2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

The Design of Multi-Lead-Compensators for Stabilization and Pole Placement in Double-Integrator Networks under Saturation

Yan Wan Sandip Roy Ali Saberi Anton Stoorvogel

Abstract— **We study decentralized controller design for stabilization and pole-placement, in a network of autonomous agents with double-integrator internal dynamics and arbitrary observation topology. We show that a simple multi-lead-compensator architecture, in particular one in which each agent uses a derivative-approximation compensator with three memory elements, can achieve both stabilization and effective pole placement. Through a scaling argument, we also demonstrate that the multi-lead-compensator can stabilize the double-integrator network under actuator saturation constraints. The multi-lead-compensator design is practical for modern dynamical network applications, in that it subdivides actuation effort and complexity among the agents and in many cases is robust to agent failure.**

I. INTRODUCTION

Through our efforts in studying control tasks in several modern dynamical networks [1], [2], [3], [4], [5], [6], we are convinced that network structure (i.e., the sensing/interaction interconnection structure among the agents) is critical in driving network dynamics, and hence must be exploited in controller design. Due to the crucial role played by the network structure, novel decentralized controller architectures that have the following two features are badly needed in dynamical network control applications: 1) control complexity and actuation are roughly equally contributed by all the agents, and 2) the controller can address control/algorithmic tasks in networks with very general sensing and/or interaction topologies and constraints such as actuator saturation. In this work, we introduce a novel decentralized dynamic controller that can guide network dynamics with arbitrary sensing structures—the very simple *multi-lead-compensator* controller. Specifically, we show that the design of the multi-lead-compensator—precisely, an LTI (linear time-invariant) decentralized state-space controller with a small number of memory elements used in each channel, that approximates a multiple-derivative feedback—allows *stabilization* and *pole-placement* in an autonomous-agent network model with a general sensing structure. We also adapt the design for stabilization under actuator saturation.

To motivate the network stabilization and pole placement problem addressed here, let us briefly review two bodies of literature: the recent efforts on autonomous-agent network control, as well as historical efforts in decentralized control.

The first three authors are with the School of Electrical Engineering and Computer Science, Washington State University. The fourth author is with the Department of Electrical Engineering, Mathematics and Computer Science at the University of Twente. The work was supported by National Science Foundation grants ECS-0528882 and ECCS-0725589, NAVY grants ONR KKK777SB001 and ONR KKK60SB0012, and National Aeronautics and Space Administration grant NNA06CN26A.

The many recent works on autonomous-agent network control are fundamentally derived from prominent work by Chua and his colleagues [7], [8] on network synchronization (see also the literature in the Physics community on synchronization, e.g.[9]). Chua in [7] gave the necessary and sufficient condition for a network with identical agents to achieve synchronization; and in [8] gave a graph interpretation of the condition when the network is diffusive, i.e., the sensing structure is described by a Laplacian matrix. Fax and Murray in [10] and Pogromsky in [11] brought forth control interpretations to the synchronization tasks in a diffusive network, and thus gave conditions for synchronization through control. We notice that network synchronization through control is closely connected to the network stabilization task, with only the distinction that network synchronization is concerned with the the stability of an invariant set while network stabilization is concerned with the stability of an equilibrium point. Thus very similar methods apply to both types of network problems, and in fact with this understanding, various other control tasks such as formation, agreement, alignment, tuning, consensus, and distributed partitioning [12], [13], [6], [14], [15] have been addressed in essentially similar ways.

Attempts have also been made to design stabilizing controllers in the classical decentralized literature. In a seminal work [16], Wang and Davison give an implicit sufficient and necessary condition for the existence of a stabilizing time-invariant dynamic controller for a general decentralized system. Decentralized controller [4] for network stabilization, and similarly other network control tasks, is difficult due to the limitations imposed by the sensing structures. The book [17] and related works view the network interconnections as disturbances, and provide controller designs based on that assumption. In an alternate direction, building on [16], Corfmat and Morse studied stabilization of *complete* systems in [18]. In their work, applying non-dynamic controller to all but one channel of a complete system makes the system controllable and observable from the remaining channel, and then a dynamical controller is applied to this single channel to achieve stabilization and also pole-placement for the whole system.

Now let us emphasize the contribution of the multi-lead-compensator controller design with respect to both the autonomous-agent network control and the existing decentralized controller design literature. We notice that the approaches in the existing literature (e.g., [10], [18], [17]) do not serve our goal of finding controller architectures that are suitable for many modern network control applications, for two reasons. 1) These existing designs are impractical to

978-1-4244-4524-0/09/$25.00 ©2009 AACC

implement: either the decentralized stabilization is achieved by hiding the contribution of network connections [17] (while in fact the network connections are critical in modern applications), or by making a single channel dominant in terms of actuation and complexity [18]. 2) Many studies, for instance those giving conditions for stabilization in the recent autonomous-agent-network literature, only work for a limited subclass of sensing structures such as ones specified by a Laplacian matrix [11], [10]. Hence, we are motivated to study the simple multi-lead-compensator architecture as an alternative. We will show that this controller can be designed for stabilization in networks with general sensing structures, using roughly equal actuations at each agent. Moreover, we stress here that the multi-lead-compensator architecture allows practical high performance design (pole placement), which is missing in the literature, as a further major contribution of this work. We focus here on the important class of double-integrator networks (networks with agents whose internal dynamics are double integrators), which are common models for autonomous-agent network applications (see e.g., [6], [15]). Our design for the double-integrator network is also illustrative of the design for much more general plants, which is applicable not only to autonomous-agent networks but ones with hardwired interconnections such as population dynamics [1].

