2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

Static Decentralized Control of a Single-Integrator Network with Markovian Sensing Topology

Sandip Roy[1]   Ali Saberi[1,2]

Abstract— **Static stabilization of a decentralized discrete-time single-integrator network that is subject to Markovian variation in the communication/sensing topology is considered. In particular, we develop conditions on the Markovian topology that are sufficient for mean-square stabilization using a decentralized static controller that has no knowledge of the underlying Markov state. Our analysis exposes a deep connection between decentralized control of single-integrator networks with Markovian topology and those with fixed topology: static stabilization of the network with Markovian topology is possible whenever the steady-state time-average of the Markovian topology is amenable to static (fixed-topology-based) decentralized control.**

I. INTRODUCTION

Groups of coordinating autonomous agents are often subject to stochastic fluctuations in their sensing/communication capabilities, that may significantly impact their ability to complete required tasks. Our aim in this article is to explore static decentralized control of a network of agents with single-integrator dynamics that is subject to Markovian fluctuations in its sensing topology.

Stochastic failures in communication and/or sensing capabilities are significant in a range of distributed-control applications, including autonomous-vehicle control [1], distributed protocol design [2], and infrastructure-network (e.g., electric power system and air traffic system) management (see, e.g., [3] for a discussion of communications in power system control). In the vehicle-control context, the article [1] considers a packet-loss model for communication, and notes that the vehicle dynamics with communication can then be viewed as a *Markovian jump-linear system* (MJLS). The article then develops a *linear matrix inequality* (LMI) condition for centralized dynamic mean-square stabilization (that further assumes knowledge of whether a packet loss has occurred). In designing distributed protocols for computing (e.g., agreement protocols), the possibility for stochastic and/or deterministic communication losses has been widely considered (see [2] for a synopsis). Queueing models for communication in electric power-system control (in particular, third-party load frequency control) have recently been developed in [4], and the ability to achieve control goals has been tied with the communication system's behavior using simulations.

We also consider the role of stochastic communication and/or sensing in the control of networks, but approach this problem specifically from a graph-theoretic and decentralized perspective. In particular, in the context of a model with simple and autonomous agent dynamics, we expose in a quite general manner the connection between the stochastic communication/sensing topology and decentralized stabilizability. The following are two specific contributions of our work:

a): For networks of agents with single-integrator dynamics (which are pertinent in, e.g., autonomous-vehicle and protocol-design applications [5], [6], [7]) and a stochastically-fluctuating communication topology, we develop graph-theoretic conditions for *distributed* static stabilization in a mean-square sense. As far as we know, our work constitutes a first study of distributed controller design for networks whose communications fluctuate in a stochastic (in particular, Markovian) manner. We stress that our stabilization study of uncertain autonomous-agent systems constitutes a novel, design-based viewpoint: in contrast to previous work (e.g., [1]), our results indicate whether any controller can stabilize the system instead of focusing on whether a particular controller suffices. Hence, our study determines whether or not a stabilizing controller can be designed.

b): Our conditions for stabilization elucidate the essential role played by the communication graphs (topologies), and the stochastic fluctuations among them, in the distributed control of the system. In this sense, our results here build on those described in [8], in which we considered stabilization for (double)-integrator networks with fixed communication topology. A connection between distributed-agent control and dynamically-varying graphs has also been established in [11], but there the focus is on controlling the agents to achieve the desired graph rather than using the varying graph in the control of the agents.

While our primary focus in this work is the control of autonomous agents with Markovian communication topology, we also mean for it to serve as a springboard toward more general study of distributed control of Markovian jump-linear systems (MJLS). Such a study of distributed control of MJLS would be valuable for characterization of hard-interconnected networks with communication (e.g., electric

[1]School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164-2752, U.S.A.{sroy,saberi}@eecs.wsu.edu

[2]Support for this work was provided by the Office of Naval Research under grant number N000140310848.

power systems) and for systems with Markovian fluctuations in the state-update or actuation processes. Several articles have been concerned with centralized stabilization of MJLS; the article [12] presents some relevant results and provides a good review of relevant literature. To the best of our knowledge, our work is the first study of distributed control of MJLS. Although we consider a model with very simple agent dynamics in this article, we believe that the strategies here can be extended to pursue both static and dynamic control of general MJLS.

The article is organized as follows. Section II contains introductory material, including formulation of our model, expression of the model as an MJLS, and review of stabilization concepts for MJLS. In Section III, we derive the central result of this article—namely, an explicit condition on the network's graph topology for stabilization—using a perturbation-based argument. Finally, Section IV contains discussion of our results and an example.

II. Model and Problem Formulation

A. Model Formulation

We consider distributed stabilization of a network of single-integrator agents, whose communication/sensing topology is subject to Markovian fluctuation. Velocity control in autonomous-vehicle applications can often be phrased as control of a network of single-integrator agents (see, e.g., [7], [8], [15]). Single-integrator models have also been proposed for agreement protocols, in [5], [6], [9]. In these and other distributed applications, uncertainties in the communication topology (e.g., random loss of communication channels or faulty transmissions) are observed [1], [2]. Hence, we are motivated to pursue distributed control of networks of single-integrator networks with randomly-varying sensing topology.

Our model, which we call a *Markovian single-integrator network (MSIN)* comprises a network of n agents. Each agent i has scalar **state** x_i that evolves in discrete time as a single integrator, i.e., as

$$x_i[k+1] = x_i[k] + u_i[k], \tag{1}$$

where $u_i[k]$ is a control input that is determined from agent i's concurrent observation (specified below). For convenience, we define a **state vector**

$$\mathbf{x}'[k] \stackrel{\triangle}{=} \begin{bmatrix} x_1[k] & \dots & x_n[k] \end{bmatrix}$$

and an **input vector**

$$\mathbf{u}'[k] \stackrel{\triangle}{=} \begin{bmatrix} u_1[k] & \dots & u_n[k] \end{bmatrix}. \tag{2}$$

The observations made by each agent in the MSIN are governed by a stochastically-varying *sensing topology*[1]

[1]From here on, we refer to the observation structure of our model as a sensing rather than communication topology. We believe that the results discussed here can apply to both to systems with sensing capability and ones with communication capability. However, we feel that our model is a bit more realistic for sensing dynamics, since no delay is assumed in the observations.

