A Targeting Approach To Disturbance Rejection In Multi-Agent Systems

by

Yining Liu

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Appilied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2012

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This thesis focuses on deadbeat disturbance rejection for discrete-time linear multi-agent systems. The multi-agent systems, on which Spieser and Shams' decentralized deadbeat output regulation problem is based, are extended by including disturbance agents. Specifically, we assume that there are one or more disturbance agents interacting with the plant agents in some known manner. The disturbance signals are assumed to be unmeasured and, for simplicity, constant. Control agents are introduced to interact with the plant agents, and each control agent is assigned a target plant agent. The goal is to drive the outputs of all plant agents to zero in finite time, despite the presence of the disturbances. In the decentralized deadbeat output regulation problem, two analysis schemes were introduced: *targeting analysis*, which is used to determine whether or not control laws can be found to regulate, not all the agents, but only the target agents; and *growing analysis*, which is used to determine the behaviour of all the non-target agents when the control laws are applied. In this thesis these two analyses are adopted to the deadbeat disturbance rejection problem. A new necessary condition for successful disturbance rejection is derived, namely that a control agent must be connected to the same plant agent to which a disturbance agent is connected. This result puts a bound on the minimum number of control agents and constraints the locations of control agents. Then, given the premise that both targeting and growing analyses succeed in the special case where the disturbances are all ignored, a new control approach is proposed for the linear case based on the idea of integral control and the regulation methods of Spieser and Shams. Preliminary studies show that this approach is also suitable for some nonlinear systems.

Acknowledgements

I would like to express my utmost gratitude to my supervisor, Professor Daniel E. Davison, for his invaluable guidance, support, and encouragement throughout my whole graduate life at the University of Waterloo.

I would also like to thank my family and friends for their love and support. Their belief in me is the source of my strength.

Table of Contents

List of Figures

Chapter 1

Introduction

Inspired by the psychological problem of controlling a crowd of people, earlier researchers have studied the co-operative regulation of nonlinear discrete-time multi-agent plants, where the goal is to drive the outputs of all plant agents to zero in finite time [20, 19, 16, 17]. To solve this output regulation problem, these researchers have taken the approach of placing *control agents* at strategic locations to interact with the plant agents. Each control agent is responsible for zeroing a single plant agent, called its *target* plant agent. The analysis to determine if computable control laws exist to zero all targets is called *targeting analysis*, which is the first stage of the solution to the regulation problem. The second stage is called *growing analysis*, which is used to determine whether or not the control laws zero the remaining (non-target) plant agents. The idea of targeting was first introduced to stabilize crowds modelled by suggestibility theory according to the work of Spieser and Davison (see [20], [19]). Later, Shams extended their work by allowing for arbitrary propagation time through agents and non-symmetrical influence between the agents. She also proposed several easily-interpreted necessary conditions for targeting and growing analyses to succeed [16][17]. Potential application areas of this theory include the control of unmanned autonomous vehicles, traffic control, and water management. However, one drawback of this earlier work is that disturbances are not modelled. The performance of the control schemes can be greatly degraded if disturbances are introduced.

With the above motivation, this thesis extends the regulation work by focusing on the disturbance rejection problem for discrete-time multi-agent plants. Unknown, unmeasurable, but constant disturbances are introduced to a discrete-time multi-agent plant. The new objective is *deadbeat disturbance rejection*, where we want all the outputs of plant agents to be zeroed in finite time despite the presence of the disturbances. Two problems are tackled. The first one deals with the question of where control agents should be placed, given knowledge of the disturbance agent locations, and the second one deals with the development of control laws (given the location of the control agents) to achieve deadbeat disturbance rejection. For the first problem, a simple but key result is that, for targeting and growing to work, a control agent must be connected to the same plant agent to which a disturbance agent is connected. This result implies that the number of control agents should be no less than the number of disturbance agents in the system. With the help of this result and the previous necessary conditions of Shams, it is possible to determine the minimal number of control agents needed to achieve deadbeat disturbance rejection using the targeting framework. For the second problem, a new double-loop control approach is proposed. An inner loop is based on the earlier regulation results, while an outer loop consisting of integrators and feedback are used to cancel out the disturbance effects. We show that the new disturbance rejection approach works well for linear systems, and likely for a specific class of nonlinear systems.

Although this work is theoretical, we can envision various application areas. Work on social-psychological systems motivated this whole line of research, and provides a good illustration of how it can be used in practice. For example, consider a group of police officers trying to stabilize the psychological state of a crowd of people, while at the same time, some malicious troublemakers keep spreading rumours to influence the crowd in some other manner. To counter the malicious rumours, the police do not need to directly interact with every individual member of the crowd since people within the crowd are interacting with each other. So, instead, the police just need to interact with a few strategic people in the crowd, and focus on controlling the attitudes of a subset of the crowd; if the "right" people are chosen, then the entire crowd could conceivably be controlled. This strategy illustrates the idea of the targeting approach applied to a disturbance rejection problem. This approach may also be useful in more technical fields, such as in distributed energy generation. Consider an energy network that is composed of numerous power generators. Due to weather conditions or other random reasons, some power generators may have some problems that can be treated as disturbances. To control thousands of power generators with disturbances is a very challenging task. The new approach enables us to assign minimal control agents at the right locations, and simplify the problem. Indeed, we can envision that this work may be useful in any multi-agent problem where disturbances arise: the food distribution for chain restaurants, logistics management, mobile sensor networks, etc.

This thesis focuses on the disturbance rejection problem, but it is tightly connected to stabilization/regulation research. We introduce different control agents to interact with separate subsections of the plant network, so the results of this thesis fit into a *decentralized control* framework [1, 4, 18, 23]. Likewise, although not emphasized in this thesis, communication and coordination among agents relates this work to the *co-operative control* community [5, 6, 8, 11, 14, 13]. Other researchers have considered disturbance rejection problems arising in multi-agent systems [22, 24, 25, 9, 7], so this thesis also connects to that research field. A key strength of this work is that it provides a simple way to assign control agents in multi-agent systems with disturbances. This result is scalable to large-scale systems. Another strength, not explored in detail in this thesis, is that each control agent is not required to sense all the states of the plant agents, which helps to reduce the sensing workload per control agent when compared to centralized disturbance rejection schemes.

A brief overview of the chapters of this thesis is given in the following. Chapter 2 first introduces notation and terminology. Then it gives the system model, extending the work of Shams [15] so that disturbance agents are incorporated. Eight assumptions are given, some identical to those in earlier work, some modified to account for disturbance rejection, and one new mild assumption introduced. This chapter provides basic knowledge for the remainder of the thesis.

Chapter 3 formalizes the concepts of *deadbeat output regulation* and *deadbeat disturbance rejection* and states the two problems being considered. As the new algorithm includes targeting and growing analyses from the regulation results, Chapter 4 reviews the analyses for multi-agent systems without disturbances. An example is given to illustrate the application of targeting and growing analyses.

The main technical contribution of this thesis is contained in Chapter 5, where a solution to the problems is provided in the case where all dynamics are linear. The results are explained with a few examples. In Chapter 6, other linear examples are given to show how this algorithm works for various network structures.

Chapter 7 attempts to extend this work to the control of nonlinear multi-agent systems. In this chapter, a class of nonlinear systems is defined by introducing three constraints. Examples are used to illustrate how these constraints were chosen. Then a conjecture is given stating that if a nonlinear multi-agent system meets the three constrains, the algorithm from Chapter 5 achieves deadbeat disturbance rejection.

Chapter 8 summarize the whole thesis with some directions for future research. Some proofs and Matlab code are given in the Appendix.

Chapter 2

System Model

This chapter describes the model of the multi-agent plant and the associated disturbance agents. Notation and terminology are introduced, then eight assumptions are explained. Although the main results of this thesis are developed for the linear case, we provide a more general nonlinear model in this chapter.

2.1 Notation and Terminology

Consider a plant that is composed of n agents, denoted

$$O_1, \ldots, O_n,$$

which influence one another in a known way. In addition, m control agents, denoted

$$X_{n+1}, \ldots, X_{n+m},$$

are introduced at specific locations with the goal of regulating the system, in the sense that the outputs of all n plant agents are driven to zero in finite time. In this thesis, a plant agent whose output is driven to zero in finite time is said to be *zeroed*. As shown in Figure 2.1, which illustrates a three-agent plant with one control agent, directed edges are used to indicate interactions among agents. For the example in Figure 2.1, O_2 and O_3 directly influence each other, while O_1 is influenced by control agent X_4, but X_4 is not influenced by O_1. Central to our work is that each control agent X_i is assigned a *target* plant agent, denoted T_i. The idea is that X_i focuses on zeroing just its target T_i instead of focusing simultaneously on multiple plant agents; in practical applications this targeting idea has an obvious appeal. In Figure 2.1, the target T_4 of X_4 is O_3; notice that a control agent and its target are connected by a red curved arrow.

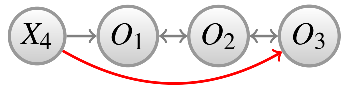

Figure 2.1: A three-agent plant with one control agent.

For the disturbance rejection problem discussed in this thesis, assume there are p (with $1 \leq p \leq n$) disturbance agents

$$D_1, \ldots, D_p$$

at fixed known locations in the system. For simplicity, suppose that no two disturbance agents are connected to the same plant agent, and that a control agent is not allowed to be directly connected to a disturbance agent. Hence, we can number the plant agents such that disturbance agent D_i (for $1 \leq i \leq p$) is connected to plant agent O_i. Define the *neighbours* of O_i (for $1 \leq i \leq n$), denoted $\mathcal{N}(O_i)$, to be the set of (plant, control, and disturbance) agents that directly influence O_i, i.e.

$$\begin{aligned} \mathcal{N}(O_i) &= \{O_j : \text{ there is a directed edge from } O_j \text{ to } O_i\} \\ &\cup \{X_j : \text{ there is a directed edge from } X_j \text{ to } O_i\} \\ &\cup \{D_j : \text{ there is a directed edge from } D_j \text{ to } O_i\}. \end{aligned}$$

Similarly define $\mathcal{N}(X_i)$ (for $n+1 \leq i \leq n+m$) to be the set of neighbours of X_i, i.e.

$$\begin{aligned} \mathcal{N}(X_i) &= \{O_j : \text{ there is a directed edge from } O_j \text{ to } X_i\} \\ &\cup \{X_j : \text{ there is a directed edge from } X_j \text{ to } X_i\}. \end{aligned}$$

In Figure 2.1, the set of neighbours of X_4 and O_1 are $\mathcal{N}(X_4) = \emptyset$ and $\mathcal{N}(O_1) = \{X_4, O_2\}$, respectively.

