

# Distributed stability tests for large-scale systems with limited model information

Frederik Deroo <sup>\*a†</sup>, Martin Meinel<sup>b†</sup>, Michael Ulbrich<sup>b</sup>, Sandra Hirche<sup>a</sup>

**Abstract**—Privacy concerns spark the desire to analyze large-scale interconnected systems in a distributed fashion, i.e. without a central entity having global model knowledge. Two different approaches are presented to analyze stability of interconnected linear time-invariant systems with limited model knowledge. The two algorithms implement sufficient stability conditions and require information exchange only with direct neighbors thus reducing the need to share model data widely and ensuring privacy. The first algorithm is based on an M-matrix condition, the second one on Lyapunov inequalities. Both algorithms rely on distributed optimization using a dual decomposition approach. Numerical investigations are used to validate both approaches.

## I. INTRODUCTION

Large-scale interconnected systems have attracted a lot of research recently in the field of system and control theory in attempts to handle new technological challenges in terms of ever-increasing system sizes and to make use of advanced communication technologies. The system class represents numerous practical applications, including traffic systems [1], vehicle formations [2], the power system [3] and other types of distribution and infrastructure networks, just to name a few. Because centralized methods are generally not applicable to this type of systems, the system theory community is constantly looking for new distributed approaches. While the design of distributed control laws has received a lot of attention, the distributed analysis of large-scale systems, e.g. with regards to stability, has seen only few results.

The desire to test stability distributedly, meaning that there is no central model of the system and that the subsystems only share their information with a relatively small part of the overall system, is important for several reasons. Our main motivation is model data privacy of the subsystems in a scenario when competitors need to form an interconnected dynamical system (e.g. in an electrical power grid) but are hesitant to share their exact model data with a central entity or all other participants. Another point is that the computation effort can be distributed among the subsystems, and additionally that these distributed methods are generally more flexible when a system changes because only parts of the system need to adapt. Last, a distributed approach promises better scalability.

\*Corresponding author.

†These authors contributed equally to this work.

<sup>a</sup>F. Deroo and S. Hirche are with the Institute for Information-oriented Control, Technische Universität München, D-80333 München, Germany, fred.deroo@tum.de, hirche@tum.de

<sup>b</sup>M. Meinel and M. Ulbrich are with the Chair of Mathematical Optimization, Department of Mathematics, Technische Universität München, Boltzmannstr. 3, D-85747 Garching b. München, Germany, meinel@ma.tum.de, mulbrich@ma.tum.de

The work is supported by the German Research Foundation (DFG) within the Priority Program SPP 1305 “Control Theory of Digitally Networked Dynamical Systems” and the TUM Munich School of Engineering.

## A. Related work

Traditionally, large-scale dynamical systems are analyzed from a centralized point of view, assuming that there is a central entity with access to the whole system model or at least to the whole interconnection structure. For example, already in the 1970s large-scale system stability was treated in [4]. They derive sufficient stability conditions under the assumption that every subsystem is dissipative and that both the supply rate function of each subsystem and the overall interconnection structure are centrally known. The conditions are then posed in the form of LMIs or definiteness requirements. Similarly, in [5], sufficient stability tests are given where a matrix condition is constructed which relates the ratio of stability of the local dynamics of each subsystem with the effect of the incoming interconnections. The matrix is then required to be a so-called M-matrix which can be tested e.g. using linear programming. Both of these methods have in common that the stability test itself is done centrally. In [6], the two approaches are relaxed and it is pointed out that for the class of positive linear systems, stability can be evaluated distributedly. The approach of applying distributed optimization in a control context has been used for example in a model predictive setting, see [7], [8], but is generally restricted to control design. In terms of stability analysis, the results in [9] show that a distributed stability test is possible for positive systems using distributed linear programming. Other results on scalable and distributed system analysis are given in [10]. The authors present scalable stability conditions for interconnected heterogeneous LTI systems represented by transfer functions that are feedback coupled through a bipartite graph. The sufficient conditions they derive are based on the Nyquist stability criterion, and the conditions are posed in such a way that subsystems only need to exchange information with their neighbors. The conditions are generalized in [11] using integral quadratic constraints (IQC). However, in both cases the derived conditions need to be evaluated for all frequencies  $\omega$ . The concept of block-diagonal stability, a generalization of diagonal stability [12], and a notion that is adopted in this article, is treated in [13] and [14], however not from a distributed perspective.

## B. Contribution

The main contributions of this article are two different distributed tests for stability for the general class of interconnected LTI systems using limited model information. There are no restrictions on the systems themselves and only limited information exchange with neighboring systems is necessary. The first stability test is based on the aforementioned M-

Matrix test from [5]. Instead of testing if a matrix is an M-matrix in a centralized fashion, we formulate an optimization problem that contains the stability condition as constraints. The second stability test is based on the Lyapunov inequality and is analogously formulated as an optimization problem. More specifically, we consider block-diagonal Lyapunov stability [13] because of its structural properties. Both optimization problems have a structure which can be exploited using the Distributed Nesterov Algorithm (DNA) from [15] in combination with dual decomposition. As a result we present two different distributed stability tests that allow us to check if an LTI system is asymptotically stable using only neighborhood information in an iterative distributed optimization scheme. As mentioned in Section I-A, the literature so far provides similar results only for the special case of positive systems, and most of the literature provides stability analysis methods only from a centralized perspective. Furthermore, an additional contribution of the article is the general idea of using advanced distributed optimization techniques to analyze systems distributedly. We stress that the use of distributed optimization techniques is not mainly aimed at reducing computational effort but the idea is to obtain methods that preserve model data privacy for the subsystems as far as possible. This is achieved in so far that the subsystems only need to share their dynamic model with their direct neighbors, and not globally or centrally. In addition, the advantages and disadvantages of each test are discussed and numerical experiments validate the presented methods.

The remainder of this article is organized as follows: In Section II, we give the problem formulation. Two different distributed stability tests are presented in Sections III and IV. The results of numerical investigations are given in Section V, before concluding with a summary in Section VI.

**Notation:**  $I_N \in \mathbb{R}^{N \times N}$  is the identity matrix,  $1_N \in \mathbb{R}^{N \times 1}$  is a vector with ones. The spectrum of a matrix  $A$  is  $\lambda(A)$  while the real part of a complex number  $x$  is  $\text{Re}(x)$ . An entry-wise positive vector  $d \in \mathbb{R}^N$  is denoted by  $d > 0$ . The Frobenius inner product of two matrices  $A, B$  is  $\sum_i \sum_j A_{ij} B_{ij} = A \bullet B$ . The set of symmetric matrices is  $\mathbb{S}^n$ .

## II. PROBLEM FORMULATION

We consider an LTI system consisting of  $N$  subsystems. The dynamics of subsystem  $i$  are written as

$$\dot{x}_i(t) = A_{ii}x_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}x_j(t), \quad i = 1, \dots, N, \quad x_i(0) = x_{i,0}, \quad (1)$$

where  $x_i \in \mathbb{R}^{n_i}$  is the state,  $A_{ii} \in \mathbb{R}^{n_i \times n_i}$  represents local dynamics, and  $A_{ij} \in \mathbb{R}^{n_i \times n_j}$  represents interaction with neighbors. In the following, the blocks  $A_{ii}$  and  $A_{ij}$  will be referred to as (*dynamic*) *model data*.

By concatenation of the subsystems' states, the overall interconnected system is written compactly as

$$\dot{x}(t) = Ax(t), \quad x(0) = x_0, \quad (2)$$

where  $x = [x_1^T, \dots, x_N^T]^T \in \mathbb{R}^n$  is the state. The matrix  $A$  consists of the blocks  $A_{ij}$ . The subsystems form a partition of the states of the system, so it must hold that  $\sum_{i=1}^N n_i = n$ .

To define the set of neighbors of subsystem  $i$ , we consider the directed interconnection graph  $\mathcal{G}_I(\mathcal{V}_I, \mathcal{E}_I)$  associated with the matrix  $A$ . The vertex set  $\mathcal{V}_I$  is given by the set of subsystems  $\mathcal{V}_I = \{1, \dots, N\}$ , and the edge set  $\mathcal{E}_I$  contains the edge  $(j, i) \in \mathcal{E}_I$  iff any entry of  $A_{ij} \neq 0$ . The set of nodes whose states influence the state of node  $i$  is defined as

$$\mathcal{N}_{\text{in},i} = \{j | (j, i) \in \mathcal{E}_I\},$$

and the set of neighboring nodes as

$$\mathcal{N}_i = \{j | (i, j) \in \mathcal{E}_I \text{ and/or } (j, i) \in \mathcal{E}_I\}.$$

Additionally, we define the undirected version of  $\mathcal{G}_I$  as  $\mathcal{G}_{I,u} = (\mathcal{V}_{I,u}, \mathcal{E}_{I,u})$  with  $\mathcal{V}_{I,u} = \mathcal{V}_I$  and  $\mathcal{E}_{I,u} = \mathcal{E}_I \cup \mathcal{E}_I^T$  where  $\mathcal{E}_I^T$  is the edge set of the transpose graph of  $\mathcal{G}_I$ .

The division into the  $N$  subsystems and the resulting interconnection graph  $\mathcal{G}_I$  is motivated by the desire to preserve model data privacy, and the use of only limited model information. Classically, the structuring of the overall system (2) into  $N$  subsystems (1) is motivated by physics, function or geography, e.g. in a multi-agent system, or an interconnected system. In contrast, in our setting subsystem (1) may comprise multiple physical, functional or geographically distant components which, however, are willing to share their model data completely among each other, i.e. the clustering of the overall system is induced by privacy constraints.