In particular, each agent's observations constitute a time-varying linear function of the concurrent state vector, with the time-dependence governed by an *underlying Markov chain*. Formally, the observations made by agent i at time k are given by

$$\mathbf{y}_i[k] = G_i(\sigma[k])\mathbf{x}[k],$$

where $\sigma[k]$ represents the time-k **status** (state) of an **underlying Markov chain** and the **graph matrix** $G_i(\sigma[k])$ is a matrix of dimension $m_i \times n$. We assume that the underlying Markov chain takes on b statuses (i.e., $\sigma[k] \in 1, \dots, b$) and is governed by the $b \times b$ stochastic matrix P. We find it convenient to define a **full observation vector**

$$\mathbf{y}'[k] \stackrel{\triangle}{=} \begin{bmatrix} \mathbf{y}_1[k] & \dots & \mathbf{y}_n[k] \end{bmatrix}$$

and **full graph matrices**

$$G(\sigma) = \begin{bmatrix} G_1(\sigma) \\ \vdots \\ G_n(\sigma) \end{bmatrix}$$

$\sigma \in 1, \dots, b$. In this notation, $\mathbf{y}[k] = G(\sigma[k])\mathbf{x}[k]$.

We assume a static decentralized linear time-invariant (LTI) controller is used in the MSIN. That is, each agent's input is computed from its concurrent state as

$$\mathbf{u}_i[k] = \mathbf{k}_i'\mathbf{y}_i[k], \tag{3}$$

where the m_i-component vector $\mathbf{k}_i$ is denoted as agent i's **control gain**. For convenience, we define a **control matrix**

$$K = \begin{bmatrix} \mathbf{k}_1 & & \\ & \ddots & \\ & & \mathbf{k}_n \end{bmatrix}$$

In this notation, $\mathbf{u}[k] = K\mathbf{y}[k]$. We stress that the controller used in the MSIN *has no knowledge of the underlying status of the Markov chain*, and hence uses a gain that is not dependent on this status.

In summary, an MSIN is a network of single-integrator agents with Markovian sensing topology and static distributed control. We note that an MSIN is defined completely by the transition matrix P, sensing topologies $[G] \stackrel{\triangle}{=} G(1), \dots, G(b)$, and controller K.

We note that an MSIN can be viewed as a particular example of an MJLS, since the input-to-output dynamics constitute a linear system whose parameters (in this case, the observation topology) are governed by an underlying Markov chain. We are interested in the static decentralized control of this MJLS.

MSIN models are applicable in such contexts as autonomous-vehicle control and agreement-protocol design. Due to space constraints, we are unable to provide application-oriented examples here. We refer the reader to an extended version of this paper [10] for several examples; a small illustrative example is given in Section IV.

B. Notions of Stabilization

In this article, we are concerned with stabilization of an MSIN using the described static decentralized controller. We assume in our development that the control matrix K is constant in time, or in other words that *the controller has no knowledge of the underlying Markov chain's state*. Stabilization of MJLS with a static controller that has no dependence on the underlying state has been considered in, e.g., [12], [13]; in this literature, stabilization with such a controller is known as *simultaneous stabilization*.

Stabilization of MJLS (and other stochastic systems) is often phrased in terms of moment-convergence of the closed-loop systems, or alternately according to probabilistic convergence of the system state sequence (see, e.g., [12], [13], [14], [16], [17] for studies of stabilization and control of MJLS). We also seek conditions for stabilization under uncertainty, so it is worthwhile for us to review relevant definitions for stabilization of stochastic systems, in the context of MSIN.

Definition 1: An MSIN is said to be stabilizable in a mean-square sense if there exists a static decentralized control matrix K such that, for any initial state $\mathbf{x}[0]$ and any initial probability distribution for the underlying Markov chain, $\lim_{k\to\infty} E(||x[k]||_2^2) = 0$.

A couple notes are in order about this definition:

- The articles [12], [13], [14] distinguish among several notions of mean-square stability and stabilization, including among asymptotic mean-square stabilization (the notion presented here), exponential mean-square stabilization, and stochastic mean-square stabilization (which is concerned with the time-integrated square error). These have been shown to be identical for MJLS, so we only consider one notion here.
- Mean-square stabilization can be generalized to δ-moment stabilization, which is concerned with convergence of $\lim_{k\to\infty} E(||x[k]||_2^\delta)$. We do not pursue this generalization here.

We also consider a second notion of stabilization:

Definition 2: An MSIN is said to be almost surely stabilizable if there exists a static decentralized control matrix K such that, for any initial state $\mathbf{x}[0]$ and any initial probability distribution for the underlying Markov chain, $P\left\{\lim_{k\to\infty} ||x[k]||_2 = 0\right\} = 1$.

It is well known (see [13]) that, for MJLS (and hence MSIN), mean-square stabilizability implies almost-sure stabilizability. In this article, we will develop conditions on the network graph for which the MSIN is mean-square stabilizable (as always, using a distributed controller), and hence also almost-sure stabilizable.

C. Phrasing Stabilization Conditions in Terms of Moment Recursions

In order to develop network-theoretic conditions for the decentralized mean-squared stabilization of an MSIN, we must characterize the limiting values of second moments of state variables. It is well known that moments and cross-moments of MJLS—and hence MSIN—state variables satisfy certain linear recursions (see, e.g., [14], [18]), and hence mean-square stabilizabilion of an MSIN can be phrased as a simultaneous stabilization problem for a deterministic linear system. In this section, we present the recursions for the first and second moments of state variables (without proof) for an MSIN[2]. We then present conditions for mean-square stabilization of the MSIN in terms of the spectrum of the moment recursion matrices.

We first note that the state update, observation process, and control law can be combined as follows:

$$\mathbf{x}[k+1] \quad = \quad (I + KG(\sigma[k]))\mathbf{x}[k]. \qquad (4)$$

Equation 4 is a form for the state dynamics from which convenient expressions for the first and second moments can be found.

