Engineering Applications of Artificial Intelligence 56 (2016) 157–174

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

A hierarchical consensus method for the approximation of the consensus state, based on clustering and spectral graph theory

Rita Morisi, Giorgio Gnecco *, Alberto Bemporad

IMT School for Advanced Studies, Piazza S. Francesco, 19, 55100 Lucca, Italy

ARTICLE INFO

Article history:
Received 15 April 2016
Received in revised form
21 June 2016
Accepted 28 August 2016
Available online 17 September 2016

Keywords:
Consensus problem
Approximation
Hierarchical consensus
Clustering
Spectral graph theory

ABSTRACT

A hierarchical method for the approximate computation of the consensus state of a network of agents is investigated. The method is motivated theoretically by spectral graph theory arguments. In a first phase, the graph is divided into a number of subgraphs with good spectral properties, i.e., a fast convergence toward the local consensus state of each subgraph. To find the subgraphs, suitable clustering methods are used. Then, an auxiliary graph is considered, to determine the final approximation of the consensus state in the original network. A theoretical investigation is performed of cases for which the hierarchical consensus method has a better performance guarantee than the non-hierarchical one (i.e., it requires a smaller number of iterations to guarantee a desired accuracy in the approximation of the consensus state of the original network). Moreover, numerical results demonstrate the effectiveness of the hierarchical consensus method for several case studies modeling real-world networks.

1. Introduction

The theory of complex systems deals with the study of the behavior of systems made of several agents (or units) that interact among each other; typical examples are social (Del Vicario et al., 2016) and economic (Battiston et al., 2016) networks, physical systems made of interacting particles (Castellano et al., 2009), and biological (Pastor-Satorras et al., 2015) and ecological (Vivaldo et al., 2016) systems. In all these cases, one has often to deal with a large number of units, which have no global knowledge about the structure of the whole system, as their interactions are limited to their neighbors in the network. Control problems on such systems are strongly influenced by structural properties of their graph of interconnections, described, e.g., in terms of a weighted/unweighted adjacency or graph-Laplacian matrix (Mesbahi and Egerstedt, 2010; Liu et al., 2011). In particular, several studies (see, e.g., Lovisari and Zampieri, 2012 for a tutorial) deal with the analysis of the conditions under which a complex system has all its agents reach asymptotically a common state, called consensus state (i.e., they agree asymptotically with the same opinion) and, in case of a positive answer, with investigating the rate of convergence to the consensus state. It is well-known (see, e.g., Boyd et al., 2004; Lovisari and Zampieri, 2012) that such a convergence rate is related to the spectral properties of the graph of interconnections (e.g., the ones of a transition probability matrix one can associate to it). The work (Boyd et al., 2004) optimizes such properties by solving a suitable convex optimization problem, called *Fastest Mixing Markov-Chain* (FMMC) problem. In our previous work (Gnecco et al., 2015), we optimized a suitable trade-off between the rate of convergence to the consensus state and the sparsity of the graph of interconnections, which is a way to insert in the model a possible cost of communication associated with each link used. In more details, the optimization problem considered in Gnecco et al. (2015) (which is a substantial extension of the conference paper, Gnecco et al., 2014) is an l_1-norm (convex) regularization of the FMMC problem, called FMMC-$l_1(\eta)$ problem, where $\eta > 0$ is a regularization parameter. Its main contributions are some theoretical results about the choice of η to avoid triviality of the resulting optimal solution, and an interpretation of the FMMC-$l_1(\eta)$ problem as a robust version of the FMMC problem, in which one is allowed to select only nominal weights associated with the edges of the graph, as such weights enter the model together with an intrinsic relative uncertainty, which cannot be removed unless the nominal values are chosen to be equal to 0. A (nonconvex) l_0-pseudo-norm regularized version of the FMMC problem is also analyzed in Gnecco et al. (2015). Some ways to restrict the search for its optimal solution to suitable feasible solutions are also investigated therein. Finally, numerical results demonstrate the effectiveness of both regularized approaches (with computational advantages for the convex case) in achieving – as desired – a "good" trade-off between sparsity of the network and its rate of convergence to the consensus state.

* Corresponding author.
E-mail addresses: rita.morisi@imtlucca.it (R. Morisi), giorgio.gnecco@imtlucca.it (G. Gnecco), alberto.bemporad@imtlucca.it (A. Bemporad).

http://dx.doi.org/10.1016/j.engappai.2016.08.018
0952-1976/

The approach followed in this paper is substantially different from Gnecco et al. (2015), although the goal is similar. In more details, the main idea of the present work is the following: for a fixed network topology, we aim at speeding up consensus using a "hierarchical" approach, whose theoretical motivation relies on spectral properties of the agents' network. Our approach is based on dividing the original connected graph into many connected subgraphs, which are expected (due to spectral graph theory arguments, Chung, 1997) to have "good" spectral properties. In this case, the rate of convergence to the "local" consensus state (i.e., the consensus state of each subgraph) is faster than the one to the "global" consensus state of the original graph. In a second phase, the resulting approximations of the local consensus states of the subgraphs are mixed to get (up to a certain tolerance) global consensus on an auxiliary graph, whose nodes are selected nodes of the subgraphs (one for each subgraph), and for which "good" spectral properties are still expected (again, due to spectral graph theory arguments). To generate the subgraphs, we apply both a technique known as spectral clustering (von Luxburg, 2007), and a second ad hoc technique that we call *nearest supernode approach*, which are both expected to extract sufficiently "dense" subgraphs (e.g., made of a single cluster of nodes, with each node connected directly to several other nodes of the same cluster). For such subgraphs, the rate of convergence to the local consensus state is relatively fast (since the second-largest eigenvalue modulus of the transition probability matrix associated with each such subgraph is relatively small). In this way, in the hierarchical approach, one fixes the sparsity of the graph, then speeds up the approximation of its consensus state possibly even more than through the resolution of the FMMC problem, since the latter does not allow for a hierarchical solution. It is worth noting that, in case the original graph is not sparse, one can still apply the hierarchical consensus method described in the paper after a preliminary step of edge sparsification (this could be achieved, e.g., applying the algorithms detailed in Batson et al., 2013), to construct another graph with a very similar spectral behavior, but with a (typically much) smaller number of edges. Then, the hierarchical consensus method could be applied directly to this sparsified graph. It has to be remarked that approaches similar to the one presented in this paper have been proposed also in Epstein et al. (2008) and Li and Bai (2012). In such works, the multi-agent system is also decomposed into a hierarchical structure. Nevertheless, neither Epstein et al. (2008) nor Li and Bai (2012) consider techniques that exploit spectral graph theory arguments for the generation of the subgraphs. Hence, compared with Epstein et al. (2008) and Li and Bai (2012), the main original contribution of the present work lies on the techniques we adopt to determine the different connected subgraphs, and on the theoretical motivations we provide for such techniques, based on spectral graph theory arguments. In addition to this, we perform an extensive numerical evaluation of the hierarchical consensus method on several case studies modeling real-world networks, achieving in most cases better performance with respect to a non-hierarchical consensus method.

The paper is structured as follows. Section 2 presents an introduction to the consensus problem, and provides an overview of the hierarchical consensus method. Section 3 provides some theoretical arguments supporting the method, based on spectral graph theory. Section 4 describes clustering techniques used by the method, whereas Section 6 provides a study of its approximation of the global consensus state. In Section 7, numerical examples are presented. Section 8 provides a refinement of the basic setting, based on the results of the numerical examples. Finally, Section 9 offers conclusions.

2. An overview of the hierarchical consensus method

Let $G = (V, E)$ be a connected undirected graph with $N = |V|$ nodes and $|E|$ edges. In the context of the paper, the nodes represent agents (or units), which locally interact among each other. Such an interaction is governed by non-negative weights associated with the edges, which have to be chosen in a suitable way. Assuming a linear time-invariant model and describing each agent as a 1-dimensional dynamical system, the consensus problem refers to the investigation of the convergence to the consensus state (see the next formula (2)), for the following linear dynamical system:

$$\underline{x}(t+1) = P\underline{x}(t), \tag{1}$$

where the column vector $\underline{x}(t) \in \mathbb{R}^N$ contains the states (opinions) of the N agents at a generic discrete time instant t, while $P \in \mathbb{R}^{N\times N}$ is a symmetric doubly stochastic matrix (i.e., $P_{ij} \geq 0$ for all $i, j = 1, \ldots, N$, $P\underline{1} = \underline{1}$, and $P = P^T$, where $\underline{1}$ is the N-dimensional vector whose components are all equal to 1). Moreover, $P_{ij} = 0$ when the two nodes i and j are different and are not linked by an edge. Due to the stated assumptions, P can be interpreted as the matrix of transition probabilities associated with a finite-states Markov chain, possibly containing self-loops, since $P_{ii} \geq 0$ for all $i \in 1, \ldots, N$. If all the diagonal entries of P are positive and the weighted graph associated with P is connected, then it is well-known (see, e.g., Lovisari and Zampieri, 2012) that, for the ith component $x^{(i)}(t)$ of $\underline{x}(t)$, one has

$$x^{(i)}(t) \xrightarrow{t\to\infty} \frac{1}{N}\underline{1}^T\underline{x}(0), \quad \forall\, i = 1, \ldots, N, \tag{2}$$

with $\underline{x}(0)$ being the vector of the initial opinions of the agents. The expression $\Sigma = \frac{1}{N}\underline{1}^T\underline{x}(0)$, which is the average of the initial opinions of the agents, is the consensus state of the system.[1]

It is well-known (see, e.g., Como et al., 2012; Fagnani, 2014) that, at any discrete time instant t, the distance from the consensus state can be bounded from above as a function of the second-largest eigenvalue modulus $\mu(P)$ of the matrix P, in the following way:

$$\left\|\underline{x}(t) - \frac{1}{N}\underline{1}\underline{1}^T\underline{x}(0)\right\|_2^2 \leq \mu^{2t}(P)\|\underline{x}(0)\|_2^2, \tag{3}$$

where $\|\cdot\|_2$ denotes the l_2 norm.[2] For a given P, this rate of convergence cannot be improved, since there exist choices of the initial state $\underline{x}(0)$ for which a better rate cannot be obtained. Using (3), the rate of convergence to the consensus state was optimized in Boyd et al. (2004) by solving a suitable convex optimization problem, whose optimization variables are the entries of the matrix P. Differently from that approach, in the paper we intend to speed up consensus by considering local consensus subproblems formulated on different subgraphs $G_m = (V_m, E_m)$ of the original

[1] Since in the paper we are dealing with undirected graphs, hence with symmetric transition probability matrices, the consensus state is the average of the initial opinions of the agents. Without this assumption, the consensus state belongs only to the convex hull of the set of such opinions. To distinguish between these two situations, the consensus problem considered in this paper is sometimes called "average" consensus problem (Lovisari and Zampieri, 2012).

[2] The proof of (3) is as follows (see also Como et al., 2012). The matrix P has the eigendecomposition $P = \frac{1}{N}\underline{1}\underline{1}^T + \sum_{j=1}^{N-1}\lambda_j\underline{v}_j\underline{v}_j^T$, where the eigenvalues are 1 and, for $j = 1, \ldots, N-1$, λ_j (with $|\lambda_j| \leq \mu(P)$). The corresponding unit-norm and orthogonal eigenvectors are $\frac{1}{\sqrt{N}}\underline{1}$ and, for $j = 1, \ldots, N-1$, $\underline{v}_j$. Then, using also (1), one gets

$$\left\|\underline{x}(t) - \frac{1}{N}\underline{1}\underline{1}^T\underline{x}(0)\right\|_2^2 = \left\|\sum_{j=1}^{N-1}\lambda_j^t\underline{v}_j\underline{v}_j^T\underline{x}(0)\right\|_2^2 = \sum_{j=1}^{N-1}\left\|\lambda_j^t\underline{v}_j\underline{v}_j^T\underline{x}(0)\right\|_2^2 \leq \mu^{2t}\,\|\underline{x}(0)\|_2^2.$$

network, then by considering another consensus problem on an auxiliary graph $G_{aux} = (V_{aux}, E_{aux})$, whose nodes are selected nodes of the subgraphs, each one representative of the associated subgraph (hence, called "supernode" in the method investigated in the paper). In our approach, once the graph G is given, the choice of the matrix P is fixed and is determined following the procedure described in Garin et al. (2010) (see the Appendix for details). Such a procedure, indeed, is guaranteed to generate a doubly stochastic and symmetric matrix P for which (2) holds. The same procedure of construction is used for the doubly stochastic and symmetric matrices P_m and P_{aux} associated, respectively, with the generic subgraph G_m, and with the auxiliary graph G_{aux}.

The *hierarchical consensus method* investigated in the paper is made of the two following consecutive phases. In the first phase, one divides the original graph into many connected subgraphs G_m, each one evolving according to a state equation of the form (1). This phase can be easily parallelized. The subgraphs are generated in such a way to increase the rate of convergence to the local consensus state, with respect to the rate of convergence to the global convergence state of the original network G. The second phase consists in determining an auxiliary graph G_{aux} that connects, depending on the topology of the original graph, selected nodes of the subgraphs above. The consensus state determined on this auxiliary graph is the same, up to a certain tolerance, of the one determined in the original network G (details are provided in Section 6). In a similar way as in Boyd et al. (2004), we ground our analysis on the upper bound (3) about the distance from the local/global consensus state. Indeed, that formula is used to determine the rate of convergence to the global/local consensus state on each graph/subgraph considered, and on the auxiliary graph. In the last two cases, of course, the matrix P, the number of nodes N and the vectors $\underline{x}(0)$ and $\underline{x}(t)$ in (3) have to be replaced by the corresponding expressions valid for each subgraph G_m and for the auxiliary graph G_{aux}. In the remaining of the work, beside the already-introduced notation $\mu(P)$, we use the notations $\mu(P_m)$ and $\mu(P_{aux})$ to indicate, respectively, the second-largest eigenvalue moduli of the transition probability matrices P_m and P_{aux}. Sometimes, the short-hand notations μ, μ_m and μ_{aux} are used instead of $\mu(P)$, $\mu(P_m)$, and $\mu(P_{aux})$.