**Definition 1.** *Given system (1),(2), a system analysis method is considered to use limited model information if subsystem  $i$  has knowledge of  $A_{ii}$ , and all  $A_{ij}, A_{ji}$  with  $j \in \mathcal{N}_i$ , and it exchanges information only with subsystems  $j \in \mathcal{N}_i$ .*

In other words, a method of this type uses no centralized dynamic model and no central decision maker. Instead, the knowledge about the dynamic model data is distributed among the subsystems in that each subsystem only knows its row and column. Concretely, the subsystems know their own dynamic model (1) and parts of the dynamic model of their neighbors. Methods using limited model information ensure privacy because the subsystems need to share their dynamic model (1) only with a small subset of other subsystems, and there is no central entity that knows the overall system.

The goal of this work is to decide about the stability of (2) with limited model information as defined in Definition 1.

**Remark 1.** *The possible clustering into larger subsystems raises several important questions such as the influence of different clusterings on the conservativeness of the subsequent stability tests, and optimal clustering. This will be future work.*

## III. DISTRIBUTED TEST FOR CONNECTIVE STABILITY

In this section we present a distributed test to analyze if a system is connectively stable.

**Definition 2.** [5] *System (2) is connectively stable if it is stable in the sense of Lyapunov for all interconnection terms  $\alpha A_{ij}$  with  $\alpha \in [0, 1]$  and  $A_{ij}$  defined in (1).*

Connective stability therefore implies that the system remains asymptotically stable even if interconnection strengths are changed to smaller values. A well-known tool to establish connective stability are vector Lyapunov functions. For completeness we shortly recapitulate the foundations [5]. The idea

of vector Lyapunov functions is to construct a Lyapunov function for the overall interconnected system as a weighted sum of individual Lyapunov functions of the isolated subsystems  $\dot{x}_{i,\text{iso}} = A_{ii}x_{i,\text{iso}}$ . The main tool to determine whether a vector Lyapunov function exists are M-matrices.

**Lemma 1.** [5] *A matrix  $W \in \mathbb{R}^{N \times N}$  with nonpositive off-diagonal elements is an M-matrix if there exists a vector  $d \in \mathbb{R}^N$  with strictly positive entries such that  $Wd > 0$ .*

The following assumption is made in the literature [5].

**Assumption 1.** *The decoupled individual subsystems are asymptotically stable, i.e.  $\text{Re}(\lambda(A_{ii})) < 0$ , and all eigenvalues of  $A_{ii}$  are distinct for all  $i \in \{1, \dots, N\}$ .*

The regular transformation matrices  $T_i \in \mathbb{R}^{n_i \times n_i}$  are defined which transform the isolated system  $i$  described by  $\dot{x}_{i,\text{iso}} = A_{ii}x_{i,\text{iso}}$  into  $\dot{\tilde{x}}_{i,\text{iso}} = \tilde{A}_{ii}\tilde{x}_{i,\text{iso}}$ . Given the second part of Assumption 1,  $T_i^{-1}A_{ii}T_i$  has a real Jordan form and the transformation mapping is  $x_{i,\text{iso}} = T_i\tilde{x}_{i,\text{iso}}$ . This transformation is used because it reduces conservativeness in the subsequent stability test [5]. With that, the following matrix  $W = [w_{ij}]$  is constructed as

$$w_{ij} = \begin{cases} \sigma_M^i, & i = j \\ -\sqrt{\max(\lambda(\tilde{A}_{ij}^T \tilde{A}_{ij}))}, & j \in \mathcal{N}_{\text{in},i} \\ 0, & \text{else,} \end{cases} \quad (3)$$

where  $\sigma_M^i = -\max(\text{Re}(\lambda(\tilde{A}_{ii})))$  and  $\tilde{A}_{ij} = T_i^{-1}A_{ij}T_j$ . With that matrix, the following theorem is formulated.

**Theorem 1.** [5] *Given Assumption 1 the system (2) is connectively stable if there is a vector  $d > 0$  such that  $Wd > 0$  with  $W$  defined in (3).*

Theorem 1 allows us to construct a test matrix to formulate a condition for connective stability, and its construction requires only limited model information. The sparsity structure of  $W$  is identical to the block sparsity structure of  $A$  as described by the graph  $\mathcal{G}_l$ . However, the test if the matrix  $W$  is an M-matrix, i.e. if there exists a  $d > 0$  such that  $Wd > 0$  is a global problem. Hence, we need to find a distributed method to find a suitable  $d$ , if possible. To achieve this, a distributed optimization problem is formulated that includes the stability condition as a constraint and then we employ distributed optimization techniques such that only limited model information according to Definition 1 is required. The presented distributed optimization approach is based on the results in [15]. The information exchange topology of the employed algorithm is determined by the structure in the constraints, which corresponds to the graph  $\mathcal{G}_{l,u}$ .

Therefore we can reformulate the stability condition as the following optimization problem with convexity parameters  $\sigma_\delta, \sigma_{d_i} > 0$

$$\min_{(d,\delta) \in \mathbb{R}^{N+1}} f(d,\delta) = -\delta + \frac{\sigma_\delta}{2} \delta^2 + \sum_{i=1}^N \frac{\sigma_{d_i}}{2N} d_i^2 \quad (4a)$$

$$\text{s.t. } -Wd + \gamma\delta 1_N \leq 0, \quad (4b)$$

$$-d + \delta 1_N \leq 0, \quad (4c)$$

where  $\gamma > 0$  is arbitrary. For the choice of  $\sigma_\delta, \sigma_{d_i} > 0$  see Remark 2 below.

Problem 4 is inspired by phase-1 problems from linear programming that are solved to find a feasible solution to a given set of linear constraints. Here, the strongly convex part  $-\delta + \frac{\sigma_\delta}{2} \delta^2$  in (4a) alone guarantees that a vector  $d^* > 0$  which satisfies  $Wd^* > 0$  is the candidate for an optimal solution of (4), as in this case there exists a  $\delta^* > 0$  with  $-\delta^* + \frac{\sigma_\delta}{2} \delta^{*2} < 0$ . Finally, we add the term  $\sum_{i=1}^N \frac{\sigma_{d_i}}{2N} d_i^2$  in (4a) to obtain a strongly convex objective function  $f(d,\delta)$  which in turn yields a differentiable dual objective function.

In the following theorem we show that even with the addition of this strongly convex term in  $d$ , the optimal objective function value of (4) indicates the existence of a vector  $d > 0$  such that  $Wd > 0$ . The conversion from a feasibility problem to an equivalent optimization problem allows us to apply distributed optimization algorithms.

**Theorem 2.** *There exists a  $d > 0$  such that  $Wd > 0$  if and only if there exists a feasible point  $(d,\delta)$  for (4) with a negative optimal objective function value  $f^* < 0$ .*

*Proof.*  $\Rightarrow$ : Let  $Wd > 0$  hold for some  $d = \tilde{d} > 0$ . Set  $\lambda_1 = \min_i(\tilde{d}_i)$ , and  $\lambda_2 = \min_i((W\tilde{d})_i)$ . Then  $\tilde{\delta} := \min(\lambda_1, \lambda_2/\gamma) > 0$ . Further,  $(d,\delta) = (0,0)$  and  $(d,\delta) = (\tilde{d}, \tilde{\delta})$  are feasible for problem (4). Hence, by convexity, for all  $t \in [0,1]$ ,  $(d,\delta) = (t\tilde{d}, t\tilde{\delta})$  is feasible for (4). Now consider the objective function  $\Phi(t)$  at  $(t\tilde{d}, t\tilde{\delta})$

$$\Phi(t) := -t\tilde{\delta} + \frac{\sigma_\delta}{2} t^2 \tilde{\delta}^2 + \sum_{i=1}^N \frac{\sigma_{d_i}}{2N} t^2 \tilde{d}_i^2.$$

Since  $\Phi'(0) = -\tilde{\delta} < 0$ , we see that if we choose  $0 < t \leq 1$  sufficiently small, then  $\Phi(t) < \Phi(0) = 0$ . We then set  $(d,\delta) = (t\tilde{d}, t\tilde{\delta})$  which has the desired properties.

$\Leftarrow$ : We now show that if there exists a feasible point  $(\tilde{d}, \tilde{\delta})$  of (4) with negative objective function value, then  $\tilde{\delta} > 0$  and  $W\tilde{d} > 0$ . In fact, a negative objective function value can only be achieved if  $\tilde{\delta} > 0$ . Now

$$\tilde{d} \geq \tilde{\delta} 1_N > 0, \quad W\tilde{d} \geq \gamma\tilde{\delta} 1_N > 0.$$

Hence, setting  $d = \tilde{d}$  finishes the proof.  $\square$

In the following we describe how the optimization problem can be solved distributedly using the DNA from [15] in combination with dual decomposition. To derive the dual problem of (4) consider the corresponding Lagrangian

$$\begin{aligned} \mathcal{L}(d,\delta,\lambda,\mu) &= -\delta + \frac{\sigma_\delta}{2} \delta^2 + \sum_{i=1}^N \frac{\sigma_{d_i}}{2N} d_i^2 + \lambda^T (-Wd + \gamma\delta 1_N) \\ &\quad + \mu^T (-d + \delta 1_N) \\ &= (-1 + \sum_{i=1}^N (\gamma\lambda_i + \mu_i))\delta + \frac{\sigma_\delta}{2} \delta^2 \\ &\quad + \sum_{i=1}^N \left( -\sum_{j=1}^N \lambda_j W_{ji} - \mu_i \right) d_i + \frac{\sigma_{d_i}}{2N} d_i^2. \end{aligned}$$

The Lagrangian is clearly separable in  $\delta$  and in  $d_i$ . The corresponding dual function is given by

$$\begin{aligned} \varphi(\lambda,\mu) &= \min_{\delta \in \mathbb{R}} \left\{ (-1 + \sum_{i=1}^N (\gamma\lambda_i + \mu_i))\delta + \frac{\sigma_\delta}{2} \delta^2 \right\} + \\ &\quad \sum_{i=1}^N \min_{d_i \in \mathbb{R}} \left\{ \left( -\sum_{j=1}^N \lambda_j W_{ji} - \mu_i \right) d_i + \frac{\sigma_{d_i}}{2N} d_i^2 \right\}, \quad (5) \end{aligned}$$

which can be evaluated in parallel.