To find these recursions, it is necessary to define a $0-$ -1 indicator vector $\mathbf{q}[k]$ for the state of the underlying Markov chain. That is, we define $\mathbf{q}[k]$ to be a t-component vector that is all zeros, except for a single unity entry at location $\sigma[k]$. The analyses of, e.g., [18] provide recursions for $E(\mathbf{q}[k] \otimes \mathbf{x}[k])$ and $E(\mathbf{q}[k] \otimes (\mathbf{x}[k]^{\otimes 2})$ for the MSIN, which can straightforwardly be used to develop expressions for first and second moments and cross-moments of state variables at each time. We present these recursions here, without proof:

$$E(\mathbf{q}[k+1] \otimes \mathbf{x}[k+1]) = HE(\mathbf{q}[k] \otimes \mathbf{x}[k]) \qquad (5)$$

and

$$E(\mathbf{q}[k+1] \otimes \mathbf{x}[k+1]^{\otimes 2}) = H_2 E(\mathbf{q}[k] \otimes \mathbf{x}[k]^{\otimes 2}), \qquad (6)$$

where

$$H = \begin{bmatrix} p_{11}(I+KG(1)) & \dots & p_{t1}(I+KG(t)) \\ \vdots & & \vdots \\ p_{1t}(I+KG(1)) & \dots & p_{tt}(I+KG(t)) \end{bmatrix}$$

and

$$H_2 = \begin{bmatrix} p_{11}(I+KG(1))^{\otimes 2} & \dots & p_{t1}(I+KG(t))^{\otimes 2} \\ \vdots & & \vdots \\ p_{1t}(I+KG(1))^{\otimes 2} & \dots & p_{tt}(I+KG(t))^{\otimes 2} \end{bmatrix}.$$

We refer to Equation 5 as the **mean recursion** and call H the **mean recursion matrix**. Similarly, we refer to Equation 6 as the **second-moment recursion** and call H_2 the **second-moment recursion matrix**.

We note that the first- and second-moments of state variables can be found in terms of the entries of $E(\mathbf{q}[k] \otimes \mathbf{x}[k])$ and $E(\mathbf{q}[k] \otimes (\mathbf{x}[k]^{\otimes 2})$, and hence it is not surprisng that mean-square stabilizability of an MSIN can be phrased in terms of the eigenvalues of the mean recursion matrix

[2]Formally, we only need consider the second-moment recursion to develop conditions for stabilization, but we consider the first moment recursion because it is used in our proof in the next section and because the first-moment recursion provides necessary conditions for stabilization.

and second-moment recursion matrix. In the following two theorems, we provide a necessary condition for stabilization in terms of the mean recursion matrix, and a necessary and sufficient condition for stabilization in terms of the second-moment recursion matrix. We note that the concepts behind these theorems are well-known (see, e.g., [14], [17]), and so we omit the proofs.

Theorem 1: An MSIN is mean-square stabilizable only if there exists a (distributed) control matrix K such that the eigenvalues of the mean recursion matrix H fall strictly within the unit circle.

Theorem 2: An MSIN is stabilizable if and only if there exists a control matrix K such that the eigenvalues of H_2 lie strictly within the unit circle.

III. Our Main Result: A Network-Theoretic Condition for Mean-Square Stabilization

We develop an explicit sufficient condition on the the sensing topology for mean-square stabilization, by designing K so that the eigenvalues of the mean recursion matrix and the second-moment recursion matrix fall within the unit circle. In particular, we use perturbation (eigenvalue sensitivity) arguments to find rather general conditions on an MSIN for which a stabilizing control matrix can be designed. That is, we give conditions on an MSIN such that the eigenvalues of the mean recursion matrix and second moment recursion matrix, respectively, can be placed strictly within the unit circle using some controller. These conditions provide an explicit means for checking whether decentralized stabilization of an MSIN can be achieved.

Our ensuing result will clarify that stabilizability of an MSIN is deeply connected with the structure of the steady-state time-average of the graph matrix. Therefore, it is useful for us to introduce some terminology that permits us to characterize the time-averaged graph matrix, and in turn present the main theorem of this article:

- We denote the steady-state probability vector for the underlying Markov chain by π. That is, π' is the (unique) left eigenvector of P with unity eigenvalue: $\pi' P = \pi'$. We also define the b-component vector λ to contain the b eigenvalues of P.
- We define a set of b square matrices $(L_1, \ldots, L_b)$ of equal dimension to be π**-full-rank**, if $\pi_1 L_1 + \ldots + \pi_b L_b$ has full rank.

We also require the following notation to present the theorem:

- We use the notation $\{L\}_q$ to denote that qth principal minor of the square matrix L.

We are now ready to present the main result. For ease of presentation, we restrict ourselves to the case where agents make scalar observations, and consider the multi-observation case (which is not much different) in the following discussion section.

Theorem 3: If there exists a permutation matrix P such that, for the "rearranged" graph matrices $\widehat{G}(1) = PG(1)P^{-1}, \ldots, \widehat{G}(b) = PG(b)P^{-1}$, the sets $(\{\widehat{G}(1)\}_q, \ldots, \{\widehat{G}(b)(q)\}_q)$ are π-full-rank for $q \in 1, \ldots, n$, then a static decentralized controller can be designed such that the eigenvalues first- and second-moment recursion matrices of the $MSIN$ are strictly within the unit circle.

Hence, by invoking Theorem 2 in addition to Theorem 3, we can find explicit sufficient condition for stabilization of an MSIN:

Theorem 4: If there exists a permutation matrix P such that, for the "rearranged" graph matrices $\widehat{G}(1) = PG(1)P^{-1}, \ldots, \widehat{G}(b) = PG(b)P^{-1}$, the sets $(\{\widehat{G}(1)\}_q, \ldots, \{\widehat{G}(b)(q)\}_q)$ are π-full-rank for $q \in 1, \ldots, n$, then a static decentralized controller that stabilizes the MSIN (in a mean-square sense) can be designed.

We shall prove Theorem 3 in several steps. Without loss of generality, we assume throughout the proof that the the permutation matrix P in Theorem 3 is the identity matrix: if it is not, we have only to permute the full graph matrices according to P and apply the ensuing argument.