2.2 The System Model

The dynamics of the system are introduced in this section. Denote the scalar output signal of O_i (for $1 \leq i \leq n$) at time k by $y_i[k]$, and that of D_i (for $1 \leq i \leq p$) by $d_i[k]$. Define $Y_i[k]$ (for $1 \leq i \leq n+m$) to be the *set* of output signals of all neighbours (except for disturbance agents) of O_i or X_i, i.e., for plant agents

$$Y_i[k] = \{y_j[k] : O_j[k] \in \mathcal{N}(O_i) \text{ or } X_j[k] \in \mathcal{N}(O_i)\},$$

and for control agents

$$Y_i[k] = \{y_j[k] : O_j[k] \in \mathcal{N}(X_i) \text{ or } X_j[k] \in \mathcal{N}(X_i)\}.$$

Again taking Figure 2.1 as an example, we have $Y_2[k] = \{y_1[k], y_3[k]\}$. Denote the state of O_i (for $1 \leq i \leq n$) by $x_i[k]$, and the state of D_i (for $1 \leq i \leq p$) by $z_i[k]$. Finally, denote the scalar control signal of X_i by $u_i[k]$ (for $n+1 \leq i \leq n+m$).

The dynamics of O_i (for $1 \leq i \leq p$) are taken to be

$$x_i[k+1] = f_i(x_i[k], Y_i[k], d_i[k]) \tag{2.1}$$

$$y_i[k] = h_i(x_i[k]), \tag{2.2}$$

while the dynamics of O_i (for $p+1 \leq i \leq n$) are

$$x_i[k+1] = f_i(x_i[k], Y_i[k]) \tag{2.3}$$

$$y_i[k] = h_i(x_i[k]). \tag{2.4}$$

Finally, the dynamics of disturbance agent D_i (for $1 \leq i \leq p$) are

$$z_i[k+1] = z_i[k] \tag{2.5}$$

$$d_i[k] = z_i[k], \tag{2.6}$$

that is, the disturbance d_i is constant.

2.3 Assumptions

For the system with dynamics (2.1)–(2.6), eight assumptions are proposed for the disturbance rejection problem. The first four assumptions are taken directly from [17]:

Assumption A_1: There is at least one path from each control agent to its associated target.

Assumption A_2: Control agents can communicate among themselves, with no time delays.

Assumption A_3: Each control agent can sense the state of any plant agent or control agent.

Assumption A_4: Each control agent targets a specific plant agent, but there are no duplicate targets. Hence, in total there are m distinct targets.

These four assumptions are exploited for targeting analysis. (More details about targeting and growing analyses are reviewed in Chapter 4.) Assumption A_1 is an elementary requirement of the targeting approach, i.e., X_i needs to be able to influence T_i if it is to successfully zero T_i. Otherwise, targeting can never succeed, as the control agents have no way to affect the target agents. Assumption A_2 simplifies targeting analysis because an outcome of targeting analysis is a set of communication requirements among the control agents, and if we assume any communication is possible, then it is not necessary to perform additional analysis to determine, for example, if X_3 can communicate with X_8 via some other communication channel (say, through X_5). Assumption A_3 is likewise for simplification purposes. An outcome of targeting analysis is a list of states that each control agents needs to sense, and if we assume that all such combinations are possible, further analysis is not required. Finally, Assumption A_4 makes targeting analysis easier and avoids redundancy in control efforts.

The next assumption is an extension of an assumption in [17], and deals with how long it takes for signals to travel through the plant network. This mild assumption basically requires that the signal propagation times not depend on the values of the particular signals. To be more explicit, define the *propagation time* of d_i through agent O_i (for $1 \leq i \leq p$), denoted $\delta_i^d \geq 1$, to be the time required for a change in d_i to propagate through the dynamics (2.1)–(2.2) to result in a change in y_i. Likewise, for any $O_j \in \mathcal{N}(O_i)$ or $X_j \in \mathcal{N}(O_i)$, define $\delta_{ji} \geq 1$ to be the time required for a change in y_j to propagate through the dynamics (2.1)–(2.2) or (2.3)–(2.4) and to result in a change in y_i.

Assumption A_5: The dynamics (2.1)–(2.2) are such that the propagation time δ_i^d is independent of $d_i[\cdot]$. Also, the dynamics (2.1)–(2.2) and (2.3)–(2.4) are such that the propagation time δ_{ji} is independent of $y_j[\cdot]$.

Assumption A_5 implies that the propagation time along any *path* in the plant is constant. It is equivalent to requiring that the relative degree of the system be constant. If there is a path from X_i to O_j (for $n+1 \leq i \leq n+m$ and $1 \leq j \leq n$), say

$$X_i \to O_\alpha \to O_\beta \to \cdots \to O_\zeta \to O_j,$$

denote by $\Delta(X_i, O_j)$ the propagation time required for a change in u_i to result in a change in y_j. Evidently $\Delta(X_i, O_j)$ is the the sum of the propagation times from one agent to another along the path, i.e.,

$$\Delta(X_i,O_j) = \Delta(X_i,O_\alpha)+\Delta(O_\alpha,O_\beta)+\Delta(O_\beta,O_\gamma)+\cdots+\Delta(O_\upsilon,O_\zeta)+\Delta(O_\zeta,O_j).$$

If there are multiple paths from X_i to O_j, then $\Delta(X_i, O_j)$ is the smallest such sum. If there are no paths from X_i to O_j, we define $\Delta(X_i, O_j) = \infty$. As an example, consider the simple queue system in Figure 2.1 again, where the control agent X_4 targets O_3. Assume the following dynamics:

$$O_1 \quad : \quad x_1[k+1] = x_1^2[k] + y_2[k] + u_4[k],\ y_1[k] = x_1[k] \tag{2.7}$$
$$O_2 \quad : \quad x_2[k+1] = y_3^2[k] + y_1[k],\ y_2[k] = x_2[k] \tag{2.8}$$
$$O_3 \quad : \quad x_3[k+1] = y_2[k],\ y_3[k] = x_3[k]. \tag{2.9}$$

Assumption A_5 is satisfied with, for example, $\Delta(X_4, O_3) = \delta_{41} + \delta_{12} + \delta_{23} = 1+1+1 = 3$.

The next assumption also appears, but in a weaker form, in [17], and is vital to the targeting approach. The assumption is related to a notion of controllability since it requires that each control agent, considered by itself, be able to control the output of its target. The assumption does *not* imply that control agents are able to simultaneously control their targets.

Assumption A_6: For $n+1 \leq i \leq n+m$, consider control agent X_i and its target $T_i = O_j$ (for some $1 \leq j \leq n$). If all control signals other than that of control agent X_i are presumed to be known for all time, then the control signal $u_i[k]$ can be found (possibly dependent on the state of various plant, control, and disturbance agents at time k and on the other presumed-known control signals) to force $y_j[k+\Delta(X_i,O_j)] = v_i[k]$ where $v_i[k] \in R$ is arbitrary but known.

As an example, consider the simple queue system in Figure 2.1 once again, with the dynamics (2.7)–(2.9). Then, Assumption A_6 is satisfied since (through iteration of the above equations)

$$x_3[k+3] = x_2^2[k] + x_1^2[k] + x_2[k] + u_4[k], \tag{2.10}$$

and therefore the control signal required for the assumption is

$$u_4[k] = v_4[k] - x_2^2[k] - x_1^2[k] - x_2[k].$$

Note that Assumption A_6 is more restrictive than a similar assumption in [17]; the difference lies in the role of $v_i[k]$, which is taken to be zero in [17] but assumed to be arbitrary above. This restriction represents a cost of being able to accommodate external disturbances.

The next assumption deals with the behaviour of non-target agents. It is a slight extension of a similar assumption in [17], and it plays a central role in growing analysis. It is essentially equivalent to assuming that there are no zero dynamics in the plant.

Assumption A_7: The dynamics of O_i in (2.1)–(2.2) satisfy the property that, if all of the signals in $Y_i \cup \{d_i\}$ are fixed at zero, except for one (call that one y_j), then, for any $\bar{k} \geq 0$, $y_i[k] = 0$ (for $k \geq \bar{k}$) implies $y_j[k] = 0$ (for $k \geq \bar{k}$). Likewise, the dynamics of O_i in (2.3)–(2.4) satisfy the property that, if all of the signals in Y_i are fixed at zero, except for one (call that one y_j), then, for any $\bar{k} \geq 0$, $y_i[k] = 0$ (for $k \geq \bar{k}$) implies $y_j[k] = 0$ (for $k \geq \bar{k}$).

The system in Figure 2.1 with dynamics (2.7)–(2.9) also satisfies Assumption A_7. To see this, consider, for instance, O_2 with $\mathcal{N}(O_2) = \{O_1, O_3\}$. If $y_2[k] = 0$ for $k \geq \bar{k}$, (2.8) implies $y_3^2[k] + y_1[k] = 0$ (for $k \geq \bar{k}$). In turn, this equation implies that if $y_3[\cdot]$ is zero, then necessarily $y_1[k] = 0$ (for $k \geq \bar{k}$) and, conversely, if $y_1[\cdot]$ is zero, then necessarily $y_3[k] = 0$ (for $k \geq \bar{k}$). If targeting succeeds in this example, we certainly have $y_3[k] = 0$ (for $k \geq \bar{k}$). Similarly, (2.9) shows that $y_2[k] = 0$ (for $k \geq \bar{k}$). So from the previous analysis, we can assure that $y_1[k] = 0$ (for $k \geq \bar{k}$), which means that all the plant agents are zeroed.

Next, Assumption A_8 ensures that disturbance agents always introduce non-trivial disturbance signals. (In terms of jargon, we say that a signal $g[\cdot]$ is *non-zeroed* if for all $\hat{k} \geq 0$, there is a $k \geq \hat{k}$ where $g[k] \neq 0$.)

Assumption A_8: For $1 \leq i \leq p$ and any non-zeroed $d_i[\cdot]$, the solution to $x_i[k+1] = f_i(x_i[k], 0, d_i[k])$, $y_i[k] = h_i(x_i[k])$ is itself non-zeroed.

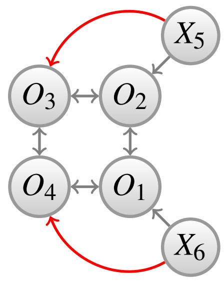

Figure 2.2: A four-agent square plant with two control agents and two disturbance agents.