Let us also discuss several recent works that have in fact addressed network topology and controller *design*, for high performance. Of interest, Boyd and his coworkers have used linear matrix inequality (LMI) techniques to optimize the Fiedler eigenvalue of a Laplacian matrix through design of an associated graph [20] (and hence to shape e.g. an associated single integrator network dynamics). In complement, building on a classical result of Fisher and Fuller [21], we have taken a structural approach to performance optimization through graph-edge and static decentralized controller design [2], [22], [23], [24]. This meshed control-theory and algebraic-graph-theory strategy has yielded designs for several families of network-interaction models and performance criteria, and has also permitted as to address the partial design problem. This paper shows that even very simple dynamical controllers when properly used can give significant freedom in shaping a network's response (for instance, allowing pole placement).

Finally, we also address controller design for double integrator networks with saturation. Let us briefly review the limited literature on controlling dynamical networks under saturation, to give context to this work. Stoorvogel and coworkers have given sufficient conditions for existence of decentralized controllers under saturation, in particular showing that stabilization is possible when the open-loop decentralized plant has 1) closed-left-half plane eigenvalues, 2) decentralized fixed modes only in the open left-half-plane, and 3) jw-axis axis eigenvalues with algebraic multiplicity 1 [27]. However, this study does not address the (obviously crucial) case where jw-axis eigenvalues are repeated, nor does it give a practical controller design. Meanwhile, our group has been able to give a sufficient condition on the observation topology under which a double-integrator network (which has repeated and defective $j\omega$-axis eigenvalues) can be stabilized under saturation, by focusing on a proportional-derivative control scheme [6]. As an alternative, for double-integrator networks with symmetric relative-position observations, Ren shows that a diffusive controller with a hyperbolic tangent nonlinearity can achieve a consensus or synchronization task [15]. Here, we present a systematic low-gain *dynamical controller design* for arbitrary double-integrator networks with actuator saturation. The design comprehensively addresses the problem of decentralized control in the presence of actuator saturation for the important double-integrator network model, and also focuses analysis and design efforts for more general decentralized plants with repeated $j\omega$-axis eigenvalues.

The remainder of the article is organized as follows. In Section II, we give the philosophy of our design and formulate the double-integrator network model. Section III addresses design of controllers for stabilization and pole-placement in the double-integrator network model, while Section IV addresses the case with actuator saturation.

II. Philosophy and Problem Formulation

In this section, we first discuss the philosophy underlying our multi-lead-compensator design, and then introduce the double-integrator network (without and with saturation) model formally.

Our multi-lead-compensator design is based on 1) construction of a high-gain feedback of multiple derivatives up to degree 2 to place the close-loop poles in desired locations in the OLHP (open left half plane); and 2) approximation of the multiple-derivative controller with lead compensators. The philosophy is that for double integrator networks, high gain feedback of output derivatives up to the degree of 2 can permit each agent to recover its local state information [4], and hence permit pole placement and stabilization. We emphasize that the novelty of the design resides in the concept that one higher derivative (here, the second derivative, or in other words the derivative equal to the *relative degree* of the local plant) is being used in feedback. This feedback concept has not been used in the literature. Moreover, since derivative controllers are not implementable due to their unbounded high frequency gains, we implement the derivative controllers using lead compensators. The lead compensators produce poles close to those of the derivative controller and also extra poles far inside the OLHP, and hence achieve stabilization and high performance. In this controller architecture, the control actions are distributed among all agents rather than being centered at a single agent. This architecture also has the advantage that it is more robust to network failures/attacks, and hence is practical for modern network applications. Finally, through a scaling argument (i.e., a slowing-down of the closed-loop dynamics), we can also address stabilization

[1]The more general case requires use of the special coordinate basis for linear systems, see our previous work on multiple-delay control for further details [19].

under saturation.

Next, let us introduce the **double-integrator network**, i.e. a decentralized system comprising n autonomous **agents** with double-integrator internal dynamics whose (scalar) observations are linear combinations of multiple agents' states. Precisely, we assume that each agent i has internal dynamics $\ddot{x}_i = u_i$, where we refer to x_i as the **position state** of agent i and $\dot{x}_i$ as the **velocity state**, and u_i is the input to agent i. For notational convenience, we define $\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$ and refer to it as the **full position state** of the network, and also define $\mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix}$. We assume each agent i makes a scalar observation $y_i = \mathbf{g}_i^T \mathbf{x}$, i.e. that its observation is a linear combination of the position states of various agents. We find it convenient to stack the observations into a vector, i.e. $\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$. We also stack vectors $\mathbf{g}_i^T$ to form a matrix $G = \begin{bmatrix} \mathbf{g}_1^T \\ \vdots \\ \mathbf{g}_n^T \end{bmatrix}$, which we refer to as the **topology matrix** since it captures the sensing/communication among the agents. In short, a double-integrator network comprises n agents that together have the dynamics

$$\begin{aligned} \ddot{\mathbf{x}} &= \mathbf{u} \\ \mathbf{y} &= G\mathbf{x}, \end{aligned} \tag{1}$$

where each agent i makes an observation y_i and can set the input u_i. We also consider a double-integrator network with agents that are subject to actuator saturation. That is, we consider a network of n agents that have the dynamics

$$\begin{aligned} \ddot{\mathbf{x}} &= \sigma(\mathbf{u}) \\ \mathbf{y} &= G\mathbf{x}, \end{aligned} \tag{2}$$

where $\sigma()$ represents a standard saturation function applied elementwise. Let us call this model the **double-integrator network with saturation**.

Our goal is to design a linear decentralized controller (mappings from each y_i to u_i) to stabilize the double-integrator-network's dynamics. As a further step, we seek a pole-placement controller, i.e. one that achieves the classical controller design goal of placing the eigenvalues of the closed-loop dynamics at desirable locations. This decentralized controller design problem for the double-integrator network is widely applicable, see [6].