Step 1: Lots of notation: Before detailing the proof, it is useful for us to define some more notation that is used in the proof:

- We shall often consider the spectrum of the first- and second-moment recursion matrices when a particular controller K is used (for a fixed $[G]$ and P). Hence, we find it useful to parametrize H and H_2 by K, using the notation $H(K)$ and $H_2(K)$, respectively.
- We use the notation $K(q)$ to describe a controller for which only the first q diagonal entries are non-zero.
- We use the notation $\{L\}_q$ for the qth leading principal minor of the square matrix L.
- We use the notation $\{L\}_{q,r}$ for the matrix comprising the first q rows and first r columns of L.
- We find it valuable to analyze certain sub-matrices of H, that capture interactions among subsets of the agents. (These are equivalent to principal minors.) Hence, we consider the matrices

$$\{H(K)\}^q = \begin{bmatrix} p_{11}(I_q + \{K\}_q\{G_1\}_q) & \ldots & p_{b1}(I_q + \{K\}_q\{G_b\}_q) \\ \vdots & & \vdots \\ p_{1b}(I_q + \{K\}_q\{G_1\}_q) & \ldots & p_{bb}(I_q + \{K\}_q\{G_b\}_q) \end{bmatrix} \quad (7)$$

- Consider the matrices $\pi_1\{G_1\}_q + \ldots + \pi_b\{G_b\}_q$, $q \in 1, \ldots, n$. By assumption, each of these matrices is full rank. Hence, for any unit vector $\mathbf{x}$, $(\pi_1\{G_1\}_q + \ldots + \pi_b\{G_b\}_q)\mathbf{x}$ is non-zero. We define ϵ as the smallest 2-norm of any vector $(\pi_1\{G_1\}_q + \ldots + \pi_b\{G_b\}_q)\mathbf{x}$ (for any q in $1, \ldots, n$.
- We define $\delta_1 = 1 - \lambda(max)$, where $\lambda(max)$ is the magnitude of the subdominant eigenvalue of P.
- We define ρ as the induced norm of the matrix $\begin{bmatrix} p_{11}G(1) & \ldots & p_{b1}G(b) \\ \vdots & & \vdots \\ p_{1b}G(1) & \ldots & p_{bb}G(b) \end{bmatrix}$.

Step 2: Eigenanalysis of the open-loop system: Consider an open-loop MSIN (i.e., an MSIN for which $K = 0$). The eigenvalues of the first-moment recursion matrix H are the entries of the vector $\lambda \otimes \mathbf{1}_n$. Hence, exactly n eigenvalues of H are 1, while the remaining eigenvalues have magnitude strictly less than 1. The n repeated unity eigenvalues are simple. Any vector of the form $\pi \otimes \mathbf{v}$, where $\mathbf{v}$ has n components, is a right eigenvector of H corresponding to the repeated unity eigenvalue.

To see why the above eigenanalysis is correct, we note that the first-moment recursion matrix for the open-loop system is given by $P' \otimes I_n$. The results in the lemma above then follow directly from basic properties of the Kronecker product (see, e.g., [19]).

Step 3: Characterization of Unity Eigenvector of $\{H(K(q))\}^{q+1}$: In Step 4, we will use a sequential controller design to move the eigenvalues of H into the unit circle. It turns out that this design process requires characterization of a particular eigenvector of each matrix $\{H(K(q))\}^{q+1}$, $1 \leq q \leq n-1$ for small control gains K, and so we first characterize these eigenvectors.

In particular, notice that $\{H(K(q))\}^{q+1}$ has at least one unity eigenvalue, with corresponding left eigenvector $\mathbf{1}' \otimes \mathbf{e}'(k+1)$ (where $\mathbf{e}(k+1)$ refers to a $k+1$-component indicator vector with the final entry non-zero. Also, let us assume that we have chosen $K(q)$ so that the remaining eigenvalues of $\{H(K(q))\}^{q+1}$ are non-unity. (It will become clear that we can do so, from our sequential design process.) Let us attempt to characterize the right eigenvector of $\{H(K(q))\}^{q+1}$ corresponding to the single unity eigenvalue. To do so, first note that, since $\{H(K(q))\}^{q+1}$ is a perturbation of $\{H(0)\}^{q+1}$, the right eigenvector of $\{H(K(q))\}^{q+1}$ corresponding to the unity eigenvector is close to a vector in the unity right eigenspace of of $\{H(0)\}^{q+1}$. That is, the right eigenvector of interest is close to a vector of the form $\pi \otimes \mathbf{v}$, where $\mathbf{v}$ is some non-zero $q+1$-component vector. To figure out the particular vector $\pi \otimes \mathbf{v}$ that the right eigenvector of interest is close to, let us again view $\{H(K(q))\}^{q+1}$ as a perturbation of $\{H(0)\}^{q+1}$, and note that one of the $q+1$ unity eigenvalues of $\{H(0)\}^{q+1}$ does not change when the controller is implemented: that is, the sensitivity of this eigenvalue to the perturbation is zero. Thus, for any small enough $K(q)$, the unity right eigenvector of $\{H(K(q))\}^{q+1}$ is arbitrarily close to the vector of the form $\pi \otimes \mathbf{v}$ for which $\begin{bmatrix} p_{11}\{G(1)\}_{q,q+1} & \dots & p_{b1}\{G(b)\}_{q,q+1} \\ \vdots & & \vdots \\ p_{1b}\{G(1)\}_{q,q+1} & \dots & p_{bb}\{G(b)\}_{q,q+1} \end{bmatrix} \pi \otimes \mathbf{v}$ is orthogonal to the unity left eigenspace of $\{H(0)\}^{q+1}$. That is, we wish to identify the vector of the form $\pi \otimes \mathbf{v}$ such that $(\mathbf{1}' \otimes \mathbf{w}') \begin{bmatrix} p_{11}\{G(1)\}_{q,q+1} & \dots & p_{b1}\{G(b)\}_{q,q+1} \\ \vdots & & \vdots \\ p_{1b}\{G(1)\}_{q,q+1} & \dots & p_{bb}\{G(b)\}_{q,q+1} \end{bmatrix} \pi \otimes \mathbf{v}$ equals 0. With just a bit of algebra, we then obtain that $|| \sum_{i=1}^{b} \pi_i \{G(i)\}_{q,q+1} \widehat{\mathbf{v}}_i ||_2$, where $\widehat{\mathbf{v}}_i$ is the part of the unity right eigenvector corresponding to underlying state i scaled by π_i, becomes arbitrarily close to zero for small enough $K(q)$. Furthermore, since each $\widehat{\mathbf{v}}_i$ becomes arbitrarily close to a constant $\mathbf{v}$, and since $|| \sum_{i=1}^{b} \pi_i \{G(i)\}_{q+1,q+1} \mathbf{v} ||_2 \geq \epsilon$ for any $\mathbf{v}$, we find that $|| \sum_{i=1}^{b} \pi_i E(i) \{G(i)\}_{q+1,q+1} \widehat{\mathbf{v}}_i ||_2$, where $E(i)$ is a square matrix with a single unity entry in the bottom right corner and zeros elsewhere, must be strictly greater than zero for small enough $K(q)$. Conceptually, what we have shown is that the right eigenvector corresponding to the unity entry cannot be perpendicular to the perturbation vector that is added when feedback at agent $q+1$ is implemented. This comes about because of the characteristics of the unity right eigenvector, as well as π-full-rank requirement.