Consider another example to illustrate the eight assumptions. The dynamics of the square plant in Figure 2.2 are the following:

$$O_1 \;:\; x_1[k+1] = y_2[k] + 2x_1[k] + y_4[k] + u_6[k], y_1[k] = x_1[k] \tag{2.11}$$
$$O_2 \;:\; x_2[k+1] = y_3[k] + 3y_1[k] + u_5[k], y_2[k] = x_2[k] \tag{2.12}$$
$$O_3 \;:\; x_3[k+1] = x_3[k] + 2y_2[k] + 4y_4[k], y_3[k] = x_3[k] \tag{2.13}$$
$$O_4 \;:\; x_4[k+1] = y_1[k] + y_3[k], y_4[k] = x_4[k]. \tag{2.14}$$

First, according to the construction of the system, we notice that there is a path $X_5 \to O_2 \to O_3$ from X_5 to O_3 and another path $X_6 \to O_1 \to O_4$ from X_6 to O_4, which satisfies Assumption A_1. Meanwhile, there are no duplicate targets in this system, as X_5 targets O_3 and X_6 targets O_4. This confirms Assumption A_4. For Assumption A_5, the propagation times are all constants, with

$$\Delta(X_5, O_1) = \delta_{52} + \delta_{21} = 1 + 1 = 2 \tag{2.15}$$
$$\Delta(X_5, O_2) = \delta_{52} = 1 \tag{2.16}$$
$$\Delta(X_5, O_3) = \delta_{52} + \delta_{23} = 1 + 1 = 2 \tag{2.17}$$
$$\Delta(X_5, O_4) = \delta_{52} + \delta_{23} + \delta_{34} = 1 + 1 + 1 = 3 \tag{2.18}$$
$$\Delta(X_6, O_1) = \delta_{61} = 1 \tag{2.19}$$
$$\Delta(X_6, O_2) = \delta_{61} + \delta_{12} = 1 + 1 = 2 \tag{2.20}$$
$$\Delta(X_6, O_3) = \delta_{61} + \delta_{12} + \delta_{23} = 1 + 1 + 1 = 3 \tag{2.21}$$
$$\Delta(X_6, O_4) = \delta_{61} + \delta_{14} = 1 + 1 = 2. \tag{2.22}$$

Hence, Assumption A_5 is satisfied. Then, considering $T_5 = O_3$ and the propagation time $\Delta(X_5, O_3) =$

2, it is easy to determine an expression for $y_5[k+2]$:

$$\begin{aligned} y_3[k+2] &= x_3[k+2] \\ &= x_3[k+1]+2x_2[k+1]+4x_4[k+1] \\ &= x_3[k]+2x_2[k]+4x_4[k]+2x_3[k]+6x_1[k]+2u_5[k] \\ &\quad +4x_3[k]+4x_1[k] \\ &= 10x_1[k]+2x_2[k]+7x_3[k]+4x_4[k]+2u_5[k]. \end{aligned} \tag{2.23}$$

Setting (2.23) to a given but arbitrary $v_5[k]$ yields

$$u_5[k] = (v_5[k]-10x_1[k]-2x_2[k]-7x_3[k]-4x_4[k])/2. \tag{2.24}$$

As Assumption A_6 supposes that all the other states of various plant at time k are presumed to be known for all time, it is possible to compute the control law $u_5[k]$. Similarly, setting the expression of $y_4[k+2]$ to a given but arbitrary $v_6[k]$ yields

$$\begin{aligned} y_4[k+2] &= x_4[k+2] \\ &= x_1[k+1]+x_3[k+1] \\ &= x_3[k]+2x_2[k]+4x_4[k]+x_2[k]+2x_1[k]+x_4[k]+u_6[k] \\ &= 2x_1[k]+3x_2[k]+x_3[k]+5x_4[k]+u_6[k] \\ &= v_6[k], \end{aligned} \tag{2.25}$$

which leads to the control law of $u_6[k]$:

$$u_6[k] = v_6[k]-2x_1[k]-3x_2[k]-x_3[k]-5x_4[k]. \tag{2.26}$$

Thus, Assumption A_6 is confirmed by these expressions of the control laws.

Assumption A_7 also holds for this example. First consider O_1 with $Y_1 = \{O_2, O_4\}$. Suppose we have $y_4[k] = 0$ (for $k \geq \bar{k}$). This means that (2.11) becomes

$$0 = 0+2y_2[k]+0.$$

Therefore, $y_2[k] = 0$ for any $k \geq \bar{k}$. Likewise, if $y_2[k] = 0$ (for $k \geq \bar{k}$), then $y_4[k] = 0$ (for $k \geq \bar{k}$). Similar arguments for O_2, O_3, and O_4 verify that Assumption A_7 holds.

Having stated all the assumptions for the disturbance rejection problem in multi-agent systems, we next formally pose two problems.

Chapter 3

Problem Statement

As presented in [17], the essence of deadbeat output regulation is the following:

Deadbeat Output Regulation: A set of computable control laws $u_{n+1},\ldots,u_{n+m}$ are said to provide *deadbeat output regulation* if there is a $\lambda > 0$ with the property that all initial conditions on $O_1,\ldots,O_n$,

$$y_1[k] = y_2[k] = \cdots = y_n[k] = 0, \text{ for } k \geq \lambda.$$

In this thesis, we extend this concept to *disturbance rejection*:

Deadbeat Disturbance Rejection: A set of computable control laws $u_{n+1},\ldots,u_{n+m}$ are said to provide *deadbeat disturbance rejection* if that there is a $\lambda > 0$ with the property that for all initial conditions on $O_1,\ldots,O_n$ and for all (constant) disturbance signals $d_1,\ldots,d_p$,

$$y_1[k] = y_2[k] = \cdots = y_n[k] = 0, \text{ for } k \geq \lambda.$$

The definition of *computable* control laws is reviewed below (see Definition 1).

The two problems considered in this thesis are as follows.

Problem 1: For a given plant with known disturbance agent locations, determine how many control agents are needed, where their locations should be, and which plant agents should be chosen as their targets, in order to successfully obtain deadbeat disturbance rejection.

Problem 2: For a given plant with known disturbance agent locations and for a given set of control agent locations with known targets, find, if possible, control laws to successfully obtain deadbeat disturbance rejection.

We now review the term *computable control laws*, first introduced in [15] for the deadbeat output regulation problem:

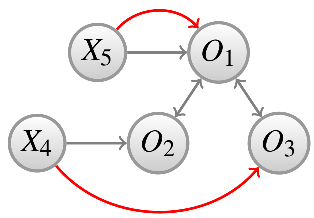

Figure 3.1: A three-agent queue plant with two control agents and one disturbance agent.

Definition 1 [15]: *The control laws $u_{n+1}[\cdot], \ldots, u_{u+m}[\cdot]$ are computable if, for every time k, there exists a permutation of $(u_{n+1}[k], \ldots, u_{u+m}[k])$ such that each entry (say, $u_q[k]$) in the permutation can be determined using only*
(a) the values of the entries to the left of $u_q[k]$,
(b) control signal data at time $k-1$ or earlier, and / or
(c) state data from time k or earlier.

In words, this definition requires that control laws be causal and, in cases where $m > 1$, the control laws can be solved sequentially without the need for solving simultaneous (possibly nonlinear, in general) equations. Consider the example in Figure 3.1 where the dynamics of the three-agent queue system are as follows:

$$x_1[k+1] = x_2[k] + x_3[k] + u_5[k] \tag{3.1}$$
$$y_1[k] = x_1[k] \tag{3.2}$$
$$x_2[k+1] = x_1[k] + u_4[k] \tag{3.3}$$
$$y_2[k] = x_2[k] \tag{3.4}$$
$$x_3[k+1] = -x_3[k] + 2x_1[k] \tag{3.5}$$
$$y_3[k] = x_3[k]. \tag{3.6}$$
$$\tag{3.7}$$

Note that $\Delta(X_5, O_1) = 2$ and $\Delta(X_4, O_3) = 3$. We can determine control laws for X_4 and X_5 by setting $y_3[k+3] = 0$ and $y_1[k+1] = 0$, which results in

$$u_4[k] = x_2[k] - 4x_1[k] + 2.5x_3[k] + u_5[k] - u_5[k+1] \tag{3.8}$$
$$u_5[k] = -3x_1[k-1] + x_3[k-1] - u_4[k-1]. \tag{3.9}$$

From (3.8), we know that the control law of X_4, as written, is not computable, as the expression includes the term $u_5[k+1]$, which is not allowed for a computable control law. (Note that $u_5[k+1]$ can be eliminated in (3.8) by substituting in for $u_5[k+1]$ using (3.9); however, after doing this substitution the new expression for $u_4[k]$ now depends on $u_4[k]$, which again violates the definition of computability.)

In contrast, consider the system in Figure 2.2 with the following dynamics:

$$\begin{aligned}
x_1[k+1] &= x_2[k] + x_4[k] + u_6[k-1] && (3.10)\\
y_1[k] &= x_1[k] && (3.11)\\
x_2[k+1] &= x_1[k] + x_3[k] + u_5[k] && (3.12)\\
y_2[k] &= x_2[k] && (3.13)\\
x_3[k+1] &= x_2[k] + x_3[k] + x_4[k] && (3.14)\\
y_3[k] &= x_3[k] && (3.15)\\
x_4[k+1] &= x_1[k] + x_3[k] + x_4[k] && (3.16)\\
y_4[k] &= x_4[k]. && (3.17)
\end{aligned}$$

The propagation time from each agent to any of its neighbours is one, except for $\Delta(X_6, O_1) = 2$, which implies that $\Delta(X_5, O_3) = 2$ and $\Delta(X_6, X_4) = 3$. Set $x_3[k+2] = 0$ and $x_4[k+3] = 0$ to yield the control laws

$$\begin{aligned}
u_5[k] &= -2x_1[k] - x_2[k] - 3x_3[k] - 2x_4[k] && (3.18)\\
u_6[k] &= -5x_1[k] - 3x_2[k] - 7x_3[k] - 6x_4[k] - 2u_5[k] - u_6[k-1]. && (3.19)
\end{aligned}$$

After substituting (3.18) into (3.19), the control laws are computable since only state information (at time k and $k-1$) and control signal information (at time $k-1$) appear on the right-hand side.

Before dealing with the two problems, in Chapters 5 and 6, we first review targeting and growing analyses in the next chapter since the new algorithm for deadbeat disturbance rejection problem is based on the previous research of targeting and growing in [15].

Chapter 4

Review of Targeting And Growing Analyses

This chapter reviews targeting and growing analyses as presented in [17] in the context of the deadbeat regulation problem. The disturbance rejection problem is not considered here, although we heavily exploit the results of this chapter in Chapter 5. In the context of the regulation problem [17], targeting analysis is used to determine whether or not computable control laws for $X_{n+1},\ldots,X_{n+m}$ can be found to simultaneously zero all m targets,[1] while growing analysis is used to determine the behaviour of all the non-target plant agents when those control laws are applied. Successful deadbeat regulation is realized if (i) targeting analysis reveals that such control laws exist, and (ii) growing analysis reveals that those control laws also zero all non-target agents. We briefly review targeting and growing analyses below.

4.1 Targeting

Targeting analysis proceeds, for each $n+1 \leq i \leq n+m$ and for j defined by $T_i = O_j$, by iterating through the system equations to compute an expression for $y_j[k+\Delta(X_i,O_j)]$. Then, upon forcing $y_j[k+\Delta(X_i,O_j)] = 0$, we can solve (by Assumption A_6) for $u_i[k]$, which will be (in general) dependent on other control signals. The goal of targeting analysis is to solve these m equations for $u_{n+1}[k],\ldots,u_{n+m}[k]$; if the equations can be solved without the need to simultaneously solve (in general, nonlinear) equations, and if the control laws are causal, we say that targeting *succeeds*. The analysis (2.23)–(2.26) of the square system in Figure 2.2 carried out in the previous chapter

[1] Assumptions A_1 and A_6 guarantee that each control agent, considered by itself, is able to zero its one target. It does *not* follow that computable control laws can necessarily be found so that the control agents all simultaneously zero their targets.

demonstrates how targeting analysis works (assuming we set all the disturbance signals to zero so that we specialize to the deadbeat regulation case).