III. Stabilization and Pole-Placement for the Double-Integrator Network

For the double-integrator network, it is necessary and sufficient for stabilization that G has full rank [6], regardless of whether centralized or decentralized control is considered and regardless of whether a linear or a non-linear time-varying (NLTV) controller is used (see also [16], [25]). Here, we will demonstrate not only stabilization but effective pole placement for arbitrary full rank G using the most limited of these schemes, namely an LTI state-space decentralized controller. In fact, we will show that a very simple controller—one that has third-order dynamics at each channel—suffices. The controller that we use may be viewed as applying first- and second-order lead compensation, or in other words approximating feedback of the first- and second- derivatives of the local observation, at each channel. This architecture, though very simple, is novel and specifically of use in the decentralized control context. Structurally, the novelty of the controller lies in that (approximate) feedback of derivatives up to and *including* the relative degree of the local dynamics is used. This is in contrast to the centralized setting, where controllers (whether designed using the observer-plus-state-feedback paradigm or in other ways) at their essence feed back derivatives up to one less than the plant's relative degree to achieve stabilization and pole placement [26].

We note that the dynamical controller design presented here enhances our existing work on stabilizing double-integrator networks [6], which is at its essence derived from Fisher and Fuller's classical result [21]. In fact, the proof of our main result in this paper also relies on this result. Thus, for the reader's convenience, let us describe the classical result of Fisher and Fuller here, before introducing and proving our main result.

Theorem 1: **(Fisher and Fuller)** Consider an $n \times n$ matrix A. If the matrix A has a nested sequence of n principal minors that all have full rank, then there exists a diagonal matrix K such that the eigenvalues of KA are in the open left half plane.

The following theorem, our main result, formalizes that stabilization and pole-placement can be achieved generally in the double-integrator network using third-order compensators at each channel. The proof of the theorem makes explicit the compensator design. Specifically, we describe how to design a controller so that sets of n closed-loop eigenvalues can be placed arbitrarily near to two desired locations (closed under conjugation) in the complex plane, while the remaining $3n$ eigenvalues are placed arbitrarily far left in the complex plane. Here is a formal statement:

Theorem 2: Consider a double-integrator network with arbitrary invertible graph-matrix G. Proper LTI compensators of order 3 can be applied at each channel, so as to place n eigenvalues each close to two desirable locations in the complex plane while driving the remaining $3n$ eigenvalues arbitrarily far left in the complex plane. Specifically, consider using a compensator at each agent i with transfer function $h_i(s) = k_o + \frac{k_1 s}{1+\epsilon\lambda_{fi} s} + \frac{k_2 s^2}{1+\epsilon s\lambda_{di}+\epsilon^2 s^2 \lambda_{zi}}$, and say that we wish to place n closed-loop eigenvalues at each of the roots of $s^2 + \alpha s + \beta$. By choosing k_2 sufficiently large, $k_1 = \alpha k_2$, and $k_o = \beta k_2$, and choosing λ_{fi}, λ_{di}, and λ_{zi} appropriately, n closed-loop eigenvalues can be placed arbitrarily close to each root of $s^2 + \alpha s + \beta$ as ϵ is made small, while the remaining $3n$ eigenvalues can be moved arbitrarily far left in the complex plane (in particular, having order $\frac{1}{\epsilon}$).

Proof:

The proof is in two steps. In the first step, we show that decentralized feedback of the observation and its first two derivatives can be used to to place the $(2n)$ closed-loop eigenvalues arbitrarily near to two locations in the complex plane, and in fact there is a parameterized family of controllers of this form that suffice. In the second step, we use this result to construct proper third-order LTI compensators at each channel that achieve the pole-placement specification given in the theorem statement.

Step 1: Let us study the closed-loop eigenvalues of the system when the (decentralized) control law $\mathbf{u}(t) = k_0\mathbf{y}(t) + k_1\dot{\mathbf{y}}(t) + k_2\ddot{\mathbf{y}}(t)$, where $k_0 = \beta k_2$ and $k_1 = \alpha k_2$, is used. The state-space representation of the closed loop system in this case is $\dot{\mathbf{X}} = A_c\mathbf{X}$, where $\mathbf{X} = \begin{bmatrix}\dot{\mathbf{x}}\\ \mathbf{x}\end{bmatrix}$ and $A_c = \begin{bmatrix}(I-k_2G)^{-1}k_1G & (I-k_2G)^{-1}k_0G\\ I & 0\end{bmatrix}$. Using the notation $\begin{bmatrix}X_1\\ X_2\end{bmatrix}$ for a right eigenvector of A_c, we have $A_c\begin{bmatrix}X_1\\ X_2\end{bmatrix} = \lambda\begin{bmatrix}X_1\\ X_2\end{bmatrix}$, which implies that $X_1 = \lambda X_2$, and $\lambda(I-k_2G)^{-1}\alpha k_2GX_2 + (I-k_2G)^{-1}\beta k_2GX_2 = \lambda X_1$. The latter yields: $\lambda\alpha k_2GX_2 + \beta k_2GX_2 = \lambda^2(I-k_2G)X_2$, or $(\lambda\alpha k_2 + \beta k_2 + \lambda^2 k_2)GX_2 = \lambda^2X_2$. This means that X_2 must be an eigenvector of G with, say, eigenvalue λ_i. In this notation, we have $(\lambda\alpha k_2 + \beta k_2 + \lambda^2k_2)\lambda_i = \lambda^2$, or $\lambda^2 - \frac{\alpha k_2\lambda_i}{1-k_2\lambda_i}\lambda - \frac{\beta k_2\lambda_i}{1-k_2\lambda_i} = 0$. Thus, the closed-loop eigenvalues are the roots of the characteristic equations $\lambda^2 - \frac{\alpha k_2\lambda_i}{1-k_2\lambda_i}\lambda - \frac{\beta k_2\lambda_i}{1-k_2\lambda_i}$, for $i = 1,\ldots,n$. Hence, by making k_2 sufficiently large, the coefficients of the characteristic equation can be made arbitrarily close to the coefficients of the quadratic equation $\lambda^2 + \alpha\lambda + \beta = 0$. From the continuous dependence of roots on parameters, the closed-loop poles thus come arbitrarily close to the roots of this characteristic equation, as desired. To summarize, when the presented multi-derivative compensator is used, the closed-loop eigenvalues can be made arbitrarily close to the two desired locations in the complex plane, for all k_2 sufficiently large.

