioned in Subsection 3.3, we assume the post-contingency power balancing rule is decoupled for each island so that we can study the power redistribution in each of them separately. In particular, we consider the proportional control $\mathbb{G}_\alpha$ that the post-contingency power injection is adjusted proportionally to compensate the power imbalance caused by the cut set outages.

Consider an island $\mathcal{I}$ and let $E_\mathcal{I} := \{(i, j) \in E \ : \ i \in \mathcal{I} \text{ and } j \in \mathcal{I}\}$ denote the set of edges whose both end points belong to $\mathcal{I}$ before contingency. As outlined in Subsection 3.3, we partition the failed lines into tie line failures $\mathcal{E}_{\text{tie}}$ and internal line failures $\mathcal{E}_{\text{int}}$. Rewriting (15) for a specific surviving line $\ell \in E_\mathcal{I} \setminus \mathcal{E}_{\text{int}}$ in island $\mathcal{I}$ yields

$$\Delta f_\ell = \sum_{\hat{\ell} \in \mathcal{E}_{\text{int}}} K_{\ell\hat{\ell}}^{\mathcal{E}_{\text{int}}} f_{\hat{\ell}} \ + \ \sum_{\hat{\ell} \in \mathcal{E}_{\text{tie}}} f_{\hat{\ell}} \sum_{k : \alpha_k > 0} \alpha_k \Big(D_{\ell, kj(\hat{\ell})} + \sum_{\hat{\ell}' \in \mathcal{E}_{\text{int}}} K_{\ell\hat{\ell}'}^{\mathcal{E}_{\text{int}}} D_{\hat{\ell}', kj(\hat{\ell})} \Big). \tag{22}$$

It is observed that the impact of cut set outages consists of two terms, one capturing the impact of the internal line failures in $\mathcal{I}$ and the other capturing the impact of the tie line failures, as shown in (22). In order

to describe the impact boundary of such cut set outages, we will study the conditions so that (22) is zero.

Decompose the pre-contingency subnetwork corresponding to the island $\mathcal{I}$ into blocks. For any surviving line ℓ and failed internal line $\hat{\ell} \in \mathcal{E}_{\text{int}}$ pair, Proposition 4.4 indicates that $K_{\ell\hat{\ell}}^{\mathcal{E}_{\text{int}}} = 0$ if ℓ and $\hat{\ell}$ are in different blocks of $\mathcal{G}_{\mathcal{I}}$. Therefore, if a surviving line ℓ is not in the same block as any outaged internal lines, (22) simplifies to

$$\Delta f_\ell = \sum_{\hat{\ell} \in \mathcal{E}_{\text{tie}}} f_{\hat{\ell}} \sum_{k:\alpha_k > 0} \alpha_k D_{\ell, kj(\hat{\ell})}$$

This suggests that the impact of multiple outaged tie lines is separable. In the following lemma we state the *Simple Path Criterion*, cf. [58, Theorem 1], which provides a sufficient condition for the PTDF introduced in Subsection 3.1 to be zero in purely topological terms.

Lemma 4.6 (Simple Path Criterion)**.** *For any line $\ell \in E$ and two nodes $j, k \in V$, if ℓ there is no simple path in the power network $G = (V, E, \boldsymbol{b})$ connecting nodes j and k that contains line ℓ, then the PTDF $D_{\ell, jk} = 0$.*

Leveraging this criterion and the notion of participating bus introduced in Subsection 3.3, we can derive a sufficient condition, summarized in the following proposition [58, Corollary 6], for a surviving line ℓ not to be impacted by a cut set outage.

Proposition 4.7 (Characterization for cut set failures)**.** *Consider a cut set $\mathcal{E}$ outage that disconnects the power network into two or more islands with decoupled power balancing rules. Let $\mathcal{I}$ denote one such island under the proportional control $\mathbb{G}_\alpha$, and let $\mathcal{E}_{\mathcal{I}} = \mathcal{E}_{\text{tie}} \cup \mathcal{E}_{\text{int}}$ be the set of outaged lines with one or both endpoints in this island. For any surviving line ℓ, if ℓ is not in the same block as any outaged internal lines $\hat{\ell} \in \mathcal{E}_{\text{int}}$ and ℓ is not on any simple path connecting the endpoint of an outaged tie line $j(\hat{\ell})$ and a participating bus k with $\alpha_k > 0$ for any $\hat{\ell} \in \mathcal{E}_{\text{tie}}$, then the post-contingency flow on line ℓ remains unchanged.*

We can reformulate the latter result in even simpler terms in the case of a single bridge line outage, obtaining the following simplified characterization for the surviving lines whose post-contingency flow remains unchanged, see [58, Corollary 2].

Corollary 4.8 (Characterization for bridge failures)**.** *Consider the outage of a bridge line $\hat{\ell} = (\hat{i}, \hat{j})$ with nonzero flow $f_{\hat{\ell}} \neq 0$ that disconnects the power network into two islands. Let $\mathcal{I}$ be the island that contains node $\hat{j}$. Under the proportional control $\mathbb{G}_\alpha$, for any line ℓ in $\mathcal{I}$ if there is no simple path in $\mathcal{I}$ that contains ℓ from the node $\hat{j}$ to a participating bus $k \in \mathcal{I}$ with $\alpha_k > 0$, then the post-contingency flow on line ℓ remains unchanged.*

5 Improving network reliability by bridge-block decomposition refinement

In this section we outline a new method to refine the bridge-block decomposition of a given power network by means of *line switching actions*. Even though we usually think of transmission lines as static assets, network operators have the ability to remotely control circuit breakers that open and close them, effectively removing them from service and thus modifying the network topology. This procedure, that goes under the name of *transmission line switching*, is a mature power system technology that is already commonly used by network operators. A large body of literature has been devoted to the optimization problems that stem from this network flexibility, see our discussion about Optimal Transmission Switching (OTS) in Section 1.2.

In this paper we want to propose a new way to improve the network reliability leveraging the flexibility offered by these transmission line switches. Our strategy is inspired by Lemma 2.1, which suggests that switching off a subset of the currently active lines is a way to refine the bridge-block decomposition of a network. In Figs. 3 and 4 we briefly illustrated how a significant bridge-block decomposition refinement is possible for the IEEE 118-bus network by switching off only three lines. Such switching actions can be taken both proactively, as semi-permanent network design decision, but also as temporary response to a contingency or an unusual load/generation profile. As summarized by our results in Section 4, a finer bridge-block decomposition greatly improves the network robustness against failure propagation, since it automatically avoids long-distance propagation of line failures. More specifically, an immediate consequence of Proposition 4.4 is that any non-cut set line outage does not propagate outside the bridge-bridge block(s) to which the disconnected lines belong.

Our approach is particularly promising in view of the fact that most power grids have a trivial bridge-block decomposition, making a global propagation of line failures very likely in these networks. Indeed, all the considered IEEE test networks have a single massive bridge-block and possibly a few additional blocks consisting of a single vertex, see Fig. 3, Table 2 in Section 6, and Fig. 5(a).

The crucial step is thus, given a power network G, identifying a subset of lines $\mathcal{E} \subset E$ to be temporarily removed from service by means of switching actions, so that the newly obtained network $G^{\mathcal{E}}$ has a "good" bridge-block decomposition $\mathcal{BB}(G^{\mathcal{E}})$ while still remaining connected. In this section, we formulate an optimization

problem to find the best subset of lines to switch off that would guarantee a desired network bridge-block decomposition without creating congestion in the remaining lines. However, any line switching action causes a power flow redistribution on the remaining lines (exactly like line outages do) and may cause overloads. It is then crucial to carefully assess the impact of each set of switching actions on the network flows and avoid creating any congested line. A brute-force search to find such best switching actions and recalculate the line flows for each instance is infeasible, since there are exponentially many such subsets of lines $\mathcal{E} \subset E$ to consider.

Aiming to reduce the number of possible subsets of lines to consider, our method takes a different approach and "works backwards". Indeed, we first identify a "good" network partition $\mathcal{P}$ whose clusters we want to become its bridge-blocks and then, using these clusters, we choose which of the cross-edges between them to remove so that $\mathcal{P}$ actually becomes the network bridge-block decomposition. We thus propose an algorithm that consists of two steps, each one tackling one of these two sub-problems. This section is structured as follows:

- In Subsection 5.1 we introduce the *Optimal Bridge-Block Identification* (OBI) problem, which aims at identifying a good network partition based on the current network flow configuration,

- In Subsection 5.2 we tackle the second sub-problem by formulating the *Optimal Bridge Selection* (OBS) problem, which identifies the best subset of lines to be switched off that refines the network bridge-block decomposition and simultaneously minimizes the network congestion level,

- In Subsection 5.3 we discuss some practical considerations regarding the algorithm implementation and present a faster recursive variant.

5.1 Identifying candidate bridge-blocks using network modularity

The bridge-block decomposition refinement should ideally require a minimum number of line switching actions. For this reason, it is thus crucial to leverage the structure of the existing network as much as possible when identifying the candidate bridge-blocks. In this subsection we describe in detail how we tackle this first sub-problem using the notion of network modularity.

Consider a power network $G = (V, E, \boldsymbol{b})$ with a fixed power injection configuration $\boldsymbol{p}$. Aiming to find a network partition $\mathcal{P} = \{\mathcal{V}_1, \ldots, \mathcal{V}_{|\mathcal{P}|}\}$ into clusters with the most suitable properties to become bridge-blocks via line switching, we introduce the following *Optimal Bridge-Blocks Identification (OBI)* problem:

$$\max_{\mathcal{P} \in \Pi(G)} \frac{1}{2M} \sum_{r=1}^{|\mathcal{P}|} \sum_{i,j \in \mathcal{V}_r} \left(|f_{ij}| - \frac{F_i F_j}{2M} \right), \qquad \text{(OBI)} \qquad (23)$$

where $\boldsymbol{f} = (f_{ij})_{(i,j) \in E}$ is the current power flow pattern determined by the injections $\boldsymbol{p}$ as prescribed by (3), $F_i := \sum_{j \in V} |f_{ij}|$ is the sum of absolute line flow incident to vertex i and $M := \frac{1}{2} \sum_{i \in V} F_i = \frac{1}{2} \sum_{i,j \in V} |f_{ij}|$ is the total absolute line flows over the network. When we restrict the subset of feasible partitions to $\Pi_k(G)$, we refer to problem (23) as the OBI-k problem.

Having in mind the optimal line switching that we intend to perform in the second step of our algorithm, it is important to remember that each line that we decide to switch off will cause a power flow redistribution on the remaining lines. Aiming to prevent any of these lines to become congested, it is thus crucial to identify in this first step a "good" partition into clusters such that (i) a smaller number of switching actions to create the desired bridge-block decomposition and that (ii) the power flow redistribution ensuing the switching actions has a minimal impact on the rest of the network and does not create congestion.

The OBI problem (23) is formulated precisely to find a partition with these features. Indeed, its objective function indirectly favors partitions that have:

(F1) very few (and/or low-weight) cross-edges between clusters;

(F2) clusters are balanced in terms of total net power and with very modest power flows from/towards the neighboring clusters.

These features follows from the fact that the objective function of (23) is a power-flow-weighted version of the *network modularity* introduced by [102, 106]. Network modularity is a key notion that has been introduced to tackle a classical problem in complex networks theory, namely unveiling the underlying community structure in a given network. Informally, the proposed optimal block identification problem is a community detection problem on a transmission network topology while using in a crucial way its electrical properties. The network modularity $Q(\mathcal{P})$ associated with a partition $\mathcal{P}$ is a scalar between $-1/2$ and 1 [15] that quantifies the quality of such a partition: a large value indicates a stronger "community structure", in the sense that there are more (and in the weighted case high-weight) edges within each cluster in $\mathcal{P}$ than across them.

Broadly speaking, a partition that yields a high network modularity is one where the connectivity within each cluster is larger than the one expected in a random network with the same nodal properties. More specifically,

the modularity of a network partition measures the total weight of the internal edges within each cluster against the same quantity in a network with the same number of nodes and nodal properties, but in which the edges are placed randomly. Indeed, the term $\frac{F_i F_j}{2M}$ should be interpreted as the likelihood of finding edge (i, j) in a random graph conditioned on having the node generalized degree sequence $F_1, \ldots, F_n$.

For the OBI problem we take as the objective function the network modularity where each edge weight is equal to absolute power flow on the corresponding line. In view of the definition of network modularity, the optimal partitions for the OBI problem are those with few lines between clusters and modest power flowing on them. We also remark that using the absolute power flows as weights in the OBI problem implicitly makes the resulting bridge-block decomposition adaptive to the possibly changing power injection and flow patterns. Such a feature would not be present if we used an unweighted network modularity as objective function or if we used other static network quantities, e.g. the line susceptances $\{\beta_\ell\}_{\ell \in E}$, as weights.

The notion of network modularity has been extended to deal with negative edge weights in [47]: in this extension the high-modularity partitions tend to assign the two nodes at the endpoints of an edge with a negative weight to different clusters. In power networks, however, the flow signs are just an artifact of the (arbitrary) directions chosen for the network edges – in fact, one could always choose edge directions so that all the flows are positive. Our goal to identify the network partitions with modest flows across different clusters *regardless of their directions* makes the *absolute values* of the flows the sensible choice for the edge weights.