We find it convenient to define a notation for $K(q)$ being small enough to guarantee this non-orthogonality. In particular, we define $\overline{k}$ as the maximum allowed magnitude for any element in $K(q)$, such that $|| \sum_{i=1}^{b} \pi_i E(i) \{G(i)\}_{q+1,q+1} \widehat{\mathbf{v}}_i ||_2$ is guaranteed to be greater than or equal to $\frac{\epsilon}{2}$.

Step 4: Stabilization Proof, First-Moment Recursion: We are now ready to design a control matrix K such that the eigenvalues of the first-moment recursion lie strictly within the unit circle, given that the premises of Theorem 3 hold. In particular, we shall prove that there exists a controller K^* with diagonal entries that satisfy $|k_n| << |k_{n-1}| << \dots << |k_1|$ such that closed-loop system is stable. (Our justification will clarify exactly how each k_{i+1} must be chosen with respect to k_i, to achieve stability). Our choice of controller turns out to be prescient, in that it also guarantees stability of the second moment-recursion matrix, as we will show in the next section.

We prove the result using a recursive and perturbation-based approach. We shall recursively design the entries of K^*, so that all but $n-q$ eigenvalues of $H(K^*(q))$ are strictly within the unit circle, for each $q \in 1, \dots, n$.

First, consider use of the controller $K^*(1)$. Let us choose $k_1 = -min(\frac{\delta_1}{\alpha_1 \rho}, \overline{k}) sign((\pi_1 \{G(1)\}_1 + \dots + \pi_b \{G(b)\}_1))$, where α_1 is a scaling constant that will be shown to be finite shortly. Viewing the control action as a perturbation on $H(0)$, we note that the subdominant eigenvalue moves by at most a distance of $k_1 \rho$, plus terms that are second- or higher order in k_1. Hence, there clearly is a constant α_1 that guarantees that none of the non-unity eigenvalues of $H(0)$ moves outside the unit circle, when the controller $K^*(1)$ is used. Thus, we need only consider the impact of the controller on the unity eigenvalues of $H(0)$.

It is easy to see (by constructing the appropriate left eigenvectors) that $n-1$ of the unity eigenvalues of $H(0)$ remain at the origin upon control by $K^*(1)$. The perturbed version of the remaining eigenvalue is the largest eigenvalue of $\{H(K^*(1))\}^1$. However, the eigenvalues of $\{H(K^*(1))\}^1$ can be viewed as perturbations of the eigenvalues of P' by $k_1 \begin{bmatrix} p_{11}\{G(1)\}_1 & \dots & p_{b1}\{G(b)\}_1 \\ \vdots & & \vdots \\ p_{1b}\{G(1)\}_1 & \dots & p_{bb}\{G(b)\}_1 \end{bmatrix}$. Hence, the largest eigenvalue of $\{H(K^*(1))\}^1$ is, for small enough k_1, arbitrarily close to

$$1 + k_1\mathbf{1}'\begin{bmatrix} p_{11}\{G(1)\}_1 & \dots & p_{b1}\{G(b)\}_1 \\ \vdots & & \vdots \\ p_{1b}\{G(1)\}_1 & \dots & p_{bb}\{G(b)\}_1 \end{bmatrix}\pi \quad (8)$$
$$= 1 + k_1(\pi_1\{G(1)\}_1 + \dots + \pi_b\{G(b)\}_1).$$

From the π-full-rank assumption, $(\pi_1\{G(1)\}_1 + \dots + \pi_b\{G(b)\}_1)$ is non-zero (and real), and so we can choose k_1 to place this eigenvalue is strictly within the unit circle. In particular, the eigenvalue will be negative, real, and bounded away from unity by at least ϵk_1, minus second- and higher-order terms. Thus, there clearly exists a non-zero constant α_{1a} such that this eigenvalue is bounded away from unity by at least $\delta_2 = \alpha_{1a}\epsilon k_1$. Also, note that $\{H(K^*(1))\}^2$ has the same eigenvalues as $\{H(K^*(1))\}^1$, except for having b more eigenvalues that are the eigenvalues of P (including an eigenvalue at the origin). This concludes the first step in the recursion.