Step 2: We now consider using a compensator at each channel i with transfer function $h_i(s) = k_o + \frac{k_1s}{1+\epsilon\lambda_{fi}s} + \frac{k_2s^2}{1+\epsilon s\lambda_{di}+\epsilon^2s^2\lambda_{zi}}$, where the gains k_0, k_1, and k_2 are those determined in Step 1, λ_{fi}, λ_{di}, and λ_{zi} are constants to be designed, and ϵ is a positive constant that will be designed sufficiently small after the other parameters have been designed. We note that this controller requires at most three memory elements at each channel to implement.

Substituting for the controllers' transfer functions, one immediately find the closed-loop characteristic polynomial. In particular, the closed-loop system's poles are values s such that $Q(s) = (I+\epsilon s\Lambda_f)(I-k_2G+\epsilon s\Lambda_d+\epsilon^2s^2\Lambda_z)s^2 - (I+\epsilon s\Lambda_d+\epsilon^2s^2\Lambda_z)k_1Gs - (I+\epsilon s\Lambda_f)(I+\epsilon s\Lambda_d+\epsilon^2s^2\Lambda_z)k_0G$ lose rank, where Λ_f, Λ_d, and Λ_z are diagonal matrices with ith diagonal entry given by λ_{fi}, λ_{di}, and λ_{zi}, respectively. We notice that the closed-loop system has $5n$ poles (counting multiplicities).

To continue, we note that $Q(s)$ can be written as $Q(s) = s^2I - k_0G - sk_1G - s^2k_2G + \epsilon M_1(s) + \epsilon^2M_2(s)$, where $M_1(t)$ and $M_2(s)$ do not depend on ϵ. Let us first consider the $2n$ values s for which $s^2I - k_0G - sk_1G - s^2k_2G$ loses rank. We note that these are precisely the closed-loop poles when the derivative-based controller is used, and so these values of s are in two groups of n, arbitrarily near to the two desired pole locations. It follows easily from perturbation arguments that, thus, n poles of the closed-loop system upon lead-compensator control (values s such that $Q(s)$ loses rank) are arbitrarily close to each desired pole location.

What remains to be shown is that the remaining poles are order $\frac{1}{\epsilon}$ and indeed can be placed in the left-half-plane. To see this, let us rewrite the Laplace-domain expression in terms of $\overline{s} = \epsilon s$. Doing so, we recover that $R(\overline{s}) = \epsilon^2Q(\overline{s}) = \overline{s}^2(I+\Lambda_f\overline{s})(I-k_2G+\overline{s}\Lambda_d+\overline{s}^2\Lambda_z) + \epsilon N_1(\overline{s}) + \epsilon^2N_2(\overline{s})$. To characterize the values $\overline{s}$ such that $R(\overline{s})$ and hence $Q(\overline{s})$ lose rank, let us first consider $T(\overline{s}) = \overline{s}^2(I+\Lambda_f\overline{s})(I-k_2G+\overline{s}\Lambda_d+\overline{s}^2\Lambda_z)$. We recognize that $T(\overline{s})$ loses rank at $\overline{s} = 0$ with multiplicity $2n$, as well as at the $3n$ values $\overline{s}$ such that $(I+\Lambda_f\overline{s})(I-k_2G+\overline{s}\Lambda_d+\overline{s}^2\Lambda_z)$ loses rank. We note that these $3n$ values are non-zero as long as $I-k_2G$ is made full rank (which we shall shortly guarantee), and we choose Λ_f, Λ_d, and Λ_z full rank and $k_2 \neq 0$, as we shall do. In this case, we see immediately from perturbation arguments that the polynomial $R(\overline{s})$ loses rank at $2n$ values $\overline{s}$ that approach the origin as ϵ is made small, as well as at $3n$ other values $\overline{s}$ that approach the $3n$ non-zero points in the complex plane where $(I+\Lambda_f\overline{s})(I-k_2G+\overline{s}\Lambda_d+\overline{s}^2\Lambda_z)$ loses rank, as ϵ is made small. Rewriting all these values in terms of s rather than $\overline{s}$, we see that the closed-loop system has $2n$ poles that are close to the origin in that they do not grow as fast as $\theta(\frac{1}{\epsilon})$ (and which we have already characterized to be close to two desired locations in the complex plane), as well as $3n$ poles of order $\frac{1}{\epsilon}$ if the poles of $(I+\Lambda_f)(I-k_2G+\overline{s}\Lambda_d+\overline{s}^2\Lambda_z)$ are nonzero (as we will show shortly).

Finally, let us construct the controller so that the values $\overline{s}$ for which $(I+\Lambda_f\overline{s})(I-k_2G+\overline{s}\Lambda_d+\overline{s}^2\Lambda_z)$ loses rank are all in the OLHP, and hence complete the proof. Clearly, n of these values are the $\overline{s}$ such that $(I+\Lambda_f\overline{s})$ loses rank. We can make these values negative and real by choosing each λ_{fi}, and hence Λ_F, positive and real.