5.1.1 Related classical clustering problems

One might argue that a simpler approach for bridge-block identification could have been solving a weighted *minimum k-cut problem*

$$\arg \min_{\mathcal{P} \in \Pi(G)} \sum_{r=1}^{|\mathcal{P}|} \mathrm{Cut}(\mathcal{V}_r), \tag{24}$$

where $\mathrm{Cut}(\mathcal{V}_r) := \sum_{i \in \mathcal{V}_r, \, j \in \mathcal{V}_r^c} |f_{ij}|$. This combinatorial optimization problem can be solved efficiently, especially in the case $k = 2$, as illustrated in [131]. However, such a min-cut solution would often not be a satisfactory partition for the power systems application at hand. Indeed, in many instances the min-cut solution separates one or very few vertices from the rest of the network and such trivial partitions will yield a rather unbalanced bridge-block decomposition with limited failure localization potential. On the contrary, it is very unlikely that the OBI problem yields trivial partitions. Indeed, using the weighted network modularity (23) as objective function favors partitions with "balanced clusters", which in our setting means with small total net power $\sum_{i \in \mathcal{V}_r} p_i$, $r = 1, \ldots, |\mathcal{P}|$. This feature of high modularity partition has been proved rigorously: as shown by [115] and [13], for a fixed number k of clusters the OBI problem (23) has the same solution of the following *balanced-cut problem*

$$\arg \min_{\mathcal{P} \in \Pi_k(G)} \sum_{r=1}^{k} \left(\mathrm{Cut}(\mathcal{V}_r) + \frac{\mathrm{Vol}(\mathcal{V}_r)^2}{2M} \right), \tag{25}$$

where we used the classical notion of *volume* of a cluster, which in our context is equal to $\mathrm{Vol}(\mathcal{V}_r) = \sum_{i \in \mathcal{V}_r} F_i$. This equivalence suggests that an optimal partition that maximizes modularity favors sparsely interconnected clusters with balanced volumes. This is particularly evident in the case of $k = 2$ target clusters, in which (25) is equivalent to

$$\arg \min_{\mathcal{V} \subset V} \left(\mathrm{Cut}(\mathcal{V}) + \frac{\mathrm{Vol}(\mathcal{V})\mathrm{Vol}(\mathcal{V}^c)}{2M} \right). \tag{26}$$

The second term $\mathrm{Vol}(\mathcal{V})\mathrm{Vol}(\mathcal{V}^c)$ is maximal when $\mathrm{Vol}(\mathcal{V}) = \mathrm{Vol}(\mathcal{V}^c) = M$ and thus favors network partition with two clusters of roughly equal volume. In contrast, the first term favors a partition of the graph in which few edges with small weights are removed. This is reminiscent of the NCut problem introduced in [127], in which the objective is to minimize the ratio

$$\mathrm{NCut}(\mathcal{P}) = \frac{\mathrm{Cut}(\mathcal{V}, \mathcal{V}^c)}{\mathrm{Vol}(\mathcal{V})} + \frac{\mathrm{Cut}(\mathcal{V}, \mathcal{V}^c)}{\mathrm{Vol}(\mathcal{V}^c)} \propto \frac{\mathrm{Cut}(\mathcal{V}, \mathcal{V}^c)}{\mathrm{Vol}(\mathcal{V})\mathrm{Vol}(\mathcal{V}^c)} \tag{27}$$

over all possible bipartitions $\mathcal{P} = \{\mathcal{V}, \mathcal{V}^c\}$. Whenever a cluster-balancing condition is introduced in the objective function either in the form of (25), (26) or (27) the min-cut problem become NP-hard, see [141] for details on NCut and [14] for details on network modularity. Efficient algorithms are thus necessary to find good locally optimal network partitions with reasonable computational costs. Numerous methods have been proposed in the literature, for a review see [43, 44, 111, 23]. These include greedy algorithms [9, 24, 103, 142], extrenal optimization [11, 33], simulated annealing [51, 95, 97], spectral methods [146, 104, 117], sampling techniques [122] and mathematical programming [1]. In the present work we focus on two of these state-of-the-art algorithms, namely the "fastgreedy" method of [24] and the spectral clustering method [104, 105], which we briefly review later in Section 6. These two methods are particularly suitable for our purposes because they allow for recursive

partitioning. In the next section we use this feature in a crucial way, by devising an algorithmic procedure that progressively refines the block decomposition by splitting existing blocks in smaller ones, thus allowing to take switching actions in a sequential fashion.

5.1.2 Intentional controlled islanding

In this subsection, we briefly give an overview of another failure mitigation strategy that has some similarities with the candidate bridge-blocks identification described earlier in this section.

In the midst of a cascading failure, it is sometimes recommended to avoid catastrophic wide area blackouts by means of *intentional controlled islanding* (ICI) [112, 148]. In other words, by means of line switching actions, they temporarily partition the network into islands to prevent the cascading failure from propagating further and/or creating very imbalanced grid sections. Moreover, such an intentional islanding can enhance faster restoration of the full grid later and avoid transient stability issues during reconnection [149, 148].

The network operator ideally would want each of these islands to have:

- a net power as close as possible to zero, in the sense that power supply and demand should be almost equal, so that only minimal load shedding or generators adjustment are needed;

- enough capacity to safely sustain the power flows within the island.

A lot of research has been devoted to the problem of finding a network partition with such properties, using either efficient search algorithms [75, 112], mixed integer programming [40, 109], graph theory [129], or spectral clustering techniques [39, 135]. Similar methods have also been used in [89, 150, 139] to group together synchronized generators in case of frequency disturbance events.

These island design principles are clearly very much aligned with those we outlined for bridge-blocks, see (F1) and (F2) in the previous section. In view of this similarity, we expect an optimal islands design based on weighted network modularity to be very effective, but a detailed exploration of this other application is out of the scope of the present paper.

Dynamically creating a bridge-block decomposition is proposed in [7] as a less drastic emergency measure in alternative to controlled islanding. The authors also classical spectral k-way clustering techniques to identify the optimal clusters to be transformed into bridge-blocks.

5.2 Building the bridge-block decomposition via optimal switching actions

In the previous subsection we formulated the OBI problem, which for a given power network identifies a partition with desirable features that we want to transform into the "backbone" of the network bridge-block decomposition. In this subsection we focus on the second sub-problem, namely how to, given such a partition $\mathcal{P}$, reconfigure the network topology so that the resulting network (i) has a bridge-block decomposition at least as fine as $\mathcal{P}$ and (ii) has minimum line congestion.

The feasibility and effectiveness of such a network reconfiguration crucially depends on the "quality" of the chosen partition $\mathcal{P}$. Nonetheless, all the results of this section do not rely on any specific features of $\mathcal{P}$ and they are valid for a general partition of G. The only additional assumption that we tacitly make in the rest of the section is the following:

(A1) for every cluster $\mathcal{V} \in \mathcal{P}$, the subgraph of G induced by $\mathcal{V}$ is connected.

Note that this is not really a stringent assumption: indeed, for any partition $\mathcal{P}$ that does not satisfy it, we can always consider the finer partition $\mathcal{P}'$ in which all the connected components of all its subgraphs are considered as separate clusters. Assumption (A1) is useful as it links the connectedness of the network with that of its reduced graph, as illustrated by the following lemma.

Lemma 5.1 (Connectedness of the reduced graph). *If G is a connected graph, then the reduced graph $G_{\mathcal{P}}$ is connected for any partition $\mathcal{P}$. The converse is true only when the partition $\mathcal{P}$ satisfies (A1).*

To any partition $\mathcal{P}$ is associated a subset $E_c(\mathcal{P}) \subset E$ of cross-edges, i.e., the edges whose endpoints belong to different clusters of $\mathcal{P}$ (cf. Subsection 2.1). Given a partition $\mathcal{P}$ of G, we are interested in finding a subset $\mathcal{E} \subset E_c(\mathcal{P})$ of cross-edges that, if removed, yield a network $G^{\mathcal{E}}$ whose bridge-block decomposition is at least as fine as $\mathcal{P}$, i.e.,

$$\mathcal{BB}(G^{\mathcal{E}}) \succeq \mathcal{P}. \tag{28}$$

Each of these subsets of cross-edges corresponds to a different set of transmission lines that we could temporarily disconnect by means of switching actions, which yields a transmission network topology with the desired bridge-block decomposition. The next proposition shows that the condition (28) is equivalent to requiring that the reduced graph induced by the partition $\mathcal{P}$ on the post-switching network $G^{\mathcal{E}}$ is a tree.

Proposition 5.2 (Reduced graph characterization)**.** *Let $\mathcal{P}$ be a vertex partition of a connected graph G. Then, $\mathcal{BB}(G) \succeq \mathcal{P}$ if and only if the reduced graph $G_{\mathcal{P}}$ is a tree.*

Proof. Assume by contradiction that $G_{\mathcal{P}}$ is not a tree and thus there exists a non-trivial cycle in $G_{\mathcal{P}}$. Since we assume all the clusters of $\mathcal{P}$ induce connected subgraphs, there exists also a non-trivial cycle c in G that includes vertices from at least two different clusters of $\mathcal{P}$. Recall that by definition of bridge-block decomposition $\mathcal{P}_{\mathrm{circuit}}(G) = \mathcal{BB}(G)$ (cf. Subsection 2.1), all the vertices on the cycle c must belong to the same bridge-block, say $B \in \mathcal{BB}(G)$. But then $\mathcal{BB}(G)$ cannot be finer than $\mathcal{P}$ since the bridge block B is not contained in any cluster in $\mathcal{P}$, since it has vertices from at least two different clusters of $\mathcal{P}$ and thus $B \nsubseteq V$ for every $V \in \mathcal{P}$.

The reverse implication can be proved along the same lines, assuming by contradiction that $\mathcal{BB}(G^{\mathcal{E}}) \nsucceq \mathcal{P}$ and leveraging again the connection between the circuits in G and those in the reduced graph $G_{\mathcal{P}}$. □

In view of Proposition 5.2, we can assume that $G_{\mathcal{P}}$ is not a tree, otherwise is suffices to take $\mathcal{E} = \emptyset$ and the target condition (28) is automatically satisfied. When the reduced graph $G_{\mathcal{P}}$ is a not a tree (possibly with/because of multiple edges), there are many valid subsets of cross-edges $\mathcal{E} \subset E_c(\mathcal{P})$ that satisfy the target property (28). Indeed, there exists a suitable subset of cross-edges for each spanning forest of the reduced graph $G_{\mathcal{P}}$. Aiming to have the least impact possible on the network and, in particular, avoiding its disconnection, we exclude all the subsets $\mathcal{E} \subset E_c(\mathcal{P})$ that are cut sets for the graph G and thus we can focus on the spanning trees of the reduced graph $G_{\mathcal{P}}$.

Among all these possible choices for the set $\mathcal{E}$, we want to pick the one that has the least impact on the surviving network in terms of line congestion. We define as *congestion* on a transmission line $\ell \in E$ the ratio between the absolute power flow on that line and its (thermal) capacity C_ℓ, i.e., $|f_\ell|/C_\ell \geq 0$.

Assuming the power injections $\boldsymbol{p}$ do not change, after the removal of the lines in any such subset $\mathcal{E} \subset E_c(\mathcal{P})$, the power redistributes on the remaining $|E \setminus \mathcal{E}|$ lines as described in Subsection 3.2, yielding a new vector of new power flows $\boldsymbol{f}^{\mathcal{E}}$. We define the *network congestion level* $\gamma(\mathcal{E})$ as the maximum line congestion in the surviving graph $G^{\mathcal{E}}$, i.e.,

$$\gamma(\mathcal{E}) := \max_{\ell \in E \setminus \mathcal{E}} |f_\ell^{\mathcal{E}}|/C_\ell. \tag{29}$$

With a minor abuse of notation, we use $\gamma(\emptyset)$ to denote to the network congestion level of the original network where no lines have been removed yet, i.e., $\mathcal{E} = \emptyset$. Power networks must be operated so that each line flow is strictly below the capacity of the corresponding line, meaning that the network congestion level should not be larger than one.

The problem of finding the best set $\mathcal{E}$ of switching actions that yields a network $G^{\mathcal{E}}$ with minimum network congestion and bridge-block decomposition as fine as $\mathcal{P}$ can be formalized as the following *Optimal Bridge Selection* (OBS) problem:

$$\min_{\mathcal{E} \subset E_c(\mathcal{P})} \quad \gamma(\mathcal{E}) \qquad \text{(OBS)} \tag{30a}$$

$$\text{s.t.} \quad (G^{\mathcal{E}})_{\mathcal{P}} \text{ is a tree.} \tag{30b}$$

The constraint (30b), by requiring the reduced graph $(G^{\mathcal{E}})_{\mathcal{P}}$ of the post-switching network $G^{\mathcal{E}}$ to be a tree (and not a forest), implicitly guarantees that $(G^{\mathcal{E}})_{\mathcal{P}}$ is connected and therefore so is $G^{\mathcal{E}}$ by virtue of Lemma 5.1. In other words, for any feasible set $\mathcal{E}$ for (30) the resulting post-switching network $G^{\mathcal{E}}$ is still connected.

In view of Proposition 5.2, constraint (30b) ensures that the bridge-block decomposition of the surviving network $G^{\mathcal{E}}$ is at least as fine as $\mathcal{P}$, but it can possible be finer. This happens when the removal of the cross-edges in $\mathcal{E}$ accidentally creates additional bridges in $G^{\mathcal{E}}$ that the original network did not have. By construction these new bridges were not cross-edges and thus must have been *inside* one or more clusters of $\mathcal{P}$. This means that the switching actions $\mathcal{E}$ involuntarily split each of these clusters into two or more smaller bridge-blocks, and thus the resulting bridge-block decomposition $\mathcal{BB}(G^{\mathcal{E}})$ is finer than $\mathcal{P}$. This happens also in the example that we illustrate in the next subsection, see Fig. 5. If $G_{\mathcal{P}}$ is not a tree, the OBS problem (30) has always at least one feasible nontrivial solution $\mathcal{E} \neq \emptyset$ for any target partition $\mathcal{P}$. Indeed, $G_{\mathcal{P}}$ has at least one spanning tree as long as it is connected, property guaranteed by assumption (A1) and by Lemma 5.1.

It could be the case that even for the optimal set $\mathcal{E}^*$ of switching actions, at least one line gets congested, i.e.,

$$\gamma(\mathcal{E}) \geq 1, \quad \text{for every feasible subset } \mathcal{E}.$$

This is a rare occurrence and, in fact, an optimal solution $\mathcal{E}^*$ with congestion level $\gamma(\mathcal{E}^*) < 1$ exists in most cases (see Section 6 for our numerical results for IEEE test networks), especially if the original network was not heavily congested, i.e., $\gamma(\emptyset) \napprox 1$. Furthermore, as we illustrate in the same section, it is often the case that the network congestion level *decreases* after taking the optimal switching actions, i.e.,

$$\gamma(\mathcal{E}^*) < \gamma(\emptyset).$$

A possible explanation for this counter-intuitive fact is the presence of loop-flows involving edges in two or more clusters, which are then removed by the switching actions, hence possibly reducing the congestion on the newly created bridges or inside the clusters.

Since the optimal solution of the combinatorial OBS problem (30) may not be unique, in which case we heuristically select the one that minimizes the total number of congested lines in the post-switching network.

Lastly, we remark that an optimization problem similar to OBS is proposed in [7] to refine the bridge-block decomposition as emergency measure. In their formulation, however, the objective is to minimize the total weight of the spanning tree $(G^{\mathcal{E}})_{\mathcal{P}}$ of the reduced graph rather than minimizing the congestion in the post-switching network.

5.3 Full algorithm and its recursive variant

We now present in full our proposed algorithm by combining the two steps described in the previous subsections. Furthermore, we discuss some practical considerations regarding the implementation of the OBS problem. In particular, we show how the computational complexity of (30) can be dramatically reduced by adopting a strategy where the bridge-block decomposition is progressively made finer by recursively splitting the existing bridge-blocks in smaller ones.

By combining the OBI and OBS problems described in the previous subsections we obtain the following procedure to refine the bridge-block decomposition of a given power network G with power injections $\boldsymbol{p}$. This algorithm also requires in input the target number of bridge-blocks $b \in \mathbb{N}$.


