Systems & Control Letters 94 (2016) 37–45

Contents lists available at ScienceDirect

Systems & Control Letters

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

Robust asymptotic model matching and its application to output synchronization of heterogeneous multi-agent systems

Lijun Zhu [a], Zhiyong Chen [b,*], Richard Middleton [b]

[a] Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong, China

[b] School of Electrical Engineering and Computer Science, The University of Newcastle, Callaghan, NSW 2308, Australia

ARTICLE INFO

Article history:
Received 4 March 2015
Received in revised form
18 April 2016
Accepted 20 April 2016

Keywords:
Multi-agent systems
Cooperative control
Heterogeneous systems
Robust control
Output regulation

ABSTRACT

Output synchronization of heterogeneous multi-agent systems has been one of the most interesting cooperative control problems. This paper first gives a brief survey of the research on the problem from which we see that the problem can be solved in a two-step manner with the aid of a properly designed local reference for each agent: (i) a controller is designed for each agent to achieve the trajectory regulation of the agent output to its associated reference; (ii) network collaboration is added to achieve consensus among references. In the presence of system uncertainties, the robust trajectory regulation problem in (i) can be solved by an internal model design. In this paper, we formulate a novel robust asymptotic model matching problem which is less conservative than trajectory regulation and can be solved by a static controller not relying on an internal model. Moreover, network collaboration is designed in (ii) within the so-called output communication setting such that consensus among references occurs concurrently with robust asymptotic model matching. As a result, output synchronization of heterogeneous multi-agent systems is achieved with a novel approach.

1. Introduction

The research interest in cooperative control of multi-agent systems was initially inspired by nature phenomena such as animals' group behaviors [1]. The theoretical progress has also contributed to applications in many areas such as formation control of autonomous vehicles [2], distributed sensor networks, load balancing [3] and power networks [4]. The main characteristics of cooperative control of multi-agent systems is that it relies on only local communication between physically or virtually connected neighboring agents. Due to this distributed control mechanism, the system is less vulnerable to agent failure. This paper is concerned with one of the most interesting problems in cooperative control of the multi-agent systems, that is, the output synchronization problem, which aims to seek agreement on agents' outputs. An output synchronization problem is also called a consensus problem with the former focusing more on agent dynamics and the later on network communication mechanism [5].

The output synchronization problem has been widely studied for multi-agent systems of simple and homogeneous dynamics, e.g., single or double-integrators [6,7], where it is also called a consensus problem. A general output synchronization of homogeneous network is formulated on a group of identical dynamics described by $\dot{x}_i = Ax_i + Bu_i,\ \ y_i = Cx_i + Du_i,\ \ i = 1, \ldots, N$; see [8–10], among others. The input u_i for each agent is designed to achieve agreement among the outputs $y_i, i = 1, \ldots, N$. When the network is synchronized, the control input vanishes and all agents have their behaviors automatically governed by the "homogeneous kernel", i.e., $\dot{x}_i = Ax_i$.

Conversely, a heterogeneous network is composed of non-identical agents whose dynamics are described by $\dot{x}_i = A_ix_i + B_iu_i,\ \ y_i = C_ix_i + D_iu_i,\ \ i = 1, \ldots, N$. As distinct from the homogeneous scenario, "kernel" dynamics are not automatically defined for an agreed behavior. Therefore, it is unavoidable to manually specify a set of homogeneous reference dynamics for each agent, e.g., $\dot{z}_i = A_oz_i + B_ov_i,\ \ \hat{y}_i = C_oz_i$, with the auxiliary control input v_i. It is expected that when synchronized, all agents have their behaviors governed by the specified "homogeneous kernel", i.e., $\dot{z}_i = A_oz_i$.