Next, let us consider the $2n$ values $\overline{s}$ such that $(I-k_2G+\overline{s}\Lambda_d+\overline{s}^2\Lambda_z)$ loses rank. Since we have assumed Λ_z is full rank, we can equivalently find $\overline{s}$ such that $\overline{s}^2I + \Lambda_z^{-1}\Lambda_d\overline{s} + \Lambda_z^{-1}(I-k_2G)$ loses rank. To continue, we note that all principal minors of $I-k_2G$ are full rank for all k_2 except those in a particular finite set, i.e. for all k_2 except those that are inverses of eigenvalues of the principal minors of G. Hence, for any design with k_2 large enough, $I-k_2G$ has a nested sequence of principal minors of full rank. Using any such design, we thus can choose Λ_z to place the eigenvalues of $\Lambda_z^{-1}(I-k_2G)$ at positive real values, according to the classical result of Fisher and Fuller (quoted as Theorem 1 above). Finally, let us choose $\Lambda_d = \Lambda_z^{-1}$. In this case, we see from simple eigenanalysis that the values $\overline{s}$ for which rank is lost are the solutions of the n scalar equations $\overline{s}^2 + \overline{s} + \lambda_i$, where each λ_i is an eigenvalue of $\Lambda_z^{-1}(I - k_2 G)$. Thus, we obtain that all solutions $\overline{s}$ are in the OLHP. □

■

We have thus demonstrated a multi-lead-compensator decentralized controller design for an arbitrary double-integrator network, which achieves stabilization as well a certain *group* pole placement. Let us stress that this group pole-placement capability gives us wide freedom to shape the dynamical response (in terms of settling and robustness properties), including by guaranteeing phase margin in the design through an inverse optimality argument (see [4]). It is worth noting that further design of the $\theta(1)$ poles is possible through consideration of heterogeneous derivative controller, however this further effort does not provide much further improvement in design performance and so we omit the details.

The multi-lead-compensator introduced here is promising for modern network applications because it is tailored to distribute controller effort and exploit the network topology (in particular by obtaining the local state at each agent through a combination of uniform feedback control and derivative estimation). Let us conclude the discussion of the multi-lead-compensator by discussing several conceptualizations and extensions of the design:

1) Although we have presented the design assuming each agent makes a single observation for convenience, the result automatically generalizes to the case where each agent may make multiple observations. In particular, whenever stabilization is possible (see [6] for conditions derived from [16]), the above pole-placement design can be applied as follows. First, the multiple observations made by the agent should be combined linearly, with each observation weighted by a randomly chosen constant. It follows automatically that this reduced (single-input, single-observation) double-integrator network can be stabilized (with probability 1), and hence the design technique introduced above can be applied.
2) The design that we have presented can be interpreted as comprising an estimator and a state-feedback controller. Specifically, if the pure derivative controller is used (with k_2 large), the agents can be viewed as immediately obtaining their local state (in particular, by rearranging their initial conditions in such a way that the second-derivative estimate and hence local state estimate are precise); thus, state feedback control can be used. In practice, such an immediate estimation/rearrangement of initial conditions is not implementable. Instead, the lead compensator design achieves estimation (in part through rearrangement of the state) at a faster time scale than the state feedback response, but not immediately. We note that such use of fast observers is classical, and so our design naturally fits the estimation-plus-state-feedback paradigm. However, the design is fundamentally different from the traditional one, in that the state estimation is only possible when the feedback is in force—that is, the estimation and control tasks are *NOT* decoupled.
3) In dynamical network stabilization tasks, robustness to agent failure is an important concern. For instance, in an autonomous vehicle network, if the failure of a single agent can spoil the stability of the entire network, the viability of a stabilization design really falls in question. Let us briefly expand on this particular robustness problem. In this case, by agent failure, we mean there exist certain agents that can not be observed by all other agents. Robustness in the presence of agent failure problem is concerned with whether the rest of the agents can still achieve stability using the original controller design. Mathematically, this is the problem of whether, if all rows and columns in the sensing structure associated with the failed agents are removed, stability in the reduced dimensional system remains. We notice that current literature on decentralized controller design does not address this important issue. For instance, the dominant channel design in [18] is highly sensitive to the failure of the dominant channel, due to the significant role played by this single channel in stabilization. In the contrast, since in our lead-compensator design, all agents contribute roughly equally to the stabilization task, this design appears to be more robust to agent failure. For broad classes of sensing structures, e.g., those for which $I - k_2 G$ is strictly *D-stable* through a diagonal (sign-pattern) scaling, stability is maintained in the presence of any number of agent failures. This is because the eigenvalues of the principal submatrices of $\Lambda_z^{-1}(I - k_2 G)$ remain in the OLHP and hence so do those of the closed-loop state matrix. Clearly, subclasses of G that satisfy the above include strictly D-stable, positive definite, and grounded Laplacian topology matrices (see Figure 1 for a full illustration).
4) We stress that the value of the gain k_2 needed for stabilization or approximate pole placement is dependent on the structure of the graph matrix G: we refer the reader to our earlier work [4] for precise bounds on the gain. Because the actuation capability is distributed among the agents, and because the gain parameter can be tuned based on the network structure, the gain often need not be large to achieve stabilization and approximate pole placement.

IV. Stabilization under Actuator Saturation

In this section, we show that a multi-lead-compensator can semi-globally stabilize a double-integrator network under saturation, for an arbitrary graph matrix G. We stress again that the double-integrator network has $2n$ poles at the origin. Hence, the result is an expansion of the sufficient condition for the existence of a stabilizing controller provided in [27].

In the following Theorem 3, we show the design of the stabilizing multi-lead-compensator using a low-gain strategy.