Algorithm 1: One-shot bridge-block decomposition refinement

Input : The power network $G = (V, E, \boldsymbol{b})$, the set of power injections $\boldsymbol{p}$, the target number of bridge-blocks b

1 Calculate the current line flows $\boldsymbol{f}$ on the network G induced by the injections $\boldsymbol{p}$;
2 Calculate the current bridge-block decomposition $\mathcal{BB}(G)$;
3 Consider in isolation the largest bridge-block in size of the current bridge-block decomposition, say $\mathcal{V} \in \mathcal{BB}(G_i)$;
4 Solve a OBI-b problem for $\mathcal{V}$ in isolation to find its optimal b-partition $\mathcal{P}^*$;
5 Solve the OBS problem (30) for $\mathcal{V}$ in isolation with target b-partition $\mathcal{P}^*$, obtaining an optimal subset $\mathcal{E}_i^*$;
6 Calculate the new congestion level $\gamma^* = \gamma(\mathcal{E}^*)$;
7 Calculate the new set of active lines to be $E' := E \setminus \mathcal{E}_i^*$, set $G' = (V, E', \boldsymbol{b}')$, and calculate its bridge-block decomposition $\mathcal{BB}(G')$;

**Output** : G', $\mathcal{BB}(G')$, γ^*

Fig. 5 illustrates the steps of Algorithm 1 with $b = 4$ when used to refine the bridge-block decomposition of the IEEE 73-bus network.
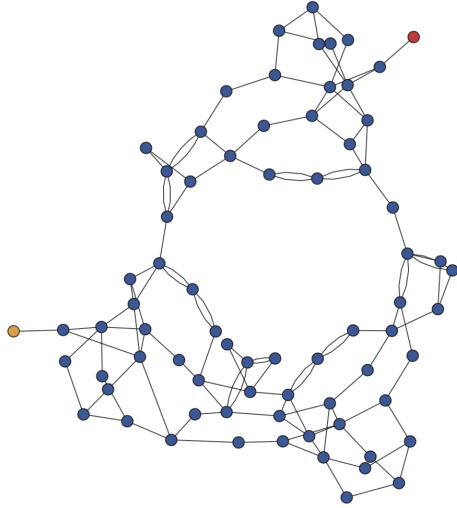
(a) The graph G corresponding to IEEE 73 network with its bridge-block decomposition $\mathcal{BB}(G)$ consisting of three bridge-blocks, two of which are trivial.
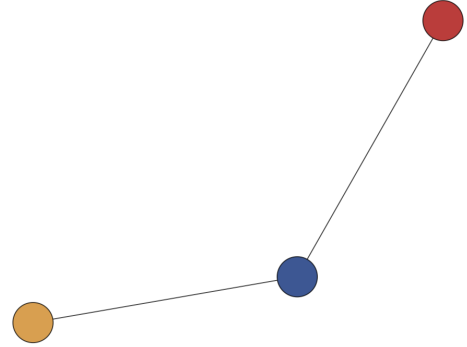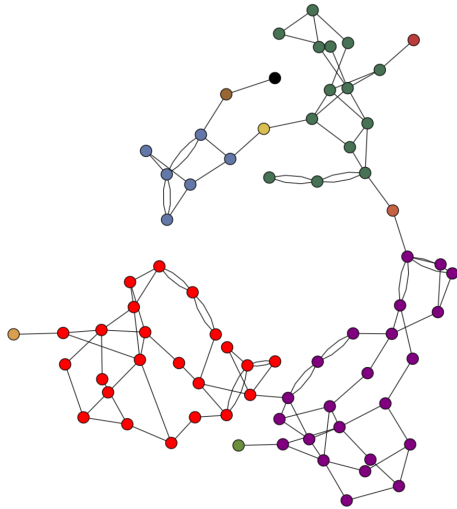
(b) The bridge-block tree of G, i.e., the reduced graph $G_{\mathcal{BB}(G)}$ corresponding to the bridge-block decomposition of the network in (a)
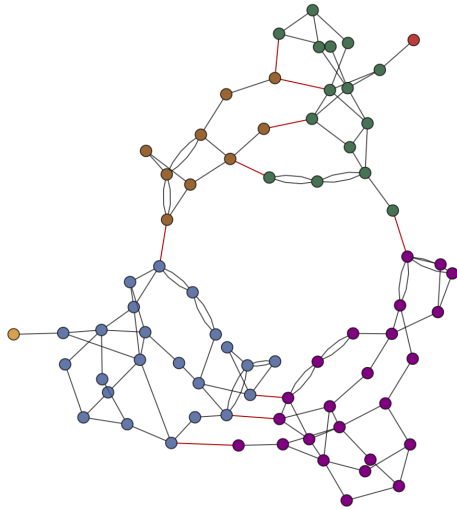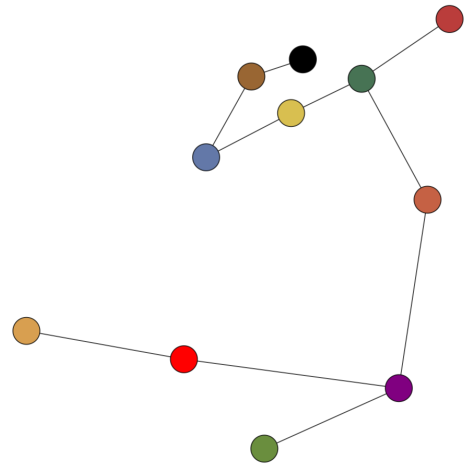
(c) The biggest bridge-block of the network in (a) is split into $b = 4$ clusters by solving an OBI-4 problem, obtaining a finer partition $\mathcal{P}^*$. The cross-edges of $\mathcal{P}^*$ inside the original bridge-block are highlighted in red.
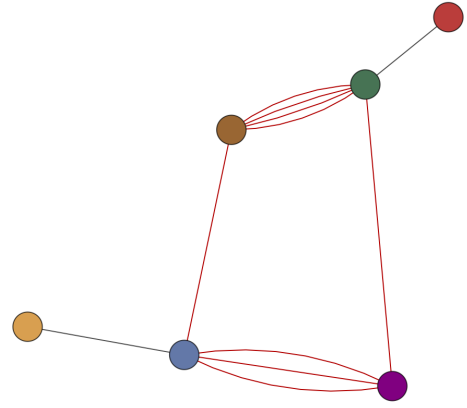
(d) The reduced graph $G_{\mathcal{P}^*}$ corresponding to the optimal partition $\mathcal{P}^*$ in (c), with the set $E_c(\mathcal{P}^*)$ of its cross-edges highlighted in red. Each feasible set of switching actions corresponds to a spanning tree of $G_{\mathcal{P}^*}$.

(e) The network G' obtained after switching off the optimal subset $\mathcal{E}^*$ consisting of 6 cross-edges. The vertex colors display its bridge-block decomposition $\mathcal{BB}(G')$, which is finer than the original one $\mathcal{BB}(G)$.

(f) The bridge-block tree of G', i.e., the reduced graph $G'_{\mathcal{BB}(G')}$ corresponding to the new bridge-block decomposition $\mathcal{BB}(G')$ of the post-switching network G' in (e).

Figure 5: One-shot algorithm at work. Even if the goal was to split the biggest bridge-block into $b = 4$ smaller ones, the algorithm returned a finer bridge-block decomposition consisting of 11 bridge-blocks.

Solving the one-shot bridge-block decomposition refinement presents several issues. Firstly, it is hard to determine a priori a reasonable target number b of bridge-blocks, which is required as input of the algorithm: a small value would only yield a minor refinement of the network bridge-block decomposition, while a large one might yield only network with congested lines, i.e., $\gamma^* > 1$. Secondly, solving the OBS at the step 3 might be computationally very expensive. As mentioned earlier, the feasible sets $\mathcal{E} \subset E_c(\mathcal{P})$ for the OBS problem (30) are in one-to-one correspondence with the spanning trees of the reduced multi-graph $G_\mathcal{P}$, which has at least b nodes.

Unfortunately, the OBS problem is not a minimum spanning tree problem, since the network congestion level appears in the objective function. For every spanning tree T of $G_\mathcal{P}$, we need to recalculate the power flows $\boldsymbol{f}^\mathcal{E}$ on the network $G^\mathcal{E}$ obtained from the original one by removing the cross-edges that do not belong to that spanning tree, i.e., $\mathcal{E} = E_c(\mathcal{P}) \setminus T$. In view of this fact, the bottleneck of the above algorithm is precisely finding all such spanning trees of $G_\mathcal{P}$. The time required to generate all spanning trees of a graph G can be expressed as $O(g(b, |E_c(\mathcal{P})|) + \tau(G_\mathcal{P})h(b, |E_c(\mathcal{P})|))$ where g, h are functions that are specific to the algorithm under consideration (see [22] for a detailed overview), but it is largely dominated by $\tau(G_\mathcal{P})$, the number of spanning trees of $G_\mathcal{P}$, that increases exponentially in the graph size b.

This latter issue disappears if the selected target partition $\mathcal{P}$ consists of only two clusters, say $\mathcal{V}_1$ and $\mathcal{V}_2$, as the reduced multi-graph $G_\mathcal{P}$ has a number of spanning trees at most linear in the number of edges of the original network G. Indeed, in this case any spanning tree for $G_\mathcal{P}$ consists of a single cross-edge in $E_c(\mathcal{P})$. Therefore, selecting a subset $\mathcal{E} \subset E_c(\mathcal{P})$ of cross-edges between $\mathcal{V}_1$ and $\mathcal{V}_2$ feasible for the OBS problem is equivalent to choosing the unique cross-edge $e \in E_c(\mathcal{P})$ to keep, which become the unique bridge of the reduced graph $(G^\mathcal{E})_\mathcal{P}$. Thus the number of feasible sets $\mathcal{E}$ for the OBS problem is precisely equal to that of the cross-edges, and, trivially, $|E_c(\mathcal{P})| \leq |E|$. Restricting ourselves to bi-partitions $\mathcal{P} \in \Pi_2(G)$, the OBS problem (30) becomes

$$\min_{e \in E_c(\mathcal{P})} \gamma(E_c(\mathcal{P}) \setminus \{e\}). \tag{31}$$

where the constraint (30b) disappears because $(G^\mathcal{E})_\mathcal{P}$ is always trivially a tree in this case. This optimization problem is much simpler and faster to solve than the original one: instead of optimizing over a complex combinatorial object, namely the family of spanning trees of the reduced graph $G_\mathcal{P}$, the feasible set is just the collection of cross-edges $E_c(\mathcal{P})$.

This fact suggests the idea that it may be computationally more efficient to refine the bridge-block decomposition progressively, splitting the current largest bridge-block into two smaller bridge-blocks in a recursive fashion. We thus propose a second algorithm that leverages this idea. Besides drastically simplifying the structure of the OBS problem to be solved, this recursive approach embeds a natural stopping rule for the bridge-block decomposition refinement. As we hinted at already, it is impossible to estimate a priori how much the bridge-block decomposition of a power network can be refined without creating congestion on any line. In this respect, a procedure that increases the number of blocks by one at each step while monitoring the network congestion level can help find the best trade-off between the number of bridge-blocks and network congestion level. We introduce the following stopping rule. We keep refining the bridge-block decomposition splitting the largest bridge-block in a recursive fashion as long as:

(i) the current maximum number of splits is less than a predefined threshold $i_{\max} \in \mathbb{N}$, and

(ii) the obtained network congestion level γ^* is not larger than a threshold $\delta \in (0, 1]$ describing the maximum tolerable congestion that the network operator can set.

The high-level structure of the proposed recursive procedure is shown in Algorithm 2.

Note that in Step 6 we select the current largest bridge-block to be further split, so that all the bridge-blocks progressively become smaller, indirectly ensuring that, when the algorithm stops, the sizes of the resulting bridge-blocks are similar. However, we could tweak the algorithm to select either (i) the bridge-block with the lowest congestion level (inside which is probably safer to switch off lines without creating overloads) or (ii) the one with the highest potential to be clustered in terms of modularity score. Both these approaches, however, could possibly lead to ineffective bridge-block decompositions, where there are still very large bridge-blocks and several small (trivial) ones.

The OPS problem needs to be solved several times in this recursive variant (and not only once as prescribed in the one-shot variant, Algorithm 1), which means that Algorithm 2 requires possibly many more power flows calculations. However, a remarkable feature of the recursive approach is that all these power flow redistribution calculations can be done "locally" (i.e., inside the bridge-block that is being split) and thus become progressively faster as the size of the largest bridge-block decreases. More specifically, let $\mathcal{V}$ be the bridge-block selected to be further split in Step 6 of Algorithm 2. By virtue of Proposition 4.4, any switching action (which is effectively equivalent to line failure in this context) inside the bridge-block $\mathcal{V}$ does not affect the line flows in any other bridge-block $\mathcal{V}' \neq \mathcal{V}$ of the current bridge-block decomposition. Therefore, only the line flows inside $\mathcal{V}_i$ needs to be recalculated and this can be done by solving the DC power flow equations (3) only for the subgraph induced

---
Algorithm 2: Recursive bridge-block decomposition refinement

    Input   : A power network $G = (V, E, \boldsymbol{b})$, a maximum number of iterations $i_{\max}$, and a congestion threshold δ

1 Set $i = 0$;
2 Set $E_0 = E$ and $G_0 = (V, E_0)$;
3 Calculate the line flows $\boldsymbol{f}_0$ and the current network congestion level $\gamma_0^* := \gamma(\emptyset)$;
4 **while** $\gamma_i^* < \delta$ **and** $i < i_{\max}$ **do**
5   Calculate the current bridge-block decomposition $\mathcal{BB}(G_i)$;
6   Consider in isolation the largest bridge-block in size of the current bridge-block decomposition, say $\mathcal{V} \in \mathcal{BB}(G_i)$;
7   Solve a OBI-2 problem for $\mathcal{V}$ in isolation to find its optimal bipartition $\mathcal{P}_i^*$;
8   Solve the simplified OBS problem (31) for $\mathcal{V}$ in isolation with target bipartition $\mathcal{P}_i^*$, obtaining an optimal subset $\mathcal{E}_i^* \subset E_c(\mathcal{P}_i^*)$ and the new network congestion level $\gamma_i^* = \gamma(\mathcal{E}_i^*)$;
9   Calculate the new set of active lines to be $E_{i+1} := E_i \setminus \mathcal{E}_i^*$ and set $G_{i+1} = (V, E_{i+1})$;
10   Increase i by 1;
11 **end**
    Output : i, G_i, $\mathcal{BB}(G_i)$, γ_i^*
---

by $\mathcal{V}$ and the corresponding local power injections. The only caveat is that we need to adjust the net power at all the nodes of $\mathcal{V}$ that are endpoints of bridges to account for the outgoing flow.

Adopting this recursive splitting strategy, the bridge-blocks become smaller at every iteration, which in turn means that fewer flows (and congestion levels) need to be recalculated at each step when solving the OBS problem (31), making it increasingly faster to solve.

The recursive structure also makes Step 10 of Algorithm 2 faster: the bridge-block decomposition does not have to be recomputed globally, since all the bridge-blocks but the one considered in this iteration will be unchanged and we only need to identify the new bridge-blocks created locally. We remark that this step cannot be skipped as the optimal set of switch action may accidentally split the considered bridge-block into more than two bridge-blocks.