Concluding, we aim at exploiting (3) to find subgraphs with "good" spectral properties, i.e., with fast convergence rate to each local consensus state. A similar remark holds for the auxiliary graph. In more details, starting from the connected agents' network G, we aim at dividing it into different connected subgraphs G_m, with a smaller second-largest eigenvalue modulus $\mu(P_m)$ than the one $\mu(P)$ associated with the original network. In fact, the smaller $\mu(P_m)$, the faster the convergence rate to the local consensus state. In particular, on each subgraph we estimate the time needed by the agents involved in that subgraph to reach the local consensus state up to a certain tolerance, using the upper bound in (3). After each subgraph has reached a sufficiently good approximation of its local consensus state, an auxiliary network with a number of nodes equal to the number of subgraphs previously generated is created. To each node of the auxiliary graph, we associate an initial opinion proportional to the approximate local consensus state previously computed, inserting the number of nodes of the corresponding subgraph inside the proportionality factor. The consensus problem is now considered on the auxiliary graph, and the upper bound in (3) (stated in this case in terms of the second-largest eigenvalue modulus $\mu(P_{aux})$) is used now to evaluate the convergence rate to the consensus state of the auxiliary graph. The whole procedure is such that this is the same, up to a certain tolerance, as the consensus state of the original graph (see Section 6 for details).

It follows from spectral graph theory arguments that "good" spectral properties needed for a successful application of the hierarchical consensus method are expected in the case of "dense" graphs/subgraphs (see Section 3 for the details). Concerning the first phase, in Sections 5 and 5.1 we describe two clustering techniques useful to divide the graph into many "dense" subgraphs; the first one consists in applying a technique known as spectral clustering, which is briefly reviewed in Section 5, while the second one, called *nearest supernode approach*, is proposed specifically for the hierarchical consensus method. Before doing this, in the next section we provide a more technical motivation of such a method, using spectral graph theory arguments.

3. Spectral graph theory arguments supporting the hierarchical consensus method

Spectral graph theory (Chung, 1996) is concerned with the analysis of graphs in terms of spectral properties of associated matrices, such as the adjacency matrix and the Laplacian matrix (Chung, 1997). In particular, it studies the relation between graph properties and the spectrum of the normalized Laplacian matrix L_{norm}, defined as follows. Given a matrix $W \in \mathbb{R}^{N\times N}$ of non-negative weights (weight matrix), one defines at first the weighted degree $d_{W,i}$ of the generic node i of the graph G as $d_{W,i} = \sum_{j=1}^{N} W_{ij}$. The weight matrix W possibly contains self-loops, i.e., there could be indices i for which $W_{ii} > 0$. In the following, we assume $d_{W,i} > 0$, $\forall\, i = 1, \ldots, N$. Then, the elements of the matrix L_{norm} are defined as follows:

$$L_{\mathrm{norm},ij} = \begin{cases} 1 - \dfrac{W_{ii}}{d_i}, & \text{if } i = j, \\ -\dfrac{W_{ij}}{\sqrt{d_i d_j}}, & \text{if } i \neq j. \end{cases}$$

Equivalently, in terms of the diagonal weighted degree matrix D_W (whose diagonal elements are the $d_{W,i}$'s), one has

$$L_{\mathrm{norm},ij} = I - D_W^{-\frac{1}{2}} W D_W^{-\frac{1}{2}}. \tag{4}$$

A basic property of the matrix L_{norm} is that it is symmetric and positive-semidefinite, and its eigenvalues, ordered non-decreasingly, satisfy $0 \le \xi_1(L_{norm}) \le \xi_2(L_{norm}) \le \ldots \le \xi_N(L_{norm}) \le 2$. Moreover, the multiplicity of 0 as an eigenvalue of L_{norm} is equal to the number of connected components of the graph G. In the specific case of the consensus problem, one has $W{=}P$, and $D_W{=}D_P$ is the $N \times N$ identity matrix. Hence, (4) reduces to $L_{\mathrm{norm},ij} = I - P$, and the eigenvalues of the two matrices are related through

$$\xi_i(L_{norm}) = 1 - \lambda_{N-i}(P), \quad \text{for } i = 1, \ldots, N. \tag{5}$$

Hence, the second-largest eigenvalue modulus $\mu(P)$ of the matrix P can be expressed as

$$\mu(P) = \max\left\{ \left|1 - \xi_2(L_{norm})\right|, \left|1 - \xi_N(L_{norm})\right| \right\}. \tag{6}$$

In practice, one can often simplify formula (6), restricting the attention to the second-smallest eigenvalue $\xi_2(L_{norm})$ of the normalized Laplacian matrix L_{norm}. Indeed, when $\left|1 - \xi_N(L_{norm})\right| \ge \left|1 - \xi_2(L_{norm})\right|$ (which is always the case when $\xi_2(L_{norm}) \ge 1$), one can define a new transition probability matrix P', whose elements are related to those of P as follows (see also Chung, 1997, Section 1.5):

$$P'_{ij} = \begin{cases} 2P_{ii}, & \text{if } i = j, \\ \dfrac{P_{ij}}{2}, & \text{if } i \neq j. \end{cases}$$

For the associated normalized Laplacian matrix L'_{norm}, one obtains

$\xi_2(L'_{\text{norm}}) = \frac{\xi_2(L_{\text{norm}})}{2}$ and $\xi_N(L'_{\text{norm}}) = \frac{\xi_N(L_{\text{norm}})}{2}$, and finally, for the second-largest eigenvalue modulus $\mu(P')$ of the matrix P', one gets the expression

$$\mu(P') = \max\left\{\left|1 - \xi_2(L_{,\text{norm}})\right|, \left|1 - \xi_N(L_{,\text{norm}})\right|\right\} = 1 - \xi_2(L_{,\text{norm}}) = 1 - \frac{\xi_2(L_{\text{norm}})}{2}. \tag{7}$$

Summarizing, apart from a possible replacement of P with P', formulas (6) and (7), combined with formula (3), show that the rate of convergence to the consensus state increases when increasing the second-smallest eigenvalue $\xi_2(L_{\text{norm}})$ of the normalized Laplacian matrix. In the following, we report two basic results from spectral graph theory that provide insights about graphs/subgraphs for which $\xi_2(L_{\text{norm}})$ is large ("good" spectral properties) or it is small ("bad" spectral properties). As explained later, such insights are essential for the effectiveness of the hierarchical consensus method.

The first basic result from spectral graph theory needed in the following is Cheeger's inequality (Chung, 1996) (see the next formula (9)), which provides lower and upper bounds on $\xi_2(L_{\text{norm}})$. Given a subset of nodes S (with $\emptyset \neq S \neq V$) of the graph associated with a weight matrix W, and the complementary set of nodes $S' = V \backslash S$, one denotes the sum of the weights of all the edges joining nodes in S with nodes in S' by $W(S, S') = \sum_{i \in S, j \in S'} W_{ij}$. Moreover, one defines the volumes of S and S' as $\text{vol}_W(S) = \sum_{i \in S} d_{W,i}$, $\text{vol}_W(S') = \sum_{j \in S'} d_{W,j}$ (in the specific case $W=P$, one has $\text{vol}_P(S) = |S|$ and $\text{vol}_P(S') = |S'|$). Then, Cheeger's constant is defined as

$$\Phi(W) = \min_{S \neq \emptyset, V} \frac{W(S, S')}{\min\left\{\text{vol}_W(S), \text{vol}_W(S')\right\}}. \tag{8}$$

Finally, Cheeger's inequality[3] states the following:

$$\frac{\Phi^2(W)}{2} \leq \xi_2(L_{\text{norm}}) \leq 2\Phi(W). \tag{9}$$

Cheeger's inequality allows one to identify easily some kinds of graphs for which $\xi_2(L_{\text{norm}})$ is small. This happens, e.g., when the graph G is made of several "clusters" of nodes (i.e., subsets of nodes for which the sum of the weights of the edges connecting nodes in the same subset is large, but the sum of the weights of the edges connecting nodes in different subsets is small), and these clusters have comparable and sufficiently large volumes. Indeed, in this case, choosing the set S of nodes to be equal to one of the clusters, one gets a small value of $W(S, S')$, whereas $\min\left\{\text{vol}_W(S), \text{vol}_W(S')\right\}$ is large. Hence, for this kind of graph, Cheeger's constant $\Phi(W)$ is small, and $\xi_2(L_{\text{norm}})$ is small, due to (9). It is worth noting that this argument does not apply when there is only one such cluster.

We now introduce the second basic result from spectral graph theory, which is useful for the motivation of the hierarchical consensus method. To do this, one defines at first the diameter $\text{diam}(G)$ of the graph G associated with the weight matrix W as the maximum over the lengths of all the shortest paths between any pair of nodes, where the length of each path is defined as the sum of all its weights. Then, the second basic result is the following lower bound on $\xi_2(L_{\text{norm}})$ (Chung, 1997, Chapter 1)[4]:

[3] In Chung (1996), Cheeger's inequality is stated and proved at first for the case of unweighted graphs without self-loops, then it is extended to the present case of weighted graphs, possibly containing also self-loops.

[4] Chung (1997, Chapter 1) provides the proof of the bound (10) for the case of unweighted graphs without self-loops, but the proof technique and the result extend directly to the case of weighted graphs, possibly with self-loops.

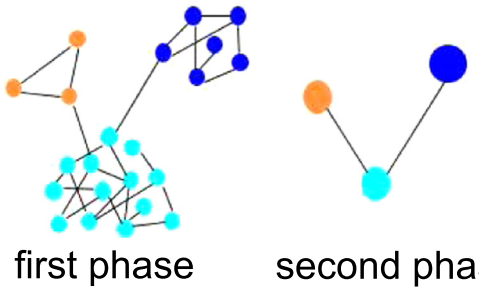

Fig. 1. For a simple example: the subgraphs and the auxiliary graph determined, respectively, in the first phase and in the second phase of the hierarchical consensus method investigated in the paper. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

$$\xi_2(L_{\text{norm}}) \geq \frac{1}{\text{diam}_W(G)\text{vol}_W(G)}. \tag{10}$$

From (10), one can infer that, $\text{vol}_W(G)$ being the same, $\xi_2(L_{\text{norm}})$ has a larger lower bound when G is "dense", i.e., it is a single "cluster" of nodes, for which $\text{diam}_W(G)$ is small.

Concluding, Cheeger's inequality (9) and the bound (10), combined with formulas (3), (6) and (7), show that, in order to achieve a fast convergence to the global/local consensus state, one should avoid, e.g., situations in which the graph/subgraph is made of two or more clusters "poorly" connected (i.e., such that the sum of the weights of the edges connecting them is small), since in this case, the second-smallest eigenvalue of the Laplacian matrix (hence the rate of convergence to the global/local consensus state) is small. Conversely, a single cluster is more effective. The same kind of considerations holds for the auxiliary graph.

Fig. 1 illustrates, by a simple example, how these ideas from spectral graph theory can be applied to the hierarchical consensus method investigated in this paper. In the situation described in the figure, the convergence to the consensus state of the original graph (the one in the upper left corner) is slow. In fact, the information related to the different opinions flows slowly, e.g., from the group of agents in light blue and either the group of agents in orange or the one in dark blue, since these groups are poorly connected to the former (indeed, there is only one edge among the dark blue/orange agents and the light blue ones). However, extracting a set of denser subgraphs may lead to a faster convergence rate to the local consensus state in each subgraph, since in that case the information would flow faster than in the original graph (for the example considered, the ideal case would be associated with the extraction of the three subgraphs shown with different colors in Fig. 1). Then, in the specific case, the second phase deals with a small and sufficiently dense auxiliary graph, for which the rate of convergence to the associated consensus state is fast.

4. Clustering techniques used by the hierarchical consensus method

In this section, we describe two techniques (i.e., *spectral clustering* and the *nearest supernode approach*) that we exploit to identify clusters of nodes, a problem whose importance for the first phase of the hierarchical consensus method has been illustrated in Section 3. We introduce the following notation. We indicate with M the number of subgraphs we intend to generate in the first phase, using either spectral clustering or the nearest supernode approach in the first phase of the method. Thus, from the original network G, we determine M different subgraphs G_m with $m = 1, \ldots, M$. As already mentioned, the goal of the hierarchical consensus method is to generate the subgraphs G_m in such a way

that the rate of convergence to the local consensus state inside each subgraph is faster than the one to the global consensus state in the original graph G. It is worth remarking that, by construction, also the auxiliary graph G_{aux} is expected to be dense. Indeed, due to (10), its volume is equal to the number of subgraphs (hence, it is small when this number is small), whereas its diameter is expected to be small, due to the rule used for the construction of its edges (which is detailed in Section 6).

The choice of the two clustering techniques described in the following subsections reflects this goal, since they are expected to generate dense subgraphs/auxiliary graph, and is motivated as follows. In more details, spectral clustering has been chosen because of its connection with spectral graph theory, which motivates the hierarchical consensus method itself, as shown in the previous section. More precisely, the optimization problem solved by spectral clustering (see, e.g., von Luxburg, 2007, formula (11)) is a relaxation of another optimization problem (called *Ncut* minimization, see, e.g., von Luxburg, 2007, formula (7)), which is strongly related to properties of the normalized Laplacian matrix of the graph, and is formulated in terms of quantities appearing, e.g., in Cheeger's inequality (9). However, spectral clustering presents also some drawbacks in the case of a "large" graph, since it requires the knowledge of the whole graph, and solving the associated optimization problem becomes increasingly difficult as the size of the graph grows. Hence, the (less computationally expensive and more distributed) nearest supernode approach has been adopted in the paper as a possible way to solve these negative issues.

5. Spectral clustering

Spectral clustering (see von Luxburg, 2007 for a tutorial) is a method able to determine clusters inside a graph, by exploiting the eigenvalues and eigenvectors of the normalized Laplacian matrix L_{norm} of the graph G.[5] As shown in von Luxburg (2007), if a connected component has a structure with k "apparent" clusters, the first k eigenvalues of the normalized Laplacian matrix are close to 0, while starting from the $(k+1)$th eigenvalue, their values are in general significantly larger than 0.[6] Without going into details, the algorithm works as follows. Starting from a weight matrix W and the number k of clusters that are supposed to be present in the graph, one computes orthogonal eigenvectors $\underline{u}_1, \ldots, \underline{u}_k$ associated with the first k eigenvalues $\xi_1(L_{norm}), \ldots, \xi_k(L_{norm})$ of the normalized Laplacian matrix. Then, one subsequently builds a matrix $U \in \mathbb{R}^{N\times k}$ containing the vectors $\underline{u}_1, \ldots, \underline{u}_k$ as columns and, for each $i = 1, \ldots, N$, associates the ith node of the graph with the i-th row of U. Finally, the points so-obtained are clustered through the k-means clustering algorithm, which forms the last part of the method.