Now assume that we have designed a controller $K^*(q)$ such that $n-q$ eigenvalues of the closed-loop system are strictly within the unit circle, $|k_2|, \dots, |k_n| \leq |k_1|$, and the subdominant eigenvalue of $H(K^*(q))$ is bounded away from unity by δ_{q+1}. Let us consider design of the controller $K^*(q+1)$. To do so, first note that $\{H(K^*(q))\}_{q+1}$ has a single eigenvalue at unity, remaining eigenvalues within the unit circle, and subdominant eigenvalue bounded away from unity by δ_{q+1}. Next, choose $k_{q+1} = \pm\frac{\delta_{q+1}}{\alpha_{q+1}\rho}$, where α_{q+1} is an appropriately-chosen constant and the sign of k_{q+1} will also also be decided shortly. Now consider $\{H(K^*(q+1))\}_{q+1}$ as a perturbation of $\{H(K^*(q))\}_{q+1}$. Notice that the subdominant eigenvalue of $\{H(K^*(q))\}_{q+1}$ moves by at most $k_{q+1}\rho$ (plus perhaps higher-order terms) upon perturbation, so there exists a finite α_{q+1} such that this and smaller eigenvalues cannot move outside the unit circle. Now let's think about how the perturbation affects the unity eigenvalue of $\{H(K^*(q))\}_{q+1}$. As before, let's use the notation $\widehat{\mathbf{v}}' = \begin{bmatrix}\widehat{\mathbf{v}}'_1 & \dots & \widehat{\mathbf{v}}'_b\end{bmatrix}$ for the unity right eigenvector of $\{H(K^*(q))\}_{q+1}$. In this notation, the unity eigenvalue can be seen (using eigenvalue sensitivity arguments) to move to $1 + k_{q+1}(\pi_1 e'(q+1)\{G(1)\}_{q+1}\mathbf{v}_1 + \dots + \pi_b e'(q+1)\{G(b)\}_{q+1}\mathbf{v}_b)$. Because we have chosen $|k_1|, \dots, |k_{q+1}| \leq \overline{k}$, it is guaranteed that the unity eigenvalue moves by at least $k_{q+1}\frac{\epsilon}{2}$, minus some higher order terms. Thus, there clearly exists and $\alpha_{(q+1)a} > 0$ such that the eigenvalue of interest of $\{H(K^*(q+1))\}_{q+1}$ is bounded away from unity by a distance of $\delta_{q+2} = k_{q+1}\alpha_{(q+1)a}\frac{\epsilon}{2}$. Finally, we can choose the sign of k_{q+1} so that this subdominant eigenvalue is real and within the unit circle. Thus, $\{H(K^*(q+1))\}_{q+1}$ has all eigenvalues strictly within the unit circle, and so $H(K^*(q+1))$ has all but $n-(q+1)$ eigenvalues within the unit circle. Hence, we have proved by induction that a controller can be designed so that all eigenvalues of H are strictly within the unit circle.

Step 5: Stabilization Proof, Second-Moment Recursion: We now show that the static gain matrix K can be designed so that the eigenvalues of H_2 fall in the unit circle. Our approach builds on the analysis of the first-moment recursion from Step 4, so we begin with several claims regarding the closed-loop first-moment recursion. We only give brief explanations of these claims, since their justifications can be obtained easily from (or in similar fashion to) our analysis in Step 4.

In particular, we claim that we can choose $k_1, \dots, k_n$ such that

1) all eigenvalues of H fall strictly in the unit circle,
2) $k_1^2 << k_n$,
3) for any given set of arbitrarily small open balls in the complex plane around the eigenvalues of P, exactly n of the eigenvalues of H lie in each ball.
4) each left eigenvector of H can be made arbitrarily close to a vector of the form $\mu' \otimes \mathbf{w}'$, where μ' is a left eigenvector of P' and $\mathbf{w}'$ is a length-n vector. Similarly, each right eigenvector of H can be made arbitrarily close to a vector of the form $\nu \otimes \mathbf{v}$, where ν is a right eigenvector of of P'.

Let us briefly discuss why the claims are true. Claim 1 is the main result proved in Step 4. To justify Claim 2, notice from Step 4 that, in order to guarantee that the eigenvalues of H fall in the unit circle, we require the magnitude of k_1 to be small enough and that the magnitude k_{i+1} to be less than fixed fraction of k_i (and then we correctly choose the signs of these gains). Hence, by choosing k_1 to be small compared to the required fraction $\frac{k_n}{k_1}$, we can guarantee that $k_1^2 << k_n$. Claims 3 and 4 are direct consequences of the interpretation of the control action as a perturbation on the patrix $P' \otimes I$. (Notice, again, that we can choose K to be arbitrarily small and still guarantee that the eigenvalues of H fall strictly within the unit circle, so the perturbation analysis is germane.)

Now let us consider the the second-moment recursion matrix H_2. Recall that

$$H_2 = \begin{bmatrix} p_{11}(I+KG(1))^{\otimes 2} & \dots & p_{b1}(I+KG(b))^{\otimes 2} \\ \vdots & & \vdots \\ p_{1b}(I+KG(1))^{\otimes 2} & \dots & p_{bb}(I+KG(b))^{\otimes 2} \end{bmatrix}. \quad (9)$$

This expression for H_2 can further be rewritten as

$$H_2 = P' \otimes I \otimes I + \begin{bmatrix} p_{11}(I \otimes KG(1)) & \dots & p_{b1}(I \otimes KG(b)) \\ \vdots & & \vdots \\ p_{1b}(I \otimes KG(1)) & \dots & p_{bb}(I \otimes KG(b)) \end{bmatrix} \quad (10)$$
$$+ \begin{bmatrix} p_{11}(KG(1) \otimes I) & \dots & p_{b1}(KG(b) \otimes I) \\ \vdots & & \vdots \\ p_{1b}(KG(1) \otimes I) & \dots & p_{bb}(KG(b) \otimes I) \end{bmatrix} \quad (11)$$
$$+ \begin{bmatrix} p_{11}(KG(1) \otimes KG(1)) & \dots & p_{b1}(KG(b) \otimes KG(b)) \\ \vdots & & \vdots \\ p_{1b}(KG(1) \otimes KG(1)) & \dots & p_{bb}(KG(b) \otimes KG(b)) \end{bmatrix}$$

We choose K so that the four claims above are achieved. In this case, notice that following are true:

- The matrix H_2 is a perturbation of $P' \otimes I \otimes I$ by three terms that depend on the control matrix K, as seen in Equation 10.