Figs. 6 and 7 visualize the various steps that Algorithm 2 takes when set with $i_{\max} = 2$ and applied to the same network considered in Fig. 5, namely the IEEE 73-bus network. It is clear that the OBS problem is essentially trivial for the recursive algorithm in both steps, see Figs. 6(d) and 7(b), while it was more involved for the one-shot algorithm, cf. Fig. 5(d).
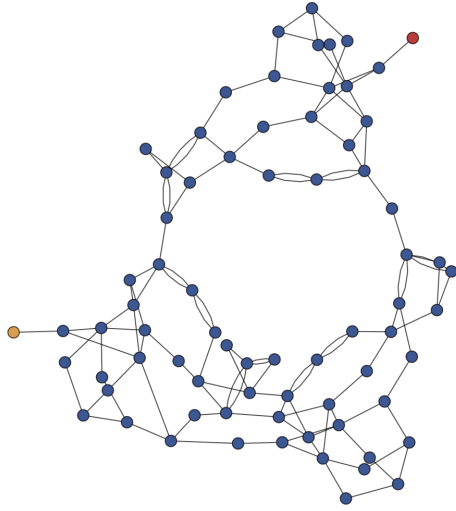
(a) The graph G_0 corresponding to IEEE 73 network with its bridge-block decomposition $\mathcal{BB}(G_0)$ consisting of three bridge-blocks, two of which are trivial.
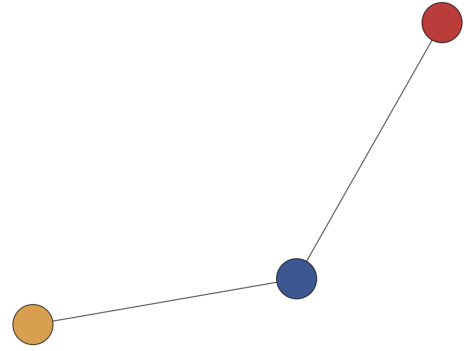
(b) The bridge-block tree of G_0, i.e., the reduced graph $(G_0)_{\mathcal{BB}(G_0)}$ corresponding to the bridge-block decomposition of the network in (a)
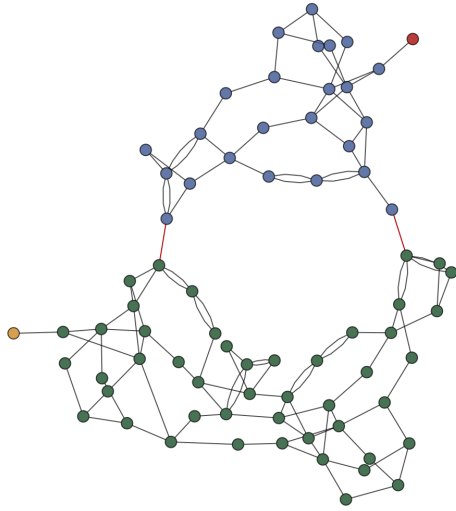
(c) The biggest bridge-block of the network in (a) is split into 2 clusters by solving a simplified OBI-2 problem, obtaining a finer partition $\mathcal{P}_0^*$. The cross-edges of $\mathcal{P}_0^*$ inside the original bridge-block are highlighted in red.
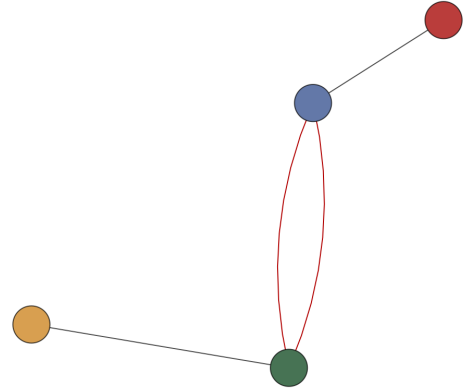
(d) The reduced graph $G_{\mathcal{P}_0^*}$ corresponding to the optimal partition $\mathcal{P}_0^*$ in (c), with the set $E_c(\mathcal{P}_0^*)$ of its cross-edges highlighted in red. The spanning trees of $G_{\mathcal{P}_0^*}$ (and thus the feasible switching actions) to consider do not grow exponentially with the size of $G_{\mathcal{P}_0^*}$, being in one-to-one correspondence with $E_c(\mathcal{P}_0^*)$.

(e) The network G_1 obtained after switching off the cross-edges in $\mathcal{E}_0^*$. The vertex colors display its bridge-block decomposition $\mathcal{BB}(G_1)$, which is finer than the original one $\mathcal{BB}(G_0)$.

(f) The bridge-block tree of G_1, i.e., the reduced graph $(G_1)_{\mathcal{BB}(G_1)}$ corresponding to the new bridge-block decomposition of the post-switching network G_1 in (e).

Figure 6: Recursive algorithm at work – first iteration $i = 1$

(a) The biggest bridge-block of the network G_1 in Fig. 6(e) is split into 2 clusters by solving a simplified OBI-2 problem, obtaining a finer partition $\mathcal{P}_1^*$. The cross-edges of $\mathcal{P}_1^*$ inside the original bridge-block are highlighted in red.

(b) The reduced graph $(G_1)_{\mathcal{BB}(G_1)}$ corresponding to the bridge-block decomposition of the network in (a)

(c) The network G_2 obtained after switching off the cross-edges in $\mathcal{E}_1^*$. The vertex colors display its bridge-block decomposition $\mathcal{BB}(G_2)$, which is finer than the previous one $\mathcal{BB}(G_1)$
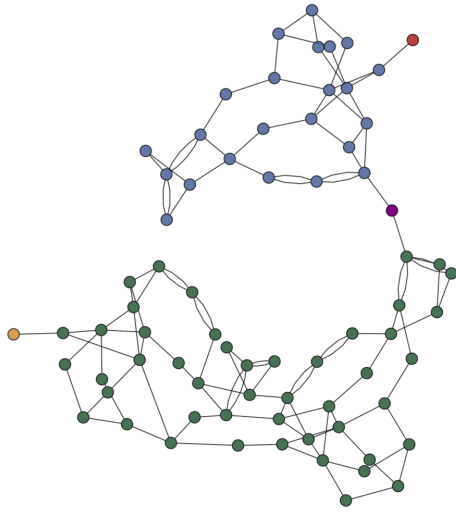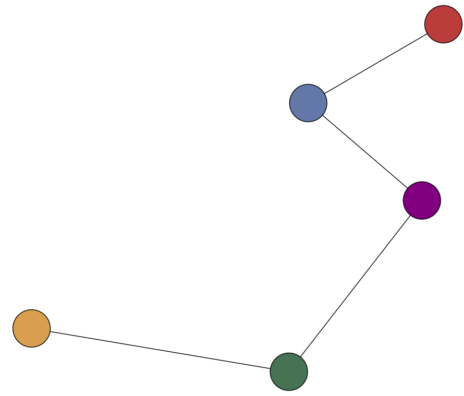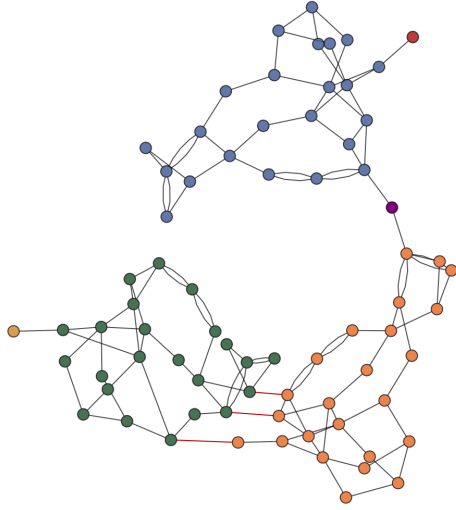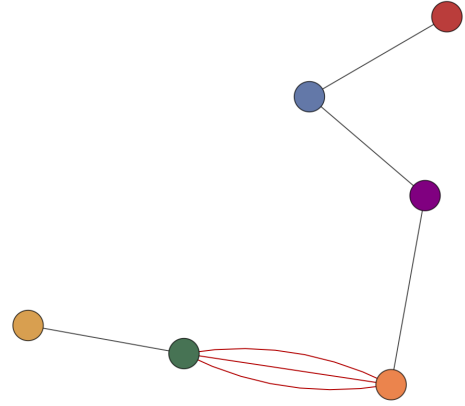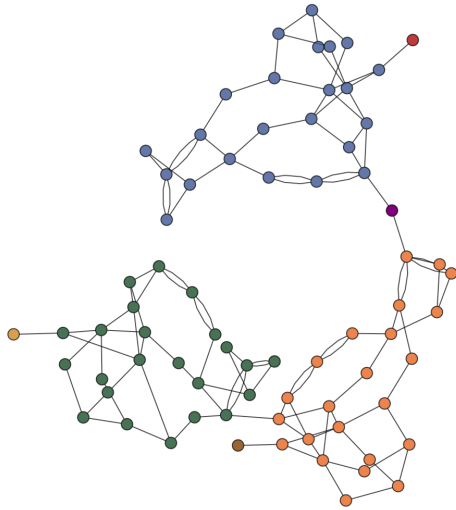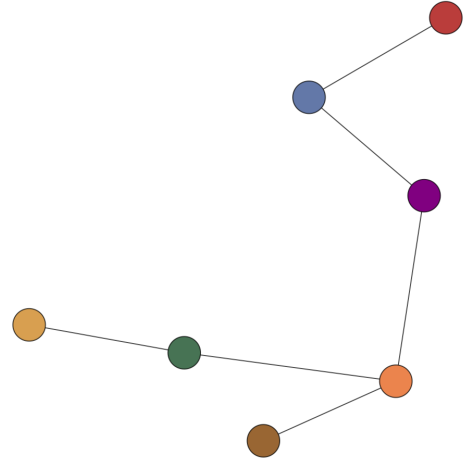
(d) The bridge-block tree of G_2, i.e., the reduced graph $(G_2)_{\mathcal{BB}(G_2)}$ corresponding to the new bridge-block decomposition of the post-switching network G_2 in (c).

Figure 7: Recursive algorithm at work – second iteration $i = 2$

Both the one-shot and the recursive algorithms focus on the scenario where the power injections are fixed. However, in practical settings the network operator could *jointly* reconfigure the network topology by means of switching actions *and* adjust (some of) the power injections, either by shedding load or by adjusting generator levels. In this scenario, the OBS problem could potentially find a solution with an even lower network congestion level, but analyzing this joint optimization problem or proposing heuristics leveraging this additional injection flexibility is beyond the scope of this paper.

We remark that one such possible strategy, in the spirit of [87], is aiming for local power adjustments and only change the power injections at the endpoints of each line that is being switched off by an amount exactly equal to its original flow on that line. This strategy would guarantee that the power flows on the entire surviving network will be unchanged and no congestion will be created. However, there might not be enough flexibility in terms of injection adjustment in each pair of endpoints. Alternatively, one can solve a global OPF problem for the whole network after the switching actions, hence ensuring that no line will be overloaded. Even better results could be obtained when considering the line switching and power injection adjustment jointly in the same optimization problem.

6 Numerical examples

This section is devoted to the implementation of the bridge-block decomposition refinement strategy introduced in Section 5. In Subsection 6.1 we first review the structural properties of several power grids used in the literature as test networks, from which it becomes evident the need of finer and more homogeneous bridge-blocks for these networks. In Subsection 6.2 we introduce two fast approximations algorithms to find locally optimal solutions for the OBI problem (23) and compare their performance on IEEE test networks in terms of the clusters' size and cross-edges. Then, in Subsection 6.3, we compare the performances of the one-shot algorithm and of its recursive variant for the same test networks.

6.1 Preliminary analysis of power systems topologies

In order to show the potential benefits of applying our method for improving network reliability, it is important to first understand the relevant structural properties of typical power networks, in particular their bridge-block decomposition. In Table 2 below we report statistics about the original topologies of various standard test networks taken from [4, 25]. For each of these networks, we find all its bridges and calculate its bridge-block decomposition. Modulo some minor variations, all the networks exhibit the same topological features, namely:

- They have only one giant bridge-block that comprises a vast majority of the network nodes;

- They have substantial number of edges that are bridges (on average roughly 18%), each of them connecting a trivial bridge-block (consisting of one or two nodes) to the aforementioned giant bridge-block.

These features are also easily recognizable for the IEEE 118-bus network displayed in Fig. 3. From these statistics it is clear that most power networks have an essentially trivial bridge-block decomposition and would thus perform very poorly in terms of failure localization. In particular, a line failure inside the unique giant bridge-block could propagate to roughly 80% of the other lines (i.e., those in the same bridge-block). Moreover, we note that all these networks already have a surprisingly large fraction (on average $\sim 18\%$) of lines that are bridges, which the switching actions prescribed by our method would increase only marginally.

Test network case	# of edges	# of bridges	% of edges that are bridges	# of bridge-blocks	Size(s) of non-trivial bridge-blocks
case14_ieee	20	1	5.00	2	13
case30_ieee	41	3	7.32	4	27
case39_epri	46	11	23.91	12	28
case57_ieee	80	1	1.25	2	56
case73_ieee_rts	120	2	1.67	3	71
case89_pegase	210	16	7.62	17	73
case118_ieee	186	9	4.84	10	109
case162_ieee_dtc	284	12	4.22	13	150
case179_goc	263	43	16.35	44	136
case200_activ	245	72	29.39	73	128
case240_pserc	448	58	12.95	59	182
case300_ieee	411	89	21.65	90	206, 3, 3
case588_sdet	686	229	33.38	230	357
case793_goc	913	290	31.76	291	500
case1354_pegase	1991	561	28.18	562	791
case1888_rte	2531	964	38.09	965	918, 5
case2000_goc	3633	445	12.25	446	1555
case2736sp_k	3273	627	19.16	628	2109
case2737sop_k	3274	627	19.15	628	2110
case2746wp_k	3281	637	19.41	638	2109
case2746wop_k	3309	607	18.34	608	2139
case2848_rte	3776	1410	37.34	1411	1421, 7, 5, 3
case2869_pegase	4582	778	16.98	779	2088
case3120sp_k	3693	731	19.79	732	2382, 8
case3375wp_k	4161	826	19.85	827	2536, 3
case9241_pegase	16049	1665	10.37	1666	7558, 7, 5, 3

Table 2: Summary of bridge and bridge-block statistics for various test networks. A bridge-block is considered to be non-trivial is it consists of strictly more than two nodes.

6.2 Bridge-block candidate identification

The OBI problem that we introduced in Section 5.1 to find good candidate bridge-blocks is a variant of the well-known weighted network modularity maximization. However, this optimization problem cannot be solved exactly, being NP-hard, especially for large power networks and we need to resort to locally optimal partition. Before presenting our numerical findings, we first briefly review the two approximation algorithms that we implemented, namely the *fastgreedy method* and the *spectral clustering*.

Fastgreedy. A very fast approximation algorithm for optimizing modularity on large networks was proposed by [24]. It uses a bottom-up hierarchical approach that tries to optimize the modularity in a greedy manner. Initially, every vertex belongs to a separate community, and communities are merged iteratively such that each merger is locally optimal (i.e., yields the largest increase in the current value of modularity). We use the Python library igraph and the built-in implementation of fastgreedy modularity optimization algorithm described in [24], which also implements some improvements proposed by [142].

Spectral clustering. Spectral clustering was first used in the context of modularity maximization by [104, 105] and has become one of the most popular modern clustering algorithms, being simple to implement and very efficient to solve using standard linear algebra software. In our implementation of this method we use two matrices that simultaneously capture the network structure and flow pattern, the modularity matrix and the weighted Laplacian matrix. They can be both calculated from the graph $G = (V, E)$ with edge weights equal the absolute power flows $\{|f_{ij}|\}_{(i,j)\in\mathcal{E}}$ as follows:

- the *modularity matrix* B, whose (i, j) entry is $B_{i,j} = |f_{ij}| - \frac{F_i F_j}{2M}$, and

- the *weighted Laplacian matrix* L, whose entries are $L_{i,j} = -|f_{ij}|$ if $i \neq j$ and $L_{i,i} = -\sum_{j\neq i} L_{i,j}$.