Spectral clustering has found several applications in engineering (Frias-Martinez and Frias-Martinez, 2014; Langone et al., 2015). For its specific application to the hierarchical consensus method, spectral clustering requires ones to fix the number of clusters to be detected, i.e., the number M of subgraphs to be generated in the first phase of the hierarchical clustering method. If one has some prior knowledge about the topology of the original graph, one can provide to the algorithm the information about the number of clusters that are expected to exist, and choose this as the number M of subgraphs to be generated. For our investigation, we intend to apply spectral clustering to the network of agents by evaluating the effects of different numbers of subgraphs we require the method to generate. In fact, in Secton 7, we test the method on graphs with and without a clear cluster structure; thus, we try different options for M, in order to find the ones that lead to the best results.

5.1. Nearest supernode approach

Besides spectral clustering, in the paper we consider also another clustering technique, which is more distributed and less expensive from a computational point of view. Indeed, especially for the case of networks with a huge number of agents, the application of spectral clustering can be difficult, since this technique requires the computation of selected eigenvectors of the normalized Laplacian matrix, whose number of elements potentially grows quadratically with the number of agents. Hence, here we propose a second clustering method, which we call *nearest supernode approach*, potentially able to overcome this issue, and to produce results comparable with the ones achieved by spectral clustering (a numerical comparison of the two methods, confirming this expectation, is reported later in Section 7).

The following rules are used by the proposed nearest supernode approach to determine suitable subgraphs G_m of the original connected graph G:

(a) starting from G, one fixes the number M of subgraphs to be generated;
(b) M nodes, called *supernodes*, are generated; they are used to create the subgraphs. These supernodes can be generated according to one of the four following procedures: either they are randomly sampled (without repetition) from the set of nodes V, or they are selected according to their ranking with respect to one of the three following node centrality measures. More precisely, one can consider the first M nodes with the highest degree, with the highest betweenness centrality, or with the highest clustering coefficient (see Newman, 2010 for a detailed description of these node centrality measures);
(c) the set of nodes V is partitioned into two disjoint subsets SN and ON, where SN is the set of supernodes determined above, while $ON = V \backslash SN$ is the set containing all the other nodes of the graph G;
(d) each node $i \in ON$ is assigned to one or more supernodes, according to the following procedure. First, shortest paths (SPs) (based on the unweighted adjacency matrix of the graph G) between i and all the supernodes are determined. If i has length(SP) = 1 to only one supernode, then it is assigned to that supernode. Otherwise, if different supernodes with length(SP) = 1 to i exist, then i is randomly associated with one of such nearest supernodes. If no supernodes with length(SP) = 1 to i exists, then i is either assigned to the nearest supernode with length(SP) > 1, if only one such nearest supernode exists, or it is randomly assigned to one of the nearest supernodes with length(SP) > 1, if more than one such supernode exist. In this situation, all the nodes belonging to the selected shortest path are associated with the selected nearest supernode.

It is worth remarking that the procedure presented above assigns all the nodes belonging to the set ON to at least one supernode, since the graph G is connected. Moreover, there are no restrictions on the number of nodes assigned to the subgraph associated with a generic supernode. Thus, subgraphs with possibly

[5] There exists also one version of spectral clustering that uses the unnormalized Laplacian matrix (von Luxburg, 2007), but the focus of the paper is on the normalized case, due to the results from spectral graph theory presented in Section 3.

[6] This result can be proved, e.g., using matrix perturbation techniques, starting from the "ideal case" in which the clusters are disconnected, and form k distinct connected components of the graph (in this case, the eigenvalue 0 of the normalized Laplacian matrix has multiplicity exactly equal to k).

different numbers of nodes are generated; in general, supernodes with high centrality are expected to generate subgraphs with larger numbers of nodes than supernodes with small centrality. Finally, the procedure just described prevents the subgraphs generated to be disconnected. This depends on the fact that it allows some nodes to be possibly shared by different subgraphs.

6. Consensus on the subgraphs and on the auxiliary graph

For each connected graph G, one can apply either clustering method described in Section 4 to extract the M subgraphs G_m. Once the subgraphs have been generated, each subgraph evolves independently according to formula (1) (with obvious changes in notation, to adapt it to that subgraph[7]). This phase requires a number of iterations of (1) sufficiently large to allow all the subgraphs to approximate the local consensus state within a desired accuracy. This number of steps is approximately equal to the one required by the subgraph $G_{\hat{m}}$ with the largest value $\mu(P_{\hat{m}})$ among the second-largest eigenvalue moduli $\mu(P_m)$ of all the subgraphs, because such subgraph is the one with the smallest rate of convergence to the local consensus state (see formula (3)). After the subgraph $G_{\hat{m}}$ has reached the desired approximation of its local consensus state, the second phase of the hierarchical consensus method starts, by determining an auxiliary graph G_{aux} with a number of nodes equal to M (one node for each subgraph). The nodes in the auxiliary graph are connected depending on the edges of the original graph. More precisely, two nodes i and j of G_{aux} are connected by an edge in G_{aux} if and only if at least two nodes of G belonging to the subgraphs associated, respectively, with i and with j, are connected by an edge in G. Once the auxiliary graph is built, it also evolves according to formula (1) (again, with obvious changes in notation, to adapt it to the auxiliary graph), and an approximation of its consensus state is determined, after some number of iterations. With a proper initialization of the states of the nodes belonging to the auxiliary graph (see Section 6.2), this is also an approximation, up to a desired tolerance, of the global consensus state of the original graph G.

It is worth remarking that when the nearest supernode approach is used in the first phase of the hierarchical consensus method, all the subgraphs G_m and the auxiliary graph G_{aux} are guaranteed to be connected. Spectral clustering, instead, does not provide such a guarantee. Nevertheless, when spectral clustering is used, preliminary experiments have shown that the subgraphs G_m and the auxiliary graph G_{aux} are usually connected, at least for small choices of M.

In the next subsections, we provide a detailed analysis of conditions under which, the desired approximation of the global consensus state of G being the same, the hierarchical consensys method requires a total number of iterations smaller than the one required by the direct evaluation of formula (1) on the original graph G. Hereafter, this second procedure is called *non-hierarchical consensus method*. Due to the discussion above, we assume in the analysis that all the graphs/subgraphs are connected.[8]

6.1. Approximation of the global consensus state through the hierarchical consensus method

Since the graphs and subgraphs involved in the hierarchical consensus method have in general different numbers of nodes, it is useful to consider in the analysis the l_∞-norm rather than the l_2-norm of the vectors of initial opinions. In this way, the elements of such vectors are more easily comparable. Moreover, as it is shown later in Section 6.3, using the l_∞-norm also allows one to translate some upper bounds valid for the graph G to upper bounds valid for G_m and G_{aux}. In order to perform the approximation error analysis using the l_∞-norm, we recall that, given a generic vector $\underline{z} \in \mathbb{R}^N$, its l_2-norm and l_∞-norm are related by the following inequalities: $\| \underline{z} \|_\infty \le \| \underline{z} \|_2 \le \sqrt{N} \| \underline{z} \|_\infty$. This, combined with the bound (3), provides

$$\left\| \underline{x}(t) - \frac{1}{N} \underline{1}\underline{1}^T \underline{x}(0) \right\|_\infty^2 \le \mu^{2t}(P) |V| \| \underline{x}(0) \|_\infty^2, \tag{11}$$

which is the main tool used for the next approximation error analysis (with obvious changes in notations, similar bounds hold for each G_m, and for G_{aux}).

The time needed to reach a desired accuracy in the approximation of the consensus state of the graph G through both the non-hierarchical and the hierarchical consensus methods is estimated in the following way:

(a) A tolerance $\varepsilon > 0$ is fixed; this tolerance represents, for both methods, the desired accuracy in the approximation, in the l_∞-norm, of the consensus state of the original graph G.

(b) The number of iterations needed by formula (1) applied to the original graph G to reach the consensus state up to the tolerance ε is bounded from above by choosing the smallest value T of t for which

$$\mu^{2t}(P) |V| \| \underline{x}(0) \|_\infty^2 \le \varepsilon^2 \tag{12}$$

(see formula (11)). Note that, to compute T, here we are assuming that $\| \underline{x}(0) \|_\infty$ is known, but we are not assuming the values of every single element of the vector $\underline{x}(0)$ to be also known. This assumption could be relaxed replacing $\| \underline{x}(0) \|_\infty$ in (12) with an upper bound. However, as shown later in Section 6.3, this relaxation is not essential for the analysis.

(c) A similar kind of analysis is applied to every single subgraph, evaluating an upper bound $t_{1^\circ\text{phase}}$ on the number of iterations needed to reach the local consensus state, in this case up to the tolerance $\frac{\varepsilon}{2}$, still with respect to the l_∞-norm. In fact, since the hierarchical consensus method is made of two consecutive phases (the first one with each subgraph evolving independently according to formula (1), and the second one involving the auxiliary graph), and since the l_∞-norm is used in the analysis, one can fix a tolerance equal to $\frac{\varepsilon}{2}$ for each of the two phases of the method, in order to achieve the desired accuracy ε in the approximation of the global consensus state of the original graph G. Without a significant loss of generality, as it is detailed later in Section 6.3, the upper bound $t_{1^\circ\text{phase}}$ can be computed considering only the behavior of the subgraph with the largest μ_m.

(d) At time $t_{1^\circ\text{phase}}$, the auxiliary graph is considered, then the matrix P_{aux} and the corresponding second-largest eigenvalue modulus μ_{aux} are determined, and an upper bound $t_{2^\circ\text{phase}}$ on the number of iterations needed to reach the accuracy $\frac{\varepsilon}{2}$ in the approximation of its consensus state is computed similarly to items (b) and (c), still with respect to the l_∞-norm. The vector of initial opinions of the agents associated with the nodes of the auxiliary graph is constructed in such a way that the consensus state of such a graph approximates the global consensus state of the graph G within the accuracy $\frac{\varepsilon}{2}$ (see Section 6.2 for the precise construction of such a vector of

[7] The possible presence of nodes in common among different subgraphs does not create problems for the independent evolution of each subgraph according to the corresponding form of formula (1), since one can associate with each subgraph an independent copy of each shared node.

[8] There is no loss in generality in assuming this because, due to formula (5), for a disconnected graph/subgraph, the second-largest eigenvalue modulus of the associated transition probability matrix is equal to 1, which corresponds to a rate of convergence to the global/local consensus state equal to 0 (i.e., no convergence in general to such state).

initial opinions).

Summarizing, the time needed by the first phase of the hierarchical consensus method to terminate is equal to $t_{1^\circ\text{phase}}$, and depends mainly on the expression $\mu_{\max}:=\max_{m=1,\dots,M}\{\mu_m\}$, whereas the time needed by the second phase to terminate is equal to $t_{2^\circ\text{phase}}$, and depends mainly on μ_{aux}. It follows that the hierarchical consensus method has a better performance guarantee than the non-hierarchical consensus method if the following condition is met:

$$t_{1^\circ\text{phase}} + t_{2^\circ\text{phase}} < T. \tag{13}$$

More details about this comparison are provided in Section 6.3.

6.2. Construction of the vectors of initial opinions, and asymptotic analysis

In this subsection, we aim at studying the solution computed by the hierarchical consensus method when the numbers of iterations of both its phases are sufficiently large (ideally, when both $t_{1^\circ\text{phase}}$ and $t_{2^\circ\text{phase}}$ tend to infinity, or equivalently, when the tolerance ε tends to 0), to verify that it can really provide a good approximation of the global consensus state of the graph G, if the initial opinions of the nodes belonging to the subgraphs and to the auxiliary graph are chosen properly.

Since the first phase of the hierarchical consensus method allows for the presence of overlaps of nodes, i.e., it may happen that the same node is shared by different subgraphs, it is important to deal with such shared nodes properly (this issue is present only if the nearest supernode approach is applied in the first phase, since there are no shared nodes when the spectral clustering is applied). In the following, we suppose that when evolving each subgraph, one knows which nodes are shared with other subgraphs, and the number of such node-sharing subgraphs (this is a mild assumption, since this information could be provided by the agents associated with the nodes). Denoting by $N_m = |V_m|$ the number of nodes of each subgraph G_m, we define the vectors $\underline{x}_m(0) \in \mathbb{R}^{N_m}$ ($m = 1, \dots, M$) of initial opinions of the agents belonging to the subgraphs in the following way. If a component of $\underline{x}_m(0)$ (say the p-th component, denoted by $x_m^{(p)}(0)$) refers to a node of G which is not shared with other subgraphs (say the node q), then we set

$$x_m^{(p)}(0) = x^{(q)}(0), \tag{14}$$

where the right-hand side refers to the q-th component of $\underline{x}(0)$. It the node p is shared, say, by M_p subgraphs, then we set

$$x_m^{(p)}(0) = \frac{x^{(q)}(0)}{M_p}. \tag{15}$$

In this way, the opinions of the agents associated with nodes shared by various subgraphs are rescaled. Now, at the end of the first phase, when $t_{1^\circ\text{phase}}$ is sufficiently large, one has, for all $m = 1, \dots, M$, and for each ith component of $\underline{x}_m(t_{1^\circ\text{phase}})$,

$$x_m^{(i)}(t_{1^\circ\text{phase}}) \simeq \frac{1}{|V_m|}\underline{1}_m^T \underline{x}_m(0), \tag{16}$$

where $\underline{1}_m$ denotes a vector of all 1 s, of the same dimension $|V_m|$ as $\underline{x}_m(0)$, and formula (16) holds since (2) can be applied in the analysis. Hence, the local consensus state of each subgraph is equal to the average of the initial opinions of the agents associated with that subgraph, possibly rescaling the values of the opinions of the agents associated with nodes shared by different subgraphs.[9] Without loss of generality, in the following we assume that the supernodes are associated with $i=1$ in formula (16).

At this point, at the beginning of the second phase of the hierarchical consensus method (i.e., at time $t = t_{1^\circ\text{phase}}$), we define the vector $\underline{x}_{\text{aux}}(t_{1^\circ\text{phase}}) \in \mathbb{R}^M$ of initial opinions of the agents associated with the nodes of the auxiliary graph as follows:

$$\underline{x}_{\text{aux}}(t_{1^\circ\text{phase}}) = \left[|V_1|\frac{M}{N}x_1^{(1)}(t_{1^\circ\text{phase}}), \dots, |V_M|\frac{M}{N}x_M^{(1)}(t_{1^\circ\text{phase}}) \right]^T, \tag{17}$$

i.e., the opinion of the agent associated with the mth supernode is rescaled by the factor $|V_m|\frac{M}{N}$. Finally, by a similar analysis, the consensus state of the auxiliary graph (which is achieved within an arbitrary accuracy if $t_{2^\circ\text{phase}}$ is sufficiently large) is the average of such opinions, and is equal to $\frac{\sum_{m=1}^M |V_m|\frac{M}{N}x_m^{(1)}(t_{1^\circ\text{phase}})}{M} \simeq \frac{\sum_{m=1}^M |V_m|\frac{M}{N}\frac{1}{|V_m|}\underline{1}_m^T\underline{x}_m(0)}{M} = \frac{1}{N}\underline{1}^T\underline{x}(0)$, where the last expression is just the desired global consensus state of the original graph G.