Due to the uniqueness of the minimizer  $d_i(\lambda, \mu_i)$  and  $\delta(\lambda, \mu)$  in (5) it follows that the gradient of the concave dual function is given by (cf. Theorem 3.1 in [16])

$$\begin{aligned}\nabla_{\lambda_i} \varphi(\lambda, \mu) &= \nabla_{\lambda_i} \mathcal{L}(d(\lambda, \mu), \delta(\lambda, \mu), \lambda, \mu) \\ &= \gamma \delta(\lambda, \mu) - \sum_{j=1}^N W_{ij} d_j(\lambda, \mu_j), \\ \nabla_{\mu_i} \varphi(\lambda, \mu) &= \nabla_{\mu_i} \mathcal{L}(d(\lambda, \mu), \delta(\lambda, \mu), \lambda, \mu) \\ &= \delta(\lambda, \mu) - d_i(\lambda, \mu_i).\end{aligned}$$

Moreover, it can be shown that the gradient of the dual function is Lipschitz continuous with Lipschitz constant [16]

$$L = \sum_{i=1}^N \frac{N(\sum_{j=1}^N W_{ji}^2 + 1)}{\sigma_{d_i}} + \frac{N(1 + \gamma^2)}{\sigma_{\delta}}. \quad (6)$$

Finally, the DNA can be applied to maximize the dual function in parallel to obtain the following algorithm. Moreover, the DNA can be implemented with event-based communication to reduce the communication effort as detailed in [15], but for ease of presentation, this aspect is left out in this article.

**Algorithm 1.** For  $k \geq 0$  do in parallel:

1) Given  $\lambda^k$  and  $\mu^k$  compute

$$\begin{aligned}\delta^{k+1} &= \arg \min_{\delta \in \mathbb{R}} \left\{ (-1 + \sum_{i=1}^N (\gamma \lambda_i^k + \mu_i^k)) \delta + \frac{\sigma_{\delta}}{2} \delta^2 \right\}, \\ d_i^{k+1} &= \arg \min_{d_i \in \mathbb{R}} \left\{ - \left( \sum_{j=1}^N \lambda_j^k W_{ji} + \mu_i^k \right) d_i + \frac{\sigma_{d_i}}{2N} d_i^2 \right\},\end{aligned}$$

2) Compute

$$\begin{aligned}\nabla_{\lambda_i} \varphi(\lambda^k, \mu^k) &= \gamma \delta^{k+1} - \sum_{j=1}^N W_{ij} d_j^{k+1}, \\ \nabla_{\mu_i} \varphi(\lambda^k, \mu^k) &= -d_i^{k+1} + \delta^{k+1}.\end{aligned}$$

3) Find

$$\begin{aligned}\tilde{\lambda}_i^k &= \arg \max_{\lambda \geq 0} \left\{ \nabla_{\lambda_i} \varphi(\lambda^k, \mu^k) \lambda - \frac{L}{2} (\lambda - \lambda_i^k)^2 \right\}, \\ \tilde{\mu}_i^k &= \arg \max_{\mu \geq 0} \left\{ \nabla_{\mu_i} \varphi(\lambda^k, \mu^k) \mu - \frac{L}{2} (\mu - \mu_i^k)^2 \right\},\end{aligned}$$

4) Find

$$\begin{aligned}t_i^k &= \arg \max_{t \geq 0} \left\{ -\frac{L}{2} t^2 + \left( \sum_{j=0}^k \frac{j+1}{2} \nabla_{\lambda_i} \varphi(\lambda^j, \mu^j) \right) t \right\}, \\ v_i^k &= \arg \max_{v \geq 0} \left\{ -\frac{L}{2} v^2 + \left( \sum_{j=0}^k \frac{j+1}{2} \nabla_{\mu_i} \varphi(\lambda^j, \mu^j) \right) v \right\},\end{aligned}$$

5) Set

$$\begin{aligned}\lambda_i^{k+1} &= \frac{k+1}{k+3} \tilde{\lambda}_i^k + \frac{2}{k+3} t_i^k, \\ \mu_i^{k+1} &= \frac{k+1}{k+3} \tilde{\mu}_i^k + \frac{2}{k+3} v_i^k,\end{aligned}$$

Starting with  $(\lambda_0, \mu_0) = (0, 0)$  the following convergence result for Algorithm 1 holds.

**Theorem 3.** Taking  $k = \lceil \sqrt{8L/\varepsilon} \rceil - 1$  with  $\varepsilon > 0$  and Lipschitz constant  $L$  defined by (6), then after iteration  $k$  of Algorithm 1 an approximate solution to problem (4) is

$$(\hat{d}, \hat{\delta}) := \sum_{j=0}^k \frac{2(j+1)}{(k+1)(k+2)} (d^{j+1}, \delta^{j+1}),$$

which satisfies the following bounds on the primal gap

$$-\varepsilon \left\| \begin{pmatrix} \lambda^* \\ \mu^* \end{pmatrix} \right\|^2 \leq -\hat{\delta} + \frac{\sigma_{\delta}}{2} \hat{\delta}^2 + \sum_{i=1}^N \frac{\sigma_{d_i}}{2N} \hat{d}_i^2 - f^* \leq 0, \quad (7)$$

as well as the following bound on the constraint violation

$$\left\| \begin{bmatrix} -W\hat{d} + \gamma\hat{\delta}1_N \\ -\hat{d} + \hat{\delta}1_N \end{bmatrix}^+ \right\| \leq \varepsilon \left\| \begin{pmatrix} \lambda^* \\ \mu^* \end{pmatrix} \right\|, \quad (8)$$

where  $f^*$  is the optimal function value of problem (4) and  $[\cdot]^+$  is the componentwise projection onto the nonnegative real numbers. Moreover,  $\lambda^*$  and  $\mu^*$  are optimal dual multipliers, i.e., they maximize the concave dual function  $\varphi(\lambda, \mu)$  in (5).

*Proof.* For ease of notation we define

$$f(d, \delta) = -\delta + \frac{\sigma_{\delta}}{2} \delta^2 + \sum_{i=1}^N \frac{\sigma_{d_i}}{2} d_i^2,$$

$$W(d, \delta) = \begin{pmatrix} -Wd + \gamma\delta 1_N \\ -d + \delta 1_N \end{pmatrix}, \quad \Lambda = (\lambda^T, \mu^T)^T.$$

It can be shown that the following inequality holds (cf. Lemma 3.3 and Remark 3.8 in [16])

$$-\|\Lambda^*\| \left\| \begin{bmatrix} W(\hat{d}, \hat{\delta}) \end{bmatrix}^+ \right\| \leq f(\hat{d}, \hat{\delta}) - f^* \leq f(\hat{d}, \hat{\delta}) - \varphi(\hat{\lambda}, \hat{\mu}), \quad (9)$$

where  $(\hat{\lambda}, \hat{\mu}) := (\tilde{\lambda}^k, \tilde{\mu}^k)$ . Moreover, applying Theorem 3.4 in [16] we have

$$f(\hat{d}, \hat{\delta}) - \varphi(\hat{\lambda}, \hat{\mu}) \leq \min_{\Lambda \in \mathbb{R}^{2N}} \left\{ \frac{2L}{(k+1)^2} \|\Lambda\|^2 - \langle W(\hat{d}, \hat{\delta}), \Lambda \rangle \right\}.$$

It is straight forward to show that the optimal solution of the right-hand side of the above is obtained at  $\Lambda = (k+1)^2 / (4L) [W(\hat{d}, \hat{\delta})]^+$  and it follows that

$$f(\hat{d}, \hat{\delta}) - \varphi(\hat{\lambda}, \hat{\mu}) \leq -\frac{(k+1)^2}{8L} \left\| \begin{bmatrix} W(\hat{d}, \hat{\delta}) \end{bmatrix}^+ \right\|^2 \leq 0. \quad (10)$$

Combining (9) and (10) yields

$$\frac{(k+1)^2}{8L} \left\| \begin{bmatrix} W(\hat{d}, \hat{\delta}) \end{bmatrix}^+ \right\|^2 - \|\Lambda^*\| \left\| \begin{bmatrix} W(\hat{d}, \hat{\delta}) \end{bmatrix}^+ \right\| \leq 0$$

and inequality (8) follows immediately as well as (7).  $\square$

A similar proof for this convergence rate is given in [17].

As the norm of the optimal dual multipliers  $\lambda^*$  and  $\mu^*$  is not known beforehand, the lower bound on the primal gap in (7) and the upper bound on the constraint violation in (8) cannot be evaluated directly with the results from the distributed algorithm in order to decide if a system is connectively stable. Therefore, we propose the following algorithm.

**Algorithm 2.** 1) Choose a minimum accuracy parameter  $\varepsilon_{\min}$ , an initial accuracy parameter  $\varepsilon_0$  and run Algorithm 1. Set  $\varepsilon_1 = \alpha \varepsilon_0$ , where  $\alpha \in (0, 1)$ .

2) Rerun Algorithm 1.

3) If  $\varepsilon_i > \varepsilon_{\min}$ , set  $\varepsilon_{i+1} = \alpha \varepsilon_i$  and go back to step 2. Otherwise stop.

The subsystems then need to evaluate if the obtained series of objective function values of all steps converge to a negative value, or if a convergence to 0 occurs.

In other words, with Algorithm 2, the subsystems decide on connective stability if  $f(\hat{d}(\varepsilon), \hat{\delta}(\varepsilon)) \not\rightarrow 0$ . In this case the subsystems observe a consistent convergence to an optimal value for decreasing values of  $\varepsilon$ . If  $f(\hat{d}(\varepsilon), \hat{\delta}(\varepsilon)) \rightarrow 0$  for  $\varepsilon \rightarrow 0$ , i.e. no convergence to a value other than 0 is observed, no decision on connective stability can be made because of the known sufficiency of the condition.