- The term $\begin{bmatrix} p_{11}(KG(1)\otimes KG(1)) & \dots & p_{b1}(KG(b)\otimes KG(b)) \\ \vdots & & \vdots \\ p_{1b}(KG(1)\otimes KG(1)) & \dots & p_{bb}(KG(t)\otimes KG(t)) \end{bmatrix}$ is of higher order than the other perturbation terms, and so can be made arbitrarily small compared to the other perturbation terms by choosing small enough K (in particular, $k_1^2 << k_n$). Thus, for a properly-chosen K, this term does not change the spectrum of H_2 and need not be considered further.
- We shall sequentially consider the two remaining perturbation terms in Equation 10. In particular, we first characterize the spectrum of

$$P' \otimes I \otimes I + \begin{bmatrix} p_{11}(I\otimes KG(1)) & \dots & p_{b1}(I\otimes KG(b)) \\ \vdots & & \vdots \\ p_{1b}(I\otimes KG(1)) & \dots & p_{bb}(I\otimes KG(b)) \end{bmatrix} \quad (12)$$

Notice that this matrix has n^2b eigenvalues. It is easy to check that each eigenvalue is exactly equal to an eigenvalue of H. More specifically, each eigenvalue of H is an n-times repeated eigenvalue of $P' \otimes I \otimes I + \begin{bmatrix} p_{11}(I\otimes KG(1)) & \dots & p_{b1}(I\otimes KG(b)) \\ \vdots & & \vdots \\ p_{1b}(I\otimes KG(1)) & \dots & p_{bb}(I\otimes KG(b)). \end{bmatrix}$ If we choose K small enough, the corresponding left eigenvectors can be made arbitrarily close to vectors of the form $\mu' \otimes \overline{\mathbf{w}}' \otimes \mathbf{w}'$, where $\mathbf{w}'$ is any n-component vector (in particular, the vectors $\overline{\mathbf{w}}'$ for a particular repeated eigenvalue can be any n-dimensional vector space.) Similarly, the right eigenvector can be made arbitrarily close to a vector of the form $\nu \otimes \overline{\mathbf{v}} \otimes \mathbf{v}$, for any $\overline{\mathbf{v}}$.
- Now let us consider the further purturbation by $\begin{bmatrix} p_{11}(KG(1)\otimes I) & \dots & p_{b1}(KG(b)\otimes I) \\ \vdots & & \vdots \\ p_{1b}(KG(1)\otimes I) & \dots & p_{bb}(KG(b)\otimes I) \end{bmatrix}$ to obtain H_2. Noticing that the left and right eigenvectors before perturbation are arbitrarily close to the form $\mu' \otimes \overline{\mathbf{w}}' \otimes \mathbf{w}'$ and $\nu \otimes \overline{\mathbf{v}} \otimes \mathbf{v}$, where $\overline{\mathbf{w}}$ and $\overline{\mathbf{v}}$ span R^n, we see that the sensitivity of the eigenvalues to the second perturbation is (in the limit of small K) exactly the same as the sensitivity of the eigenvalues of H upon control by K. To be more precise, consider the eigenvalues of $P' \otimes I \otimes I$. These are the eigenvalues of P', each repeated n^2 times. As we have discussed, the first perturbation $\begin{bmatrix} p_{11}(I\otimes KG(1)) & \dots & p_{b1}(I\otimes KG(b)) \\ \vdots & & \vdots \\ p_{1b}(I\otimes KG(1)) & \dots & p_{bb}(I\otimes KG(b)) \end{bmatrix}$ causes each set of n^2 eigenvalues to be perturbed into n sets of n eigenvalues in a ball around the original value. The second perturbation then causes each of these subsets of n repeated eigenvalues to be perturbed, in exactly the same way (up to higher-order terms) as the perturbation from sets of n^2 to sets of n eigenvalues. Because of this sensitivity structure for the eigenvalues, we notice the following two features:
  - The eigenvalues of H_2 that are perturbations of non-unity eigenvalues of P' remain in small balls around their original values, and so cannot lie outside the unit circle.
  - The n^2 eigenvalues of H_2 that are perturbations of the unity eigenvalue of P' all lie within the unit circle. In particular, they are first perturbed (in sets of n) by small negative real amounts. The second perturbation moves eigenvalues from each set again by small negative real amounts, and hence all these eigenvalues lie strictly within the unit circle. In particular, the distance of the dominant eigenvalue from unity will be twice the distance of the dominant eigenvalue of H from unity.

Hence, the a controller can be designed so that the eigenvalues of H_2 lie within the unit circle, and so Theorem 3 has been proved.

IV. Discussion

We have shown that our ability to develop a static distributed controller for an MSIN is deeply connected with the structure of the network graph and the steady-state probability vector for the underlying Markov chain. Our result for MSINs builds on our previous study of single- and double-integrator networks with fixed topology [8]. In particular, in [8], we showed stabilization of a double-integrator network (and, implicitly, a single-integrator network) using a static distributed controller is possible if there exists a permutation of the graph matrix such that all leading principal minors are full rank. Our analysis here shows that an identical result holds for single-integrator networks with Markovian switching topologies, but with the condition phrased in terms of a weighted average of the multiple possible topologies. In other words, distributed control of the network with switching topology is possible whenever distributed control of the network with the steady-state time-averaged topology is possible. This result is sensible, since stabilizability of the network with averaged topology implies first-moment stabilizability of the original network, while higher-moment stabilizability can be guaranteed by sufficiently slowing down the system.

It is also worth discussing the generalization of our result to the case where agents can make multiple observations. Noticing that the agents can only control a single input, we see that each agent's control input is a linear combination of its output variables in each status. Hence, the MSIN is stabilizable whenever we can find combinations of the outputs (i.e., of the rows of the graph matrices) such that the resulting scalar-observation MSIN is stabilizable. The multi-observation case has been considered in more detail for deterministic integrator networks, in [8].

Illustrative Example: We briefly illustrate our main result using an example $MSIN$. Further discussion of this example—in particular its possible application to autonomous-vehicle velocity control—is given in [10].