In both cases we choose to work with their normalized counterparts, namely

$$B_n := W^{-1/2} B W^{-1/2} \qquad \text{and} \qquad L_n := W^{-1/2} L W^{-1/2},$$

where $W = \mathrm{diag}(F_1, \ldots, F_n)$, since, especially in graphs of large size or with very heterogeneous weights, it has been argued that the spectral clustering method is more effective and computationally more stable when using the normalized version of these matrices, see [12, 140].

The spectral clustering performed on the normalized Laplacian matrix L_n approximates the optimal solution of the NP-hard problem known as *Normalized Cut (NCut)*. Working with the normalized modularity matrix B_n means that the spectral clustering method approximates the partition that maximizes a renormalized version Q_n of the network modularity, namely

$$Q_n(\mathcal{P}) := \frac{1}{2M} \sum_{r=1}^{|\mathcal{P}|} \frac{1}{\sum_{k\in\mathcal{V}_r} |F_k|} \sum_{i,j\in\mathcal{V}_r} \left(|f_{ij}| - \frac{F_i F_j}{2M} \right).$$

If we fix the target number b of clusters the optimal normalized modularity clustering is equal to the normalized cut clustering, in the sense that $\arg\max_{\mathcal{P}} Q_n(\mathcal{P}) = \arg\min_{\mathcal{P}} \mathrm{NCut}(\mathcal{P})$. This follows from the identity

$$Q_n(\mathcal{P}) = \frac{b-1}{2M} - \frac{1}{2M}\mathrm{NCut}(\mathcal{P}),$$

proved in [12, 151] under the assumption that the target number b of clusters is fixed. This means that the spectral clustering method performed on either matrix B_n and L_n solve approximately the same optimization problem. However, having different eigensystems, they often yield to slightly different results. For this reason, we consider both of these two variants of the spectral clustering approximation method in the following subsection and label them as *Spectral L_n* and *Spectral B_n*.

We remark that the authors in [135] also adopt the normalized-Laplacian-based spectral clustering to power systems, proposing several choices for the edge weights, including the one we adopted in this paper, i.e., the absolute value of power flows. In the same paper, some heuristics are outlined to decide the optimal number of clusters in which to partition the network, but these leverage the eigensystem structure of the Laplacian matrix, rather than being informed by a specific target number of clusters b like in our method.

6.2.1 Performance comparison

We compare the performance of the three aforementioned clustering methods (FastGreedy, Spectral L_n and Spectral B_n) to optimally partition some standard IEEE test cases available in the pglib-opf library [4, 25]. We consider four networks of various sizes, ranging from small (39 nodes) to large (2737 nodes).

We first consider the quality of the network partitions returned by three methods by looking exclusively at graph-related quantities. In the following tables, we report for different target number b of clusters the normalized modularity score Q_n (the higher the better) of the partition returned by each algorithm, as well as the sizes of the b clusters and the fraction of edges that are cross-edges between them. We further report the number of cross-edges to be switched off to obtained a bridge-block decomposition with the identified clusters as bridge-blocks. Recall that, as we argued in Section 5.1, it is highly desirable to have a small number of cross-edges between the identified clusters, since it dramatically reduces the feasible sets of switching actions for the OBS problem.

Clustering method	runtime (sec)	b	Q_n	fraction of cross-edges	#lines to be switched off	resulting partition of the largest bridge-block
Spectral L_n	0.0192	2	0.379	0.0869	3	[10, 18]
Spectral B_n	0.0186	2	0.406	0.0652	2	[11, 17]
FastGreedy	0.0016	2	0.406	0.0652	2	[11, 17]
Spectral L_n	0.0247	3	0.486	0.152	5	[9, 9, 10]
Spectral B_n	0.0210	3	0.501	0.130	4	[8, 9, 11]
FastGreedy	0.0018	3	0.524	0.109	3	[7, 10, 11]
Spectral L_n	0.0306	4	0.513	0.173	5	[4, 7, 8, 9]
Spectral B_n	0.0291	4	0.527	0.196	5	[4, 7, 8, 9]
FastGreedy	0.0027	4	0.535	0.152	4	[4, 7, 7, 10]

Table 3: 39 EPRI

Clustering method	runtime (sec)	b	Q_n	fraction of cross-edges	#lines to be switched off	resulting partition of largest bridge-block
Spectral L_n	0.0255	2	0.220	0.032	5	[28, 81]
Spectral B_n	0.0294	2	0.390	0.032	5	[40, 69]
FastGreedy	0.0032	2	0.427	0.075	13	[50, 59]
Spectral L_n	0.0338	3	0.490	0.053	8	[27, 33, 49]
Spectral B_n	0.0341	3	0.518	0.059	9	[29, 38, 42]
FastGreedy	0.0061	3	0.527	0.102	17	[16, 34, 59]
Spectral L_n	0.0366	4	0.498	0.075	11	[9, 19, 33, 48]
Spectral B_n	0.0477	4	0.540	0.102	16	[18, 21, 29, 41]
FastGreedy	0.0381	4	0.620	0.140	23	[16, 27, 32, 34]

Table 4: IEEE 118

Clustering method	runtime (sec)	b	Q_n	fraction of cross-edges	#lines to be switched off	resulting partition of largest bridge-block
Spectral L_n	0.912	2	0	0.0008	1	[1, 917]
Spectral B_n	0.844	2	0.483	0.0107	26	[391, 527]
FastGreedy	0.443	2	0.489	0.0130	32	[415, 503]
Spectral L_n	7.536	3	0.610	0.0217	51	[168, 356, 394]
Spectral B_n	1.201	3	0.648	0.0194	46	[273, 308, 337]
FastGreedy	0.461	3	0.617	0.0198	48	[205, 298, 415]
Spectral L_n	8.167	4	0.709	0.0221	52	[187, 189, 222, 320]
Spectral B_n	5.693	4	0.709	0.0205	48	[157, 204, 261, 296]
FastGreedy	1.028	4	0.724	0.0217	52	[153, 205, 262, 298]

Table 5: 1888rte

Clustering method	runtime (sec)	b	Q_n	fraction of cross-edges	#lines to be removed	resulting partition of largest bridge-block
Spectral L_n	3.949	2	0.375	0.0076	24	[674, 1436]
Spectral B_n	2.702	2	0.379	0.0058	18	[690, 1420]
FastGreedy	0.606	2	0.461	0.0489	159	[1000, 1110]
Spectral L_n	4.394	3	0.640	0.0150	47	[659, 698, 753]
Spectral B_n	2.837	3	0.640	0.0159	50	[667, 702, 741]
FastGreedy	0.987	3	0.632	0.0516	167	[410, 590, 1110]

Table 6: Polish 2737sop

In the vast majority of the considered cases, the two spectral clustering methods yield partitions with smaller values of the normalized modularity Q_n with respect to the FastGreedy method, which, however, consistently returns partitions with a larger number of cross-edges than the other two methods. In our experiments, the FastGreedy method is considerably (2x-10x times) faster than the two spectral clustering methods.

FastGreedy has the advantage that all the partitions for various b are calculated jointly because the algorithm is nested in that each partition of size b is obtained from the corresponding one of size $b-1$ by optimally dividing one of the clusters into two. This recursive nature of the FastGreedy method makes it faster, but at times it means that partitions with unbalanced clusters sizes could be returned. On the other hand, the two spectral clustering methods almost always identify partitions with clusters of balanced size.

In the cases considered above, all methods always yield partitions with exactly b connected clusters. It is possible that the actual number of identified *connected* clusters could be strictly larger than b because some of them induce network subgraphs that are disconnected, violating our assumption (A1). This occurs very rarely as the OBI objective function severely penalizes such partitions, but even so it would not be an issue as it would lead to a finer bridge-block decomposition.

6.3 Performance of one-shot and recursive algorithms

We first illustrate the numerical results for the one-shot algorithm, i.e., Algorithm 1, for various test networks. More specifically, for each network we solve the OBS problem (30a) three times, using the (possibly different) optimal partition $\mathcal{P}$ into b clusters returned by each of the three different clustering algorithms (cf. Subsection 6.2). Recall that there is a one-to-one correspondence between the feasible subsets of cross-edges $\mathcal{E} \subset E_c(\mathcal{P})$ and the spanning trees of the reduced multi-graph $G_\mathcal{P}$. This means that the one-shot algorithm cannot be solve exactly already for partitions into $b \geq 5$ clusters, since the spanning trees of the reduced multi-graph grow exponentially with b. However, if we consider a small number of target clusters, say $b \leq 4$, we can still solve the OBS exactly by considering all possible spanning tree of $G_\mathcal{P}$.

As pre-processing step, for each of the considered networks we first run a DC-OPF to obtain the power flow configuration and calculate the initial network congestion level $\gamma(\emptyset)$ (which is possibly equal to 1 if one or more line limits are active for the OPF solution). We then run Algorithm 1 with $b = 4$ using the three different clustering methods discussed in the previous section. Table 7 below reports for each network both the initial congestion level $\gamma(\emptyset)$ and the results of the one-shot algorithm ran with $b = 4$ as target number of clusters. More specifically, for each network and each clustering algorithm, we report

- the runtime,

- the number of spanning trees of $G_\mathcal{P}$ considered,

- the optimal post-switching network congestion level $\gamma(\mathcal{E}^*)$,

- the number of lines that to be switched off for the optimal set of switching actions $\mathcal{E}^*$ and which fraction they represent on the total number of network lines.

network	partition method	runtime (sec)	# spanning trees	$\gamma(\mathcal{E}^*)$	# congested lines	# of lines switched off	% of lines switched off
case39_epri	initial network configuration	-		1.000	2	-	-
	Spectral L_n	0.238	10	0.960	0	3	6.52
	Spectral B_n	0.270	10	0.960	0	3	6.52
	FastGreedy	0.245	12	**0.833**	**0**	3	6.52
case57_ieee	initial network configuration	-		0.938	0	-	-
	Spectral L_n	6.210	220	1.050	2	13	16.25
	Spectral B_n	5.454	216	1.078	2	12	15.00
	FastGreedy	6.829	256	**0.921**	**0**	14	17.50
case73_ieee	initial network configuration	-		0.632	0	-	-
	Spectral L_n	1.330	49	1.171	1	8	6.66
	Spectral B_n	2.010	71	1.171	3	10	10.83
	FastGreedy	0.970	31	**0.723**	**0**	6	5.00
case118_ieee	initial network configuration	-		1.000	2	-	-
	Spectral L_n	1.912	80	**1.004**	**2**	10	5.38
	Spectral B_n	2.259	80	1.016	3	10	5.38
	FastGreedy	7.958	264	2.248	8	18	9.68
case179_goc	initial network configuration	-		1.000	4	-	-
	Spectral L_n	1.699	54	1.243	7	7	2.66
	Spectral B_n	10.358	312	1.035	2	13	4.94
	FastGreedy	2.103	69	**1.000**	**1**	9	3.42
case200_activ	initial network configuration	-		0.708	0	-	-
	Spectral L_n	6.541	196	0.591	0	15	6.12
	Spectral B_n	9.146	312	0.591	0	16	6.53
	FastGreedy	6.863	208	**0.591**	**0**	12	4.90
case300_ieee	initial network configuration	-		1.000	11	-	-
	Spectral L_n	3.610	120	1.220	7	12	2.92
	Spectral B_n	14.275	468	**1.058**	**5**	14	3.41
	FastGreedy	37.212	1112	1.161	6	24	5.84

Table 7: Results obtained by the one-shot algorithm with $b = 4$ target clusters using three different methods for the candidate bridge-blocks identification. We highlight in bold the best method and the resulting post-switching congestion $\gamma(\mathcal{E}^*)$ for each network.

From Table 7 we deduce that the resulting post-switching network congestion level varies a lot depending on the test network and partition method. In particular, in a few instances the maximum congestion is strictly bigger than 1, but when considering all three methods, there is always at least one solution that does not increase the congestion significantly. Furthermore, in many instances the post-switching network congestion level even decreases with respect to the initial configuration and so does the number of congested lines.

We did not run the one-shot algorithm on larger networks, as it would have required a substantially (exponentially) larger amount of time. This is not a surprise, in view of the discussion in Section 5.3 regarding the feasible region of the OBS problem. Indeed, the runtime of Algorithm 1 does not depend on the network size per-se, but rather on the the number of spanning trees of the reduced graph $G_{\mathcal{P}}$ corresponding to the optimal partition $\mathcal{P}$ identified by each of the three methods. This makes the runtime of the one-shot algorithm rather unpredictable, discouraging its usage in real time.

Let us now focus on the performance of the recursive algorithm, i.e. Algorithm 2. In Table 8 below, we report the results for the same test networks considered in Table 7 s well as a few more networks of larger size. Indeed, the running time of the recursive algorithm depends solely on the number of spanning trees that needs to be considered in the OBS problem. By design, for the recursive algorithm this number is in most cases very small even for large networks, as we can deduce from the second last column of Table 8. Indeed for the recursive algorithm the number of considered spanning trees is equal to the number of cross-edges of the selected bipartition, which is in turn equal to one plus the number of lines switched off.

For each test network, we run the recursive algorithm with $i_{\max} = 3$, reporting the runtime of each of these three iterations. This choice for $i_{\max}$ allows us to make a fair comparison with the one-shot algorithm with a target of $b = 4$ clusters, since we expect the same number of resulting bridge-blocks (modulo "spontaneous" splits in smaller bridge-blocks).

network	iteration	runtime (sec)	$\gamma(\mathcal{E}^*)$	# congested lines	# of lines switched off	% of lines switched off
case39_epri	initial configuration	-	$\gamma(\emptyset) = 1.000$	2	-	-
	1	0.091	0.794	0	2	4.35
	2	0.069	0.833	0	1	2.17
	3	0.082	0.833	0	1	2.17
	final	0.242	**0.833**	**0**	**4**	**8.70**
case57_ieee	initial configuration	-	$\gamma(\emptyset) = 0.938$	0	-	-
	1	0.371	1.038	2	10	12.5
	2	0.142	1.038	2	3	3.75
	3	0.085	1.038	2	1	1.25
	final	0.598	**1.038**	**2**	**14**	**17.5**
case73_ieee	initial configuration	-	$\gamma(\emptyset) = 0.632$	0	-	-
	1	0.076	0.778	0	1	0.83
	2	0.099	0.772	0	2	1.67
	3	0.156	0.694	0	4	3.33
	final	0.242	**0.694**	**7**	**6**	**5.83**
case118_ieee	initial configuration	-	$\gamma(\emptyset) = 1.000$	2	-	-
	1	0.278	1.011	2	5	2.69
	2	0.168	1.045	2	4	2.15
	3	0.135	1.045	2	3	1.61
	final	0.581	**1.045**	**2**	**12**	**6.45**
case179_goc	initial configuration	-	$\gamma(\emptyset) = 1.000$	4	-	-
	1	0.148	1.382	5	3	1.14
	2	0.154	1.382	5	2	0.76
	3	0.251	1.382	5	6	2.28
	final	0.553	**1.382**	**5**	**11**	**4.18**
case200_activ	initial configuration	-	$\gamma(\emptyset) = 0.708$	0	-	-
	1	0.221	0.591	0	5	2.04
	2	0.281	0.591	0	8	3.26
	3	0.141	0.605	0	3	1.22
	final	0.643	**0.605**	**0**	**16**	**6.52**
case300_ieee	initial configuration	-	$\gamma(\emptyset) = 1.000$	11	-	-
	1	0.509	1.161	5	12	2.91
	2	0.223	1.161	5	3	0.73
	3	0.386	1.197	7	8	1.95
	final	1.118	**1.197**	**7**	**23**	**5.59**
case1888_rte	initial configuration	-	$\gamma(\emptyset) = 1.000$	14	-	-
	1	4.567	0.869	0	48	1.89
	2	1.994	0.869	0	9	0.36
	3	2.694	0.869	0	24	0.95
	final	9.225	**0.869**	**0**	**81**	**3.20**
case2735_sop	initial configuration	-	$\gamma(\emptyset) = 1.000$	1	-	-
	1	12.57	0.964	0	116	3.54
	2	1.317	0.964	0	19	0.58
	3	2.145	2.637	13	33	1.00
	final	16.032	**2.637**	**13**	**168**	**5.12**

Table 8: Results obtained by the recursive algorithm stopped after 3 iterations using the FastGreedy method for the candidate bridge-blocks identification.