The determination of targeting success depends on two factors: the structure of the underlying graph in the plant, and the propagation times through the plant and control agents. For a given set of control law expressions, we can decide whether the control laws are computable or not, according to Definition 1. For example, consider the three-agent queue system in Figure 3.1 described in the previous chapter; the control laws are not computable in the sense of Definition 1. As an even simpler example, consider again (2.7)–(2.9). In this example, $m = 1$ so there is no need to worry about simultaneous control signals. In fact, Assumptions A_1, A_3 and A_6 directly imply that targeting succeeds; i.e., targeting always succeeds if $m = 1$. Indeed, forcing $y_3[k+3] = 0$ yields, from (2.10), the following control law:

$$u_4[k] = -x_2^2[k] - x_1^2[k] - x_2[k]. \tag{4.1}$$

This control law zeroes the target.

In complicated large-scale systems, a more systematic approach is helpful to determine if the control laws are computable. An algorithmic approach, based on *dependency graphs*, is advocated in [17]. The dependency graph has nodes as the signals $u_{n+1}[\cdot], \ldots, u_{n+m}[\cdot]$ at times $\ldots, k-2, k-1, k, k+1, k+2, \ldots$, with, for each i in the interval $n+1 \leq i \leq n+m$, directed edges from $u_i[l-1]$ to $u_i[l]$ (for $-\infty < l < \infty$) to indicate that $u_i[l-1]$ must be computed before $u_i[l]$ can be computed. In addition, directed edges are drawn, for any $j \neq i$ where $\Delta(X_j, T_i) < \infty$, to $u_i[l]$ from $u_j[l + \Delta(X_i, T_i) - \Delta(X_j, T_i)]$ (for $-\infty < l < \infty$) to capture the dependence of $u_i[\cdot]$ on other control signals [17]. For example, consider again the system in Figure 3.1. Recall the control laws derived earlier:

$$\begin{aligned} u_4[k] &= x_2[k] - 4x_1[k] + 2.5x_3[k] + u_5[k] - u_5[k+1] \\ u_5[k] &= -3x_1[k-1] + x_3[k-1] - u_4[k-1]. \end{aligned}$$

Based on these expressions for the control laws, we can draw the dependency graph, shown in Figure 4.1. Notice that there are loops in the dependency graph, confirming the fact that the control laws in (3.8)and (3.9) are not computable. Hence, targeting does not succeed in the plant of Figure 3.1.

The following theorem, from [15], provides two necessary and sufficient tests for targeting to succeed:

Theorem 1[15]: *For a given plant, given set of $m \geq 2$ control agents, and given targeting assignment, the following three conditions are equivalent:*
(a) Targeting succeeds.
(b) The dependency graph has no loops.
(c) For every p in the interval $2 \leq p \leq m$ and every permutation involving p of the m control

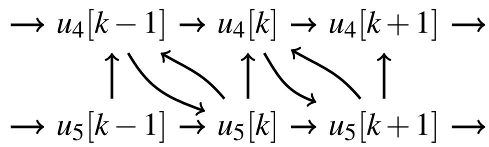

Figure 4.1: The dependency graph of the system in Figure 3.1.

agents (denoted $(\bar{X}_1, \ldots, \bar{X}_p)$ with corresponding targets $(\bar{T}_1, \ldots, \bar{T}_p)$),

$$\sum_{i=1}^{p} \Delta(\bar{X}_i, \bar{T}_i) < \sum_{i=1}^{p-1} \Delta(\bar{X}_i, \bar{T}_{i+1}) + \Delta(\bar{X}_p, \bar{T}_1). \tag{4.2}$$

Condition (b) in Theorem 1 is graphical in nature, as discussed earlier. Condition (c), in contrast, is a set of algebraic constraints. The condition generates a total of $\sum_{p=2}^{m} C_p^m \cdot (p-1)! = \sum_{p=2}^{m} m!/[p(m-p)!]$ distinct inequalities. To help understand this algebraic condition better, consider the case $m = 3$. In this situation, we have five inequalities in all. That is, for $p = 2$, condition (c) generates three distinct inequalities, namely,

$$\begin{aligned}
\Delta(X_1, T_1) + \Delta(X_2, T_2) &< \Delta(X_1, T_2) + \Delta(X_2, T_1) \\
\Delta(X_1, T_1) + \Delta(X_3, T_3) &< \Delta(X_1, T_3) + \Delta(X_3, T_1) \\
\Delta(X_2, T_2) + \Delta(X_3, T_3) &< \Delta(X_2, T_3) + \Delta(X_3, T_2),
\end{aligned}$$

while for $p = 3$, it generates two inequalities:

$$\begin{aligned}
\Delta(X_1, T_1) + \Delta(X_2, T_2) + \Delta(X_3, T_3) &< \Delta(X_1, T_2) + \Delta(X_2, T_3) + \Delta(X_3, T_1) \\
\Delta(X_1, T_1) + \Delta(X_2, T_2) + \Delta(X_3, T_3) &< \Delta(X_1, T_3) + \Delta(X_3, T_2) + \Delta(X_2, T_1).
\end{aligned}$$

These five distinct inequalities are necessary and sufficient, according to Theorem 1, for targeting to succeed when $m = 3$.

4.2 Growing

If targeting succeeds, then we have (by definition) a set of computable control laws that zero all targets. We then turn to growing analysis to determine whether the control laws that resulted from targeting analysis also happen to zero non-target agents. The growing analysis algorithm (GAA) from [17], shown in Figure 4.2, determines a set of plant agents, denoted Ω, that is guaranteed to be zeroed by the control laws. The set Ω "grows" as the algorithm proceeds. The GAA always terminates in a finite number of steps [17]. If Ω contains the entire set of plant agents, then we

Step 1: Initialize $\Omega = \{T_{n+1}, \ldots, T_{n+m}\}$.

Step 2: Determine if there exists a $O_j \in \Omega$ such that all agents in $\mathcal{N}(O_j)$, except for exactly one (call it O_q or X_q, depending on the type of agent), are elements of Ω. Then necessarily O_q (or X_q) is zeroed. Augment Ω with O_q (or X_q).

Step 3: Repeat Step 2 until either:

– all of $O_1, \ldots, O_n$ are in Ω, in which case growing succeeds, or
– no O_j can be found satisfying the condition of Step 2, and at least one plant agent does not appear in Ω, in which case growing fails.

Figure 4.2: The Growing Analysis Algorithm (GAA) from [17].

say growing *succeeds*. If both targeting and growing work, then the control laws that resulted from the targeting analysis achieve plant regulation, and the *settling time* (i.e., the number of samples required until all plant outputs are zero) is exactly [17]

$$\lambda = \max\{\Delta(X_i, T_i) : n+1 \leq i \leq n+m\}.$$

The following theorem summarizes these two facts about growing:

Theorem 2: [15] *For a given plant, given set of $m \geq 1$ control agents, and given targeting assignment, assume targeting succeeds. Then the following hold:*
(a) The GAA terminates after a finite number of iterations.
(b) If growing succeeds, then regulation of the plant is achieved with settling time $\lambda = \max\{\Delta(X_i, T_i) : n+1 \leq i \leq n+m\}$.

For the example in Figure 2.1 with dynamics (2.7)–(2.9) and control law (4.1), the GAA proceeds with $\Omega = \{O_3\}$, $\Omega = \{O_3, O_2\}$, and $\Omega = \{O_3, O_2, O_1\}$, and therefore growing succeeds. Thus, we conclude that the control law (4.1) achieves regulation with settling time $\lambda = 3$. For the example of the square system in Figure 2.2, the GAA proceeds with $\Omega = \{O_3, O_4\}$, $\Omega = \{O_3, O_4, O_2\}$, and $\Omega = \{O_3, O_4, O_2, O_1\}$, which implies that growing succeeds with settling time $\lambda = 2$. This matches the analysis of the square system in Figure 2.2 in the previous chapter. Notice that, unlike targeting analysis, growing analysis depends only on the structure of the underlying plant graph [17].

4.3 Example of Targeting and Growing Analyses

To help further understand targeting and growing analyses, a new example in Figure 4.3 with dynamics (4.3)–(4.8) is given in this section. We go through all the steps and check out the conditions in Theorem 1 and Theorem 2.

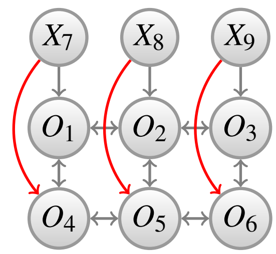

Figure 4.3: A six-agent grid plant with three control agents.

The dynamics of this example are as the following:

$$x_1[k+1] = x_2[k]+x_4[k]+u_7[k-1],\ y_1[k]=x_1[k] \quad (4.3)$$
$$x_2[k+1] = x_1[k]+x_3[k]+x_5[k]+u_8[k],\ y_2[k]=x_2[k] \quad (4.4)$$
$$x_3[k+1] = x_2[k]+x_6[k]+u_9[k-1],\ y_3[k]=x_3[k] \quad (4.5)$$
$$x_4[k+1] = x_1[k]+x_4[k]+x_5[k],\ y_4[k]=x_4[k] \quad (4.6)$$
$$x_5[k+1] = x_2[k]+x_4]k]+x_6[k],\ y_5[k]=x_5[k] \quad (4.7)$$
$$x_6[k+1] = x_3[k]+x_5[k],\ y_6[k]=x_6[k]. \quad (4.8)$$

From these dynamics, we determine that the propagation time from any agent to any of its neighbours is one, except for the following cases:

$$\Delta(X_7,O_1) = 2$$
$$\Delta(X_9,O_3) = 2.$$

Hence,

$$\Delta(X_7,O_4) = 3$$
$$\Delta(X_7,O_5) = 4$$
$$\Delta(X_7,O_6) = 5$$
$$\Delta(X_8,O_4) = 3$$
$$\Delta(X_8,O_5) = 2$$
$$\Delta(X_8,O_6) = 3$$
$$\Delta(X_9,O_4) = 5$$
$$\Delta(X_9,O_5) = 4$$
$$\Delta(X_9,O_6) = 3.$$

First, for targeting analysis, set the target outputs $y_4[x+\Delta(X_7,O_4)]$, $y_5[x+\Delta(X_8,O_5)]$ and