6.3. Non-asymptotic performance analysis

In this subsection, we provide a non-asymptotic performance analysis of the hierarchical consensus method (i.e., considering a finite number of iterations), comparing it with the non-hierarchical consensus method, and expressing condition (13) in terms of spectral properties of the graphs and subgraphs involved in the method. A similar analysis was made in Epstein et al. (2008) for an analogous hierarchical consensus method developed therein, but in that case, no spectral graph arguments like the ones provided in Section 3 of this work were presented as a motivation.

In the following, we investigate the number of iterations needed by the hierarchical consensus method to reach an approximation of the global consensus state up to the tolerance $\varepsilon > 0$. The following discussion refers to any among the graphs G, G_m, and G_{aux}, although we exemplify it at first by considering the graph G. Using (11), the minimal number of iterations of formula (1) that guarantees an approximation of the consensus state up to the desired tolerance $\varepsilon > 0$ is equal to

$$T = \max\left\{0, \frac{\log\left(\frac{\varepsilon^2}{|V|\|\underline{x}(0)\|_\infty^2}\right)}{\log\left(\mu^2\right)}\right\}. \tag{18}$$

Since $\mu < 1$, one gets $\log\left(\mu^2\right) < 0$, while $\log\left(\frac{\varepsilon^2}{|V|\|\underline{x}(0)\|_\infty^2}\right)$ could either be positive or negative. In particular, its numerator is positive when $\varepsilon > \sqrt{|V|}\,\|\underline{x}(0)\|_\infty$, from which it follows $\frac{\log\left(\frac{\varepsilon^2}{|V|\|\underline{x}(0)\|_\infty}\right)}{\log\left(\mu^2\right)} < 0$ and $T=0$. Moreover, for the case of a sufficiently small value of ε, one has $\varepsilon < \sqrt{|V|}\,\|\underline{x}(0)\|_\infty$ (hence, a positive value for T), and (18) becomes

$$T = \frac{\log\left(\frac{\varepsilon^2}{|V|\|\underline{x}(0)\|_\infty^2}\right)}{\log\left(\mu^2\right)}. \tag{19}$$

A similar kind of bound holds, with obvious changes in notations, for the subgraphs G_m and for the auxiliary graph G_{aux}. In the following, we always assume that the associated ε is sufficiently

[9] Thus, the local consensus state of a subgraph that shares some nodes with other subgraphs may be different from its local consensus state in case of no shared nodes.

(*footnote continued*)

small, in such a way that simplifications like (19) can be made. In particular, for the first phase of the hierarchical consensus method, one gets

$$t_{1^\circ\text{phase}} \leq \max_{m=1,\ldots,M} \left\{ \frac{\log\left(\frac{\varepsilon^2}{4|V_m| \left\| \underline{x}_m(0) \right\|_\infty^2} \right)}{\log\left(\mu_m^2 \right)} \right\}, \tag{20}$$

and, for its second phase,

$$t_{2^\circ\text{phase}} \leq \frac{\log\left(\frac{\varepsilon^2}{4|V_{\text{aux}}| \left\| \underline{x}_{\text{aux}}(t_{1^\circ\text{phase}}) \right\|_\infty^2} \right)}{\log\left(\mu_{\text{aux}}^2 \right)}. \tag{21}$$

At this point, we observe that an advantage of using the l_∞-norm in the analysis (with respect, e.g., to the l_2-norm) is that since the state vector in (1) is a convex combination of the opinions of the agents associated with the nodes of the graph, one gets

$$\| \underline{x}(t) \|_\infty \leq \| \underline{x}(0) \|_\infty, \quad \forall\, t = 1, 2, \ldots. \tag{22}$$

This, combined with the (definitions (14), 15), and (17), provides also the following upper bounds:

$$\| \underline{x}_m(0) \|_\infty \leq \| \underline{x}(0) \|_\infty, \quad \forall\, m = 1, \ldots, M, \tag{23}$$

$$\| \underline{x}_{\text{aux}}(t_{1^\circ\text{phase}}) \|_\infty \leq \max_{m=1,\ldots,M} \left\{ |V_m| \right\} \frac{M}{N} \| \underline{x}(0) \|_\infty, \tag{24}$$

which allow one to bound from above $t_{1^\circ\text{phase}}$ and $t_{2^\circ\text{phase}}$ in terms of $\| \underline{x}(0) \|_\infty$ as follows:

$$t_{1^\circ\text{phase}} \leq \max_{m=1,\ldots,M} \left\{ \frac{\log\left(\frac{\varepsilon^2}{4|V_m| \left\| \underline{x}(0) \right\|_\infty^2} \right)}{\log\left(\mu_m^2 \right)} \right\}, \tag{25}$$

$$t_{2^\circ\text{phase}} \leq \frac{\log\left(\frac{\varepsilon^2}{4|V_{\text{aux}}| \left(\max_{m=1,\ldots,M} \left\{ |V_m| \right\} \frac{M}{N} \right)^2 \left\| \underline{x}(0) \right\|_\infty^2} \right)}{\log\left(\mu_{\text{aux}}^2 \right)}. \tag{26}$$

For ε sufficiently small (in particular, smaller than 1, in such a way that $\log(\varepsilon)$ is negative), the right-hand sides of formulas (19), (25) and (26) are dominated, respectively, by the terms $\frac{\log(\varepsilon)}{\log(\mu)}$, $\frac{\log(\varepsilon)}{\log(\mu_{\max})}$, and $\frac{\log(\varepsilon)}{\log(\mu_{\text{aux}})}$. Then, in this situation, recalling formula (13), the hierarchical consensus method has a better performance guarantee than the non-hierarchical consensus method when the following condition holds:

$$\frac{1}{\log(\mu_{\max})} + \frac{1}{\log(\mu_{\text{aux}})} > \frac{1}{\log(\mu)}, \tag{27}$$

(here, one can notice that all the ratios involved are negative).

Concluding, at least for ε sufficiently small, (27) shows that the hierarchical consensus method is associated with a better performance guarantee than the non-hierarchical one when all the subgraphs G_m ($m = 1, \ldots, M$) and the graph G_{aux} have better spectral properties than G. As already mentioned, Section 3 motivates the use of clustering algorithms to make such subgraphs and the auxiliary graph has such good spectral properties.

7. Numerical results

In this section, the hierarchical consensus method is applied to various kinds of connected graphs G, and compared with the non-hierarchical consensus method. We report several numerical examples obtained by performing the first phase of the hierarchical consensus method applying both spectral clustering and the nearest supernode approach, with the aim of comparing numerically the two clustering methods used in the first phase. For both cases, the numerical results are evaluated by considering various choices for the number M of subgraphs extracted from the original graph G. The procedure is tested on various kinds of random graphs. Specifically, we consider, as graph models, the random geometric graph (Bollobás, 1998), the planted partition model (Mossel et al., 2015), and the preferential attachment model (Barabási and Albert, 1999). This last model generates random scale-free networks, such as the Internet, the World Wide Web, citation networks, and several real-world social networks. Hence, our goal is to compare the hierarchical and non-hierarchical consensus methods on different kinds of graphs modeling real-world networks. When applying the nearest supernode clustering method, we first fix the number M of subgraphs, then 10 tests are run for each situation studied. In fact, in the process of generating the subgraphs via that clustering method, the nodes in ON are usually assigned randomly to one of the nearest supernodes, unless there is only one such supernode. For the nearest supernode clustering method, the final results reported later in this section are empirical means and standard deviations over the 10 tests. In this way, a better comparison is obtained between the two clustering methods. For every kind of graph considered in the numerical comparison, the vector $\underline{x}(0)$ of the initial opinions of the N agents is generated as the realization of a random vector, where each component is drawn i.i.d. from the standard uniform distribution on the interval $(0, 1)$. The l_∞-norm of this vector is then used to determine the minimal number of steps of the non-hierarchical consensus method that guarantees to reach the global consensus state up to the fixed tolerance $\varepsilon > 0$ (see formula (19)). It is worth noting that, with this choice of the vector $\underline{x}(0)$, one can bound from above its l_∞-norm by the value 1, without knowing the specific realizations of its components. Formulas (25) and (26) are then used for the two phases of the hierarchical consensus method. Finally, we consider values of ε sufficiently small in order to neglect the dependence of formulas (19), (25), and (26) on the number of nodes of the subgraphs/graphs considered, and to assume that the slowest subgraph in the first phase of the consensus method is the one associated with $\mu_{\max}$.

7.1. Random geometric graph

To generate this kind of graph, N points are sampled from a 3-dimensional Gaussian distribution with mean $(0, 0, 0)$ and covariance matrix $3I \in \mathbb{R}^{3\times 3}$. A threshold is then applied on the Euclidean distance between every pair of points, connecting the two points of the pair via an edge of the graph, when the distance is smaller than the threshold. Two realizations of random geometric graphs with different numbers of nodes are considered, the first one with $N=100$ nodes, and the second one with $N=300$ nodes. The adjacency matrices of the two realizations are shown in Fig. 2. In both cases, we fix a tolerance ε equal to 10^{-6}.

When the realization of the random geometric graph with $N=100$ nodes shown in Fig. 2 (a) is considered, the second-largest

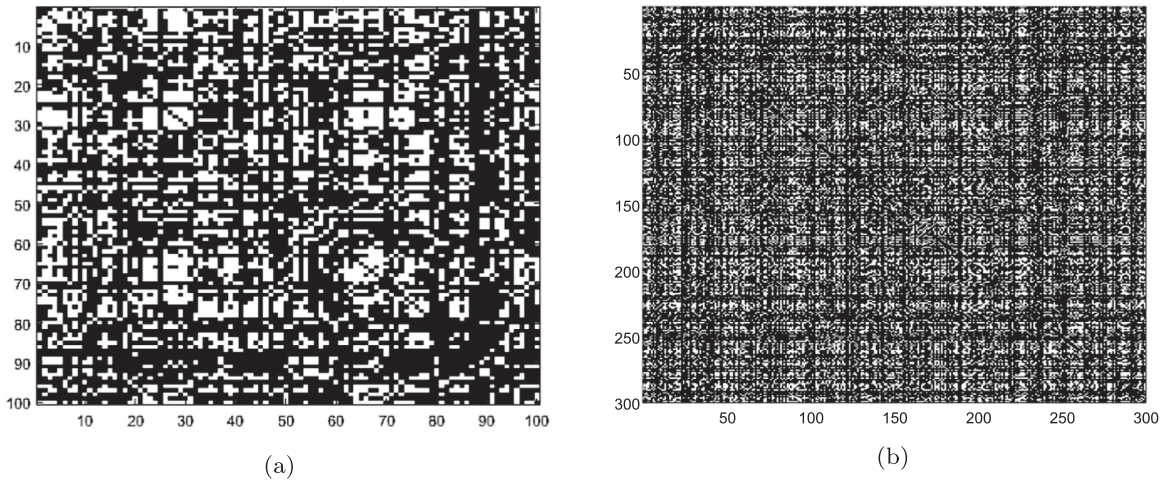

Fig. 2. Adjacency matrices of two realizations of a random geometric graph: (a) with 100 nodes; (b) with 300 nodes.

eigenvalue modulus of the transition probability matrix P associated with the graph G is equal to $\mu = 0.97$, while the number of steps required by the non-hierarchical consensus method to guarantee the tolerance ε is $T=458$. The first phase of the hierarchical consensus method is performed by considering both spectral clustering and the nearest supernode approach. In more details, when spectral clustering is applied, we require the hierarchical consensus method to extract a number of clusters $M \in \{10, 5, 2, 1\}$, while for the nearest supernode approach we choose $M \in \{20, 10, 5, 2, 1\}$. We do not require spectral clustering to determine 20 clusters, because in that case it could create disconnected subgraphs. Clearly, for both clustering methods, the result obtained for $M=1$ is the one achieved by the non-hierarchical consensus method. In the figures, we report also that result, in order to have a better comparison between the performances of the hierarchical and non-hierarchical consensus methods.

Fig. 3 reports the upper bound on $t_{1^\circ\text{phase}} + t_{2^\circ\text{phase}}$ derived from formulas (25) and (26), by varying the number M of subgraphs considered. For $M=1$, formula (19) is applied. On the left, the figure shows the time needed by the hierarchical consensus method when spectral clustering is applied during its first phase, while on the right, the results obtained by the nearest supernode approach are reported. For the spectral clustering, we report on the x-axis the average dimension of the subgraphs G_m, which is equal in this case to $h = \frac{N}{M}$, since no overlaps of nodes are allowed by this clustering method. Since both clustering methods require as an input the number of clusters M one wants to detect, and not their average dimension, for comparison purposes, we report h on the x-axis also for the nearest supernode approach. In this case, since overlaps of nodes are allowed among the different subgraphs (due to the procedure followed for their construction), the average dimension of the subgraphs G_m can be larger than h, although this number can be still considered as an approximate average number of nodes per subgraph. Of course, the nearest supernode approach can still create subgraphs even with a number of nodes smaller than h. The plot on the right shows the empirical mean and standard deviation, over the 10 tests, of the number of steps required by the hierarchical consensus method to guarantee the desired accuracy, when the nearest supernode approach is used in its first phase. The four types of seeds described in Section 5.1 are considered in the plot.