**Remark 2.** Regarding the choice of convexity parameters  $\sigma_{d_i}$  and  $\sigma_{\delta}$ , we use that Algorithm 1 has a complexity of  $\mathcal{O}(\sqrt{L/\varepsilon})$  according to Theorem 3. It follows that large convexity parameter  $\sigma_{d_i}$  and  $\sigma_{\delta}$  on the one hand yield a small Lipschitz constant  $L$  defined by (6) which reduces the number of iterations. On the other hand, in this case the accuracy  $\varepsilon$  has to be chosen smaller, which raises the number of iterations, as the objective function value (4a) and therefore the primal gap moves closer to zero. It follows that there is a trade-off in the choice, but that the choice has little effect on the overall computational effort.

**Remark 3.** In parallel to the work presented in this paper, a new distributed dual gradient algorithm for linearly constrained separable problems with strongly convex objective function is published in [18] where a linear convergence rate is shown. This would be a good alternative to the algorithm that we use. However, we apply the DNA as the focus here is on the development of distributed stability tests. Furthermore, the DNA facilitates an event-triggered communication [15].

Finally, Algorithm 1 can be set up and run with limited model information according to Definition 1. That means that it is completely distributed with the same structure as the original dynamical system described by the graph  $\mathcal{G}_{I,u}$ , i.e. only neighboring subsystems need to communicate during the optimization. This follows immediately from the sparsity structure of  $W$  for the computation of  $d_i^{k+1}$  and steps 2-5 of Algorithm 1. The computation of  $\delta^{k+1}$  in step 1 of Algorithm 1, which is a globally shared variable, can be done with local communication too by using a consensus algorithm [19]: Each agent computes the term  $\gamma\lambda_i + \mu_i$ , then the average of all terms is determined with only local communication in a consensus phase and knowing the size of the network  $N$  each agent obtains the sum over all individual terms and can compute  $\delta^{k+1}$  himself. In the same way, the approximate objective function value  $f(\hat{d}(\varepsilon), \hat{\delta}(\varepsilon))$  and the Lipschitz constant  $L$  defined in (6) can be computed with only local communication if the network size is known (for details we refer to [19]). Finally, the total number of subsystems is no sensitive information and does not violate privacy. Concluding this section, we now have a method to test connective stability of System (2) using limited model information.

#### IV. DISTRIBUTED LYAPUNOV STABILITY ANALYSIS

In this section we present an alternative method for testing stability of large-scale systems based on the Lyapunov linear matrix inequality. It is well known that system (2) is asymptotically stable if and only if there exists a  $P \in \mathbb{S}^n$  with  $P \succ 0$  such that the Lyapunov linear matrix inequality

$$A^T P + P A \prec 0 \quad (11)$$

is satisfied. If  $P$  is a full matrix (11) has no distributed structure. Therefore, we make the following assumption.

**Assumption 2.** The solution  $P$  of (11) is restricted to be block-diagonal, where the block sizes of  $P$  are determined by the subsystem sizes  $n_i$  in (1).

Assumption 2 restricts (11) to a sufficient condition. We confine our analysis to this sufficient condition because it maintains the structure of the original system. Hence, Assumption 2 enables us to apply distributed optimization, which in turn allows us to test if system (2) is asymptotically stable using limited model information according to Definition 1.

In the following we will consider a problem with a sparsity structure induced by  $A^{\text{sym}} = (A^T \text{diag}(P^1, \dots, P^N) + \text{diag}(P^1, \dots, P^N)A)$  for any  $P^i \in \mathbb{S}^{n_i}$ . The block sparsity structure is described by the undirected interaction graph  $\mathcal{G}_{I,u}$ , while the element sparsity structure is described by the graph  $\mathcal{G}_u = (\mathcal{V}_u, \mathcal{E}_u)$ , where  $(j, i) \in \mathcal{E}_u$  iff the element  $A_{ij}^{\text{sym}} \neq 0$ .

**Remark 4.**  $\mathcal{G}_u$  has the same structure as  $\mathcal{G}_{I,u}$  if the nodes of  $\mathcal{G}_u$  that belong to the same subsystem are merged to a subsystem node. It follows that any distributed algorithm with a communication topology that equals the topology of  $\mathcal{G}_u$  uses only limited model information as defined in Definition 1.

In the following we will use a decomposition method for LMIs such that we can apply distributed optimization methods to check Lyapunov stability. In order for the decomposition method to work, we have to make the following assumption.

**Assumption 3.**  $\mathcal{G}_u$  is a chordal graph.

A chordal graph is defined to be a graph where every cycle of length  $\geq 4$  has a chord, i.e. an edge joining non-consecutive vertices of the cycle [20]. If  $\mathcal{G}_u$  does not satisfy this assumption, it is possible to chordalize  $\mathcal{G}_u$  in polynomial time [21] but then local communication with neighbors is not sufficient for the presented stability test to be applicable. In the following we check condition (11) with a block-diagonal  $P$  in a distributed way. To this end we use the same idea as in Section III by formulating an optimization problem and then applying distributed optimization methods. We start with the following problem with convexity parameters  $\sigma_{\delta}$  and  $\sigma_{P^l}$ .

$$\min_{\delta \in \mathbb{R}, P^l \in \mathbb{S}^{n_l}} f(\delta, P^l) = -\delta + \frac{\sigma_{\delta}}{2} \delta^2 + \sum_{l=1}^N \frac{\sigma_{P^l}}{2N} \|P^l\|_F^2 \quad (12a)$$

$$\text{s.t. } F(P, \delta) \succeq 0, \quad (12b)$$

$$P^l - \delta I_{n_l} \succeq 0 \text{ for } l = 1, \dots, N, \quad (12c)$$

where

$$F(P, \delta) := -A^T \text{diag}(P^1, \dots, P^N) - \text{diag}(P^1, \dots, P^N)A - \gamma \delta I_n,$$

and  $\gamma$  is arbitrary. The variable  $\delta$  is used to make the problem feasible, independent of the stability of the system. The optimal function value of problem (12) is defined as  $f^*$ .

**Theorem 4.** The inequality (11) holds for some  $P = \tilde{P} \in \mathbb{S}^n$  with  $\tilde{P} = \text{diag}(\tilde{P}^1, \dots, \tilde{P}^N)$  if and only if there exists a feasible point  $(P^1, \dots, P^N, \delta)$  for (12) with  $f^* < 0$ .

*Proof.* The proof is identical to the proof of Theorem 2 because the overall problem formulation is identical and the

objective function of (12) is also convex. Additionally, as in the proof of Theorem 2  $f^* < 0$  implies that  $\delta^* > 0$ .  $\square$

Following the approach in [22], where a semidefinite dual of the AC-OPF problem is solved distributedly, we apply the *range-space conversion method* [21] to decompose the LMI (12b). This allows the application of the distributed dual decomposition scheme presented in the previous section.

To this end, let  $\mathcal{N} = \{1, \dots, n\}$  and consider the *r-sparsity pattern* [21] of constraint (12b), which is

$$\text{SP}_{\text{LMI}} = \{(i, j) \in \mathcal{N} \times \mathcal{N} : F_{ij}(P, \delta) \neq 0$$

for some  $(P^1, \dots, P^N, \delta) \in \mathbb{S}^{n_1} \times \dots \times \mathbb{S}^{n_N} \times \mathbb{R}, i \neq j\}$ .

We use the following definitions that can be found in [21]: Let  $F \subseteq \mathcal{N} \times \mathcal{N}$  and define

$$F^\bullet = F \cup \{(i, i) : i \in \mathcal{N}\},$$

$$\mathbb{S}^n(F, 0) = \{X \in \mathbb{S}^n : X_{ij} = 0 \text{ if } (i, j) \notin F^\bullet\},$$

$$\mathbb{S}^C = \{X \in \mathbb{S}^n : X_{ij} = 0 \text{ if } (i, j) \notin C \times C\} \forall C \subseteq \mathcal{N},$$

$$\mathbb{S}_+^C = \{X \in \mathbb{S}^C : X \succeq 0\} \forall C \subseteq \mathcal{N},$$

$$J(C) = \{(i, j) \in C \times C : i \leq j\} \forall C \subseteq \mathcal{N}.$$

One has  $F(P, \delta) \subseteq \mathbb{S}^n(\text{SP}_{\text{LMI}}, 0)$  for all  $(P^1, \dots, P^N, \delta)$  and obviously the sparsity structure of  $\mathbb{S}^n(\text{SP}_{\text{LMI}}, 0)$  coincides with the sparsity structure of the adjacency matrix of  $\mathcal{G}_u$ .