$y_6[x+\Delta(X_9,O_6)]$ to zero as follows:

$$\begin{aligned}
y_4[x+3] &= x_4[k+3] \\
&= x_1[k+2]+x_4[k+2]+x_5[k+2] \\
&= x_1[k+1]+2x_2[k+1]+3x_4[k+1]+x_5[k+1]+x_6[k+1]+u_7[k] \\
&= 5x_1[k]+2x_2[k]+3x_3[k]+5x_4[k]+6x_5[k]+x_6[k]+u_7[k-1]+2u_8[k]+u_7[k] \\
&= 0 \qquad (4.9)\\
y_5[k+2] &= x_5[k+2] \\
&= x_2[k+1]+x_4[k+1]+x_6[k+1] \\
&= 2x_1[k]+2x_3[k]+3x_5[k]+x_4[k]+u_8[k] \\
&= 0 \qquad (4.10)\\
y_6[k+3] &= x_6[k+3] \\
&= x_3[k+2]+x_5[k+2] \\
&= 2x_2[k+1]+x_4[k+1]+2x_6[k+1]+u_9[k] \\
&= 3x_1[k]+4x_3[k]+x_4[k]+5x_5[k]+2u_8[k]+u_9[k] \\
&= 0. \qquad (4.11)
\end{aligned}$$

From (4.9)–(4.11), we obtain the following three control laws:

$$\begin{aligned}
u_7[k] &= -5x_1[k]-2x_2[k]-3x_3[k]-5x_4[k]-6x_5[k]-x_6[k]-u_7[k-1]-2u_8[k] \qquad (4.12)\\
u_8[k] &= -2x_1[k]-2x_3[k]-x_4[k]-3x_5[k] \qquad (4.13)\\
u_9[k] &= -3x_1[k]-4x_3[k]-x_4[k]-5x_5[k]-2u_8[k]. \qquad (4.14)
\end{aligned}$$

Looking at these control laws, we see that both $u_7[k]$ and $u_9[k]$ depend on $u_8[k]$. The associated dependency graph, shown in Figure 4.5, has no loops. Hence, by Theorem 1, the control laws (4.12)–(4.14) are computable. Alternatively, we could have used condition (c) of Theorem 1 to show that targeting succeeds. The following inequalities arise from condition (c):

$$\begin{aligned}
\Delta(X_7,O_4)+\Delta(X_8,O_5) &< \Delta(X_7,O_5)+\Delta(X_8,O_4) \qquad (4.15)\\
\Delta(X_7,O_4)+\Delta(X_9,O_6) &< \Delta(X_7,O_6)+\Delta(X_9,O_4) \qquad (4.16)\\
\Delta(X_8,O_5)+\Delta(X_9,O_6) &< \Delta(X_8,O_6)+\Delta(X_9,O_5) \qquad (4.17)\\
\Delta(X_7,O_4)+\Delta(X_8,O_5)+\Delta(X_9,O_6) &< \Delta(X_7,O_5)+\Delta(X_8,O_6)+\Delta(X_9,O_4) \qquad (4.18)\\
\Delta(X_7,O_4)+\Delta(X_8,O_5)+\Delta(X_9,O_6) &< \Delta(X_7,O_6)+\Delta(X_9,O_5)+\Delta(X_8,O_4). \qquad (4.19)
\end{aligned}$$

It is easy to verify that all five of these inequalities hold for this example, so we again conclude that targeting succeeds.

Next, we turn to growing analysis. It is routine to verify that Assumption A_7 holds. The GAA starts with $\Omega=\{O_4,O_5,O_6\}$ and Ω grows as follows:

$$\Omega=\{O_4,O_5,O_6,O_1\},\Omega=\{O_4,O_5,O_6,O_1,O_2\},\Omega=\{O_4,O_5,O_6,O_1,O_2,O_3\}.$$

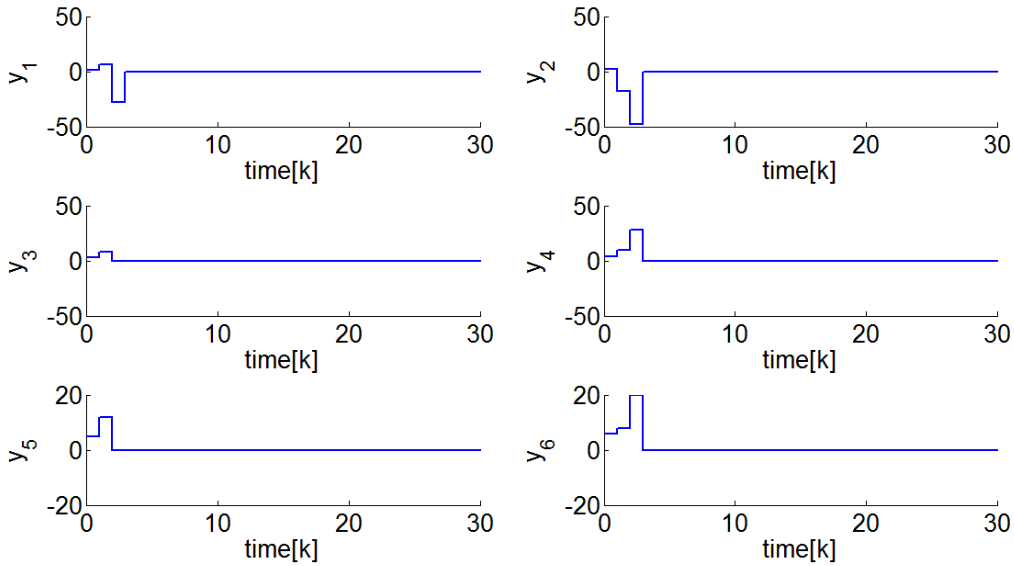

Figure 4.4: Simulation result for the example in Figure 4.3.

Note that the growing process is not unique, although the final answer is unique. For this example, Ω includes all plant agents, so growing succeeds. Hence, the computable control laws (4.12)–(4.14) achieve deadbeat output regulation, as demonstrated in Figure 4.4. The settling time is $\lambda = 3$.

In the next chapter, we discuss disturbance rejection for linear systems. Our approach is based on the targeting and growing analyses reviewed in this chapter, although extra complexities arise because of the presence of the disturbances.

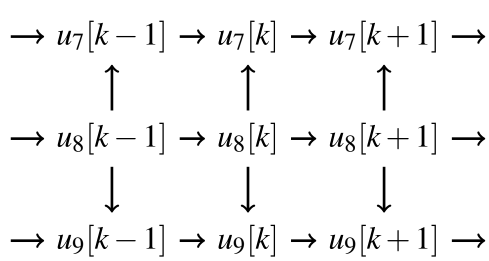

Figure 4.5: The dependency graph of the system in Figure 4.3.

Chapter 5

Designing for Deadbeat Disturbance Rejection in the Linear Case

In this chapter, the disturbance rejection problem in the linear case is discussed. We still use the ideas of targeting and growing, but to deal with the presence of disturbances, an additional feedback loop with integral action is introduced. Figure 5.5 provides an overview of the proposed scheme, and Figure 5.8 summarizes the final design algorithm.

5.1 Necessary Conditions for Control Agent Placement

This section makes a contribution towards solving Problem 1, i.e., determining the number and locations of the control agents necessary to achieve deadbeat disturbance rejection. Since our approach (described below; see Figure 5.8) ultimately involves solving the regulation problem, the four necessary conditions from [17], summarized in Theorems 3 and 4 below, still apply:

Theorem 3 [17]: *For a given plant, given set of $m \geq 2$ control agents, and given targeting assignment, assume that targeting succeeds. Then both the following hold:*
(a) Propagation times along the paths from control agents to respective targets must be, on average, less than propagation times along the paths from control agents to all other targets.

$$\frac{1}{m}\sum_{i=n+1}^{n+m}\Delta(X_i,T_i) \; < \; \frac{1}{m(m-1)}\sum_{i=n+1}^{n+m}\sum_{\substack{j=n+1\\ j\neq i}}^{n+m}\Delta(X_i,T_j). \tag{5.1}$$

(b) There are no nodes in common between a fastest path connecting X_i to T_i (for $n+1 \leq i \leq$

$n+m$) and a fastest path connecting X_j to T_j (for $n+1 \leq j \leq n+m, j \neq i$).

In practice, it is probably desirable for control agents to be close to their own target agents; this condition is also desirable because it results in a small settling time. Theorem 3(a) shows that, fortuitously, this is in fact a necessary condition for targeting to succeed. We can verify that (5.1) holds for the examples given in the previous chapters. For the system in Figure 2.2, we have

$$\begin{aligned}
\Delta(X_5, O_3) &= \delta_{52} + \delta_{23} = 1 + 1 = 2 \\
\Delta(X_5, O_4) &= \delta_{52} + \delta_{23} + +\delta_{34} = 1 + 1 + 1 = 3 \\
\Delta(X_6, O_3) &= \delta_{61} + \delta_{12} + \delta_{23} = 1 + 1 + 1 = 3 \\
\Delta(X_6, O_4) &= \delta_{61} + \delta_{14} = 1 + 1 + 2,
\end{aligned}$$

and therefore

$$\frac{1}{2}(\Delta(X_5, O_3) + \Delta(X_6, O_4)) < \frac{1}{2(2-1)}(\Delta(X_5, O_4) + \Delta(X_6, O_3)).$$

Thus, condition (a) of Theorem 3 is satisfied in this example. Similarly, condition (a) holds for the system in Figure 4.3

$$\begin{aligned}
\frac{1}{3}(\Delta(X_7, O_4) + \Delta(X_8, O_5) + \Delta(X_9, O_6)) < \frac{1}{3(3-1)}(&\Delta(X_7, O_5) + \Delta(X_7, O_6) \\
&+ \Delta(X_8, O_4) + \Delta(X_8, O_6) + \Delta(X_9, O_4) + \Delta(X_9, O_5)).
\end{aligned}$$

However, condition (a) does not hold for the system in Figure 3.1 since

$$\frac{1}{2}(\Delta(X_5, O_1) + \Delta(X_4, O_3)) = \frac{1}{2(2-1)}(\Delta(X_5, O_3) + \Delta(X_4, O_1)),$$

which implies that targeting analysis fails. This conclusion is consistent with our earlier analysis.

Condition (b) of Theorem 3 helps to effectively reduce the number of combinations of the locations of control agents and their targets that need to be considered when solving Problem 1. It is an encouraging result because it is easy to verify from the system structure. For example, we can verify that there are no intersections among the fastest paths in both systems of Figure 2.2 and Figure 4.3. On the other hand, there is a common node O_1 between the fastest paths connecting X_5 to O_1 and X_4 to O_3 in Figure 3.1. Therefore, targeting analysis fails in this case according to condition (b). Like Theorem 3(a), this condition is also a necessary, not sufficient, condition for targeting to succeed.

Theorem 4 [17]: *For a given plant, given set of $m \geq 1$ control agents, and given targeting assignment, assume that targeting succeeds. Then growing succeeds only if both the following*

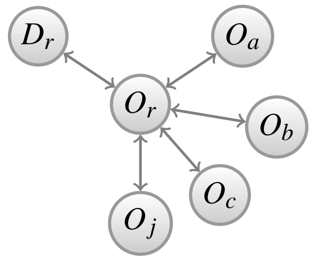

Figure 5.1: Figure for the proof of Lemma 1.

hold:
(a) Each plant agent lies on the fastest path from some X_i to its associated T_i.
(b) For each X_i, the fastest path from X_i to T_i is unique.