Concerning the random geometric graph with $N=300$ nodes, whose adjacency matrix is the one reported on the right in Fig. 2, one obtains $\mu = 0.995$. The number of steps required by the hierarchical consensus method to reach the global consensus state up

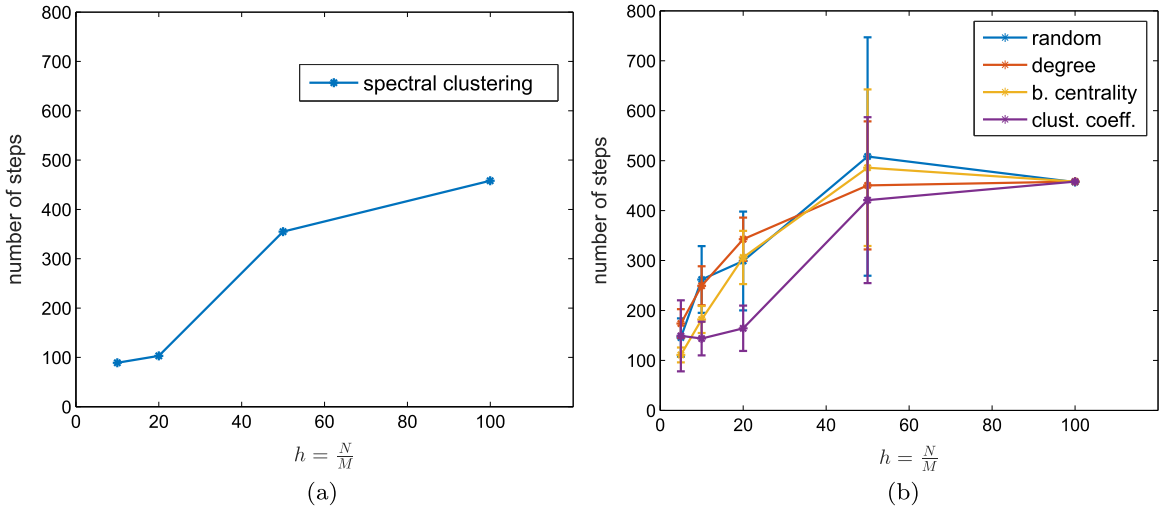

Fig. 3. Number of steps required to guarantee the desired accuracy $\varepsilon = 10^{-6}$ in the approximation of the global consensus state via the hierarchical/non-hierarchical consensus methods, for a realization of a random geometric graph with 100 nodes. In (a): spectral clustering is used during the first phase of the hierarchical consensus method. In (b): the first phase of the hierarchical consensus method is performed by applying the nearest supernode approach, selecting the supernodes according to the four different rules described in Section 5.1. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

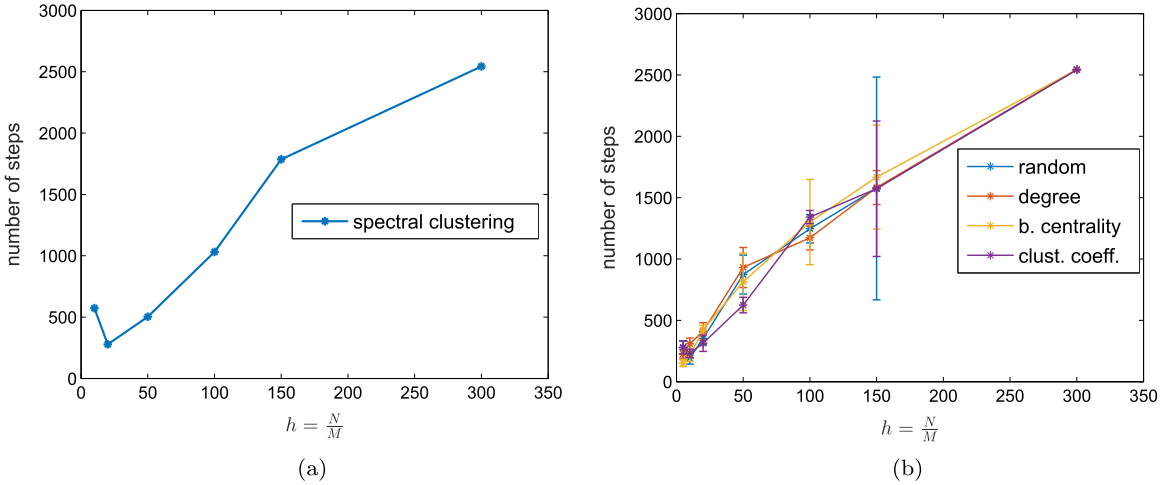

Fig. 4. Similar to Fig. 3, but for a realization of a random geometric graph with 300 nodes.

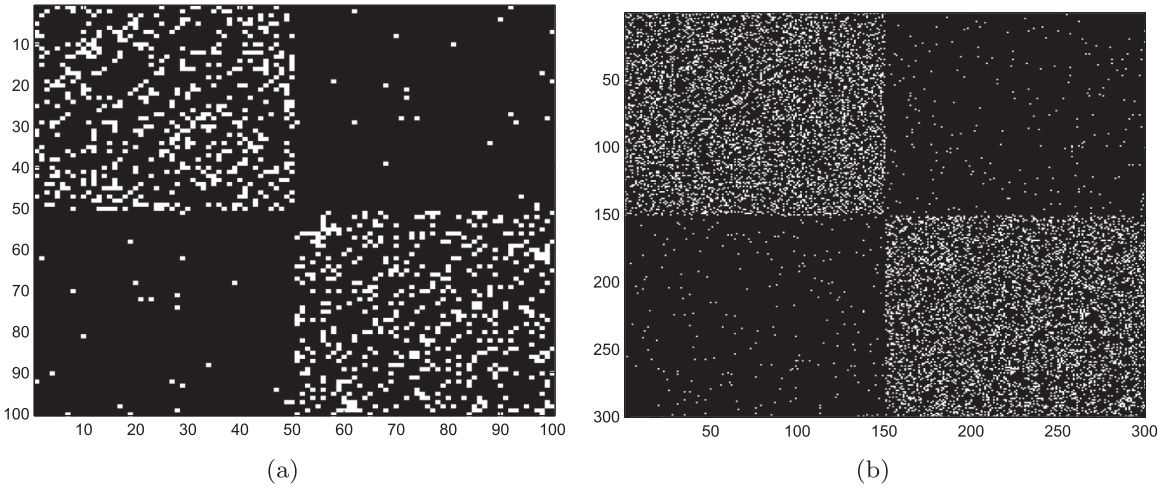

Fig. 5. Adjacency matrices of two realizations of a planted partition model with 100 nodes (a) and with 300 nodes (b). Both graphs have been generated by setting $p_{in} = 0.2$ and $p_{out} = 0.01$.

to the tolerance ε is $T=2543$. In Fig. 4 (a), the results obtained when spectral clustering is applied in the first phase are shown. In more details, a number of clusters $M \in \{30, 15, 6, 3, 2, 1\}$ are considered. Again, we do not consider a larger number of clusters (e.g., 60), because that clustering method has problems in detecting small clusters. When the nearest supernode approach is considered, the number of subgraphs used to divide the original graph, instead, is $M \in \{60, 30, 15, 6, 3, 2, 1\}$. The plot in Fig. 4 (b) shows the average over 10 tests of the results of each run. Again, the result corresponding to $M=1$ is the one obtained by the non-hierarchical consensus method.

From the plot shown in Fig. 4, we can infer that, for each type of seed used, and for each value of h, when random geometric graphs are considered, the hierarchical consensus method improves the results obtained by non-hierarchical one. In addition, when a random geometric graph with a relatively large number of nodes (300 nodes rather than 100 nodes) is considered, the proposed nearest supernode approach works often even better than spectral clustering. Indeed, when small subgraphs are generated (e.g., with an approximate average number of nodes h equal either to 5 or to 10), and even when the original graph is divided into only $M=2$ subgraphs, the nearest supernode approach provides a better performance guarantee than spectral clustering.

7.2. Planted partition model

The same procedure is applied to a planted partition model. This is a cluster-exhibiting random graph model, where nodes inside the same cluster are connected by an edge with probability p_{in}, while nodes belonging to different clusters are connected with probability p_{out}. In particular, we follow an Erdős–Rényi model generating a random graph with N nodes that exhibits two clusters: the first one with N_1 nodes, and the second one with N_2 nodes. We consider examples with two equally-sized clusters. Thus, starting from a graph with an even number N of nodes, we require each cluster to have $\frac{N}{2}$ nodes.

In the following, we examine two realizations of a planted partition model, one with $N=100$ nodes, and a larger one with $N=300$ nodes (their adjacency matrices are shown in Fig. 5). Both of them have intra-cluster probability of connection equal to $p_{in} = 0.2$, while nodes in different clusters are connected with inter-cluster probability $p_{out} = 0.01$.

For the example with $N=100$ nodes (adjacency matrix reported in Fig. 5(a)), the results are shown in Fig. 6. In particular, for the original graph G, the number of steps required to reach an approximation of the global consensus state up to a tolerance equal to $\varepsilon = 10^{-6}$ is $T=326$, while the second-largest eigenvalue modulus is equal to $\mu = 0.96$.

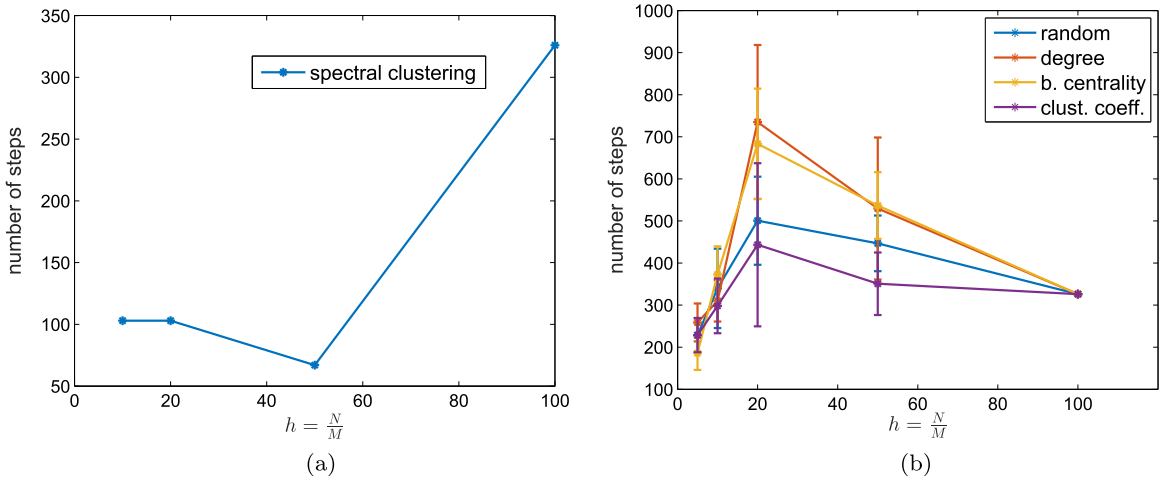

Fig. 6. Similar to Fig. 3, but for a realization of a planted partition model with 100 nodes, two equally sized clusters, $p_{in} = 0.2$, and $p_{out} = 0.01$.

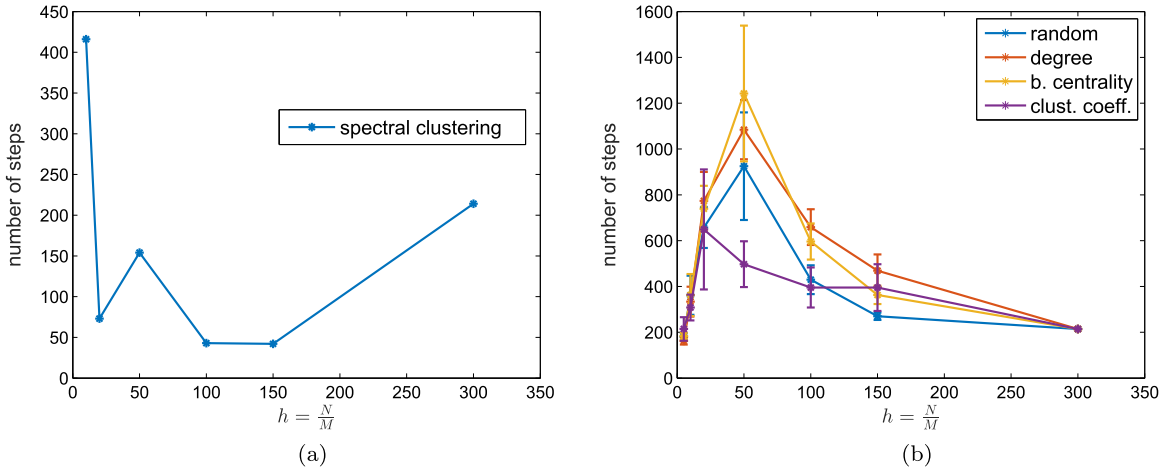

Fig. 7. Similar to Fig. 3, but for a realization of a planted partition model with 300 nodes, two equally sized clusters, $p_{in} = 0.2$, and $p_{out} = 0.01$.

When the planted partition model with $N=300$ nodes (whose adjacency matrix is shown in Fig. 5) (b) is considered, the second-largest eigenvalue modulus of the original graph G is equal to $\mu = 0.94$.

The results achieved by spectral clustering and by the nearest supernode approach are shown in Fig. 7 (a) and (b), respectively. When spectral clustering is adopted, the original graph is divided into a number of subgraphs $M \in \{30, 15, 6, 3, 2, 1\}$; while when we implement the first phase of the hierarchical consensus method by means of the nearest supernode approach, the number of subgraphs is $M \in \{60, 30, 15, 6, 3, 2, 1\}$.

When dealing with planted partition models, the nearest supernode approach does not work as well as in the case with random geometric graphs. Nevertheless, for the example with $N=100$ nodes, if the first phase is performed by spectral clustering, we obtain satisfactory results (as the plot in Fig. 6(a) shows). In fact, in this case, all the choices for the number of clusters considered to partition the original graph lead to a better performance with respect to the non-hierarchical consensus case. As expected, when the number of subgraphs to be generated is $M=2$, the method obtains the best result. Nevertheless, also the nearest supernode approach is able to achieve good results, especially when the clustering coefficient is adopted to select the supernodes, and subgraphs with a small approximate average number of nodes h (i.e., either 5 or 10) are considered (plot in Fig. 6(b)).

When a planted partition model with $N=300$ nodes is considered, again, spectral clustering works better than the nearest supernode approach, although the latter is able to achieve good results when subgraphs G_m with a sufficiently small approximate average number of nodes h (i.e., either 5 or 10), are extracted from the graph G.

7.3. Preferential attachment model

We conclude the numerical comparison of the hierarchical/non-hierarchical consensus methods by applying them to two realizations of a preferential attachment model. The graphs are generated according to the standard $G(N, m)$ model, where m (here, chosen to be equal to 2) denotes the number of edges to be inserted whenever a new node is added to the graph, while for the number N of nodes, we choose again to test the methods on both a graph with $N=100$ nodes (adjacency matrix shown in Fig. 8(a)) and one with 300 nodes (whose adjacency matrix is reported in Fig. 8 (b)).