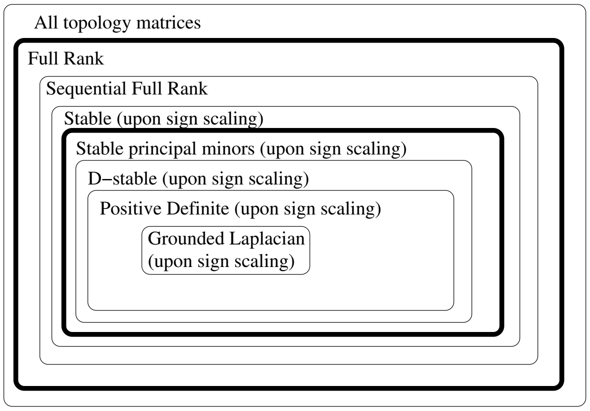

Fig. 1. We diagram several matrix classes that are of interest in representing a double integrator network's sensing topology. A multi-lead-compensator design is possible whenever the topology matrix is full rank, and a design that is robust to agent failures is possible if the topology matrix has stable principal minors to within a sign scaling (a scaling of each row by ± 1).

More specifically, we discuss a scaling of the compensator design presented in Theorem 2, that yields semi-global stabilization under actuator saturation. Here is the result:

Theorem 3: Consider a double-integrator network with input saturation, and arbitrary invertible graph-matrix G. Say that compensators $h_i(s) = k_o + \frac{k_1 s}{1+\epsilon\lambda_{fi}s} + \frac{k_2 s^2}{1+\epsilon s\lambda_{di}+\epsilon^2 s^2\lambda_{zi}}$ have been designed for each agent i according to Theorem 1, so that stabilization is achieved in the double-integrator network without saturation. Then the network with input saturation can be semiglobally stabilized using at each agent i the parameterized family of proper LTI compensators $\hat{h}_{i,\hat{\epsilon}}(s) = k_o\hat{\epsilon}^2 + \frac{\hat{\epsilon}k_1 s}{1+\frac{\epsilon}{\hat{\epsilon}}\lambda_{fi}s} + \frac{k_2 s^2}{1+\frac{\epsilon}{\hat{\epsilon}}s\lambda_{di}+\frac{\epsilon^2}{\hat{\epsilon}^2}s^2\lambda_{zi}}$ That is, for any specified ball of plant and compensator initial conditions W, there exists $\hat{\epsilon}^*(W)$ such that, for all $0 < \hat{\epsilon} \le \hat{\epsilon}^*(W)$, the compensator with the transfer function $\hat{h}_{i,\hat{\epsilon}}(s)$ at each channel achieves local stabilization of the origin and contains W in its domain of attraction.

Proof: Without loss of generality, we can limit ourselves to examining the response from the plant initial conditions, since the component of the response due to the precompensator initial conditions can be made arbitrarily small through static pre- and post-scaling of the compensator at each agent, see e.g., [28].

In order to prove that the proposed controller semiglobally stabilizes the double-integrator network under input saturation, it suffices to show that, for any bounded set of initial conditions W the ∞-norm of the input $||\mathbf{u}(t)||$ where $t \ge 0$ remains upper bounded by 1, and also the dynamics without saturation are asymptotically stable. Thus, we can verify semi-global stabilization by showing that, for any bounded set of initial conditions W the norm of the input $||\mathbf{u}(t)||$ where $t \ge 0$ scales by $\hat{\epsilon}$ and further the closed-loop dynamics without actuator saturation are asymptotically stable. Let us prove this through a spectral argument.

Let us first consider applying the new scaled controller, i.e. the controller with transfer function $\widehat{H}(s) = [diag(h_i(s))]$. In the Laplace domain, the closed-loop dynamics of the double-integrator network ignoring saturation (when this scaled controller is used) are given by

$$s^2\mathbf{X}(s) - s\mathbf{x}(0) - \dot{\mathbf{x}}(0) = k_o\hat{\epsilon}^2 G\mathbf{X}(s) + \\ (I + \frac{\epsilon}{\hat{\epsilon}}\Lambda_f s)^{-1}\hat{\epsilon}sk_1G\mathbf{X}(s) + (1 + \frac{\epsilon}{\hat{\epsilon}}\Lambda_d s + \frac{\epsilon^2}{\hat{\epsilon}^2}\Lambda_z s^2)^{-1}k_2 s^2 G\mathbf{X}(s), \tag{3}$$

where Λ_f, Λ_d, and Λ_z are diagonal matrices whose ith diagonal entries are λ_{fi}, λ_{di}, and λ_{zi}, respectively.

Let us apply the change of variables $\hat{\epsilon}\bar{s} = s$, and scale both sides of Equation 4 by $\frac{1}{\hat{\epsilon}^2}$. The closed-loop system dynamics in terms of $\bar{s}$ in the Laplace domain becomes

$$\bar{s}^2\mathbf{X}(\hat{\epsilon}\bar{s}) - \frac{1}{\hat{\epsilon}}\bar{s}\mathbf{x}(0) - \frac{1}{\hat{\epsilon}^2}\dot{\mathbf{x}}(0) = k_oG\mathbf{X}(\hat{\epsilon}\bar{s}) \\ +(I + \epsilon\Lambda_f\bar{s})^{-1}\bar{s}k_1G\mathbf{X}(\hat{\epsilon}\bar{s}) + (1 + \epsilon\Lambda_d\bar{s} + \epsilon^2\Lambda_z\bar{s}^2)^{-1}k_2\bar{s}^2G\mathbf{X}(\hat{\epsilon}\bar{s}). \tag{4}$$