Theorem 4 provides two necessary conditions for growing to succeed. Both of the conditions are easy to verify from the system structure, which is especially appealing when dealing with systems with large number of agents. It is easy to show that these two conditions hold in the examples in Figure 2.2, Figure 4.3, and Figure 3.1.

In addition to these four necessary conditions, we include a new necessary condition for the disturbance case:

Lemma 1: *For targeting and growing to succeed, a control agent must be connected to each plant agent to which a disturbance agent is connected.*

Proof: Consider a disturbance agent D_r (for $1 \leq r \leq p$) connected to plant agent O_r. Suppose $O_a, O_b, O_c, \ldots, O_j$ are the plant agent neighbours of O_r, as indicated in Figure 5.1.

We use a contradiction argument. To this end, suppose that targeting and growing succeed and that there is *not* a control agent connected to O_r. The dynamics of O_r have the form (2.1)–(2.2), that is,

$$x_r[k+1] = f_r(x_r[k], Y_r[k], d_r[k]), \text{for } k \geq 0 \quad (5.2)$$
$$y_r[k] = h_r(x_r[k]). \quad (5.3)$$

Since growing succeeds, all the plant agent neighbours are zeroed, i.e., $Y_r[k] = 0$ for all $k > \bar{k}$. Consequently,

$$x_r[k+1] = f_r(x_r[k], 0, d_r[k]), \text{for } k > \bar{k} \quad (5.4)$$
$$y_r[k] = h_r(x_r[k]). \quad (5.5)$$

If $d_r[\cdot]$ is non-zeroed, then (5.4)-(5.5) imply, via Assumption A_7, that $y_r[\cdot]$ is non-zeroed, and therefore O_r is not zeroed, contradicting our supposition that growing succeeds. □

The necessary condition of Lemma 1, like those of Theorems 3 and 4, puts a bound on the

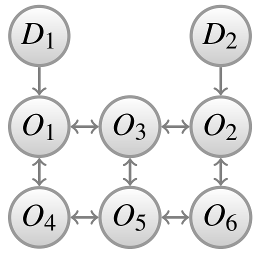

Figure 5.2: A six-agent grid plant with two disturbance agents.

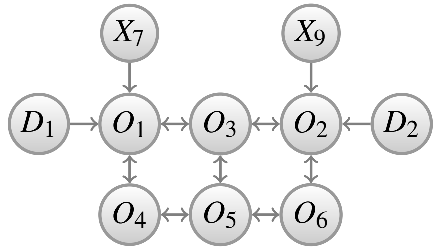

Figure 5.3: Control agents for a six-agent grid plant with two disturbance agents.

minimum number of control agents, and constrains the locations of control agents, needed to achieve deadbeat disturbance rejection. For example, Lemma 1 tells us immediately that at least as many control agents are needed as there are disturbance agents (i.e., $m \geq p$ must hold).

To illustrate the necessary conditions for Problem 1, we give an example of how to locate the control agents for the system shown in Figure 5.2. From Lemma 1, we know that control agents must be connected to plant agents to which disturbance agents are connected. This implies that at least two control agents are needed, as illustrated in Figure 5.3. We now have to choose targets for these control agents, and determine if any additional control agents are needed. For notational simplicity, assume for this example that all propagation times through individual agents are one. We quickly deduce that it is impossible to assign targets to X_7 and X_9 while satisfying condition (a) of Theorem 4 and condition (b) of Theorem 3, unless a third control agent is included. Including a control agent on O_3 gives the extra flexibility to satisfy these two conditions. In fact, the conditions can be satisfied using the targeting assignment shown in Figure 5.4. Moreover, targeting and growing analyses succeed for this arrangement, so we conclude that at least three control agents are needed for this example. We will revisit this example later.

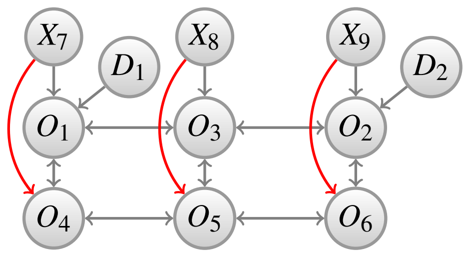

Figure 5.4: A six-agent grid plant with two disturbance agents and three control agents.

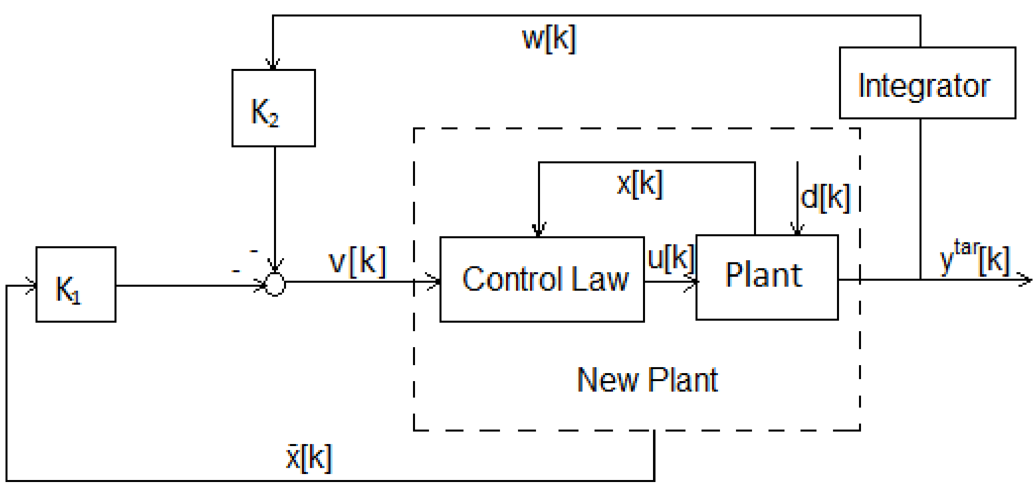

Figure 5.5: Block diagram of the proposed disturbance rejection scheme.

5.2 Proposed Control Approach

Recall Problem 2 proposed in Chapter 3: For a given plant with known disturbance agent locations and for a given set of control agent locations with known targets, find, if possible, control laws to successfully obtain deadbeat disturbance rejection. This section explains a control approach to solve this problem.

A high-level block diagram of the proposed approach is presented in Figure 5.5. Focus on the dashed box. The “Plant” refers to the collection of networked plant agents, with control input

$$u[k] = [u_{n+1}[k], \ldots, u_{n+m}[k]]^T,$$

disturbance input

$$d[k] = [d_1[k], \ldots, d_p[k]]^T,$$

and plant state

$$x[k] = [x_1^T[k], \ldots, x_n^T[k]]^T.$$

The plant output is the set of output signals from just the target agents, namely

$$y^{tar}[k] = [y_{n+1}^{tar}[k], \ldots, y_{n+m}^{tar}[k]]^T,$$

where $y_i^{tar}[k]$ is the output of agent T_i (for $n+1 \leq i \leq n+m$). In Figure 5.5, “Control Law” generates the control input signal $u[k]$, which is derived through targeting analysis, very much like in the regulation case, except with the introduction of signals denoted $v_i[k]$ to accommodate the disturbances. The dashed box can be considered as a “New Plant” which has the input

$$v[k] = [v_{n+1}[k], \ldots, v_{n+m}[k]]^T,$$

the new plant states

$$\bar{x}[k] = [\bar{x}_1^T[k], \ldots, \bar{x}_n^T[k]]^T,$$

and the output $y^{tar}[k]$. For the integral loop in Figure 5.5, signal

$$w[k] = [w_{n+1}[k], \ldots, w_{n+m}[k]]^T$$

is the output of the “Integrator” block:

$$w[k+1] = w[k] + y^{tar}[k]. \tag{5.6}$$

(Given the disturbance signals are constants, we use integral control to reject all disturbances in steady-state. By taking the outputs of the target agents to be the inputs of the integrators, we can expect to force the target outputs to be zero. Notice that the discrete-time integrators are different from the ones in continuous time, as shown in Figure 5.6.)

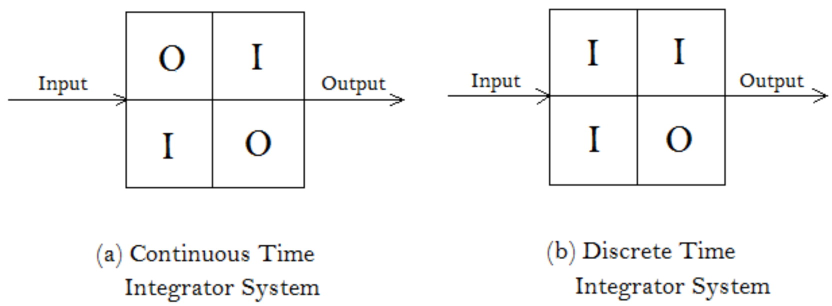

Figure 5.6: State-space realizations for integrators in both continuous time and discrete time.

For simplicity, we now invoke the restriction that all the dynamics are linear. Hence, the dynamics of plant agent O_i (for $1 \leq i \leq n$) can be written as

$$x_i[k+1] = A_i x_i[k] + B_i y[k] + B_i^u u[k] + B_i^d d_i[k] \tag{5.7}$$
$$y_i[k] = C_i x_i[k], \tag{5.8}$$

where

$$y[k] \triangleq [y_1[k], \ldots, y_n[k]]^T. \tag{5.9}$$

If some agent does not directly influence plant agent O_i, then the associated elements of B_i, B_i^u, or B_i^d are zero. For example, if there is no disturbance attached to O_i, for $p+1 \leq i \leq n$, necessarily $B_i^d = 0$.

To solve Problem 2, we assume that the number and location of control agents, as well as targeting assignment, have been chosen so that

- both targeting and growing succeed in the special case where the disturbances are all zero, and
- a control agents is connected to each plant agent to which a disturbance agent is connected (as per Lemma 1).