By comparing Tables 7 and 8 it is clear that among the two algorithms, the one-shot algorithm is the one with slightly better performance, as it yields lower network congestion level for almost all the test networks considered. On the other hand, the recursive algorithm still yields good results (even when using only the FastGreedy method for the first subproblem), but it is orders of magnitude faster.

7 Concluding remarks & future directions

In this paper, we proposed a novel spectral representation of power systems which captures the topological and physical interactions within a power grid. Particularly, we illustrated that the graph Laplacian is of importance to characterize the power distributions. This spectral perspective further provides a simple, exact and clear representation of the power redistribution after line failures, where we introduced the network block decomposition to precisely characterize the boundary of the region a line failure can impact.

A finer block decomposition of the power grid creates more, smaller management regions whose operations can be isolated from each other in various aspects, bringing benefits such as disruption localizability, optimization parallelization, and more to the control and management of power systems. Therefore, we introduced a two-stage procedure to optimally modify the network topology for a finer block decomposition. Specifically, we first identify the candidate power network blocks by solving a weighted version of the network modularity problem and then identify the line switching action that has the least impact on the surviving network with respect to line congestion. This two-stage procedure is validated with numerical experiments.

In the rest of this section, we present a collection of examples on how network block decomposition offers powerful analytical guarantees and benefits in different power system applications beyond the setting of this paper, and how we can leverage these properties to improve the current industry practices in these applications.

Real-time failure localization and mitigation. Reliability is a crucial part for the development of sustainable modern power systems. Current industry practice relies on simulations based contingency analysis which is usually constrained by computational power [130]. Since the cascading failures exhibit a complicated and non-local propagation pattern, such analysis tends to omit the structural properties of failures and may fail to control and mitigate the failure in real time. There are topological models adopted from complex network analysis, however, such models are relatively simple thus cannot reveal physical properties [68].

As introduced in Sections 2 and 4, network block decomposition leads to clear and well-defined boundaries that can localize the impact of failures. This suggests a new technique for grid reliability by switching off a small number of transmission lines so that finer-grained network blocks are created. In [60], a real-time distributed control strategy utilizing the localizability from network block decomposition is proposed. Such control strategy is guaranteed to (i) always prevent successive failures from happening; and (ii) localize the impact of initial failures as much as possible. Simulation results show that the controller significantly improves the overall grid reliability.

Security constrained OPF. Security constrained OPF (SC OPF) is another approach to improve grid reliability that works by adding a set of contingencies and requiring the operating point to be safe under any of such contingency. These additional constraints greatly increase the size of the problem [17]. In practice, the SC OPF is solved with contingency filtering techniques [18] to reduce the problem size and/or iterative methods such as Alternating Direction Method of Multipliers (ADMM) [21] for distributed computations.

With a finer network block decomposition, both approaches can be significantly accelerated. The block induces a natural way to decompose and decouple the SC OPF, reducing the number of constraints and variables in each region. In addition, the messages necessary to pass across are reduced, leading to a fast convergence for the local optimizations. Therefore, there is huge potential in accelerating the computation of SC OPF when the network can be decomposed into comparable blocks.

Distributed AC OPF. Optimal Power Flow (OPF) underlies the efficient and stable operation of power grids, reducing the generation cost and transmission loss while maintaining the system stability in steady state. AC OPF problems are challenging because of their highly non-convex nature stemming from the nonlinear relationship between voltages and the complex powers. To overcome this challenge, semidefinite programming (SDP) relaxation of the AC OPF problem, where the optimization variable is transformed to a matrix $\mathbf{V}$ that needs to be semidefinite positive, i.e. $\mathbf{V} \succeq 0$, has been proposed; see [98, 86] for surveys to a considerable literature. However, the standard SDP solvers do not scale well with the network size, necessitating a distributed solution where multiple agents communicate with each other and solve the relaxed problem together. This has proved difficult because the $\mathbf{V} \succeq 0$ constraint is globally coupled, rendering ADMM to be inapplicable. Nevertheless, it is shown in [29] that if the power grid exhibits a block decomposition and each block has an nonempty interior, then the partial matrix with known entries specified by these blocks is chordal and thus can be completed to a feasible global solution $\mathbf{V} \succeq 0$ (see. e.g., [91]). This enables the application of ADMM if the SDP optimization is decomposed in accordance with the network block structure and allows the optimization to be distributed to each block. Therefore, a finer network block decomposition helps reduce the overall operation cost by allowing AC OPF to be solved more efficiently.

Monitoring of price manipulation. Aggregators are playing an increasingly important role in the construction of a morden power grid, especially with the higher penetration level of renewable energy resources. The aggregators install, monitor and manage household energy devices such as rooftop solar panels and electric vehicle charging stations, and coordinate with utilities and Idependent System Operators (ISOs) to provide renewable generation and demand response. In addition to the benefits of aggregators, however, aggregators have the potential to manipulate prices through strategic curtailment for more profit [121]. To quantify the potential for aggregators to exercise market power, in [121] an algorithmic framework is given to approximately solve a bilevel quadratic optimization problem. Their method relies on decomposing the optimization problem into local parts and solve approximately with dynamic programming, thus only works for radial networks. It can be shown that similar bound holds in general networks if the block size of the network is upper bounded. Therefore, it is possible to extend the analysis from radial networks to general networks with proper block decomposition.

Acknowledgements

References

[1] G. Agarwal and D. Kempe. Modularity-maximizing graph communities via mathematical programming. *The European Physical Journal B*, 66(3):409–418, Dec 2008.

[2] O. Alsaç, B. Stott, and W. F. Tinney. Sparsity-oriented compensation methods for modified network solutions. *IEEE Transactions on Power Apparatus and Systems*, PAS-102(5):1,050–1,060, May 1983.

[3] Awerbuch and Shiloach. New connectivity and MSF algorithms for shuffle-exchange network and PRAM. *IEEE Transactions on Computers*, C-36(10):1258–1263, Oct. 1987.

[4] S. Babaeinejadsarookolaee, A. Birchfield, R. D. Christie, C. Coffrin, C. DeMarco, R. Diao, M. Ferris, S. Fliscounakis, S. Greene, R. Huang, C. Josz, R. Korab, B. Lesieutre, J. Maeght, D. K. Molzahn, T. J. Overbye, P. Panciatici, B. Park, J. Snodgrass, and R. Zimmerman. The Power Grid Library for Benchmarking AC Optimal Power Flow Algorithms. *arXiv:1908.02788*, 2019.

[5] R. Bacher and H. Glavitsch. Network topology optimization with security constraints. *IEEE Transactions on Power Systems*, 1(4):103–111, 1986.

[6] R. Bacher and H. Glavitsch. Loss reduction by network switching. *IEEE Transactions on Power Systems*, 3(2):447–454, 1988.

[7] J. Bialek and V. Vahidinasab. Tree-partitioning as an emergency measure to contain cascading line failures. *Submitted to IEEE Transactions on Power Systems*, 2020.

[8] D. Bienstock. *Electrical Transmission System Cascades and Vulnerability*. Society for Industrial and Applied Mathematics, Dec. 2015.

[9] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, oct 2008.

[10] S. Blumsack, P. Hines, M. Patel, C. Barrows, and E. C. Sanchez. Defining power network zones from measures of electrical distance. In *2009 IEEE Power Energy Society General Meeting*, pages 1–8, July 2009.

[11] S. Boettcher and A. G. Percus. Optimization with extremal dynamics. *Complex.*, 8(2):57–62, Nov. 2002.

[12] M. Bolla. Penalized versions of the newman-girvan modularity and their relation to normalized cuts and k-means clustering. *Phys. Rev. E*, 84:016108, Jul 2011.

[13] Z. M. Boyd, E. Bae, X.-C. Tai, and A. L. Bertozzi. Simplified energy landscape for modularity using total variation. *SIAM Journal on Applied Mathematics*, 78(5):2439–2464, 2018.

[14] U. Brandes, D. Delling, M. Gaertler, R. Goerke, M. Hoefer, Z. Nikoloski, and D. Wagner. Maximizing modularity is hard. *arXiv preprint physics/0608255*, 2006.

[15] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner. On modularity clustering. *IEEE Transactions on Knowledge and Data Engineering*, 20(2):172–188, Feb 2008.

[16] A. Brouwer and W. Haemers. *Spectra of Graphs*. Universitext. Springer New York, 2011.

[17] F. Capitanescu, J. M. Ramos, P. Panciatici, D. Kirschen, A. M. Marcolini, L. Platbrood, and L. Wehenkel. State-of-the-art, challenges, and future trends in security constrained optimal power flow. *Electric Power Systems Research*, 81(8):1731–1741, 2011.

[18] F. Capitanescu and L. Wehenkel. A new iterative approach to the corrective security-constrained optimal power flow problem. *IEEE transactions on power systems*, 23(4):1533–1541, 2008.

[19] J. G. Caputo, A. Knippel, and N. Retiere. Spectral solution of load flow equations, 2019.

[20] H. Cetinay, F. A. Kuipers, and P. Van Mieghem. A topological investigation of power flow. *IEEE Systems Journal*, 12(3):2524–2532, Sep. 2018.

[21] S. Chakrabarti, M. Kraning, E. Chu, R. Baldick, and S. Boyd. Security constrained optimal power flow via proximal message passing. In *2014 Clemson University Power Systems Conference*, pages 1–8. IEEE, 2014.

[22] M. Chakraborty, S. Chowdhury, J. Chakraborty, R. Mehera, and R. K. Pal. Algorithms for generating all possible spanning trees of a simple undirected connected graph: an extensive review. *Complex & Intelligent Systems*, 5(3):265–281, 2019.

[23] M. Chen, K. Kuzmin, and B. K. Szymanski. Community detection via maximization of modularity and its variants. *IEEE Transactions on Computational Social Systems*, 1(1):46–65, March 2014.

[24] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Phys. Rev. E*, 70:066111, Dec 2004.

[25] C. Coffrin and R. Zimmerman. Github repository pglib-opf v20.07. https://github.com/power-grid-lib/pglib-opf.

[26] E. Cotilla-Sanchez, P. Hines, C. Barrows, S. Blumsack, and M. Patel. Multi-attribute partitioning of power networks based on electrical distance. *IEEE Transactions on Power Systems*, 28(4):4979–4987, Nov 2013.

[27] E. Cotilla-Sanchez, P. D. H. Hines, C. Barrows, and S. Blumsack. Comparing the topological and electrical structure of the north american electric power infrastructure. *IEEE Systems Journal*, 6(4):616–626, Dec 2012.

[28] L. Cuadra, S. Salcedo-Sanz, J. D. Ser, S. Jiménez-Fernández, and Z. Geem. A critical review of robustness in power grids using complex networks concepts. *Energies*, 8(9):9211–9265, Aug. 2015.

[29] E. Dall'Anese, H. Zhu, and G. B. Giannakis. Distributed optimal power flow for smart microgrids. *IEEE Transactions on Smart Grid*, 4(3):1464–1475, 2013.

[30] I. Dobson, B. A. Carreras, D. E. Newman, and J. M. Reynolds-Barredo. Obtaining statistics of cascading line outages spreading in an electric transmission network from standard utility data. *IEEE Transactions on Power Systems*, 31(6):4831–4841, Nov 2016.

[31] F. Dorfler and F. Bullo. Kron reduction of graphs with applications to electrical networks. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 60(1):150–163, 2013.

[32] P. G. Doyle and J. L. Snell. Random Walks and Electric Networks. *American Mathematical Monthly*, 94(January):202, 2000.

[33] J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. *Phys. Rev. E*, 72:027104, Aug 2005.

[34] F. Dörfler and F. Bullo. Novel insights into lossless AC and DC power flow. In *2013 IEEE Power Energy Society General Meeting*, pages 1–5, 2013.

[35] F. Dörfler and F. Bullo. Synchronization in complex networks of phase oscillators: A survey. *Automatica*, 50(6):1539–1564, 2014.

[36] F. Dörfler, J. W. Simpson-Porco, and F. Bullo. Electrical networks and algebraic graph theory: Models, properties, and applications. *Proceedings of the IEEE*, 106(5):977–1005, May 2018.

[37] W. Ellens, F. Spieksma, P. Van Mieghem, a. Jamakovic, and R. Kooij. Effective graph resistance. *Linear Algebra and its Applications*, 435(10):2491–2506, nov 2011.

[38] M. K. Enns, J. J. Quada, and B. Sackett. Fast linear contingency analysis. *IEEE Transactions on Power Apparatus and Systems*, PAS-101(4):783–791, April 1982.

[39] A. Esmaeilian and M. Kezunovic. Prevention of power grid blackouts using intentional islanding scheme. *IEEE Transactions on Industry Applications*, 53(1):622–629, Jan 2017.

[40] N. Fan, D. Izraelevitz, F. Pan, P. M. Pardalos, and J. Wang. A mixed integer programming approach for optimal power grid intentional islanding. *Energy Systems*, 3(1):77–93, Mar 2012.

[41] E. B. Fisher, R. P. O'Neill, and M. C. Ferris. Optimal transmission switching. *IEEE Transactions on Power Systems*, 23(3):1346–1355, 2008.

[42] S. Fliscounakis, F. Zaoui, G. Simeant, and R. Gonzalez. Topology influence on loss reduction as a mixed integer linear programming problem. In *2007 IEEE Lausanne Power Tech*, pages 1987–1990, 2007.

[43] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75 – 174, 2010.

[44] S. Fortunato and D. Hric. Community detection in networks: A user guide. *Physics Reports*, 659:1 – 44, 2016. Community detection in networks: A user guide.

[45] A. Ghosh, S. Boyd, and A. Saberi. Minimizing Effective Resistance of a Graph. *SIAM Review*, 50(1):37–66, 2008.

[46] H. Glavitsch. Switching as means of control in the power system. *International Journal of Electrical Power & Energy Systems*, 7(2):92–100, 1985.

[47] S. Gómez, P. Jensen, and A. Arenas. Analysis of community structure in networks of correlated data. *Phys. Rev. E*, 80:016114, Jul 2009.