From Equation 5, we get

$$\mathbf{X}(\hat{\epsilon}\bar{s}) = \\ (\bar{s}^2 I - k_oG - (I + \epsilon\Lambda_f\bar{s})^{-1}\bar{s}k_1G - (1 + \epsilon\Lambda_d\bar{s} + \epsilon^2 + \Lambda_z\bar{s}^2)^{-1}k_2\bar{s}^2G)^{-1}\frac{1}{\hat{\epsilon}}\bar{s}\mathbf{x}(0) \\ +(\bar{s}^2 I - k_oG - (I + \epsilon\Lambda_f\bar{s})^{-1}\bar{s}k_1G - (1 + \epsilon\Lambda_d\bar{s} + \epsilon^2\Lambda_z\bar{s}^2)^{-1}k_2\bar{s}^2G)^{-1}\frac{1}{\hat{\epsilon}^2}\dot{\mathbf{x}}(0) \\ = \frac{1}{\hat{\epsilon}^2}((\bar{s}^2 I - k_oG - (I + \epsilon\Lambda_f\bar{s})^{-1}\bar{s}k_1G - (1 + \epsilon\Lambda_d\bar{s} + \epsilon^2\Lambda_z\bar{s}^2)^{-1}k_2\bar{s}^2G)^{-1}\hat{\epsilon}\bar{s}\mathbf{x}(0) \\ +(\bar{s}^2 I - k_oG - (I + \epsilon\Lambda_f\bar{s})^{-1}\bar{s}k_1G - (1 + \epsilon\Lambda_d\bar{s} + \epsilon^2\Lambda_z\bar{s}^2)^{-1}k_2\bar{s}^2G)^{-1}\dot{\mathbf{x}}(0)) \tag{5}$$

On the other hand, we note that using the original controller design with transfer function $H(\bar{s})$ and initial conditions $(\hat{\epsilon}\mathbf{x}(0), \dot{\mathbf{x}}(0))$ gives us (for the double integrator network ignoring saturation)

$$\overline{\mathbf{X}}(\bar{s}) = \\ ((\bar{s}^2 I - k_oG - (I + \epsilon\Lambda_f\bar{s})^{-1}\bar{s}k_1G - (1 + \epsilon\Lambda_d\bar{s} + \epsilon^2\Lambda_z\bar{s}^2)^{-1}k_2\bar{s}^2G)^{-1}\hat{\epsilon}\bar{s}\mathbf{x}(0) + \\ (\bar{s}^2 I - k_oG - (I + \epsilon\Lambda_f\bar{s})^{-1}\bar{s}k_1G - (1 + \epsilon\Lambda_d\bar{s} + \epsilon^2\Lambda_z\bar{s}^2)^{-1}k_2\bar{s}^2G)^{-1}\dot{\mathbf{x}}(0)), \tag{6}$$

where we have used the notation $\overline{\mathbf{X}}(\bar{s})$ for the Laplace transform of the state to avoid confusion.

Equations (5) and (6) together inform us that $\hat{\epsilon}\mathbf{X}(\hat{\epsilon}\bar{s})$ is $\overline{\mathbf{X}}(\bar{s})$ scaled by $\frac{1}{\hat{\epsilon}}$. Hence, clearly, the double-integrator network's response adopting controller with transfer function $\widehat{H}_{\hat{\epsilon}}(s)$ is the response of the double-integrator network using the controller with transfer function $H(\bar{s})$ with the initial conditions $(\hat{\epsilon}\mathbf{x}(0), \dot{\mathbf{x}}(0))$, scaled in amplitude by $\frac{1}{\hat{\epsilon}}$ and in frequency also by $\frac{1}{\hat{\epsilon}}$. Hence with just a little algebra, we see that the input norm $||\mathbf{u}(t)||$ is upper-bounded by a multiple of $\hat{\epsilon}$. Furthermore, we directly recover from Equations (5) and (6) that the closed-loop poles scale with $\hat{\epsilon}$ upon use of the scaled controller, and so asymptotic stability is maintained. Thus, semi-global stabilization is achieved.

■

In the above theorem, we have shown that a low-gain scaling of a (decentralized) multi-lead-compensator stabilizes a double-integrator network under saturation, for an arbitrary full-rank topology matrix G. Thus, we have fully addressed design of low-gain decentralized controllers for the double-integrator network with saturation. Let us conclude with two remarks about the design:

1) Let us distinguish the approach to design taken here with the traditional approach for centralized systems with saturation [29], [30]. In the low-gain design for centralized plants, actuation capabilities are subdivided between the observer and state feedback. In contrast, the double-integrator network requires integrated design of the entire dynamical controller, and hence we need a scaling of the full design to address control under saturation.

2) It is an open question as to whether decentralized plants with repeated $j\omega$-axis eigenvalues (and with all eigenvalues in the CLHP and all decentralized fixed modes in the OLHP) are amenable to semi-global stabilization. This first result shows that there is some promise for achieving semi-global stabilization broadly.

REFERENCES

[1] Y. Wan and S. Roy, "A scalable methodology for evaluating and designing coordinated air traffic flow management strategies under uncertainty," *IEEE Transactions on Intelligent Transportation Systems*, in press. Short version in *Proceedings of AIAA 2007 Guidance, Navigation, and Control Conference*, Hilton Head, South Carolina, 20-23 August, 2007.

[2] Y. Wan, S. Roy, and A. Saberi, "Designing spatially-heterogeneous strategies for control of virus spread," *IET Systems Biology*, vol. 2, No. 4, pp. 184–201, 2008. Short version in *Proceedings of 46th IEEE Conference on Decision and Control*, New Orleans, Louisiana, 12-14 December, 2007.

[3] Y. Wan, S. Roy, and A. Saberi, "A stochastic automaton-based algorithm for flexible and distributed network partitioning," *International Journal of Distributed Sensor Networks*, vol. 4, no. 3, pp. 223–246, 2008.