We now use targeting analysis methods to find the control laws $u[k]$, with appropriate introduction of the signals $v[k]$ for the new plant. To this end, we note that in the special case where everything is linear, the system output $y^{tar}[k]$ can always be expressed as (for $n+1 \leq i \leq n+m$ and where the various α coefficients are known scalars)

$$\begin{aligned} y_i^{tar}[k+\Delta_i] &= \sum_{n+1\le q\le n+m, q\ne i} (\alpha_u^{qi} u_q[k] + \alpha_d^{qi} d_q[k]) + \alpha_1^i x_1[k] + \alpha_2^i x_2[k] + \\ &\qquad \cdots + \alpha_n^i x_n[k] + \alpha_u^i u_i[k] + \alpha_d^i d_i[k] \qquad (5.10) \\ &\triangleq v_i[k] + \tilde{d}_i[k] \qquad (5.11) \\ &\triangleq \tilde{v}_i[k], \qquad (5.12) \end{aligned}$$

where, for notational convenience,

$$\Delta_i \triangleq \Delta(X_i, O_i),\ i = n+1, \ldots, n+m.$$

The sum term in (5.10) includes all the control signals $u_q[k]$ and the disturbance signals $d_q[k]$, other than $u_i[k]$ and $d_i[k]$, that affect $y_i^{tar}[k+\Delta_i]$. As indicated in (5.11), we define

$$\tilde{d}_i[k] \triangleq \sum_{q\ne i} \alpha_d^{qi} d_q[k] + \alpha_d^i d_i[k] \qquad (5.13)$$

as the net effect of disturbances on $y_i^{tar}[k+\Delta_i]$, while $v_i[k]$ includes the effect of all the other control agents and plant agents terms. Finally, we define

$$\tilde{v}_i[k] \triangleq v_i[k] + \tilde{d}_i[k].$$

From (5.10) and (5.12), we know that

$$\sum_{q\ne i} (\alpha_u^{qi} u_q[k] + \alpha_d^{qi} d_q[k]) + \alpha_1^i x_1[k] + \alpha_2^i x_2[k] + \cdots + \alpha_n^i x_n[k] + \alpha_u^i u_i[k] + \alpha_d^i d_i[k] = v_i[k] + \tilde{d}_i[k],$$

where

$$\tilde{d}_i[k] \triangleq \sum_{q\ne i} \alpha_d^{qi} d_q[k] + \alpha_d^i d_i[k].$$

This implies that

$$\sum_{q\ne i} \alpha_u^{qi} u_q[k] + \alpha_1^i x_1[k] + \alpha_2^i x_2[k] + \cdots + \alpha_n^i x_n[k] + \alpha_u^i u_i[k] = v_i[k],$$

which results in the following control laws $u_i[k]$ (for $n+1 \le i \le n+m$, and assuming disturbances are zero):

$$u_i[k] = (v_i[k] - \sum_{q\ne i} \alpha_n^{qi} u_q[k] - \alpha_1^i x_1[k] - \ldots - \alpha_n^i x_n[k])/\alpha_u^i. \qquad (5.14)$$

(The coefficient α_u^i is necessarily non-zero by Assumption A_6.)

Control laws (5.14) necessarily exist because (i) we are presuming that the control agent locations and targets have been chosen to ensure that targeting analysis succeeds, and (ii) Assumption A_6 holds. Note that, if we set

$$v_i[k] = 0$$

in (5.14), then the resulting control laws are exactly the same as those that would be obtained in the linear case without any disturbances.

Transforming

$$y_i^{tar}[k+\Delta_i] = \tilde{v}_i[k],$$

into the z-domain results in

$$y_i^{tar}(z) \cdot z^{\Delta_i} = \tilde{v}_i(z). \tag{5.15}$$

From (5.15), we have the transfer function

$$y_i^{tar}(z)/\tilde{v}_i(z) = 1/z^{\Delta_i}, \tag{5.16}$$

so it follows that

$$\begin{bmatrix} y_{n+1}^{tar}(z) \\ y_{n+2}^{tar}(z) \\ \vdots \\ y_{n+m}^{tar}(z) \end{bmatrix} = \begin{bmatrix} 1/z^{\Delta_1} & 0 & \cdots & 0 \\ 0 & 1/z^{\Delta_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/z^{\Delta_m} \end{bmatrix} \begin{bmatrix} \tilde{v}_{n+1}(z) \\ \tilde{v}_{n+2}(z) \\ \vdots \\ \tilde{v}_{n+m}(z) \end{bmatrix}. \tag{5.17}$$

This system captures the dynamics of the "New Plant" in Figure 5.5. To obtain a state-space realization of this system, first recognize that, for any $r \geq 1$, a state-space realization of $1/z^r$ is

$$\begin{aligned} A'_r &= \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \in R^{r \times r}, \\ B'_r &= \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \in R^{r \times 1}, \\ C'_r &= \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \in R^{1 \times r}, \\ D'_r &= 0. \end{aligned}$$

Thus, a state-space realization of the system in (5.17) is

$$\begin{aligned}\bar{x}[k+1] &= \bar{A}\bar{x}[k] + \bar{B}\tilde{v}[k] \\ &= \bar{A}\bar{x}[k] + \bar{B}v[k] + \bar{B}\tilde{d}[k] \qquad (5.18)\\ y^{tar}[k] &= \bar{C}\bar{x}[k], \qquad (5.19)\end{aligned}$$

with

$$\bar{A} = \text{block diag}\{A'_{\Delta(X_{n+1},T_{n+1})}, \ldots, A'_{\Delta(X_{n+m},T_{n+m})}\} \qquad (5.20)$$

$$\bar{B} = \text{block diag}\{B'_{\Delta(X_{n+1},T_{n+1})}, \ldots, B'_{\Delta(X_{n+m},T_{n+m})}\} \qquad (5.21)$$

$$\bar{C} = \text{block diag}\{C'_{\Delta(X_{n+1},T_{n+1})}, \ldots, C'_{\Delta(X_{n+m},T_{n+m})}\}. \qquad (5.22)$$

and

$$\begin{aligned}\bar{x}[k] = [&\tilde{v}_{n+1}[k-\Delta_1], \tilde{v}_{n+1}[k-\Delta_1-1], \ldots, \tilde{v}_{n+1}[k-1], \ldots, \\ &\tilde{v}_{n+m}[k-\Delta_m], \tilde{v}_{n+m}[k-\Delta_m-1], \ldots, \tilde{v}_{n+m}[k-1]]^T.\end{aligned}$$

A state-space realization for the augmented plant, composed of the "New Plant" (5.18)–(5.19) and the "Integrator" (5.6), is

$$x^*[k+1] = A_1^* x^*[k] + B^*(v[k] + \tilde{d}[k]) \qquad (5.23)$$

$$y^{tar}[k] = C^* x^*[k], \qquad (5.24)$$

where

$$x^* = \begin{bmatrix} \bar{x}[k] \\ w[k] \end{bmatrix}, \quad A_1^* = \begin{bmatrix} \bar{A} & O \\ \bar{C} & I \end{bmatrix}, \quad B^* = \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix}, \quad C^* = \begin{bmatrix} \bar{C} & O \end{bmatrix}.$$

We can prove that the augmented system (5.23)–(5.24) is controllable:

Lemma 2: *The pair* (A_1^*, B^*) *is controllable.*

Proof: See Appendix A.

Lemma 2 implies that

$$K = [K_1 \quad K_2]$$

can be found so that the control law

$$v[k] = -K_1\bar{x}[k] - K_2 w[k] \tag{5.25}$$

results in a stable deadbeat system, i.e., all closed-loop eigenvalues are at the origin. Indeed, there is a particularly simple form for the matrix K, dependent only on the propagation times in the plant network, that places all the eigenvalues of $A_1^* - B^*K$ at the origin:

Lemma 3: *The following forms of matrices K_1 and K_2 place all the eigenvalues of $A_1^* - B^*K$ at the origin:*

$$K_1 = \left[\begin{array}{c:c:c:c} \underbrace{1\ 1\ \cdots\cdots\ 1}_{\Delta_1} & & & \\ \hdashline & \underbrace{1\ 1\ \cdots\cdots\ 1}_{\Delta_2} & & \\ \hdashline & & \cdots\cdots & \\ \hdashline & & & \underbrace{1\ 1\ \cdots\cdots\ 1}_{\Delta_m} \end{array}\right], \tag{5.26}$$

$$K_2 = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \undersetbrace{}{0 & \cdots & 0 & 1} \end{bmatrix}. \tag{5.27}$$

Proof: See Appendix B.

The resulting closed-loop system equations are

$$\begin{aligned} x^*[k+1] &= A^*x^*[k] + B^*\tilde{d}[k]; && (5.28) \\ y^{tar}[k] &= C^*x^*[k], && (5.29) \end{aligned}$$

where

$$A^* = \begin{bmatrix} \bar{A} - \bar{B}K_1 & -\bar{B}K_2 \\ \bar{C} & I \end{bmatrix}.$$

We now obtain the main result of this thesis:

Theorem 5: *The control law obtained by substituting (5.25) into (5.14) successfully zeros, for any unknown constant disturbances, all the target agents, with the maximum settling time* $\lambda = m+n+\max\{\Delta(X_i,T_i) : n+1 \leq i \leq n+m\}$.

Proof: By (5.28)–(5.29), we have

$$\begin{aligned} x^*(z)\cdot z &= A^*x^*(z)+B^*\tilde{d}(z) \\ y^{tar}(z) &= C^*x^*(z), \end{aligned}$$

and

$$y^{tar}(z) = C^*(zI - A^* + B^*K)^{-1}B^*\tilde{d}(z). \tag{5.30}$$

As the matrix K guarantees the stability of (5.30), there in fact exists a steady-state value of $y^{tar}[\cdot]$, namely

$$\begin{aligned} & \lim_{k\to\infty} y^{tar}[k] \\ = \;& \lim_{z\to 1}(z-1)y^{tar}(z) \\ = \;& C^*(I - A^* + B^*K)^{-1}B^*\tilde{d}. \end{aligned} \tag{5.31}$$

Introduce matrices S, P, T, and Q so that

$$\begin{aligned} & \begin{bmatrix} S & T \\ P & Q \end{bmatrix} \\ = \;& (I - A^* + B^*K)^{-1} \\ = \;& \begin{bmatrix} I - \bar{A} + \bar{B}K_1 & \bar{B}K_2 \\ \bar{C} & 0 \end{bmatrix}^{-1}. \end{aligned}$$

Then,

$$\begin{aligned} & \begin{bmatrix} I - \bar{A} + \bar{B}K_1 & \bar{B}K_2 \\ \bar{C} & 0 \end{bmatrix} \begin{bmatrix} S & T \\ P & Q \end{bmatrix} \\ = \;& \begin{bmatrix} (I - \bar{A} + \bar{B}K_1)S + \bar{B}K_2P & (I - \bar{A} + \bar{B}K_1)T + \bar{B}K_2Q \\ -\bar{C}S & -\bar{C}T \end{bmatrix} \\ = \;& \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}, \end{aligned} \tag{5.32}$$

The (2,1) entry of (5.32) implies

$$\bar{C}S = 0.$$

Continuing from (5.31),

$$\begin{aligned}
\lim_{k\to\infty} y^{tar}[k] &= \begin{bmatrix} \bar{C} & 0 \end{bmatrix} \begin{bmatrix} S & T \\ P & Q \end{bmatrix} \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix} \tilde{d} \\
&= \begin{bmatrix} \bar{C}S & \bar{C}T \end{bmatrix} \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix} \tilde{d} \\
&= \begin{bmatrix} 0 & \bar{C}T \end{bmatrix} \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix} \tilde{d} \\
&= 0.
\end{aligned}$$

Finally, the maximum settling time formula includes time steps needed to form control laws $v[k]$ in addition to the time steps for control laws $u[k]$ to zero all the targets. According to

$$v[k] = -K_1\bar{x}[k] - K_2w[k],$$

control laws $v[k]$ can be formed only after the n entries of $\bar{x}[k]$ and the m entries of $w[k]$ are known, which will take at most $n+m$ time steps. Thus considering the time which control laws $u[k]$ take to zero all the target agents, the maximum settling time is

$$\lambda = m + n + \max\{\Delta(X_i, T_i) : n+1 \leq i \leq n+m\}.$$

□

In summary, the control laws obtained by substituting (5.25) into (5.14) zero all targets (by Lemma 1) and all non-targets (since growing succeeds), thereby achieving deadbeat disturbance rejection.