[48] G. Granelli, M. Montagna, F. Zanellini, P. Bresesti, R. Vailati, and M. Innorta. Optimal network reconfiguration for congestion management by deterministic and genetic algorithms. *Electric Power Systems Research*, 76(6):549–556, 2006.

[49] P. Gribik, D. Shirmohammadi, S. Hao, and C. Thomas. Optimal power flow sensitivity analysis. *IEEE Transactions on Power Systems*, 5(3):969–976, August 1990.

[50] M. Guerrero, C. Gil, F. G. Montoya, A. Alcayde, and R. Baños. Community detection in power grids by an evolutionary method. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '17, pages 279–280, New York, NY, USA, 2017. ACM.

[51] R. Guimerà and L. A. N. Amaral. Cartography of complex networks: modules and universal roles. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(02):P02001, feb 2005.

[52] T. Güler and G. Gross. Detection of island formation and identification of causal factors under multiple line outages. *IEEE Transactions on Power Systems*, 22(2):505–513, May 2007.

[53] T. Güler, G. Gross, and M. Liu. Generalized line outage distribution factors. *IEEE Transactions on Power Systems*, 22(2):879–881, May 2007.

[54] J. Guo, Y. Fu, Z. Li, and M. Shahidehpour. Direct calculation of line outage distribution factors. *IEEE Transactions on Power Systems*, 24(3):1633–1634, Aug 2009.

[55] L. Guo, C. Liang, and S. H. Low. Monotonicity properties and spectral characterization of power redistribution in cascading failures. In *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 918–925, Oct 2017.

[56] L. Guo, C. Liang, A. Zocca, S. Low, and A. Wierman. Fast-timescale control of cascading failures in power systems. *Submitted. Preprint at arXiv:2005.11319*, 2021.

[57] L. Guo, C. Liang, A. Zocca, S. Low, and A. Wierman. Line failure localization of power networks part i: Non-cut outages. *IEEE Transactions on Power Systems*, 2021.

[58] L. Guo, C. Liang, A. Zocca, S. Low, and A. Wierman. Line failure localization of power networks part ii: Cut set outages. *IEEE Transactions on Power Systems*, 2021.

[59] L. Guo, C. Liang, A. Zocca, S. H. Low, and A. Wierman. Failure localization in power systems via tree partitions. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 6832–6839, Miami Beach, FL, Dec 2018.

[60] L. Guo, C. Liang, A. Zocca, S. H. Low, and A. Wierman. Less is more: Real-time failure localization in power systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3871–3877, 2019.

[61] J. Han and A. Papavasiliou. The Impacts of Transmission Topology Control on the European Electricity Network. *IEEE Transactions on Power Systems*, 31(1):496–507, 2016.

[62] F. Harary. *Graph Theory*. Addison Wesley series in mathematics. Addison-Wesley, 1971.

[63] A. R. Hassani Ahangar, G. B. Gharehpetian, and H. R. Baghaee. A review on intentional controlled islanding in smart power systems and generalized framework for ici in microgrids. *International Journal of Electrical Power & Energy Systems*, 118:105709, 2020.

[64] A. Hauswirth, S. Bolognani, G. Hug, and F. Dörfler. Generic existence of unique Lagrange multipliers in AC optimal power flow. *IEEE Control Systems Letters*, 2(4), October 2018.

[65] K. W. Hedman, M. C. Ferris, R. P. O'Neill, E. B. Fisher, and S. S. Oren. Co-optimization of generation unit commitment and transmission switching with n-1 reliability. *IEEE Transactions on Power Systems*, 25(2):1052–1063, 2010.

[66] K. W. Hedman, R. P. O'Neill, E. B. Fisher, and S. S. Oren. Optimal transmission switching with contingency analysis. *IEEE Transactions on Power Systems*, 24(3):1577–1586, 2009.

[67] K. W. Hedman, S. S. Oren, and R. P. O'Neill. A review of transmission switching and network topology optimization. In *2011 IEEE Power and Energy Society General Meeting*, pages 1–7, 2011.

[68] P. Hines, E. Cotilla-Sanchez, and S. Blumsack. Do topological models provide good information about electricity infrastructure vulnerability? *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 20(3):033122, 2010.

[69] P. D. H. Hines, I. Dobson, E. Cotilla-Sanchez, and M. Eppstein. "dual graph" and "random chemistry" methods for cascading failure analysis. In *2013 46th Hawaii International Conference on System Sciences*, pages 2141–2150, Jan 2013.

[70] P. D. H. Hines, I. Dobson, and P. Rezaei. Cascading power outages propagate locally in an influence graph that is not the actual grid topology. *IEEE Transactions on Power Systems*, 32(2):958–967, March 2017.

[71] J. Hopcroft and R. Tarjan. Algorithm 447: efficient algorithms for graph manipulation. *Communications of the ACM*, 16(6):372–378, June 1973.

[72] Y.-F. Huang, S. Werner, J. Huang, N. Kashyap, and V. Gupta. State estimation in electric power grids: Meeting new challenges presented by the requirements of the future grid. *IEEE Signal Processing Magazine*, 29(5):33–43, 2012.

[73] J. Hörsch, H. Ronellenfitsch, D. Witthaut, and T. Brown. Linear optimal power flow using cycle flows. *Electric Power Systems Research*, 158:126–135, 2018.

[74] J. Johnson and M. Chertkov. A majorization-minimization approach to design of power transmission networks. In *49th IEEE Conference on Decision and Control (CDC)*, pages 3996–4003. IEEE, 2010.

[75] Kai Sun, Da-Zhong Zheng, and Qiang Lu. Splitting strategies for islanding operation of large-scale power systems using obdd-based methods. *IEEE Transactions on Power Systems*, 18(2):912–923, May 2003.

[76] F. Kaiser, V. Latora, and D. Witthaut. Inhibiting failure spreading in complex networks, 2020.

[77] F. Kaiser, J. Strake, and D. Witthaut. Collective effects of link failures in linear flow networks. *New Journal of Physics*, 22(1):013053, jan 2020.

[78] F. Kaiser and D. Witthaut. Topological theory of resilience and failure spreading in flow networks, 2020.

[79] M. Khanabadi, H. Ghasemi, and M. Doostizadeh. Optimal transmission switching considering voltage security and n-1 contingency analysis. *IEEE Transactions on Power Systems*, 28(1):542–550, 2012.

[80] D. J. Klein and M. Randić. Resistance distance. *Journal of Mathematical Chemistry*, 12(1):81–95, 1993.

[81] F. Koeth. *Investigations of Spectral Properties in Power Systems*. Theses, Université Grenoble Alpes, Dec. 2019.

[82] F. Koeth and N. Retiere. Spectral properties of dynamical power systems. In *2019 IEEE Milan PowerTech*, pages 1–6, 2019.

[83] H.-J. Koglin and H. Muller. Overload reduction through corrective switching actions. In *Int. Conf. Power Sys. Monitor. & Contr*, volume 24, 1980.

[84] A. S. Korad and K. W. Hedman. Robust corrective topology control for system reliability. *IEEE Transactions on Power Systems*, 28(4):4042–4051, 2013.

[85] Y. Koç, M. Warnier, R. Kooij, and F. Brazier. Structural vulnerability assessment of electric power grids. In *Proceedings of the 11th IEEE International Conference on Networking, Sensing and Control*, pages 386–391, April 2014.

[86] J. Lavaei and S. H. Low. Zero duality gap in optimal power flow problem. *IEEE Transactions on Power Systems*, 27(1):92–107, 2011.

[87] C. Liang, L. Guo, A. Zocca, S. Yu, S. H. Low, and A. Wierman. An integrated approach for failure mitigation & localization in power systems. *Electric Power Systems Research*, 190:106613, 2021.

[88] G. Lin, S. Liu, A. Zhou, J. Dai, B. Chai, B. Zhang, H. Qiu, K. Gao, Y. Song, and R. Chen. Community detection in power grids based on louvain heuristic algorithm. In *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*, pages 1–4, 2017.

[89] L. Liu, W. Liu, D. A. Cartes, and I.-Y. Chung. Slow coherency and angle modulated particle swarm optimization based islanding of large-scale power systems. *Advanced Engineering Informatics*, 23(1):45 – 56, 2009.

[90] S. Liu, Z. Lin, Y. Zhao, Y. Liu, Y. Ding, B. Zhang, L. Yang, Q. Wang, and S. E. White. Robust system separation strategy considering online wide-area coherency identification and uncertainties of renewable energy sources. *IEEE Transactions on Power Systems*, 35(5):3574–3587, 2020.

[91] S. H. Low. Convex relaxation of optimal power flow—part ii: Exactness. *IEEE Transactions on Control of Network Systems*, 1(2):177–189, 2014.

[92] Z. Lu, X. Sun, Y. Wen, G. Cao, and T. L. Porta. Algorithms and applications for community detection in weighted networks. *IEEE Transactions on Parallel and Distributed Systems*, 26(11):2916–2926, Nov 2015.

[93] Z. Lu, Y. Wen, and G. Cao. Community detection in weighted networks: Algorithms and applications. In *2013 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 179–184, March 2013.

[94] A. Marot, S. Tazi, B. Donnot, and P. Panciatici. Guided machine learning for power grid segmentation. In *2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, pages 1–6, 2018.

[95] C. P. Massen and J. P. K. Doye. Identifying communities within energy landscapes. *Phys. Rev. E*, 71:046101, Apr 2005.

[96] A. A. Mazi, B. F. Wollenberg, and M. H. Hesse. Corrective control of power system flows by line and bus-bar switching. *IEEE Transactions on Power Systems*, 1(3):258–264, 1986.

[97] A. Medus, G. Acuña, and C. Dorso. Detection of community structures in networks via global optimization. *Physica A: Statistical Mechanics and its Applications*, 358(2):593 – 604, 2005.

[98] D. K. Molzahn and I. A. Hiskens. A survey of relaxations and approximations of the power flow equations. *Foundations and Trends in Electric Energy Systems*, 4(1-2):1–221, 2019.

[99] U. Nakarmi and M. Rahnamay-Naeini. Analyzing power grids' cascading failures and critical components using interaction graphs. In *2018 IEEE Power Energy Society General Meeting (PESGM)*, pages 1–5, Aug 2018.

[100] U. Nakarmi, M. Rahnamay-Naeini, and H. Khamfroush. Critical component analysis in cascading failures for power grids using community structures in interaction graphs. *IEEE Transactions on Network Science and Engineering*, 2019.

[101] T. Nesti, A. Zocca, and B. Zwart. Emergent failures and cascades in power grids: A statistical physics perspective. *Phys. Rev. Lett.*, 120:258301, 2018.

[102] M. E. J. Newman. Analysis of weighted networks. *Phys. Rev. E*, 70:056131, Nov 2004.

[103] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Phys. Rev. E*, 69:066133, Jun 2004.

[104] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E*, 74:036104, Sep 2006.

[105] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

[106] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69:026113, Feb 2004.

[107] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, pages 849–856, Cambridge, MA, USA, 2001. MIT Press.

[108] G. A. Pagani and M. Aiello. The power grid as a complex network: A survey. *Physica A: Statistical Mechanics and its Applications*, 392(11):2688 – 2700, 2013.

[109] S. Pahwa, M. Youssef, P. Schumm, C. Scoglio, and N. Schulz. Optimal intentional islanding to enhance the robustness of power grid networks. *Physica A: Statistical Mechanics and its Applications*, 392(17):3741–3754, 2013.

[110] M. Pirani, J. W. Simpson-Porco, and B. Fidan. System-Theoretic Performance Metrics for Low-Inertia Stability of Power Networks. *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 5106–5111, 2017.

[111] M. A. Porter, J.-P. Onnela, and P. J. Mucha. Communities in networks. *Notices of the AMS*, 56(9):1082–1097, 2009.

[112] Qianchuan Zhao, Kai Sun, Da-Zhong Zheng, Jin Ma, and Qiang Lu. A study of system splitting strategies for island operation of power system: a two-phase method based on obdds. *IEEE Transactions on Power Systems*, 18(4):1556–1565, Nov 2003.

[113] J. Quirós-Tortós and V. Terzija. A graph theory based new approach for power system restoration. In *2013 IEEE Grenoble Conference*, pages 1–6, 2013.

[114] G. Ranjan, Z.-L. Zhang, and D. Boley. Incremental Computation of Pseudo-Inverse of Laplacian. In *COCOA 2014: 8th International Conference on Combinatorial Optimization and Applications*, volume 5573, pages 729–749. Springer International Publishing, 2014.

[115] J. Reichardt and S. Bornholdt. Statistical mechanics of community detection. *Phys. Rev. E*, 74:016110, Jul 2006.

[116] N. Retire, D. T. Ha, and J.-G. Caputo. Spectral Graph Analysis of the Geometry of Power Flows in Transmission Networks. *IEEE Systems Journal*, 14(2):2736–2747, 2019.

[117] T. Richardson, P. J. Mucha, and M. A. Porter. Spectral tripartitioning of networks. *Phys. Rev. E*, 80:036111, Sep 2009.

[118] J. G. Rolim and L. J. B. Machado. A study of the use of corrective switching in transmission systems. *IEEE Transactions on Power Systems*, 14(1):336–341, 1999.

[119] H. Ronellenfitsch, D. Manik, J. Horsch, T. Brown, and D. Witthaut. Dual theory of transmission line outages. *IEEE Transactions on Power Systems*, 32(5):4060–4068, Sept. 2017.

[120] H. Ronellenfitsch, M. Timme, and D. Witthaut. A dual method for computing power transfer distribution factors. *IEEE Transactions on Power Systems*, 32(2):1007–1015, 2017.

[121] N. A. Ruhi, K. Dvijotham, N. Chen, and A. Wierman. Opportunities for price manipulation by aggregators in electricity markets. *IEEE Transactions on Smart Grid*, 9(6):5687–5698, 2017.

[122] M. Sales-Pardo, R. Guimerà, A. A. Moreira, and L. A. N. Amaral. Extracting the hierarchical organization of complex systems. *Proceedings of the National Academy of Sciences*, 104(39):15224–15229, 2007.

[123] G. Schnyder and H. Glavitsch. Integrated security control using an optimal power flow and switching concepts. *IEEE Transactions on Power Systems*, 3(2):782–790, 1988.

[124] G. Schnyder and H. Glavitsch. Security enhancement using an optimal switching power flow. *IEEE Transactions on Power Systems*, 5(2):674–681, 1990.

[125] W. Shao and V. Vittal. Corrective switching algorithm for relieving overloads and voltage violations. *IEEE Transactions on Power Systems*, 20(4):1877–1885, 2005.

[126] W. Shao and V. Vittal. Bip-based opf for line and bus-bar switching to relieve overloads and voltage violations. In *2006 IEEE PES Power Systems Conference and Exposition*, pages 2090–2095, 2006.

[127] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, Aug. 2000.

[128] S. Soltan, D. Mazauric, and G. Zussman. Analysis of failures in power grids. *IEEE Transactions on Control of Network Systems*, 4(2):288–300, June 2017.

[129] S. Soltan, M. Yannakakis, and G. Zussman. *Doubly Balanced Connected Graph Partitioning*, pages 1939–1950. SIAM, 2017.