Consider the maximal cliques  $C_1, \dots, C_p$  of  $\mathcal{G}_u$  and denote by  $E_{ij}$  the  $n \times n$  symmetric matrix whose components  $(i, j)$  and  $(j, i)$  are 1 and all others are 0. Obviously, the set  $\{E_{ij} : (i, j) \in \mathcal{N} \times \mathcal{N}, i \leq j\}$  is a basis of  $\mathbb{S}^n$ . Defining the sets [21]

$$J = \bigcup_{s=1}^p J(C_s) \quad \text{and} \quad \Gamma(i, j) = \{s : i \in C_s, j \in C_s\} \forall (i, j) \in J,$$

the LMI (12b) is equivalent to

$$E_{ij} \bullet \sum_{s \in \Gamma(i, j)} W^s - E_{ij} \bullet F(P, \delta) = 0 \quad (13)$$

for  $(i, j) \in J$  and  $W^s \in \mathbb{S}_+^{C_s}$  for  $s = 1, \dots, p$ . For details we refer to Section 5.2 in [21]. It follows that problem (12) can be written as

$$\min_{\delta \in \mathbb{R}, P^l \in \mathbb{S}^{n_l}} -\delta + \frac{\sigma_\delta}{2} \delta^2 + \sum_{l=1}^N \frac{\sigma_{P^l}}{2N} \|P^l\|_F^2 \quad (14a)$$

$$E_{ij} \bullet \sum_{s \in \Gamma(i, j)} W^s - E_{ij} \bullet F(P, \delta) = 0 \text{ for } (i, j) \in J, \quad (14b)$$

$$W^s \in \mathbb{S}_+^{C_s} \text{ for } s = 1, \dots, p, \quad (14c)$$

$$\delta I_{n_l} - P^l \preceq 0 \text{ for } l = 1, \dots, N. \quad (14d)$$

To obtain a strongly convex objective function for (14) which guarantees the differentiability of the gradient of the corresponding dual objective function, we modify (14) to

$$\min_{\delta \in \mathbb{R}, P^l \in \mathbb{S}^{n_l}} -\delta + \frac{\sigma_\delta}{2} \delta^2 + \sum_{l=1}^N \frac{\sigma_{P^l}}{2N} \|P^l\|_F^2 + \sum_{s=1}^p \frac{\sigma_{W^s}}{2p} \|W^s\|_F^2 \quad (15a)$$

$$E_{ij} \bullet \sum_{s \in \Gamma(i, j)} W^s - E_{ij} \bullet F(P, \delta) = 0 \text{ for } (i, j) \in J, \quad (15b)$$

$$W^s \in \mathbb{S}_+^{C_s} \text{ for } s = 1, \dots, p, \quad (15c)$$

$$\delta I_{n_l} - P^l \preceq 0 \text{ for } l = 1, \dots, N. \quad (15d)$$

**Theorem 5.** *The inequality (11) holds for some  $P = \tilde{P} \in \mathbb{S}^n$  with  $\tilde{P} = \text{diag}(\tilde{P}^1, \dots, \tilde{P}^N)$  if and only if there exists a feasible point  $(P^1, \dots, P^N, \delta, W^1, \dots, W^p)$  for (15) with  $f^* < 0$ , where  $f^*$  denotes the optimal function value of problem (15).*

*Proof.* The proof is similar to the proof of Theorem 2 and is left out due to space constraints. The interested reader is referred to the extended online version of this article [23].  $\square$

Finally, we rewrite  $F(P, \delta)$  in a way that allows a decomposition. To this end, let  $B_l$  be defined as

$$B_l = \left\{ \sum_{i=1}^{l-1} n_i + 1, \dots, \sum_{i=1}^l n_i \right\} \times \left\{ \sum_{i=1}^{l-1} n_i + 1, \dots, \sum_{i=1}^l n_i \right\}.$$

For  $l = 1, \dots, N$  and  $(i, j) \in B_l$  we define

$$F^0 = -\gamma I_n,$$

$$F_{ij}^l = \begin{cases} \frac{1}{2}(-A^T E_{ij} - E_{ij} A) & \text{if } i < j, \\ \frac{1}{2}(-A^T E_{ji} - E_{ji} A) & \text{if } i > j, \\ -A^T E_{ij} - E_{ij} A & \text{if } i = j, \end{cases}$$

and with  $i_l := i - \sum_{s=1}^{l-1} n_s$ , we can rewrite  $F(P, \delta)$  as

$$F(P, \delta) = F^0 \delta + \sum_{l=1}^N \sum_{(i, j) \in B_l} F_{ij}^l P_{i_l j_l}^l.$$

To solve problem (15) in parallel we again employ the DNA in combination with dual decomposition. To this end, consider the following Lagrangian of problem (15)

$$\mathcal{L}(\delta, P, W, \Lambda, M)$$

$$= -\delta + \frac{\sigma_\delta}{2} \delta^2 + \sum_{l=1}^N \frac{\sigma_{P^l}}{2N} \|P^l\|_F^2 + \sum_{s=1}^p \frac{\sigma_{W^s}}{2p} \|W^s\|_F^2$$

$$+ \sum_{(i, j) \in J} \Lambda_{ij} \left( E_{ij} \bullet \sum_{s \in \Gamma(i, j)} W^s - E_{ij} \bullet F(P, \delta) \right)$$

$$+ \sum_{l=1}^N M^l \bullet (\delta I_{n_l} - P^l)$$

$$= - \underbrace{\left( \sum_{(i, j) \in J} \Lambda_{ij} E_{ij} \bullet F^0 + 1 - \sum_{l=1}^N M^l \bullet I_{n_l} \right)}_{x_\delta} \delta + \frac{\sigma_\delta}{2} \delta^2 +$$

$$\sum_{l=1}^N \left[ \sum_{(i, j) \in B_l} - \underbrace{\left( \sum_{(a, b) \in J} \Lambda_{ab} E_{ab} \bullet F_{ij}^l + M_{i_l j_l}^l \right)}_{=X_{P_{i_l j_l}}^l} P_{i_l j_l}^l + \frac{\sigma_{P^l}}{2N} \|P^l\|_F^2 \right]$$

$$+ \sum_{s=1}^p \left[ - \underbrace{\sum_{(i, j) \in J(C_s)} \Lambda_{ij} E_{ij} \bullet W^s}_{X_W^s} + \frac{\sigma_{W^s}}{2p} \|W^s\|_F^2 \right]$$

$$= -x_\delta \delta + \frac{\sigma_\delta}{2} \delta^2 + \sum_{l=1}^N \left[ -X_P^l \bullet P^l + \frac{\sigma_{P^l}}{2N} \|P^l\|_F^2 \right]$$

$$+ \sum_{s=1}^p \left[ -X_W^s \bullet W^s + \frac{\sigma_{W^s}}{2p} \|W^s\|_F^2 \right],$$

which is separable in  $\delta$ ,  $P^1, \dots, P^N$ , and  $W^1, \dots, W^p$ . The corresponding dual function is

$$\begin{aligned} \varphi(\Lambda, M) &= \min_{\delta \in \mathbb{R}, P^l \in \mathbb{S}^{n_l}, W^s \in \mathbb{S}_+^{C_s}} \mathcal{L}(\delta, P, W, \Lambda, M) \\ &= \min_{\delta \in \mathbb{R}} \left\{ -x_\delta \delta + \frac{\sigma_\delta}{2} \delta^2 \right\} \\ &\quad + \sum_{l=1}^N \min_{P^l \in \mathbb{S}^{n_l}} \left\{ -X_P^l \bullet P^l + \frac{\sigma_{P^l}}{2N} \|P^l\|_F^2 \right\} \\ &\quad + \sum_{s=1}^p \min_{W^s \in \mathbb{S}_+^{C_s}} \left[ -X_W^s \bullet W^s + \frac{\sigma_{W^s}}{2p} \|W^s\|_F^2 \right] \\ &= -x_\delta \delta(\Lambda, M) + \frac{\sigma_\delta}{2} \delta(\Lambda, M)^2 \\ &\quad + \sum_{l=1}^N \left[ -X_P^l \bullet P^l(\Lambda, M) + \frac{\sigma_{P^l}}{2N} \|P^l(\Lambda, M)\|_F^2 \right] \\ &\quad + \sum_{s=1}^p \left[ -X_W^s \bullet W^s(\Lambda, M) + \frac{\sigma_{W^s}}{2p} \|W^s(\Lambda, M)\|_F^2 \right], \end{aligned}$$

where  $\delta(\Lambda, M)$ ,  $P^l(\Lambda, M)$ , and  $W^s(\Lambda, M)$  are the unique solutions. As before  $\varphi(\Lambda, M)$  can be evaluated in parallel and is continuously differentiable due to the uniqueness of the solutions  $\delta(\Lambda, M)$ ,  $P^l(\Lambda, M)$ , and  $W^s(\Lambda, M)$ . Moreover, the gradient of  $\varphi(\Lambda, M)$  with

$$\begin{aligned} \nabla_{\Lambda_{ij}} \varphi(\Lambda, M) &= E_{ij} \bullet \sum_{s \in \Gamma(i, j)} W^s(\Lambda, M) \\ &\quad - E_{ij} \bullet \left( F^0 \delta(\Lambda, M) + \sum_{l=1}^N \sum_{(i, j) \in B_l} F_{ij}^l P_{ij}^l(\Lambda, M) \right) \text{ for } (i, j) \in J \end{aligned}$$

and

$$\nabla_{M^l} \varphi(\Lambda, M) = \delta(\Lambda, M) I_{n_l} - P^l(\Lambda, M) \text{ for } l = 1, \dots, N,$$

is again Lipschitz continuous with Lipschitz constant

$$\begin{aligned} L &= \sum_{s=1}^p p \|E_{C_s}\|^2 / \sigma_{W^s} + \sum_{l=1}^N N \left( \|\hat{F}^l\|^2 + 1 \right) / \sigma_{P^l} \\ &\quad + \left( \sum_{(i, j) \in J} (E_{ij} \bullet F^0)^2 + n \right) / \sigma_\delta, \end{aligned} \quad (16)$$

where  $E_{C_s} \in \mathbb{R}^{|J(C_s)| \times n^2}$  is the matrix that contains the rows  $E_{ij}(\cdot)^T$  for  $(i, j) \in J(C_s)$  and  $\hat{F}^l \in \mathbb{R}^{|J| \times n_l^2}$  is the matrix that contains rows  $(E_{ab} \bullet F_{i_1 j_1}^l, \dots, E_{ab} \bullet F_{i_{|B_l|} j_{|B_l|}}^l)$  for  $(a, b) \in J$ . Finally, the DNA can be applied to maximize the augmented dual function in parallel. Again, for ease of presentation the aspect of event-based communication is not presented but we refer to [15] for details.