With the aid of reference models, output synchronization is achieved through two steps. In the first step, a controller u_i is designed for each agent such that the agent output is regulated to the associated local reference in the sense of $\lim_{t\to\infty} y_i(t) - \hat{y}_i(t) = 0$. In the second step, additional network collaborative control

* Corresponding author.
E-mail addresses: ljzhu@eee.hku.hk (L. Zhu), zhiyong.chen@newcastle.edu.au (Z. Chen), richard.middleton@newcastle.edu.au (R. Middleton).

http://dx.doi.org/10.1016/j.sysconle.2016.04.003
0167-6911/

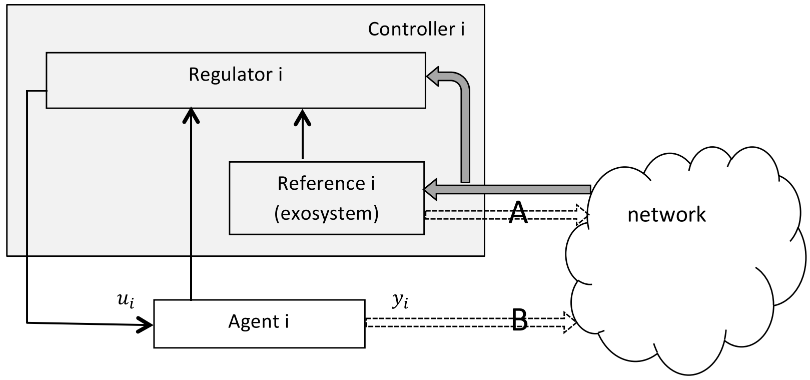

Fig. 1. Schematic diagram of local control design for each agent $i = 1, \ldots, N$. Solid line arrows represent local measurement and control signal flows; block arrows represent network communication signal flows. For state communication, we permit transmission of the internal reference state represented by the block arrow "A". For output communication, the agent output is transmitted across the network represented by the block arrow "B".

is added to v_i (and u_i if necessary) to achieve consensus among homogeneous references in the sense of $\lim_{t\to\infty} z_i(t) - z_j(t) = 0, \quad \forall i, j = 1, \ldots, N$. Next, we will review the research status on these two steps and hence introduce the motivation of this paper. The schematic diagram of control design for each agent is illustrated in Fig. 1.

I. Regulation of an agent to its local reference

When the agent dynamics involve no uncertainties, the regulation can always be achieved by a controller (called a regulator in Fig. 1) using feedforward compensation. Specifically, the regulation problem essentially aims to achieve $\lim_{t\to\infty} y_i(t) - \hat{y}_i(t) = 0$ between the agent $\dot{x}_i = A_i x_i + B_i u_i, \ y_i = C_i x_i + D_i u_i$ and the virtual reference $\dot{z}_i = A_o z_i + B_o v_i, \ \hat{y}_i = C_o z_i$. (For linear systems, $B_o v_i$ does not influence system stability and is therefore left in the second step of consensus design for discussion. With $B_o v_i$ not taken into consideration, the reference $\dot{z}_i = A_o z_i$ is called a virtual reference for convenience.) This is a standard formulation of an *output regulation problem* in literature. It is well known that a static controller $u_i = K_i(x_i - \Pi_i z_i) + \Gamma_i z_i$ solves the problem where K_i is the stabilizing matrix and $\Pi_i z_i$ and $\Gamma_i z_i$ are the feedforward compensation satisfying the Sylvester equations $\Pi_i A_o = A_i \Pi_i + B_i \Gamma_i$ and $C_i \Pi_i + D_i \Gamma_i = C_o$. Such a feedforward design was used in [11] (although an observer state $\hat{x}_i$ was implemented in the controller instead of x_i). It is worth noting that a feedforward compensation controller was studied in [12,13] for nonlinear systems. To achieve exact regulation, the feedforward compensation depends on precise calculation of the reference trajectory $\hat{y}_i(t) = C_o z_i(t)$ and its time derivatives. A main feature of the work in [12] is that the reference $\dot{z}_i = A_o z_i + B_o v_i$ is properly designed such that the relative degree between v_i and $\hat{y}_i$ is sufficiently high. As a result, the network communication signal v_i does not appear in $\hat{y}_i$ or its required time derivatives. A similar technique was also reported in [13] for a subtly different setup where the reference dynamics are more general but not explicitly specified.