[4] Y. Wan, S. Roy, A. Saberi, and A. Stoorvogel, "A multiple-derivative and multiple-delay paradigm for decentralized controller design: introduction using the canonical double-integrator network," submitted to *Automatica*. Short version in *Proceedings of AIAA 2008 Guidance, Navigation, and Control Conference*, Honolulu, Hawaii, 18-21 August, 2008.

[5] Y. Wan, S. Roy, and A. Saberi, "A new focus in the science of networks: toward methods for design," *Proceedings of the Royal Society A*, vol. 464, pp. 513–535, March, 2008.

[6] S. Roy, A. Saberi, and K. Herlugson, "Formation and alignment of distributed sensing agents with double-integrator dynamics," IEEE Press Monograph on *Sensor Network Operations*, April 2006.

[7] C. W. Wu and L. Chua, "Application of kronecker products to the analysis of systems with uniform linear coupling, " *IEEE Transactions on Circuits and Systems I*, vol. 42, no. 10, pp. 775-779, October 1995.

[8] C. W. Wu and L. Chua, "Application of graph theory to the synchronization in an array of coupled nonlinear oscillators, " *IEEE Transactions on Circuits and Systems I*, vol. 42, no. 8, pp. 494-497, October 1995.

[9] S. H. Strogatz, "Coupled Oscillators and Biological Synchronization," *Scientific American*, vol. 269, no. 6, pp. 102-109, Dec. 1993.

[10] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465-1476, September 2004.

[11] A. Pogromsky and H. Nijmeijer, "Cooperative oscillatory behavior of mutually coupled dynamical systems," *IEEE Transactions on Circuits and Systems*, Part I, vol. 48, no. 2, Feb. 2001.

[12] J. Baillieul and P. J. Antsaklis, "Control and communication challenges in networked real-time systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 9-28, Jan. 2007.

[13] S. Roy, A. Saberi, and A. Stoorvogel, "Toward a control theory for networks," *International Journal of Robust and Nonlinear Control*, vol. 17, no. 10-11, p. 897, July 2007.

[14] Z. Duan, J. Wang, G. Chen, and L. Huang, "Stability analysis and decentralized control of a class of complex dynamical networks, " *Automatica*, vol. 44, pp. 1028-1035, 2008.

[15] W. Ren, "On consensus algorithms for double-integrator dynamics," *IEEE Transactions on Automatic Control*, vol. 53, no. 6, pp. 1503-1509, Jul. 2008.

[16] S. Wang and E. J. Davison, "On the stabilization of decentralized control systems," *IEEE Transactions on Automatic Control*, vol. AC-18, pp. 473-478, October 1973.

[17] D. Siljak, *Decentralized Control of Complex Systems,* Academic Press: Boston, 1994.

[18] J. P. Corfmat and A. S. Morse, "Decentralized control of linear multivariable systems," *Automatica*, vol. 12, pp. 479-495, 1976.

[19] Y. Wan, S. Roy, A. Saberi, and A. Stoorvogel, "A multiple-derivative and multiple-delay paradigm for decentralized controller design: uniform rank systems," submitted to *Automatica*.

[20] S. Boyd, "Convex optimization of graph laplacian eigenvalues," in *Proceedings of the International Congress of Mathematicians*, vol. 3, pp. 1311-1319, 2006.

[21] M. E. Fisher and A. T. Fuller,"On the Stability of matrices and the convergence of linear iterative processes," *Proceedings of the Cambridge Philosophical Society*, vol. 45, pp. 417-425, 1958.

[22] S. Roy, A. Saberi, and P. Petite,"Scaling: a canonical design problem for networks," in the *Proceedings of the 50th American Control Conference*, Minneapolis,MN, June 2006. Extended version in the *International Journal of Control* (by S. Roy and A. Saberi), vol. 80, no. 8, pp. 1342-1353, 2007.

[23] Y. Wan, S. Roy, X. Wang, A. Saberi, T. Yang, M. Xue, and B. Malek, "On the structure of graph edge designs that optimize the algebraic connectivity," in *Proceedings of 47th IEEE Conference on Decision and Control*, Cancun, Mexico, 9–11 December, 2008.

[24] S. Roy, Y. Wan, and A. Saberi, "On time-scale designs for networks," *International Journal of Control*, in press. Short version in *Proceedings of 47th IEEE Conference on Decision and Control*, Cancun, Mexico, 9–11 December, 2008.

[25] B. D. O. Anderson and J. B. Moore, "Time-varying feedback laws for decentralized control," *IEEE Transactions on Automatic Control*, Vol. 26, No. 5, pp. 1133-1139, October 1981.

[26] A. Saberi, B. M. Chen, and P. Sannuti, *Loop Transfer Recovery: Analysis and Design*, Springer-Verlag, 1993.

[27] A. Stoorvogel, A. Saberi, C. Deliu, and P. Sannuti, "Decentralized stabilization of time-invariant systems subject to actuator saturation," *Advanced Strategies in Control Systems with Input and Output Constraints*, LNCIS series (S. Tarbouriech et al eds.), Springer-Verlag, June 2006.

[28] S. Roy, Y. Wan, A. Saberi, and B. Malek, "An alternative approach to designing stabilizing compensators for saturating linear time-invariant plants," to appear in *the Proceedings of 47th IEEE Conference on Decision and Control*, Cancun, Mexico, December 9-11, 2008. PDF available at www.eecs.wsu.edu/~ywan.

[29] Z. Lin and A. Saberi, "Semi-global exponential stabilization on linear systems subject to "input saturation" via linear feedbacks," *Systems and Control Letters*, vol. 21, pp. 225-239, 1993.

[30] Z. Lin and A. Saberi, "A semi-global low-and-high gain design technique for linear systems with input saturation – stabilization and disturbance rejection," *International Journal of Robust and Nonlinear Control*, vol. 5, pp. 381-398, 1995.