**Algorithm 3** (Distributed Lyapunov stability test).

For  $k \geq 0$  do

1) Given the necessary  $M^{l,k}$  and components  $\Lambda_{ij}^k$ , each subsystem computes

$$\delta^{k+1} = \arg \min_{\delta \in \mathbb{R}} \left\{ -x_\delta \delta + \frac{\sigma_\delta}{2} \delta^2 \right\}$$

using a consensus algorithm to determine  $x_\delta$  with local communication as described in Section III. Furthermore,

the subsystems compute in parallel:

$$\begin{aligned} P^{l,k+1} &= \arg \min_{P^l \in \mathbb{S}^{n_l}} \left\{ -X_P^l \bullet P^l + \frac{\sigma_{P^l}}{2N} \|P^l\|_F^2 \right\}, \\ W^{s,k+1} &= \arg \min_{W^s \in \mathbb{S}_+^{C_s}} \left\{ -X_W^s \bullet W^s + \frac{\sigma_{W^s}}{2p} \|W^s\|_F^2 \right\}, \end{aligned}$$

for  $l = 1, \dots, N$ ,  $s = 1, \dots, p$ , and send  $P^{l,k+1}$ , and  $W^{s,k+1}$  to their neighbors.

For  $(i, j) \in J$  and  $l = 1, \dots, N$ , the subsystems do in parallel:

2) Given  $\delta^{k+1}$ ,  $P^{l,k+1}$ , and  $W^{s,k+1}$  compute

$$\begin{aligned} \nabla_{\Lambda_{ij}} \varphi(\Lambda^k, M^k) &= E_{ij} \bullet \sum_{s \in \Gamma(i, j)} W^{s,k+1} \\ &\quad - E_{ij} \bullet \left( F^0 \delta^{k+1} + \sum_{l=1}^N \sum_{(i, j) \in B_l} F_{ij}^l P_{ij}^{l,k+1} \right), \\ \nabla_{M^l} \varphi(\Lambda^k, M^k) &= \delta^{k+1} I_{n_l} - P^{l,k+1}. \end{aligned}$$

3) Find  $Y_{ij}^k = \arg \max_{Y_{ij} \in \mathbb{R}} \left\{ \nabla_{\Lambda_{ij}} \varphi(\Lambda^k, M^k) Y_{ij} - \frac{L}{2} (Y_{ij} - \Lambda_{ij}^k)^2 \right\}$ ,

$$H^{l,k} = \arg \max_{H^l \in \mathbb{S}_+^{n_l}} \left\{ \nabla_{M^l} \varphi(\Lambda^k, M^k) \bullet H^l - \frac{L}{2} \|H^l - M^{l,k}\|_F^2 \right\}.$$

4) Find  $Z_{ij}^k = \arg \max_{Z_{ij} \in \mathbb{R}} \left\{ -\frac{L}{2} Z_{ij}^2 + \sum_{j=0}^k \frac{j+1}{2} \nabla_{\Lambda_{ij}} \varphi(\Lambda^j, M^j) Z_{ij} \right\}$ ,

$$T^{l,k} = \arg \max_{T^l \in \mathbb{S}_+^{n_l}} \left\{ -\frac{L}{2} \|T^l\|_F^2 + \sum_{j=0}^k \frac{j+1}{2} \nabla_{M^l} \varphi(\Lambda^j, M^j) \bullet T^l \right\}$$

5) Set  $\Lambda_{ij}^{k+1} = \frac{k+1}{k+3} Y_{ij}^k + \frac{2}{k+3} Z_{ij}^k$ ,

$$M^{l,k+1} = \frac{k+1}{k+3} H^{l,k} + \frac{2}{k+3} T^{l,k}.$$

6) Send  $\Lambda_{ij}^k$  and  $M^{l,k}$  to the neighbors.

A convergence result analogous to Theorem 3 can be made for Algorithm 3 but is left out due to space restrictions. The interested reader is referred to the extended version of this article [23]. Algorithm 2 now uses Algorithm 3 instead of Algorithm 1.

Due to the definition of  $X_P^l$  and  $X_W^s$  in step 1 of Algorithm 3 and the definition of  $\nabla_{\Lambda_{ij}} \varphi(\Lambda^k, M^k)$  in step 2, it follows that the communication topology of Algorithm 3 equals the communication topology of  $\mathcal{G}_u$  as we assumed  $\mathcal{G}_u$  to be chordal (Assumption 3). Hence, Algorithm 3 uses only limited model information according to Definition 1.

Moreover, all subproblems in the above algorithm have closed form solutions. For example, we derive the closed form solution for  $W^{s,k+1}$  in step 1: Consider the spectral decomposition of the symmetric matrix  $X_W^s$

$$X_W^s = Q \Sigma Q^T = (Q_+, Q_-) \begin{pmatrix} \Sigma_+ & 0 \\ 0 & \Sigma_- \end{pmatrix} \begin{pmatrix} Q_+^T \\ Q_-^T \end{pmatrix},$$

where  $\Sigma_+$  contains the non-negative eigenvalues of  $X_W^s$ . It follows that the optimal solution  $W^{s,k+1}$  is the projection on the positive semidefinite part of  $X_W^s$

$$W^{s,k+1} = \frac{Q_+ \Sigma_+ Q_+^T}{\sigma_{W^s}}.$$

Identically, solutions for  $H^{l,k}$  and  $T^{l,k}$  can be obtained.

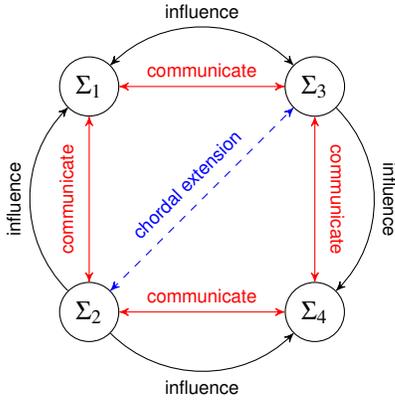


Fig. 1: Communication topology for the two stability tests: M-Matrix test (red, solid), additional links for Lyapunov test (blue, dashed) because of chordal extension.

#### A. Discussion of the two tests

In this subsection, we discuss the two tests with respect to differences, and respective advantages and disadvantages. For the sake of a clearer presentation we do not take privacy considerations or distributed computations into account here.

Both tests have in common that they use a block-diagonal quadratic Lyapunov function. The M-matrix Lyapunov function has the form  $V_M = \sum_{i=1}^N d_i x_i^T (T_i^T)^{-1} T_i^{-1} x_i$  while the Lyapunov inequality approach has the Lyapunov function  $V_L = \sum_{i=1}^N x_i^T P_i x_i$ . Since both are restricted to a block-diagonal form, both represent only sufficient conditions. Both also have in common that the summands need to represent Lyapunov functions of the individual subsystems. However, by comparing the structure of the Lyapunov functions  $V_M$  and  $V_L$  one can see that  $V_M$  has only  $N$  degrees of freedom since the transformation matrices  $T_i$  are fixed. On the other hand,  $V_L$  has  $\sum_{i=1}^N \frac{n_i(n_i+1)}{2}$  degrees of freedom, which reduces conservatism except for the special case that all systems are scalar. So the M-matrix condition is more conservative because it evaluates the stability of the overall system using scalar (worst-case) approximations of the individual subsystems. In fact, the M-matrix test can be viewed as a special case of the block-diagonal quadratic Lyapunov condition. It requires diagonal stability of the transformed system  $\tilde{A}$  that is comprised of the blocks  $\tilde{A}_{ij}$  given in Eq. (3), but using only  $N$  parameters on the diagonal instead of  $n$ . So clearly the M-matrix test is more conservative than the Lyapunov test but numerical investigations in Section V will show that it has computational advantages. However, the conservativeness of the M-matrix test is not introduced for the sake of the distributed solution. For the Lyapunov test, the conservativeness stems from the block-diagonal restriction needed for the distributed solution.

Furthermore, the Lyapunov test has the disadvantage that the communication topology needs to be a chordal graph, so there may be a need for additional communication links instead of only the ones to the physical neighbors. This is visualized in Figure 1 for a small example.

The conservativeness is illustrated with an example. The A-

matrix of a system with two subsystems is given by

$$A = \left[ \begin{array}{cc|cc} -1 & 0 & \frac{1}{2} & \frac{1}{10} \\ 0 & -\frac{1}{10} & -1 & -\frac{1}{2} \\ \hline 1 & 1 & -\frac{1}{2} & 0 \\ -\frac{3}{2} & \frac{1}{3} & 0 & -1 \end{array} \right].$$

All eigenvalues of  $A$  have negative real parts so the system is asymptotically stable, and the interconnection topology is chordal. The test matrix  $W$  according to (3) is  $W = \begin{bmatrix} 0.1640 & -1.1515 \\ -1.9234 & 0.5422 \end{bmatrix}$ . One eigenvalue of  $W$  has negative real part so the matrix is not an M-matrix. However, it is possible to determine a block-diagonal Lyapunov matrix  $P$  as

$$P = \begin{bmatrix} 49.2348 & 12.1610 & 0 & 0 \\ 12.1610 & 40.7617 & 0 & 0 \\ 0 & 0 & 35.3606 & 9.5783 \\ 0 & 0 & 9.5783 & 17.4666 \end{bmatrix}.$$

The matrix is positive definite and satisfies (11) showing asymptotic stability.

#### V. NUMERICAL SIMULATIONS

In this section the two presented approaches of testing stability are evaluated. In the first subsection both distributed algorithms are applied to test systems that satisfy both conditions, while in the second subsection only one condition is satisfied to show the reduced conservativeness. Finally, the tests are applied to a power system.

##### A. Systems satisfying the M-matrix condition

For the M-matrix test from Section III, 100 asymptotically stable systems with  $N = 25$  subsystems are randomly created, with  $n_i = 2\forall i$ , and with connection probability between subsystems of 0.1. The created systems have the following properties: The minimal real parts of the eigenvalues of  $A$  are between  $-50.6$  and  $-46.6$ , and the maximal ones are between  $-7.4$  and  $-0.001$ . There are 82 to 190 directed edges (on average 133). Also they are created such that they satisfy Assumption 3 necessary to apply the Lyapunov stability test.