[130] J. Song, E. Cotilla-Sanchez, G. Ghanavati, and P. D. Hines. Dynamic modeling of cascading failure in power systems. *IEEE TPS*, 31(3):2085–2095, 2016.

[131] M. Stoer and F. Wagner. A simple min-cut algorithm. *J. ACM*, 44(4):585–591, July 1997.

[132] B. Stott, O. Alsaç, and F. L. Alvarado. Analytical and computational improvements in performance-index ranking algorithms for networks. *International Journal of Electrical Power & Energy Systems*, 7(3):154–160, July 1985.

[133] B. Stott, J. Jardim, and O. Alsac. DC power flow revisited. *IEEE Transactions on Power Systems*, 24(3):1290–1300, 2009.

[134] J. Strake, F. Kaiser, F. Basiri, H. Ronellenfitsch, and D. Witthaut. Non-local impact of link failures in linear flow networks. *New Journal of Physics*, 21(5):053009, May 2019.

[135] R. Sánchez-García, M. Fennelly, S. Norris, N. Wright, G. Niblo, J. Brodzki, and J. Bialek. Hierarchical spectral clustering of power grids. *IEEE Transactions on Power Systems*, 29(5):2229–2237, Sept 2014.

[136] R. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, June 1972.

[137] R. E. Tarjan and U. Vishkin. An efficient parallel biconnectivity algorithm. *SIAM Journal on Computing*, 14(4):862–874, Nov. 1985.

[138] P. Van Mieghem. *Graph Spectra for Complex Networks*. CUP, 2011.

[139] V. Vittal, W. Kliemann, Y. . Ni, D. G. Chapman, A. D. Silk, and D. J. Sobajic. Determination of generator groupings for an islanding scheme in the manitoba hydro system using the method of normal forms. *IEEE Transactions on Power Systems*, 13(4):1345–1351, Nov 1998.

[140] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[141] D. Wagner and F. Wagner. Between min cut and graph bisection. In *Mathematical Foundations of Computer Science 1993*, pages 744–750, Berlin, Heidelberg, 1993. Springer Berlin Heidelberg.

[142] K. Wakita and T. Tsurumi. Finding community structure in mega-scale social networks: [extended abstract]. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 1275–1276, New York, NY, USA, 2007. ACM.

[143] X. Wang, Y. Koç, R. E. Kooij, and P. Van Mieghem. A network approach for power grid robustness against cascading failures. In *2015 7th International Workshop on Reliable Networks Design and Modeling (RNDM)*, pages 208–214, Oct 2015.

[144] X. Wang, E. Pournaras, R. E. Kooij, and P. Van Mieghem. Improving robustness of complex networks via the effective graph resistance. *The European Physical Journal B*, 87(9):221, Sep 2014.

[145] J. Westbrook and R. E. Tarjan. Maintaining bridge-connected and biconnected components on-line. *Algorithmica*, 7(1-6):433–464, June 1992.

[146] S. White and P. Smyth. *A Spectral Clustering Approach To Finding Communities in Graphs*, pages 274–285. SIAM, 2005.

[147] A. J. Wood, B. F. Wollenberg, and G. B. Sheblé. *Power generation, operation, and control.* John Wiley & Sons, 2013.

[148] H. You, V. Vittal, and X. Wang. Slow coherency-based islanding. *IEEE Transactions on Power Systems*, 19(1):483–491, 2004.

[149] H. You, V. Vittal, and Z. Yang. Self-healing in power systems: an approach using islanding and rate of frequency decline-based load shedding. *IEEE Transactions on Power Systems*, 18(1):174–181, 2003.

[150] H. You, V. Vittal, and Z. Yang. Self-healing in power systems: an approach using islanding and rate of frequency decline-based load shedding. *IEEE Transactions on Power Systems*, 18(1):174–181, Feb 2003.

[151] L. Yu and C. Ding. Network community discovery: Solving modularity clustering via normalized cut. In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, MLG '10, pages 34–36, New York, NY, USA, 2010. ACM.

[152] A. Zocca and B. Zwart. Optimization of stochastic lossy transport networks and applications to power grids. *Stochastic Systems*, 11(1):34–59, Mar. 2021.

A   Appendix

A.1   Proof of Proposition 2.2

(i) From its definition (1), it is clear that we can rewrite the Laplacian matrix L as

$$L = \sum_{(i,j)\in E} b_{ij}(\boldsymbol{e}_i\boldsymbol{e}_i^T + \boldsymbol{e}_j\boldsymbol{e}_j^T - \boldsymbol{e}_i\boldsymbol{e}_j^T - \boldsymbol{e}_j\boldsymbol{e}_i^T).$$

Therefore, for any $\boldsymbol{x} \in \mathbb{R}^n$ we have

$$\boldsymbol{x}^T L \boldsymbol{x} = \boldsymbol{x}^T \Big(\sum_{(i,j)\in E} b_{ij}(\boldsymbol{e}_i\boldsymbol{e}_i^T + \boldsymbol{e}_j\boldsymbol{e}_j^T - \boldsymbol{e}_i\boldsymbol{e}_j^T - \boldsymbol{e}_j\boldsymbol{e}_i^T) \Big)\boldsymbol{x} = \sum_{(i,j)\in E} b_{ij}\,\boldsymbol{x}^T(\boldsymbol{e}_i\boldsymbol{e}_i^T + \boldsymbol{e}_j\boldsymbol{e}_j^T - \boldsymbol{e}_i\boldsymbol{e}_j^T - \boldsymbol{e}_j\boldsymbol{e}_i^T)\boldsymbol{x}$$

$$= \sum_{(i,j)\in E} b_{ij}\left(x_i^2 - 2x_ix_j + x_j^2 \right) = \sum_{(i,j)\in E} b_{ij}(x_i - x_j)^2 \geq 0$$

(ii) In (i) we rewrote $\boldsymbol{x}^T L \boldsymbol{x}$ as the quadratic form $\sum_{(i,j)\in E} b_{ij}(x_i - x_j)^2$. This, in combination with the fact that $b_{ij} > 0$ for every line $(i,j) \in E$ suffices to conclude that L is positive semidefinite.

(iii) The row and column property immediately follows from the way the matrix L is defined, cf. (1). Since the kernel of L contains at least the nontrivial vector $\boldsymbol{1}$, the matrix is singular.

(iv) We claim that a vector $\boldsymbol{v}$ is in the null space of L if and only if $v_i = v_j$ for all nodes i,j in the same connected component. For any such pair, there exists a path $\ell_1,\ldots,\ell_r$ connecting them and for all the nodes traversed by this path (including i and j) the corresponding entries of the vector v must be equal, in view of the fact that $\boldsymbol{v}^T L \boldsymbol{v} = 0$ and that $b_{\ell_1},\ldots,b_{\ell_r} > 0$. The reverse implication is trivial. This claim implies that an orthonormal basis of the null space consists of k orthogonal unit vectors $\boldsymbol{v}_j$, $j = 1,\ldots,k$, whose entries are

$$(\boldsymbol{v}_j)_i := \frac{\boldsymbol{1}_{\mathcal{I}_j}(i)}{\sqrt{|\mathcal{I}_j|}}, \qquad i = 1,\ldots,n,$$

where $\mathcal{I}_1,\ldots,\mathcal{I}_k$ are the connected components of the graph G. Hence, the null space of L has a dimension of k and, therefore, $\text{rank}(L) = n - \dim(\ker(L)) = n - k$.

Consider now the singular value decomposition of the symmetric real matrix L, that is $L = U\Sigma U^T$. The columns of U are the orthonormal eigenvectors of L, namely $\frac{1}{\sqrt{|\mathcal{I}_1|}}\boldsymbol{1}_{\mathcal{I}_1},\ldots,\frac{1}{\sqrt{|\mathcal{I}_k|}}\boldsymbol{1}_{\mathcal{I}_k},\boldsymbol{v}_{k+1},\ldots,\boldsymbol{v}_n$. Then the pseudo-inverse $L^\dagger$ is equal to $L^\dagger = U\Sigma^\dagger U^T$, where $\Sigma^\dagger$ is the diagonal $n \times n$ matrix of rank $n - k$ obtained from Σ by replacing each the nonzero eigenvalues λ_i of L by its reciprocal $1/\lambda_i$. The first identity

$$L^\dagger = \sum_{j=k+1}^{n} \frac{1}{\lambda_j}\,\boldsymbol{v}_j\boldsymbol{v}_j^T \tag{32}$$

readily follows from $L^\dagger = U\Sigma^\dagger U^T$ in view of the orthogonality of the eigenvectors and the structure of the null space described above. For the second identity, it is enough to show that

$$\left(L + \sum_{j=1}^{k} \frac{1}{|\mathcal{I}_j|} \mathbf{1}_{\mathcal{I}_j} \mathbf{1}_{\mathcal{I}_j}^T\right)^{-1} = L^\dagger + \sum_{j=1}^{k} \frac{1}{|\mathcal{I}_j|} \mathbf{1}_{\mathcal{I}_j} \mathbf{1}_{\mathcal{I}_j}^T = \sum_{j=k+1}^{n} \frac{1}{\lambda_j} \boldsymbol{v}_j \boldsymbol{v}_j^T + \sum_{j=1}^{k} \frac{1}{|\mathcal{I}_j|} \mathbf{1}_{\mathcal{I}_j} \mathbf{1}_{\mathcal{I}_j}^T.$$

We prove that the RHS of the latter equation is the inverse matrix by direct calculation

$$\left(L + \sum_{j=1}^{k} \frac{1}{|\mathcal{I}_j|} \mathbf{1}_{\mathcal{I}_j} \mathbf{1}_{\mathcal{I}_j}^T\right) \left(\sum_{j=k+1}^{n} \frac{1}{\lambda_j} \boldsymbol{v}_j \boldsymbol{v}_j^T + \sum_{j=1}^{k} \frac{1}{|\mathcal{I}_j|} \mathbf{1}_{\mathcal{I}_j} \mathbf{1}_{\mathcal{I}_j}^T\right) = UU^T = I,$$

and the conclusion follows by uniqueness of the matrix inverse. In the special case in which G is connected, then $k = 1$, $V = \mathcal{I}_1$, and the above expression simplifies in view of the fact that $\frac{1}{|\mathcal{I}_1|} \mathbf{1}_{\mathcal{I}_1} \mathbf{1}_{\mathcal{I}_1}^T = \frac{1}{n} \mathbf{1} \mathbf{1}^T$.

(v) All the listed properties of $L^\dagger$ can be easily deduced from (ii) and identity (32).

A.2   Proof of Proposition 3.2

(i) We can directly inspect that the $(\ell, \hat{\ell})$-th element of the PTDF matrix D, i.e. $D_{\ell\ell}$, coincides with the $(\ell, \hat{\ell})$-th element of the matrix $BC^T L^\dagger C$. More specifically, using the first equation of Proposition 3.1 and choosing $\hat{\ell} = (s, t) \in E$, we have

$$(BC^T L^\dagger C)_{\ell\hat{\ell}} = \mathbf{e}_\ell^T BC^T L^\dagger C \mathbf{e}_{\hat{\ell}} = b_\ell \mathbf{e}_\ell^T C^T L^\dagger C \mathbf{e}_{\hat{\ell}} = b_\ell (\mathbf{e}_i - \mathbf{e}_j)^T L^\dagger (\mathbf{e}_s - \mathbf{e}_t) = b_\ell \left(L_{is}^\dagger + L_{jt}^\dagger - L_{it}^\dagger - L_{js}^\dagger\right) = D_{\ell\hat{\ell}}.$$

(ii) Let us consider the second expression given in Proposition 3.1 for $D_{\ell,\ell}$ in the special case where $\hat{\ell} = (s, t) = (i, j) = \ell$. We obtain

$$D_{\ell,\ell} = b_\ell \frac{\sum_{\mathcal{E}\in T(\{i,i\},\{j,j\})} \beta(\mathcal{E}) - \sum_{\mathcal{E}\in T(\{i,j\},\{j,i\})} \beta(\mathcal{E})}{\sum_{\mathcal{E}\in T} \beta(\mathcal{E})} = b_\ell \frac{\sum_{\mathcal{E}\in T(\{i\},\{j\})} \beta(\mathcal{E})}{\sum_{\mathcal{E}\in T} \beta(\mathcal{E})}, \tag{33}$$

where the last identity follows from the fact that $T(\{i,j\},\{j,i\}) = \emptyset$. The collection $T = T_E$ of all spanning trees of the graph G can be partitioned into two subsets, depending on whether the edge $\ell = (i, j)$ belongs to it or not. All the spanning trees that do contain ℓ are in one-to-one correspondence with the spanning forests in $T(\{i\}, \{j\})$. Indeed, each such spanning tree, once the edge ℓ is removed, yields a spanning forest consisting of two trees, one containing node i and the other one node j. The other subset consisting of all the spanning trees that do not contain ℓ is $T\backslash\{\ell\}$ by definition. In view of these considerations, we can rewrite the sum of all spanning tree weights as

$$\sum_{\mathcal{E}\in T} \beta(\mathcal{E}) = b_\ell \sum_{\mathcal{E}\in T(\{i\},\{j\})} \beta(\mathcal{E}) + \sum_{\mathcal{E}\in T\backslash\{\ell\}} \beta(\mathcal{E}).$$

Using this identity to rewrite the denominator appearing in the RHS of (33) we obtain

$$\frac{b_\ell \sum_{\mathcal{E}\in T(\{i\},\{j\})} \beta(\mathcal{E})}{\sum_{\mathcal{E}\in T} \beta(\mathcal{E})} = \frac{b_\ell \sum_{\mathcal{E}\in T(\{i\},\{j\})} \beta(\mathcal{E})}{b_\ell \sum_{\mathcal{E}\in T(\{i\},\{j\})} \beta(\mathcal{E}) + \sum_{\mathcal{E}\in T\backslash\{\ell\}} \beta(\mathcal{E})} = 1 - \frac{\sum_{\mathcal{E}\in T\backslash\{\ell\}} \beta(\mathcal{E})}{\sum_{\mathcal{E}\in T} \beta(\mathcal{E})}.$$

Since the collection $T\backslash\{\ell\}$ could be empty but it is always strictly included in T, we immediately deduce that $0 < D_{\ell\ell} \leq 1$.

(iii) The two identities readily follow from Proposition 3.1 in the special case $\ell = (s, t)$ combined with the definition of effective resistance given in (4).

A.3   Proof of Theorem 3.4

We prove the result by contradiction. If $D_{\ell\hat{\ell}} \neq 0$, then at least one of the two sets $T(\{i,s\},\{j,t\})$ and $T(\{i,t\},\{j,s\})$ of spanning forests appearing in the numerator of the RHS of (5) is nonempty. Suppose $T(\{i,s\},\{j,t\})$ is nonempty and take any spanning forest $F \in T(\{i,s\},\{j,t\})$. The tree in the forest F that contains buses i and s defines a path from i to s, and the other tree in F that contains j and t defines a path from j to t. Clearly these two paths do not share any vertices. If we add the lines $\ell = (i, j)$ and $\hat{\ell} = (s, t)$ to these two vertex-disjoint paths we obtain a simple cycle that contains both ℓ and $\hat{\ell}$.