When the agent dynamics involve uncertainties, exact feedforward compensation for achieving regulation becomes impossible. For example, Π_i and Γ_i cannot be precisely determined according to the Sylvester equations if the system matrices A_i, B_i, C_i, D_i contain uncertainties. Therefore, researchers appeal to the so-called internal model principle. Specifically, for linear systems, the regulation problem essentially aims to achieve $\lim_{t\to\infty} y_i(t) - \hat{y}_i(t) = 0$ between the agent $\dot{x}_i = A_i(w)x_i + B_i(w)u_i, \ y_i = C_i(w)x_i + D_i(w)u_i$ and the virtual reference $\dot{z}_i = A_o z_i, \ \hat{y}_i = C_o z_i$ (system uncertainties are explicitly represented by w). This is a standard formulation of a *robust output regulation problem* in the literature, which can be solved by a dynamic controller $u_i = k_i(\eta_i, x_i, y_i, z_i), \ \dot{\eta}_i = \zeta_i(\eta_i, x_i, y_i, z_i)$ (called a regulator in Fig. 1). In the literature on the robust output regulation problem, the virtual reference $\dot{z}_i = A_o z_i$ is called an exosystem and the dynamic compensator $\dot{\eta}_i = \zeta_i(\eta_i, x_i, y_i, z_i)$ an internal model. Such an internal model based design was used in [14] for linear SISO systems (although an observer state $\hat{x}_i$ was implemented in the controller instead of x_i). It is noted that different terminology is used in literature. For example, the real reference $\dot{z}_i = A_o z_i + B_o v_i$ itself is called an internal model in [11]. Moreover, in [11] and some other relevant works [15–17], the authors also investigated the necessity of internal model for synchronization.

It is worth mentioning that recently there was some relevant development of the internal model based design for nonlinear systems. For nonlinear systems, the situation becomes more complicated. In particular, the real reference $\dot{z}_i = A_o z_i + B_o v_i$ should be considered as the exosystem because an extra input has a complicated influence on a nonlinear system. The robust output regulation problem with this exosystem has been studied using two different approaches. In [18], when the network consensus controller $v_i = k_i(z)$ is designed, the network of *all* reference models, represented by an autonomous system $\dot{z} = A_o \otimes Iz + \text{col}(B_o k_1(z), \ldots, B_o k_N(z))$, with $z = \text{col}(z_1, \ldots, z_N)$, is regarded as the exosystem for each agent. Based on this bulky exosystem whose dimension is proportional to network size, a class of internal model is designed which solves the robust output regulation problem. An alternative approach was given in [19,20] where the problem is also formulated as a robust output regulation problem for each agent, but with its own *single* reference model as its exosystem subject to external perturbation v_i (i.e., the network influence from neighboring agents). In particular, the formulation is termed a perturbed output regulation problem. In the formulation of [19,20], for each agent, the network influence is regarded as an external perturbation, so the regulation controller for each individual agent can be designed in a decentralized manner. Moreover, the design does not rely on the network protocols (e.g., a sampled-data network consensus algorithm was adopted in [19]).

We can see from the above that, for either linear systems or nonlinear systems, robust output regulation actually aims at trajectory tracking in the sense of $\lim_{t\to\infty} y_i(t) - \hat{y}_i(t) = 0$ for $\hat{y}_i(t) = C_o z_i(t)$. A dynamic internal mode based controller is unavoidable for the robust output regulation problem when the agent dynamics contain uncertainties. The work in this paper, however, reveals that exact trajectory tracking is not always necessary for output synchronization of multi-agent systems. In fact, we will define a new problem, that is, there exists a trajectory $\sigma_i(t)$ governed by the dynamics of the reference model, i.e., $\dot{\sigma}_i = A_o \sigma_i$, such that $\lim_{t\to\infty} y_i(t) - \hat{y}_i(t) = 0$ for $\hat{y}_i(t) = C_o \sigma_i(t)$. The subtle difference between this new problem and the aforementioned robust output regulation problem is that the