To validate the results of the distributed Algorithm 1 we compare the results of our distributed algorithm with a centralized implementation in Yalmip [24]. The comparison with Yalmip is not in terms of efficiency but only serves as validation with regards to the final obtained values, because the goals of our distributed algorithm and a centralized implementation are different in terms of privacy vs. efficiency. The convexity parameters are chosen as  $\sigma_\delta = 10^{-3}$ ,  $\sigma_{d_i} = 10^{-3}$ . The values of the cost function obtained with Yalmip lie between  $-18$  and  $-4.78 \cdot 10^{-4}$ . The approximate result from the distributed algorithm is always smaller than the Yalmip result, as expected by Eq. (7). Using Algorithm 2 with  $\epsilon_0 = 10^{-1}$ ,  $\epsilon_{\min} = 10^{-5}$  and  $\alpha = 0.1$ , we obtain the objective function differences between Yalmip and Algorithm 1 that are summarized in Table I with minimum, maximum and mean values. They show that the approximated solution given by Algorithm 1 improves with decreasing  $\epsilon$ , as expected. Furthermore, the numerical computations show that with decreasing  $\epsilon$ , convergence to negative objective function is observed in all 100 cases. Hence, all systems are connectively stable by Theorem 2.

The systems are also analyzed with the distributed Lyapunov stability test. We first compare the results of optimization

TABLE I: Difference in objective function value between Yalmip and Algorithm 1: Section V-A

$\epsilon$	$\min( f_\epsilon^* - f_{\text{Yalmip}}^* )$	$\max( f_\epsilon^* - f_{\text{Yalmip}}^* )$	$\text{mean}( f_\epsilon^* - f_{\text{Yalmip}}^* )$
$10^{-1}$	0.0041	0.090	0.012
$10^{-2}$	$4.08 \cdot 10^{-4}$	0.0089	0.0012
$10^{-3}$	$4.08 \cdot 10^{-5}$	$8.91 \cdot 10^{-4}$	$1.16 \cdot 10^{-4}$
$10^{-4}$	$4.13 \cdot 10^{-6}$	$8.89 \cdot 10^{-5}$	$1.17 \cdot 10^{-5}$
$10^{-5}$	$4.5 \cdot 10^{-7}$	$8.90 \cdot 10^{-6}$	$1.19 \cdot 10^{-6}$

TABLE II: Difference in objective function value between Yalmip and Algorithm 3: Section V-A

$\epsilon$	$\min( f_\epsilon^* - f_{\text{Yalmip}}^* )$	$\max( f_\epsilon^* - f_{\text{Yalmip}}^* )$	$\text{mean}( f_\epsilon^* - f_{\text{Yalmip}}^* )$
$10^{-1}$	0.0019	0.041	0.0033
$10^{-2}$	$1.86 \cdot 10^{-4}$	0.0048	$3.89 \cdot 10^{-4}$
$10^{-3}$	$1.92 \cdot 10^{-5}$	$7.58 \cdot 10^{-4}$	$8.57 \cdot 10^{-5}$

problem (15) solved with Yalmip with the results from Algorithm 3. The convexity parameters are set to  $\sigma_\delta = 10^{-3}$ ,  $\sigma_{pl} = 10^{-3}$ ,  $\sigma_{Ws} = 10^{-5}$ .

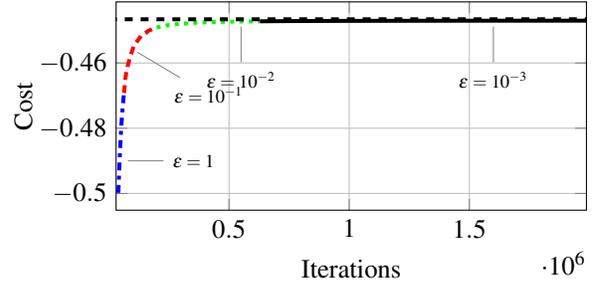
Using Algorithm 2 with  $\epsilon_0 = 10^{-1}$ ,  $\epsilon_{\min} = 10^{-3}$ , and  $\alpha = 0.1$  the differences in Table II are obtained, indicating that Algorithm 3 approximates the optimal solution well. Also, convergence to negative cost function values is observed for all 100 systems and hence, we conclude stability by Theorem 4. In conclusion, both methods identify all 100 stable systems correctly.

Furthermore, the numerical effort of both tests is compared. For the same accuracy  $\epsilon = 10^{-3}$ , the M-matrix test requires between 173956 and 283095 iterations while the Lyapunov test requires between 1736306 and 2385591. In addition to the higher number of iterations, the individual iterations of the Lyapunov test are more costly than the M-matrix test because they involve matrix operations. The solution of LMIs that is part of the Lyapunov test is inherently more costly than the linear program structure of the M-matrix test. Also the overall size of the problem is clearly smaller for the M-matrix test because it uses a scalar approximation for every subsystem while the Lyapunov test works with the complete model.

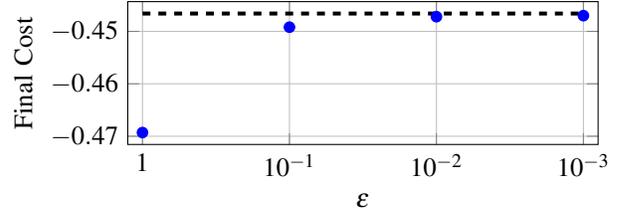
To visualize the principle behavior of the optimization algorithm and the influence of  $\epsilon$ , the cost evolution for one example system is shown in Figure 2a where the Lyapunov stability test is applied. It can be seen what the gain of the additional number of iterations caused by a smaller value of  $\epsilon$  is. The evolution starts with a negative value and then approaches the optimal value from below. In Figure 2b we see the decrease in distance from the actual optimal value which is always smaller than the respective  $\epsilon$ .

### B. Systems violating the M-matrix condition

In Section V-A it is shown that the Lyapunov stability test also identifies all systems to be asymptotically stable that satisfy the M-matrix condition. In this subsection, again 100 systems are randomly created, each with  $N = 25$  subsystems and  $n_i = 2\forall i$ , but in such a way such that they fail the M-Matrix condition but satisfy the Lyapunov condition. The minimal real parts of the eigenvalues of  $A$  lie between  $-22.03$  and  $-18.6$ , the maximal ones between  $-2.6$  and  $-0.03$ , and there are 78 to 197 directed interconnections. Also they are created such that satisfy Assumption 3.



(a) Cost evolution for different  $\epsilon$



(b) Final cost for different  $\epsilon$

Fig. 2: Algorithm behavior for different values of  $\epsilon$ : Final optimal value is  $-0.4466$  in black (dashed)

Employing Algorithm 2 with  $\epsilon_0 = 10^{-1}$ ,  $\epsilon_{\min} = 10^{-3}$  and  $\alpha = 0.1$ , the cost difference between Yalmip (values between  $-3.3$  and  $-0.068$ ) and Algorithm 3 are given in Table III. Also this procedure indicates convergence to negative cost function values in all 100 cases. Hence we conclude asymptotic stability by Theorem 4.

When the M-Matrix test is applied to the systems using Yalmip, the objective value is between  $1.4 \cdot 10^{-13}$  and  $1.76 \cdot 10^{-11}$ , i.e. the systems do not satisfy the condition for connective stability. When applying Algorithm 1 with Algorithm 2, the obtained objective functions become closer and closer to zero with decreasing  $\epsilon$  ( $10^{-3} \rightarrow 10^{-6}$ ). While one cannot guarantee that the condition may be satisfied for even smaller values of  $\epsilon$ , the subsystems have to choose an appropriate  $\epsilon_{\min}$  to make their decision. Summarizing the two subsections, the Lyapunov stability test identifies a larger class of systems at a higher numerical effort. This trade-off is summarized in Table IV. Reducing the number of iterations is the subject of current work and involves the application of an adaptive-step size rule for the DNA as proposed in [22]. Qualitatively, the advantage of the M-matrix test in terms of numerical effort is expected to remain.

Naturally, there are systems that violate both the M-matrix condition, and the block-diagonal Lyapunov stability condition. But the block-diagonal Lyapunov stability condition is clearly less conservative than the M-matrix condition at the cost of an increased numerical cost. On the other hand, the M-matrix condition establishes connective stability, i.e. robust stability. In practice, it is recommended to first check the simple M-matrix condition and only check the Lyapunov condition if the M-matrix condition fails.

### C. Application to 30 bus power system

In this subsection we apply the presented tests to a power system model. The dynamics of each subsystem follow [25]

TABLE III: Difference in objective function value between Yalmip and Algorithm 3: Section V-B

$\varepsilon$	$\min( f_\varepsilon^* - f_{\text{Yalmip}}^* )$	$\max( f_\varepsilon^* - f_{\text{Yalmip}}^* )$	$\text{mean}( f_\varepsilon^* - f_{\text{Yalmip}}^* )$
$10^{-1}$	0.0017	0.0376	0.0076
$10^{-2}$	$1.89 \cdot 10^{-4}$	0.0038	0.0011
$10^{-3}$	$2.22 \cdot 10^{-5}$	0.0035	$4.86 \cdot 10^{-4}$

TABLE IV: Comparison between the M-Matrix test and the Lyapunov test: The Lyapunov test is clearly less conservative at a larger computational effort

	M-Matrix test	Lyapunov Test
# identified stable systems out of 200	100	200
Average # iterations for $\varepsilon = 10^{-3}$	156468	1751734

$$\begin{aligned} \dot{\delta}_i(t) &= 2\pi f_i(t) \\ \dot{f}_i(t) &= -\frac{f_i(t)}{T_{P_i}} - \frac{K_{P_i}}{2\pi T_{P_i}} \left( \sum_{j \in \mathcal{N}_{in,i}} K_{S_{ij}} [\delta_i - \delta_j] \right) + \frac{K_{P_i} P_{g_i}}{T_{P_i}}, \end{aligned}$$

where  $\delta_i$  is the phase angle,  $f_i$  the frequency,  $P_{g_i}$  the generator output (input to the system),  $T_{P_i}$  the system model time constant,  $K_{P_i}$  the system gain and  $K_{S_{ij}}$  the synchronizing coefficient of the tie-line between the  $i$ th and the  $j$ th area. The parameters for all areas are identical with  $T_{P_i} = 25$ ,  $K_{P_i} = 100$ ,  $K_{S_{ij}} = 0.5$ . Since the original system is only marginally stable, we use a structured feedback for stabilization [26].