For the example with $N=100$ nodes, the second-largest eigenvalue modulus of the transition probability matrix associated

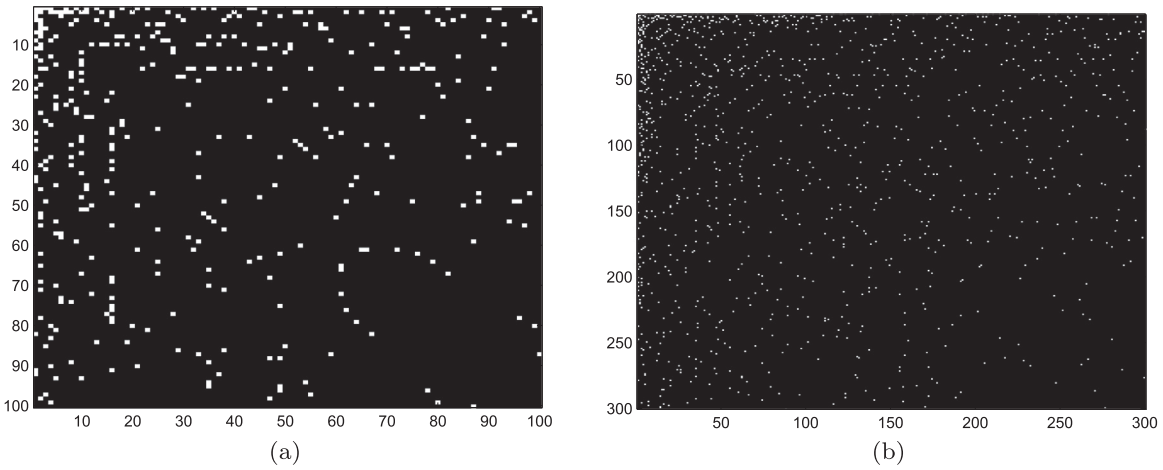

Fig. 8. Adjacency matrices of two realizations of a preferential attachment model: with 100 nodes (a); with 300 nodes (b).

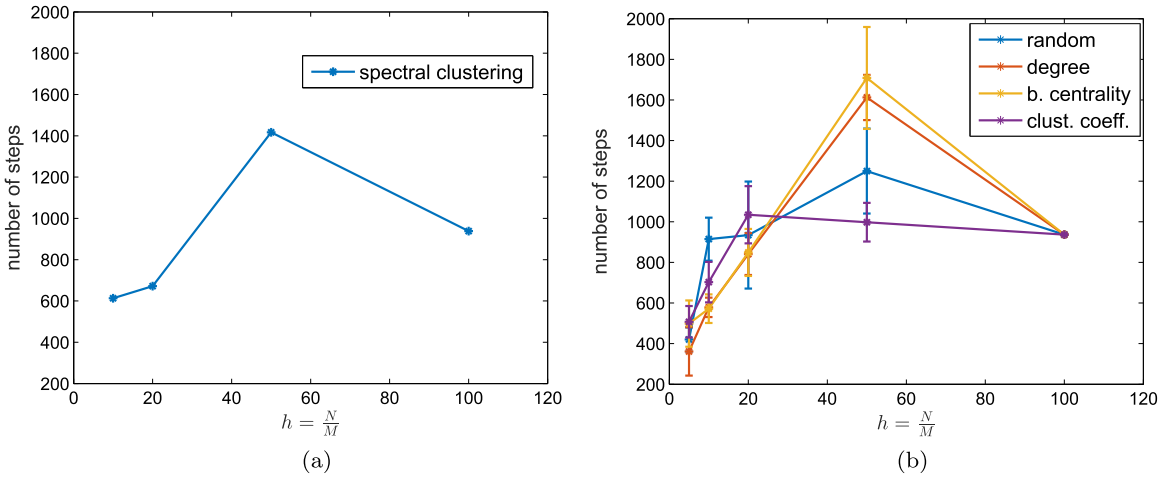

Fig. 9. Similar to Fig. 3, but for a realization of a preferential attachment model with 100 nodes and $m=2$.

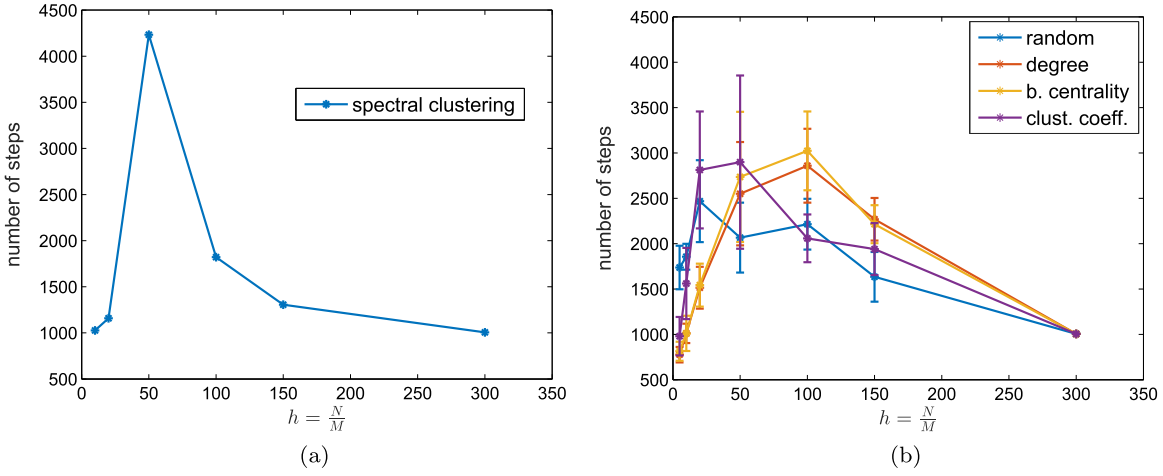

Fig. 10. Similar to Fig. 3, but for a realization of a preferential attachment model with 300 nodes and $m=2$.

with the original graph G is $\mu = 0.99$, while the number of steps required by the non-hierarchical consensus method to reach the global consensus state up to a tolerance equal to $\varepsilon = 10^{-6}$ is $T=937$. The results obtained by applying the hierarchical consensus method are shown in Fig. 9.

Finally, we perform the same numerical investigation for the example with $N=300$ nodes; the adjacency matrix is shown in Fig. 8(b). For this example, the second-largest eigenvalue modulus of the transition probability matrix of the original graph G is $\mu = 0.99$, while the number of steps needed by the non-hierarchical consensus method to reach an approximation of the global consensus state equal to $\varepsilon = 10^{-6}$ is $T=1005$. The results obtained by applying the hierarchical consensus method implemented both via spectral clustering and via the nearest supernode approach are shown in Fig. 10.

When the preferential attachment model is considered, the

hierarchical consensus method shows some problems with both the clustering methods adopted during its first phase, especially for the example with 300 nodes, modeling the case of a graph with a sufficiently large number of nodes. Nevertheless, when the smaller graph with 100 nodes is considered, good results are obtained. In particular, the nearest supernode approach with subgraphs associated with a relatively small approximate average number of nodes h is able to improve the performance with respect to the non-hierarchical consensus method. In addition, as the plot in Fig. 9(b) shows, the best results are obtained when the clustering coefficient is used for the generation of the supernodes.

8. Drawbacks and refinements of the basic version of the method

In the previous section, we have applied the hierarchical consensus method to different kinds of graphs, using two clustering methods for the extraction of the subgraphs G_m in its first phase. In this section, first, we analyze a factor, which we will call *antenna effect*, that is shown to influence strongly (and negatively) the results achieved by both clustering methods adopted to perform the first phase. Then, we propose a possible way to overcome that effect.

8.1. The antenna effect

In this subsection, we analyze the performance of the hierarchical consensus method when subgraphs G_m containing one node with degree equal to 1 are generated. We refer to this kind of situation by the term *antenna effect*. To simplify the theoretical analysis, we consider here the case of a graph like the one shown in Fig. 11 (a), which is made of a complete subgraph (in this case, made of $N-1=4$ nodes) connected by a single edge to another node with degree equal to 1. We call this kind of graph *basic antenna effect model* with N nodes. In the next subsection, we also investigate numerically other kinds of graphs showing the occurrence of the antenna effect.

We aim at studying the spectral properties of the basic antenna effect model, exploiting Cheeger's inequality (see formula (9)). We briefly recall from Section 3 that this inequality provides a lower and an upper bound on the second-smallest eigenvalue $\xi_2(L_{\text{norm}})$ of the normalized Laplacian matrix L_{norm} of a weighted graph, which possibly contains self-loops. As shown in Section 3, if a (doubly stochastic and symmetric) transition probability matrix P (or P') is used as the weight matrix, $\xi_2(L_{\text{norm}})$ is strongly related to the rate of convergence to the consensus state associated with the graph. More precisely, the larger $\xi_2(L_{\text{norm}})$, the larger such a rate of convergence.

Now, we aim at studying theoretically how the eigenvalue $\xi_2(L_{\text{norm}})$ is influenced by the occurrence of the antenna effect. To

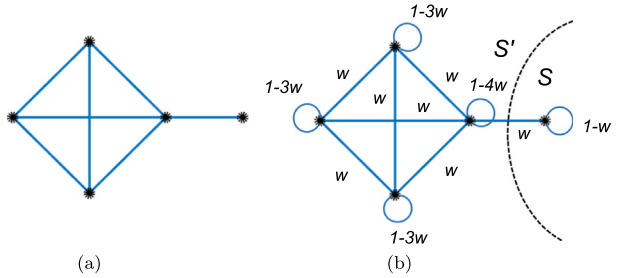

Fig. 11. (a) A nearly complete graph with one node attached to only a single node of the complete part (*basic antenna effect model*); (b) one choice for the subset S made to determine an upper bound on Cheeger's constant for the basic antenna effect model.

do this, we exploit Cheeger's inequality to find an upper bound on $\xi_2(L_{\text{norm}})$ for the basic antenna effect model with $N \geq 2$ nodes. For our investigation of the antenna effect, we do not need to compute exactly Cheeger's constant appearing inside Cheeger's inequality (which is a combinatorial problem, see formula (8)), but we limit to find an upper bound on it.

We recall that in the (proposed version of the) hierarchical consensus method, the matrix P is computed following the procedure described in the Appendix. Thus, to each edge of G one associates in P a weight w (ϵ according to the notation used in the Appendix), while a self-loop with weight $1-wd_i$ is associated in P to every vertex i, where d_i is the corresponding degree. Now, for the basic antenna effect model, the largest degree $d_{\max}(G)$ in the graph is achieved by the only node of the complete part of the graph which is connected to the node of degree 1, and is equal to $N-1$. Hence, since the weight of each self-loop has to be non-negative and smaller than or equal to 1, one obtains the bounds

$$0 \leq w \leq \frac{1}{N-1}. \tag{28}$$

Moreover, choosing the set S in the definition of Cheeger's constant as in Fig. 11(b) and using (8), one gets

$$\Phi(P) \leq \frac{w}{\min\{1, N-1\}} = w. \tag{29}$$

This, combined with formulas (9) and (28), provides the following upper bound on $\xi_2(L_{\text{norm}})$ for the basic antenna effect model with $N \geq 2$ nodes: $\xi_2(L_{\text{norm}}) \leq 2w \leq \frac{2}{N-1}$. Hence, we can conclude that, for N sufficiently large, such a graph model has a very small value of $\xi_2(L_{\text{norm}})$, and also the rate of convergence to its consensus state[10] is very small. It is also worth mentioning, instead, that, for $N \geq 3$, the complete subgraph with $N-1$ nodes inside the basic antenna effect model (i.e., the subgraph obtained disconnecting the node with degree 1, and replacing the weight $1-(N-1)w$ of the self-loop of the attached node with $1-(N-2)w$) has[11]

$$\xi_2(L_{\text{norm}}) = (N-2)w \cdot \frac{N-1}{N-2} = (N-1)w, \tag{30}$$

whose maximum value is

$$\xi_2(L_{\text{norm}}) = \frac{N-1}{N-2} \tag{31}$$

when w achieves its maximal admissible value $\frac{1}{N-2}$. When N is large, (31) simplifies to $\xi_2(L_{\text{norm}}) \simeq 1$. Hence, we can conclude that the presence of the additional node in the basic antenna effect model can decrease significantly the value of the second-smallest eigenvalue of the normalized Laplacian matrix.

From the analysis presented above, we can conclude that in situations for which the first phase of the hierarchical consensus method can produce subgraphs showing the antenna effect, it is better to keep the average number of nodes of such subgraphs small. This explains why, in the numerical results presented in Section 7, good results have been obtained several times when, e.g., subgraphs with a small approximate average number of nodes h (i.e., either 5 or 10), have been considered.

To support the theoretical analysis just presented, in the next subsection, we also investigate from a numerical point how the spectral properties of a graph can be influenced by the antenna effect.

[10] In case the basic antenna effect model is one of the subgraphs determined in the first phase of the hierarchical consensus method, this is the local consensus state of that subgraph.

[11] Formula (30) is provided, e.g., in Chung (1997, Lemma 1.7) for the case $w = \frac{1}{N-2}$ (no self-loops), whereas its extension to the presence of self-loops is straightforward.

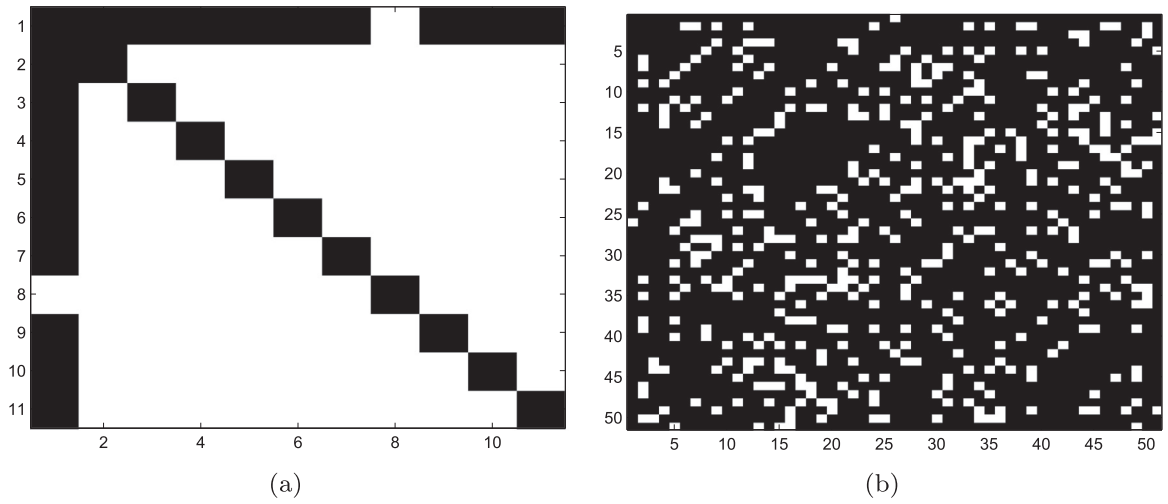

Fig. 12. A complete subgraph (a) and a sparser one (b), both connected to an additional node via one edge.