We consider an MSIN with $n = 4$ agents and $b = 5$ underlying statuses. Nominally (i.e., in underlying status 1), the agents make sensing observations according to the full graph matrix

$$G(1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 1 & -\frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & 1 & -\frac{1}{2} \\ 0 & 0 & -1 & 1 \end{bmatrix}. \qquad (13)$$

We consider control of this network when the sensing topology switches back and forth in a Markovian fashion from the nominal topology to one of the four topologies in which a single agent does not make its observations at each time-step (e.g., $G(i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 1 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$). We assume that the following Markov chain governs the switching dynamics:

$$P = \begin{bmatrix} 0.8 & 0.05 & 0.05 & 0.05 & 0.05 \\ 0.5 & 0.5 & 0 & 0 & 0 \\ 0.3 & 0 & 0.7 & 0 & 0 \\ 0.5 & 0 & 0 & 0.5 & 0 \\ 0.4 & 0 & 0 & 0 & 0.6 \end{bmatrix} \qquad (14)$$

(Here, the first status of the Markov chain represents the nominal topology, and the four other statuses represent the topologies with failures.)

For this example, it is easy to check that $\sum_{i=1}^{b} \pi_i G(i)$ equals $\begin{bmatrix} 0.93 & 0 & 0 & 0 \\ -0.44 & 0.88 & -0.440 & \\ 0 & -0.47 & 0.93 & -0.47 \\ 0 & 0 & -0.92 & 0.92 \end{bmatrix}$, and consequently that the network is π-full-rank. Hence, we know that there exists a static controller K, such that the closed-loop system is stable in a mean-square sense. In fact, we can check that the static controller $K = I$ suffices to stabilize the MSIN in a mean-square sense. In Figure IV, we plot the state of one of the agents during a sample run, along with the expectation of this agent's state and two-standard deviation intervals around the expected value. These plots illustrate that the mean value of the state approaches zero and the variance also approaches zero, so that the state converges to the origin.

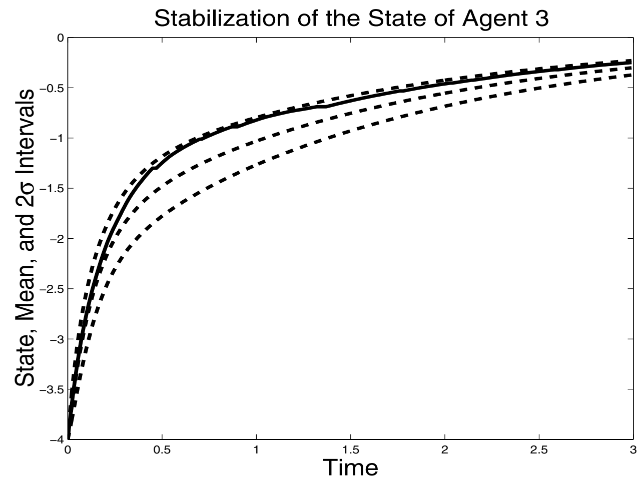

Fig. 1. The sample state trajectory for one agent (Agent 3) is plotted along with its mean value and 2-standard deviation intervals about the mean.

REFERENCES

[1] P. Seiler and R. Sengupta, "Analysis of communication losses in vehicle control problems," in *Proceedings of the American Control Conference*, pp. 1491-1496, June 2001.

[2] N. A. Lynch, *Distributed Algorithms*, Morgan Kaufmann Publishers: San Mateo, CA, 1996

[3] K. Tomsovic, D. Bakken, V. Venkatasubramanian, and A. Bose, "Designing the next generation of real-time control, communication, and computations for large power systems," submitted to the *Proceedings of the IEEE*.

[4] S. Bhowmik, K, Tomsovic, and A. Bose, "Communication models for third party load frequency control," *IEEE Transactions on Power Systems*, Vol. 19, No. 1, Feb. 2004, pp. 64-72.

[5] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, pp. 65-78, 2004.

[6] R. Olfati Saber and R. M. Murray, "Agreement problems in networks with directed graphs and switching topologies", *IEEE Conference on Decision and Control*, 2003.

[7] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fi xed and switching networks", submitted to *Automatica*, July 2003.

[8] S. Roy, A. Saberi, and K. Herlugson, 'Formation and Alignment of Distributed Sensing Agents with Double-Integrator Dynamics', accepted for publication in the IEEE Press Monograph *Sensor Network Operations*, Sep. 2004.

[9] S. Roy, A. Saberi, and K. Herlugson, 'A control-theoretic perspective on the design of distributed agreement protocols', submitted to *Automatica*, Jul. 2004.

[10] S. Roy and A. Saberi, "Static decentralized control of a single-integrator network with Markovian sensing topology", submitted to *Automatica*, Sep. 2004.

[11] M. Mesbahi, "On state-dependent dynamic graphs and their controllability properties," *IEEE Transactions on Automatic Control* (to appear).

[12] Y. Fang and K. A. Loparo, "Stabilization of continuous-time jump linear systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 10, Oct. 2002, pp. 1590-1603.

[13] Y. Fang, K. A. Loparo, and X. Feng, "Almost sure stability and δ-moment stability of jump-linear systems," *International Journal of Control*, vol. 59, no. 5, 1994, pp. 1281-1306.

[14] Y. Fang and K. A. Loparo, "Stochastic stability of jump linear systems", *IEEE Transactions on Automatic Control*, vol. 47, no. 7, Jul. 2002, pp. 1204-1208.

[15] J. A. Fax and R. M. Murray, "Information fbw and cooperative control of vehicle formations", submitted to *IEEE Transactions on Automatic Control*, April 2003.

[16] J. B. R. do Val, J. C. Geromel, and O. L. V Costa, "Uncoupled Riccati iterations for the linear quadratic control problem of discrete-time Markov jump linear systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 12, Dec. 1998, pp. 1727-1733.

[17] O. L. V Costa and M. D. Fragoso, "Comments on 'Stochastic stability of jump linear systems' ", *IEEE Transactions on Automatic Control*, vol. 49, no. 8, Aug. 2004, pp. 1414-1416.

[18] S. Roy, G. C. Verghese, and B. C. Lesieutre, "Moment-linear stochastic systems", in the *Proceedings of the First International Conference on Informatics in Control, Automation, and Robotics*, Aug. 2004.

[19] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press: New York, 1991.