The interconnection topology is taken from the IEEE 30 bus test case [27]. The interconnection topology is not a chordal graph so for the Lyapunov stability test we have to use the chordal extension of the graph.

Eventually, stability cannot be shown using the M-matrix condition. Applying the Lyapunov stability test, the value of the cost obtained with Yalmip is  $-1.7004$ . Using the distributed Algorithm 3 with Algorithm 2 with accuracy values  $\varepsilon = \{1, 10^{-1}, 10^{-2}, 10^{-3}\}$  the following corresponding objective function values are obtained:  $-1.7164, -1.7036, -1.7023, -1.7022$ . One can see that with decreasing  $\varepsilon$  convergence to a negative cost function value is observed. Thus, the distributed Lyapunov test indicates asymptotic stability of this practical example of a large-scale dynamical system based on Theorem 4.

## VI. CONCLUSIONS

In this article we present two methods to evaluate stability of large-scale systems in a distributed fashion where each subsystem only exchanges limited model information. The first approach is based on an M-matrix condition that results in a linear program, the second one is based on a Lyapunov LMI. The key to the distribution of the stability tests lies in the formulation of the stability conditions in the form of an optimization problem, thus allowing the application of distributed optimization techniques. Numerical simulations indicate that the second approach is computationally more costly but reduces conservativeness. On the other hand, the first approach guarantees connective stability. Both methods represent first approaches for general LTI systems to evaluate stability without the availability of a central, overall system model, thus ensuring privacy of the subsystem model data.

## REFERENCES

- [1] D. Helbing, "Traffic and related self-driven many-particle systems," *Reviews of modern physics*, vol. 73, no. 4, p. 1067, 2001.
- [2] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, pp. 427–438, 2012.
- [3] D. Hill and G. Chen, "Power systems as dynamic networks," in *Proc. of 2006 IEEE Int. Symposium on Circuits and Systems (ISCAS)*, 2006, pp. 722–725.
- [4] P. Moylan and D. Hill, "Stability criteria for large-scale systems," *IEEE Transactions on Automatic Control*, vol. 23, no. 2, pp. 143–149, 1978.
- [5] D. D. Siljak, *Decentralized Control of Complex Systems*, ser. Mathematics in Science and Engineering. Academic Press, 1991.
- [6] R. H. Gielen and M. Lazar, "Non-conservative dissipativity and small-gain conditions for stability analysis of interconnected systems," in *Proc. 51st Conf. on Decision and Control (CDC)*, 2012, pp. 4187–4192.
- [7] Necoara, I. and Nedelcu, V. and Dumitrache, I., "Parallel and distributed optimization methods for estimation and control in networks," *Journal of Process Control*, vol. 21, no. 5, pp. 756–766, 2011.
- [8] Scattolini, R., "Architectures for distributed and hierarchical model predictive control – a review," *Journal of Process Control*, vol. 19, no. 5, pp. 723–731, 2009.
- [9] A. Rantzer, "Distributed control of positive systems," in *Proc. 50th IEEE Conf. on Decision and Control and European Control Conf. (CDC-ECC)*, 2011, pp. 6608–6611.
- [10] Pates, R. and Vinnicombe, G., "Stability certificates for networks of heterogeneous linear systems," in *Proc. 51st IEEE Conf. on Decision and Control (CDC)*, 2012, pp. 6915–6920.
- [11] S. Z. Khong and A. Rantzer, "Scalable stability conditions for heterogeneous networks via integral quadratic constraints," in *Proc. European Control Conference (ECC)*, 2014, pp. 2863–2867.
- [12] E. Kaszkurewicz and A. Bhaya, *Matrix diagonal stability in systems and computation*. Springer, 2000.
- [13] Carlson, D. and Hershkowitz, D. and Shasha, D., "Block diagonal semistability factors and Lyapunov semistability of block triangular matrices," *Linear Algebra and its Applications*, vol. 172, pp. 1–25, 1992.
- [14] Berman, A. and Goldberg, F. and Shorten, R., "Comments on Lyapunov  $\alpha$ -stability with Some Extensions," in *Variational and Optimal Control Problems on Unbounded Domains*, Wolansky, G. and Zaslavski, A. J., Ed. American Mathematical Soc., 2014.
- [15] M. Meinel, M. Ulbrich, and S. Albrecht, "A class of distributed optimization methods with event-triggered communication," *Computational Optimization and Applications*, vol. 57, no. 3, pp. 517–553, 2014.
- [16] I. Necoara and J. Suykens, "Application of a smoothing technique to decomposition in convex optimization," *IEEE Transactions on Automatic Control*, vol. 53, no. 11, pp. 2674–2679, 2008.
- [17] Necoara, I. and Nedelcu, V., "Rate analysis of inexact dual first order methods. Application to dual decomposition," *IEEE Transactions on Automatic Control*, vol. 49, no. 5, pp. 1232–1243, 2014.
- [18] —, "On linear convergence of a distributed dual gradient algorithm for linearly constrained separable convex problems," *ARXIV*, 2014.
- [19] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [20] J. L. Gross and J. Yellen, *Handbook of graph theory*. CRC press, 2004.
- [21] S. Kim, M. Kojima, M. Mevissen, and M. Yamashita, "Exploiting sparsity in linear and nonlinear matrix inequalities via positive semidefinite matrix completion," *Math. Prog.*, vol. 129, pp. 33–68, 2011.
- [22] M. Meinel and M. Ulbrich, "A distributed algorithm with event-triggered communication for convex LMI-constrained optimization problems," 2014, Preprint, Department of Mathematics, TU München.
- [23] F. Deroo and M. Meinel and M. Ulbrich and S. Hirche, "Distributed stability tests for large-scale systems with limited model information," 2014, Preprint. [Online]. Available: [www-m1.ma.tum.de/foswiki/pub/M1/Lehrstuhl/MartinMeinel/distributed\\_stability.pdf](http://www-m1.ma.tum.de/foswiki/pub/M1/Lehrstuhl/MartinMeinel/distributed_stability.pdf)
- [24] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, 2004. [Online]. Available: <http://users.isy.liu.se/johanl/yalmip>
- [25] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE Control Systems*, vol. 22, no. 1, pp. 44–52, 2002.
- [26] F. Deroo, M. Ulbrich, B. D. O. Anderson, and S. Hirche, "Accelerated iterative Distributed Controller Synthesis with a Barzilai-Borwein Step Size," in *Proc. 51st IEEE Conf. on Decision and Control (CDC)*, 2012.
- [27] R. Christie, "IEEE 30 Bus Test Case," 1993. [Online]. Available: [http://www.ee.washington.edu/research/pstca/pf30/pg\\_tca30bus.htm](http://www.ee.washington.edu/research/pstca/pf30/pg_tca30bus.htm)



**Frederik Deroo** received his Diplom degree in Mechatronics from the University of Stuttgart, Germany in 2011, and a Master of Science degree in Engineering Science and Mechanics from the Georgia Institute of Technology in 2009. Since April 2011 he has been a research assistant at the Institute of Information-oriented control. His research interests include distributed control and analysis of interconnected systems, as well as the application of distributed optimization methods to control theory problems.



**Martin Meinel** graduated in 2004 with a diploma in mathematics at the Technical University of Munich. Since 2004 he is a member of the Chair of Mathematical Optimization at the TUM, where he is writing a doctoral dissertation on distributed optimization with event-triggered communication. During his Ph.D. he participated in the SPP-1305 priority program funded by the German Research Foundation (DFG). His research interests include distributed semidefinite and convex optimization as well as the investigation of convergence properties

using event-triggered communication in a distributed setting.



**Michael Ulbrich** received his Diploma, Doctorate, and Habilitation in Mathematics from the Technische Universität München in 1992, 1996, and 2002, respectively. From 2002 to 2006 he was a full professor at the Department of Mathematics, University of Hamburg. Since 2006 he is a full professor and head of the Chair of Mathematical Optimization, Technische Universität München. His main research interests include nonlinear optimization, optimal control, large-scale optimization, complementarity problems, variational inequalities, MPECs, inverse

optimal control, and nonsmooth analysis.



**Sandra Hirche** received the Diplom degree in Mechanical Engineering and Transport Systems from the Technical University Berlin, Germany, in 2002 and the Doctor of Engineering degree in Electrical Engineering and Information Technology from the Technische Universität München, Munich, Germany, in 2005. From 2005 to 2007 she was awarded a Postdoc scholarship from the Japanese Society for the Promotion of Science at the Fujita Laboratory, Tokyo Institute of Technology, Tokyo, Japan. From 2008 to 2012 she has been an associate professor at

Technische Universität München. Since 2013 she holds the Liesel Beckmann Chair and is director of the Institute for Information-oriented Control in the Department of Electrical Engineering and Information Technology at Technische Universität München. Her main research interests include networked dynamical systems, cooperative and distributed control, and event-triggered control with applications in human-in-the-loop systems, robotics, and infrastructure systems. She has published more than 150 papers in international journals, books and refereed conferences. Dr. Hirche has served or is serving on the Editorial Boards of the IEEE Transactions on Control Systems Technology and the IEEE Transactions on Haptics. She has received multiple awards such as the Rohde & Schwarz Award for her PhD thesis in 2005, the IFAC World Congress Best Poster Award in 2005 and together with students Best Paper Awards of IEEE Worldhaptics and IFAC Conference of Manoeuvring and Control of Marine Craft in 2009.