8.2. Numerical examples related to the antenna effect

In the following, we examine two trivial examples of graphs presenting the antenna effect. Their adjacency matrices are shown in Fig. 12; in particular, on the left in the figure, we consider a graph made of a complete subgraph with 10 nodes connected via one edge to an additional node with degree equal to 1; while the plot on the right shows the adjacency matrix of a random geometric subgraph (sparser than the complete one) with 50 nodes in total, again connected via one edge to an additional node with degree equal to 1.

We perform the following experiment. First, we consider the original graph (either the complete one with 10 nodes, or the sparser one with 50 nodes), and we compute the second-largest eigenvalue modulus of its associated transition probability matrix P. Then, we connect the additional node to one selected node of the original graph, and we compute the second-largest eigenvalue modulus of the transition probability matrix P associated with the resulting graph. We repeat this procedure selecting each time a different node of the original graph, then we compare the resulting second-largest eigenvalue moduli. To do the comparison, we compute the transition probability matrix P in two ways: first, using the method described in the Appendix, then solving the FMMC problem, which determines the optimal (i.e., smallest) value for the second-largest eigenvalue modulus when considering the non-hierarchical case. In this way, we avoid the possibility that an increase of the second-largest eigenvalue modulus obtained after the insertion of the additional node has to be ascribed to the particular method adopted to determine the transition probability matrix P.

Concerning the first example (the one reported in Fig. 12(a)), the second-largest eigenvalue modulus obtained before the insertion of the additional node is approximately equal to 0, for both the methods adopted to determine the matrix P. When the additional node is inserted (connecting it every time with a different node of the original graph), one obtains a remarkable increase in the value of the second-largest eigenvalue modulus. Fig. 13 shows the results: on the left, it reports the second-largest eigenvalue modulus of the matrix P computed through the method described in the Appendix, whereas on the right, it shows the results obtained by solving the FMMC problem. The latter produces slightly better results (smaller values of the second-largest eigenvalue modulus of P), but still in this case, the antenna effect causes a significantly large value of the second-largest eigenvalue modulus of P when the additional node is inserted. Due to the nature of the original graph (i.e., a complete one), it was expected that the results achieved when the additional node with degree 1 is inserted do not depend on the selection of the node of the original graph to which the new node is connected, as Fig. 13 shows.

Regarding the second example (the one reported in Fig. 12(b)),

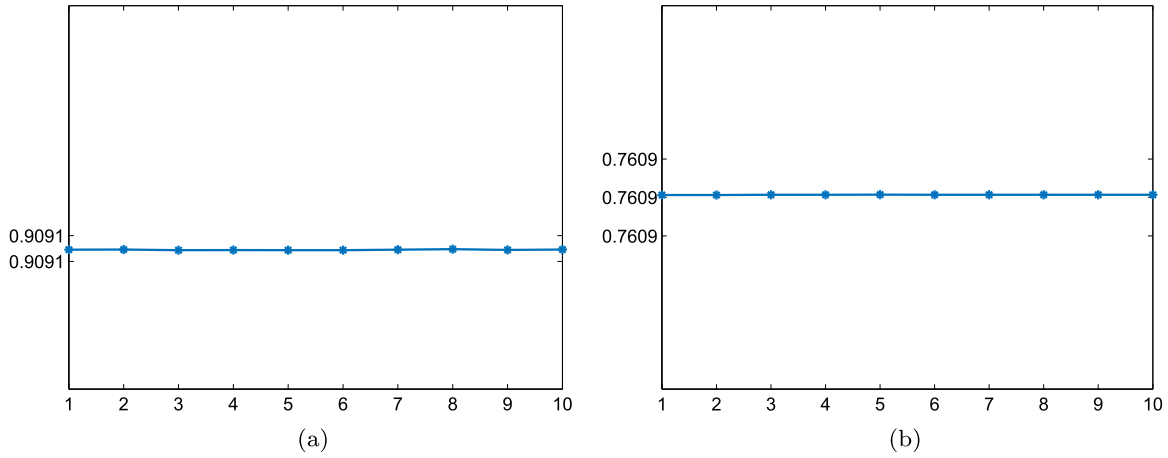

Fig. 13. Second-largest eigenvalue modulus of the transition probability matrix P associated with the graph obtained when a node with degree 1 is connected to a selected node (reported on the x-axis) of the complete subgraph in Fig. 12 (a): (a) generation of P according to the procedure detailed in the Appendix; (b) generation of P by solving the FMMC problem.

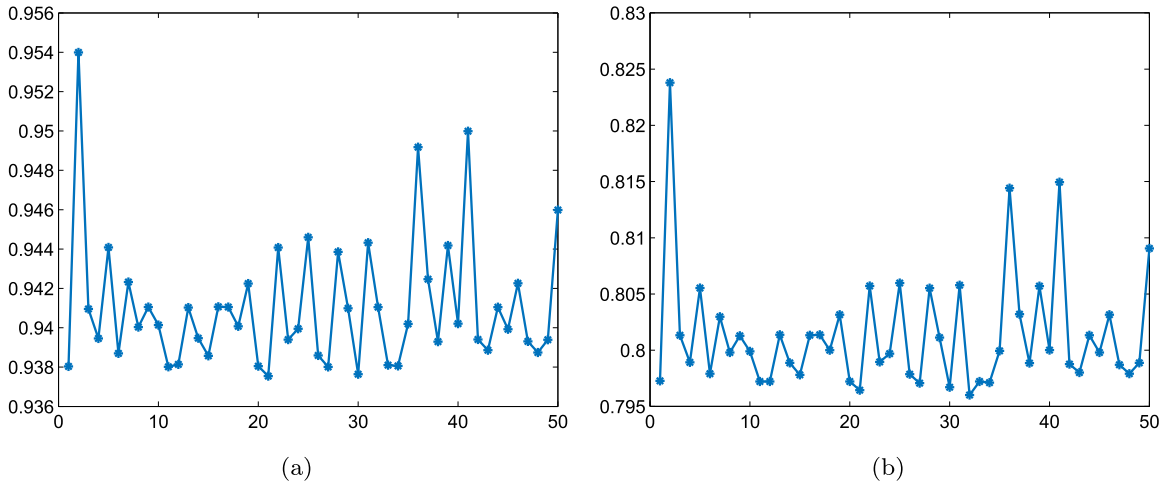

Fig. 14. Similar to Fig. 13, but for the sparse subgraph in Fig. 12 (b).

the results obtained by applying the same procedure as before are shown in Fig. 14. In this case, the second-largest eigenvalue modulus of the matrix P associated with the original graph is equal to $\mu = 0.8275$, when P is determined following the method described in the Appendix. The plot in Fig. 14(a) shows the second-largest eigenvalue moduli of the transition probability matrices P associated with the graphs obtained after connecting the new node to a selected node of the original graph. On the other hand, if we determine the transition probability matrices by solving the FMMC problem, we obtain a value of μ equal to 0.5606 for the original graph, while the second-largest eigenvalue moduli for the graphs obtained after the insertion of the additional node are shown in the plot in Fig. 14(b). Again, we can see that the insertion of the additional node increases significantly the second-largest eigenvalue modulus, demonstrating also that the antenna effect can show up also for graphs different from the basic antenna effect model, possibly containing more than one node with degree equal to 1.

Finally, we report the results obtained considering a realization of the planted partition model with $N=100$ nodes, $p_{in} = 0.2$ and $p_{out} = 0.01$. When dealing with this kind of model, we use the term "apparent cluster" to refer to each cluster obtained by the model itself in the "ideal case" $p_{in} = 1$ and $p_{out} = 0$, while we indicate with the term "subgraphs" the clusters extracted by the nearest supernode approach. In particular, to do the following numerical investigation, we apply the nearest supernode approach by selecting $M=2$ subgraphs and the clustering coefficient as the method to generate the supernodes. The adjacency matrix of the original graph is shown in Fig. 15; the second-largest eigenvalue modulus of the transition probability matrix associated with it and computed following the procedure described in the Appendix is $\mu = 0.953$.

In Fig. 16, the two subgraphs extracted from the original graph in the first phase of the hierarchical consensus method are shown. These subgraphs closely match the apparent clusters of the model, apart from the presence in the second subgraph of one node, which belongs to the first apparent cluster, and has degree 1 in the second subgraph. The second-largest eigenvalue modulus of the transition probability matrix associated with the first subgraph is equal to $\mu_1 = 0.854$, while the one associated with the second subgraph is much larger, i.e., it is $\mu_2 = 0.95$. In this case, one can see that only one node assigned to the "wrong" subgraph and whose degree is 1 in such subgraph produces a significantly large value of the second-largest eigenvalue modulus of the transition probability matrix associated with that subgraph. The numerical examples just presented provide an additional demonstration regarding the possibility of a high change (particularly, a "worsening") of the spectral properties of a graph/subgraph, if one additional node with degree 1 is inserted into the graph/subgraph. Thus, it is important to avoid the occurrence of this phenomenon, when considering the subgraphs generated in the first phase of the hierarchical consensus method, because the presence of the antenna effect could limit the effectiveness of the whole method.

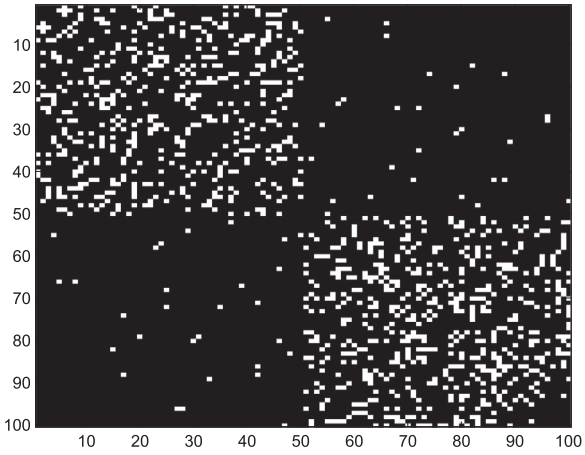

Fig. 15. Adjacency matrix of a realization of the planted partition model with 100 nodes, $p_{in} = 0.2$ and $p_{out} = 0.01$.

8.3. A possible way to overcome the antenna effect

In this subsection, we discuss a possible way to improve the results of the hierarchical consensus method, when subgraphs presenting the antenna effect are extracted from the original graph during the first phase of the method. The idea is to re-assign the nodes whose degree is equal to 1 in the subgraphs determined in the first phase of the method to subgraphs in which their degree would be higher. More precisely, once the M subgraphs G_m have been determined, we compute the degree of each node inside every single subgraph; if a node has degree equal to 1 in the subgraph it is currently assigned to, we compute its degree as if it would be re-assigned to any other subgraph, and we finally assign it to the subgraph where its degree would be maximal. In this way, we try to avoid the occurrence of nodes with degree equal to 1 in the subgraphs resulting from this modification of the first phase of the method. To investigate numerically the effectiveness of the proposed solution to overcome the antenna effect, we apply it to

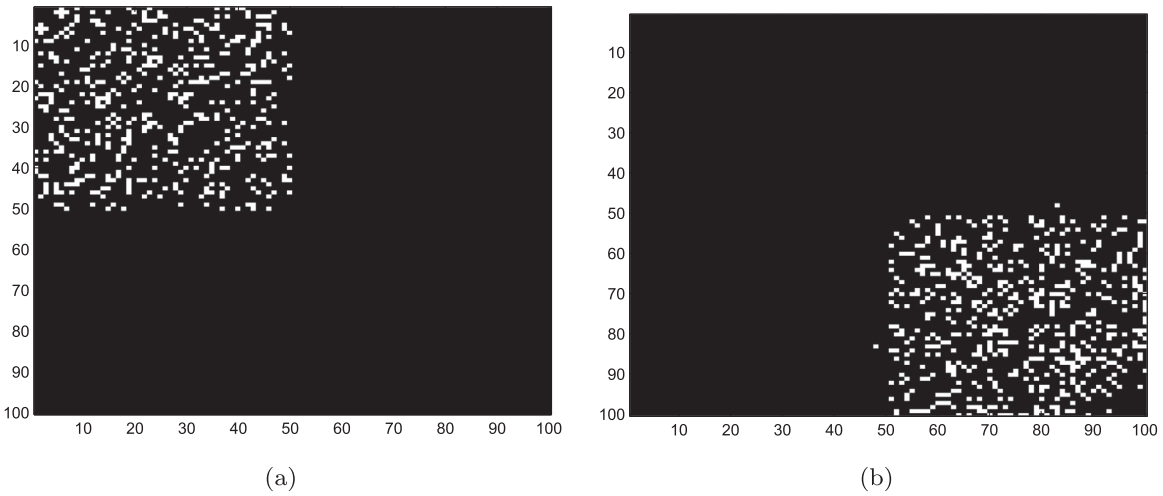

Fig. 16. Subgraphs determined by the hierarchical consensus method with $M=2$, when the nearest supernode approach is applied, and the supernodes are selected using the clustering coefficient: (a) first subgraph; (b) second subgraph. The adjacency matrix on the right shows the occurrence of the antenna effect.

the realization of the planted partition model with $N=300$ nodes, $p_{in} = 0.2$ and $p_{out} = 0.01$, which has been already considered in Fig. 7. The results achieved are shown in Fig. 17. The proposed solution is applied in combination with the nearest supernode approach, by considering all the four types of methods used to determine the supernodes. The results obtained by the original version of the hierarchical consensus method are shown in blue, whereas the ones obtained by using the solution proposed to overcome the antenna effect are shown in red. The plots shown in Figs. 17 highlight that the proposed solution of re-assigning the

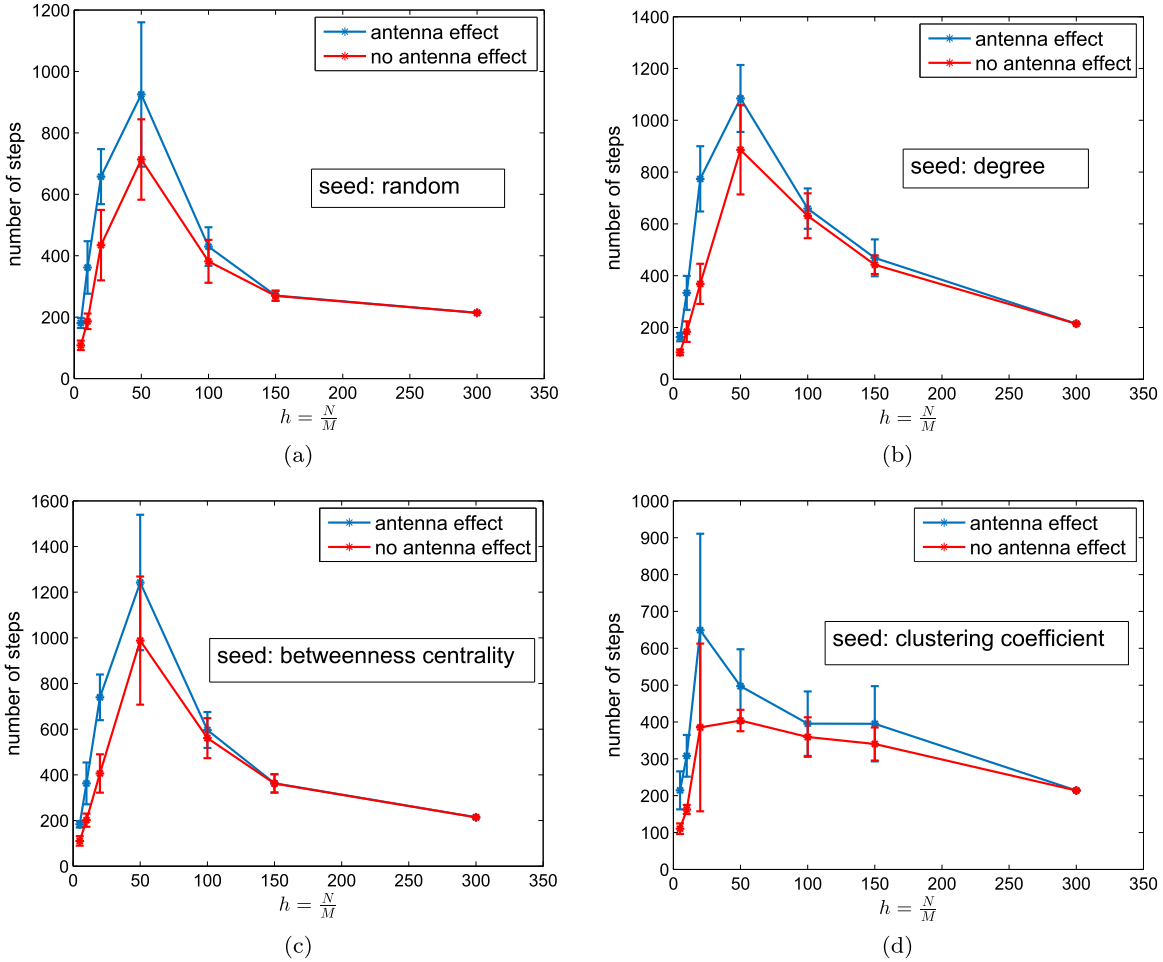

Fig. 17. Planted partition model with $N=300$ nodes, $p_{in} = 0.2$, and $p_{out} = 0.01$. In blue: number of iterations required by the original hierarchical consensus method to guarantee the given tolerance $\varepsilon = 10^{-6}$ in the approximation of the global consensus state; in red: number of iterations required to guarantee the same tolerance by the modified method, in which the nodes with degree 1 in the original subgraphs are assigned to other subgraphs. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

nodes with degree 1 in the subgraph they are assigned at the beginning remarkably improves the results. In the example reported, indeed, the number of steps of the hierarchical consensus method that guarantee the desired accuracy in the approximation of the global consensus state decreases considerably when it is combined with the modification proposed in this subsection, for all the methods used to generate the supernodes (the one based on the clustering coefficient providing the best results).

9. Discussion and conclusions

We have studied the approximation of the global consensus state of an agents' network through a hierarchical consensus method, divided into two phases. The goal of the first phase is to extract, from the original network, several subgraphs with "good" spectral properties, which guarantee fast convergence rates to their local consensus states. In the second phase, an auxiliary graph, derived from such subgraphs, is considered, to find an approximation of the global consensus state. The method has been motivated theoretically using spectral graph theory arguments and also investigated numerically, comparing it with a non-hierarchical consensus method on different kinds of graphs modeling real-world complex networks. The results of the hierarchical consensus method are satisfactory in almost all the cases studied, showing usually a better performance with respect to the non-hierarchical consensus method (i.e., a smaller number of iterations needed to guarantee the same accuracy in the approximation of the consensus state of the original network), for both the clustering methods used in the first phase (i.e., spectral clustering, and ad-hoc method, called nearest supernode approach). In more details, in the experiments conducted, both clustering methods used in the first phase of the hierarchical consensus method were able to achieve satisfactory results when realizations of random geometric graphs were considered. Indeed, in these cases, better results were obtained than the ones achieved by the non-hierarchical consensus method, as Figs. 3 and 4 show. When realizations of planted partition models were considered, the results highlight the fact that, when a cluster-exhibiting original graph was considered, the hierarchical consensus method implemented via spectral clustering had better performance than the non-hierarchical consensus method; while the nearest supernode approach revealed some drawbacks. Finally, when considering realizations of the preferential attachment model, both clustering methods showed problems. Nevertheless, the nearest supernode approach achieved better results than spectral clustering, and in some situations it was able to outperform the non-hierarchical consensus method.

In the paper we have also investigated, both theoretically and numerically, a phenomenon, called antenna effect, which could worsen, in some situations, the performance of the hierarchical consensus method itself. Then, we have suggested a possible way to overcome the antenna effect, and demonstrated experimentally its effectiveness. Finally, we have observed empirically that, when the nearest supernode approach is applied in the first phase of the hierarchical consensus method, the best results are usually obtained when the clustering coefficient is used for the choice of the supernodes. In fact, the nodes with the highest clustering coefficient are expected to be a sort of "centroids" of clusters.

For what concern possible future investigations and developments, the hierarchical consensus method could be applied to other kinds of graphs modeling real-world complex networks, and also to a stochastic version of the consensus problem. Moreover, from the numerical results obtained, we have observed that none of the two clustering methods adopted is always better than the other one. Hence, a possible improvement would be to make the choice of the clustering method automatic, possibly based on some a priori knowledge of the structure of the graph. Other possible improvements concern the development of methods for a better detection of subgraphs with "good" spectral properties. This goal could be achieved, e.g., through a more effective generation of the supernodes, a better assignment of the other nodes to the supernodes, a dynamic choice of the supernodes in case of their "bad" initial choice, and the use of splitting/merging clustering techniques. Moreover, distributed approximate versions of spectral clustering (e.g., the one developed in Heefeda et al., 2012) could be used inside the hierarchical consensus framework to overcome the drawbacks of spectral clustering mentioned in Section 4. Finally, as another possible development, one could consider a variation of the hierarchical consensus method, in which there is a continuous alternation between its two phases.

Appendix A. Determination of the transition probability matrices

Given a connected graph G, the following method, taken from Garin et al. (2010), allows one to construct a doubly stochastic and symmetric matrix P for which convergence to the consensus state in formula (1) is guaranteed. More specifically, the elements of the matrix P are defined as follows:

$$P_{ij} = \begin{cases} 1 - \epsilon d_i & \text{if } i = j, \\ \epsilon, & \text{if } (i,j) \in E \text{ and } i \neq j, \end{cases} \tag{32}$$

where $d_i = \left|\{ j \in V,\, i \neq j\colon (i,j) \in E \}\right|$ is the (unweighted) degree of the node i, and $\epsilon > 0$ is chosen in such a way that $\epsilon < \frac{1}{d_{\max}(G)}$, where $d_{\max}(G)$ is the maximum degree of any node in G. In this way, the diagonal entries of P are guaranteed to be positive. Equivalently, in terms of the (unweighted) adjacency matrix $A \in \{0,1\}^{N \times N}$ of the graph G (whose generic element $A_{ij} = 1$ if and only if $(i,j) \in E$), and of its diagonal (unweighted) degree matrix D (whose diagonal elements are the d_i's), formula (32) can be expressed as $P = I + \epsilon(A - D)$. The corresponding normalized Laplacian matrix is

$$L_{\text{norm}} = I - P = -\epsilon(A - D). \tag{33}$$

Since the normalized Laplacian matrix is symmetric and positive-semidefinite, it follows from (33) that the larger ϵ, the larger the second-smallest eigenvalue ξ associated with the normalized Laplacian matrix L_{norm}. The procedure just described is also used, with obvious changes in notation, to determine the transition probability matrices P_m of the subgraphs G_m involved in the first phase of the hierarchical consensus method, and the one P_{aux} of the auxiliary graph G_{aux}. It has to be remarked that all the matrices P_m are derived starting from submatrices of the same adjacency matrix A associated with the graph G, although the weights associated with the self-loops and the choices for ϵ may be different. Finally, when computing the transition probability matrix associated with a generic graph/subgraph, we choose

$$\epsilon = \frac{1}{d_{\max} + 1}, \tag{34}$$

where $d_{\max}$ refers to the specific graph/subgraph.[12]

[12] Another possible choice would be $\epsilon = \frac{1}{d_{\max}(G)+1}$ for every graph/subgraph, since $d_{\max}(G_m) \leq d_{\max}(G)$ for every subgraph G_m. However, the choice (34) produces a larger second-smallest eigenvalue $\xi_2(L_{\text{norm}})$ of the associated normalized Laplacian matrix L_{norm}, as it follows from (33).

References

Barabási, A.-L., Albert, R., 1999. Emergence of scaling in random networks. Science 286, 509–512.

Batson, J., Spielman, A., Srivastava, N., Teng, S.-H., 2013. Spectral sparsification of graphs: theory and algorithms. Commun. ACM 56, 87–94.

Battiston, S., Doyne Farmer, J., Flache, A., Garlaschelli, D., Haldane, A.G., Heesterbeek, H., Hommes, C., Jaeger, C., May, R., Scheffer, M., 2016. Complexity theory and financial regulation. Science 351, 818–819.

Bollobás, B., 1998. Random Graphs. Springer, Cambridge, UK.

Boyd, S., Diaconis, P., Xiao, L., 2004. Fastest mixing Markov chain on a graph. SIAM Rev. 46, 667–689.

Castellano, C., Fortunato, S., Loreto, V., 2009. Statistical physics of social dynamics. Rev. Mod. Phys. 81, 591.

Chung, F., 1996. Laplacians of Graphs and Cheeger Inequalities, vol. 2. Bolyai Mathematical Society, Keszthely, pp. 157–172.

Chung, F., 1997. Spectral Graph Theory. AMS, Providence, Rhode Island.

Como, G., Fagnani, F., Zampieri, S., 2012. I sistemi multi-agente e gli algoritmi di consenso, la matematica nella società e nella cultura: rivista dell'unione matematica italiana. Serie I 5 (1), 1–29.

Del Vicario, M., Bessi, A., Zollo, F., Petroni, F., Scala, A., Caldarelli, G., Stanley, H.E., Quattrociocchi, W., 2016. The spreading of misinformation online. Proc. Natl. Acad. Sci. 113, 554–559.

Epstein, M., Lynch, K., Johansson, K.H., Murray, R.M., 2008. Using hierarchical decomposition to speed up average consensus. In: Proceedings of the International Federation of Automatic Control (IFAC) World Congress, Seoul, Korea, pp. 612–618.

Fagnani, F., 2014. Lectures on consensus dynamics over networks. INRIA Winter Sch. Complex Netw. 214.

Frias-Martinez, V., Frias-Martinez, E., 2014. Spectral clustering for sensing urban land use using Twitter activity. Eng. Appl. Artif. Intell. 35, 237–245.

Garin, F., Schenato, L., 2010. A survey on distributed estimation and control applications using linear consensus algorithms. In: Bemporad, A., Heemels, M., Johansson, M. (Eds.), Networked control systems, Lecture Notes in Control and Information Sciences vol. 406; 2010, pp. 75–107.

Gnecco, G., Morisi, R., Bemporad, A., 2014. Sparse solutions to the average consensus problem via l_1-norm regularization of the fastest mixing Markov-chain problem. In: Proceedings of the 53rd IEEE Int. Conf. on Decision and Control (CDC), IEEE, Los Angeles, pp. 2228–2233.

Gnecco, G., Morisi, R., Bemporad, A., 2015. Sparse solutions to the average consensus problem via various regularizations of the fastest mixing Markov-chain problem. IEEE Trans. Netw. Sci. Eng. 2, 97–111.

Heefeda, M., Gao, F., Abd-Almageed, W., 2012. Distributed approximate spectral clustering for large-scale datasets. In: Proceedings of the 21st International Symposium on High-Performance Parallel and Distributed Computing (HPDC), pp. 223–234.

Langone, R., Alzate, C., De Ketelaere, B., Vlasselaer, J., Meert, W., Suykens, J.A.K., 2015. LS-SVM based spectral clustering and regression for predicting maintenance of industrial machines. Eng. Appl. Artif. Intell. 37, 268–278.

Li, X., Bai, H., 2012. Consensus on hierarchically decomposed topology to accelerate convergence and to improve delay robustness. In: Proceedings of the 24th Chinese Control and Decision Conf. (CCDC), IEEE, Taiyuan, China, pp. 597–602.

Liu, Y.-Y., Slotine, J.-J., Barabási, A.-L., 2011. Controllability of complex networks. Nature 473, 167–173.

Lovisari, E., Zampieri, S., 2012. Performance metrics in the average consensus problem: a tutorial. Annu. Rev. Control 36, 26–41.

Mesbahi, M., Egerstedt, M., 2010. Graph Theoretic Methods in Multiagent Networks. Princeton University Press, Princeton.

Mossel, E., Neeman, J., Sly, A., 2015. Reconstruction and estimation in the planted partition model. Probab. Theory Relat. Fields 162, 431–461.

Newman, M., 2010. Networks: An Introduction. Oxford University Press, Oxford.

Pastor-Satorras, R., Castellano, C., Van Mieghem, P., Vespignani, A., 2015. Epidemic processes in complex networks. Rev. Mod. Phys. 87, 925–979.

Vivaldo, G., Masi, E., Pandolfi, C., Mancuso, S., Caldarelli, G., 2016. Networks of plants: how to measure similarity in vegetable species. Sci. Rep. 11, 6. http://dx.doi.org/10.1038/srep27077, article no. 27077.

von Luxburg, U., 2007. A tutorial on spectral clustering. Stat. Comput. 17, 395–416.