As an example, let's again consider the system in Figure 5.4. Assume the following dynamics:

$$\begin{aligned}
x_1[k+1] &= x_1[k] + 2x_3[k] + x_4[k] + u_7[k] + d_1[k] && (5.33)\\
y_1[k] &= x_1[k] && (5.34)\\
x_2[k+1] &= 3x_2[k] + x_3[k] + x_6[k] + u_9[k] + d_2[k] && (5.35)\\
y_2[k] &= x_2[k] && (5.36)\\
x_3[k+1] &= x_1[k] + x_2[k] + x_5[k] + u_8[k] && (5.37)\\
y_3[k] &= x_3[k] && (5.38)\\
x_4[k+1] &= x_1[k] + x_4[k] + x_5[k] && (5.39)\\
y_4[k] &= x_4[k] && (5.40)\\
x_5[k+1] &= 2x_3[k] + x_4[k] + x_6[k] && (5.41)
\end{aligned}$$

$$
\begin{aligned}
y_5[k] &= x_5[k] && (5.42)\\
x_6[k+1] &= 3x_2[k]+x_5[k]+x_6[k] && (5.43)\\
y_6[k] &= x_6[k]. && (5.44)\\
& && (5.45)
\end{aligned}
$$

Using the propagation times

$$
\begin{aligned}
\Delta(X_7,O_4) &= 2\\
\Delta(X_8,O_5) &= 2\\
\Delta(X_9,O_6) &= 2,
\end{aligned}
$$

we determine

$$
\begin{aligned}
x_4[k+2] &= x_1[k+1]+x_4[k+1]+x_5[k+1]\\
&= 2x_1[k]+4x_3[k]+3x_4[k]+x_5[k]+x_6[k]+u_7[k]+d_1[k]\\
&= v_7[k]+d_1[k] && (5.46)\\
x_5[k+2] &= 2x_3[k+1]+x_4[k+1]+x_6[k+1]\\
&= 3x_1[k]+5x_2[k]+x_4[k]+4x_5[k]+x_6[k]+2u_8[k]\\
&= v_8[k] && (5.47)\\
x_6[k+2] &= 3x_2[k+1]+x_5[k+1]+x_6[k+1]\\
&= 12x_2[k]+5x_3[k]+x_4[k]+5x_6[k]+x_5[k]+3u_9[k]+3d_2[k]\\
&= v_9[k]+3d_2[k], && (5.48)
\end{aligned}
$$

which have the form of (5.10) and (5.12). From (5.46) – (5.48), the control laws of u_7, u_8 and u_9 are

$$
\begin{aligned}
u_7[k] &= v_7[k]-2x_1[k]-4x_3[k]-3x_4[k]-x_5[k]-x_6[k] && (5.49)\\
u_8[k] &= (v_8[k]-3x_1[k]-5x_2[k]-x_4[k]-4x_5[k]-x_6[k])/2 && (5.50)\\
u_9[k] &= (v_9[k]-12x_2[k]-5x_3[k]-x_4[k]-x_5[k]-5x_6[k])/3. && (5.51)
\end{aligned}
$$

Next, construct the augmented system (5.23)–(5.24):

$$
\begin{aligned}
x^*[k+1] &= A_1^*x^*[k]+B^*(v[k]+\tilde{d}[k]) && (5.52)\\
y^{tar}[k] &= C^*x^*[k], && (5.53)
\end{aligned}
$$

where

$$
\begin{aligned}
x^* &= \begin{bmatrix} \bar{x}[k] \\ w[k] \end{bmatrix}, \\
A_1^* &= \begin{bmatrix} \bar{A} & O \\ \bar{C} & I \end{bmatrix}, \\
B^* &= \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix}, \\
C^* &= \begin{bmatrix} \bar{C} & O \end{bmatrix},
\end{aligned}
$$

with

$$
\begin{aligned}
\bar{A} &= \text{block diag}\{A'_{\Delta(X_7,O_2)}, A'_{\Delta(X_8,O_5)}, A'_{\Delta(X_9,O_6)}\} \\
&= \text{block diag}\{A'_2, A'_2, A'_2\} \\
\bar{B} &= \text{block diag}\{B'_{\Delta(X_7,O_2)}, B'_{\Delta(X_8,O_5)}, B'_{\Delta(X_9,O_6)}\} \\
&= \text{block diag}\{B'_2, B'_2, B'_2\} \\
\bar{C} &= \text{block diag}\{C'_{\Delta(X_7,O_2)}, C'_{\Delta(X_8,O_5)}, C'_{\Delta(X_9,O_6)}\}, \\
&= \text{block diag}\{C'_2, C'_2, C'_2\},
\end{aligned}
$$

and

$$
A'_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \ B'_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \ C'_2 = \begin{bmatrix} 1 & 0 \end{bmatrix}.
$$

Using the propagation times from the three control agents to their targets, we use Lemma 3 to find K_1 and K_2 to place all eigenvalues of $A_1^* - B^*K$ at the origin:

$$
K_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}, \ K_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.
$$

The resulting control laws for $v[k]$ are

$$
\begin{aligned}
v_7[k] &= -\bar{x}_1[k] - \bar{x}_2[k] - w_1[k] \\
v_8[k] &= -\bar{x}_3[k] - \bar{x}_4[k] - w_2[k] \\
v_9[k] &= -\bar{x}_5[k] - \bar{x}_6[k] - w_3[k].
\end{aligned}
$$

Finally, substitute $v[k]$ into (5.49)–(5.51) to complete the controller design

$$
\begin{aligned}
u_7[k] &= -\bar{x}_1[k] - \bar{x}_2[k] - w_1[k] - 2x_1[k] - 4x_3[k] - 3x_4[k] - x_5[k] - x_6[k] \\
u_8[k] &= (-\bar{x}_3[k] - \bar{x}_4[k] - w_2[k] - 3x_1[k] - 5x_2[k] - x_4[k] - 4x_5[k] - x_6[k])/2 \\
u_9[k] &= (-\bar{x}_5[k] - \bar{x}_6[k] - w_3[k] - 12x_2[k] - 5x_3[k] - x_4[k] - x_5[k] - 5x_6[k])/3,
\end{aligned}
$$

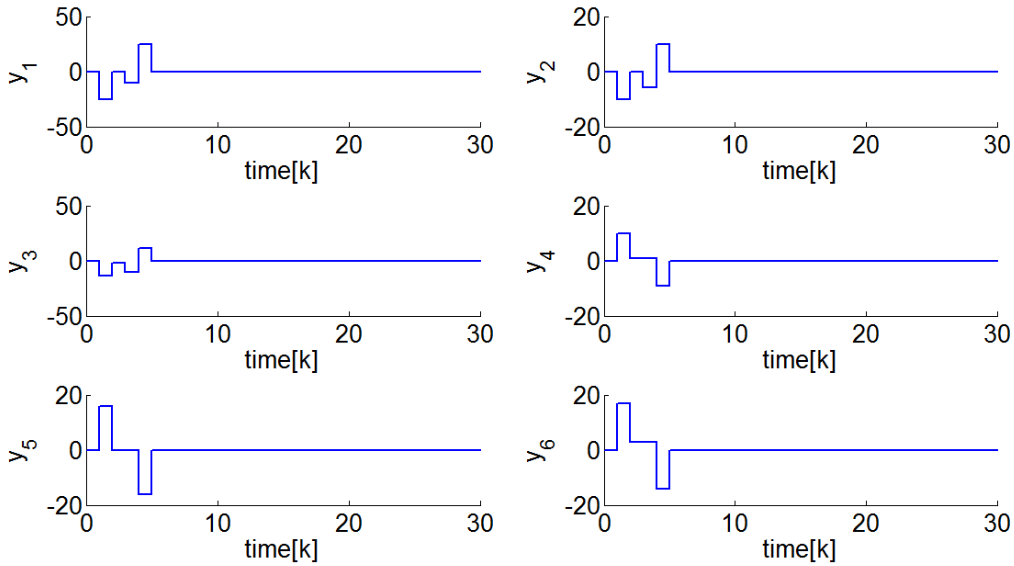

Figure 5.7: Simulation result for the example in Figure 5.4.

where

$$\bar{x}[k] = [v_7[k-2], v_7[k-1], v_8[k-2], v_8[k-1], v_9[k-2], v_9[k-1]]^T.$$

Matlab code for implementation of the controller is shown in Appendix C. Using the initial conditions and disturbances

$$\begin{aligned} x[0] &= [1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6]^T, \\ \bar{x}[0] &= [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\ w[0] &= [0\ \ 0\ \ 0]^T \\ d_1[k] &= d_2[k] = 1 \text{ (for } k \geq 0), \end{aligned}$$

simulations (see Figure 5.7) show that the control laws successfully reject constant disturbances.

5.3 Summary Of The Design Algorithm and A Detailed Example

The proposed disturbance rejection design algorithm is summarized in Figure 5.8. As a detailed example, consider the queue system in Figure 5.9. The dynamic equations of this system, with

Step 1: Choose the number of control agents, their placement, and targeting assignment to be consistent with Lemma 1 and such that both targeting and growing succeed. Note that this can always be done (e.g., in the extreme case, attach a control agent to each plant agent and assign the neighbour of X_i to be its target).

Step 2: Ignoring all disturbances, derive expressions for the control laws $u[k]$, in terms of $v[k]$, as indicated in (5.12) and (5.14).

Step 3: Find matrix $K = [K_1 \quad K_2]$ using (5.26) and (5.27) so that the eigenvalues of $A_1^* - B^*K$ are all at the origin. Set $v[k] = -K_1\bar{x}[k] - K_2 w[k]$.

Step 4: Substitute $v[k]$ into the $u[k]$ from Step 2.

Figure 5.8: Summary of the proposed disturbance rejection design algorithm.

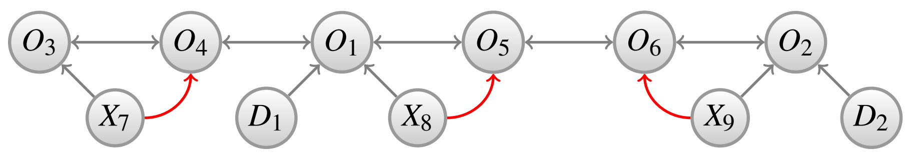

Figure 5.9: A six-agent queue plant with two disturbance agents.

$y_i[k] = x_i[k]$ for $i = 1,4,5,6$, are

$$\begin{aligned}
x_1[k+1] &= x_1[k] + x_4[k] + 2x_5[k] + u_8[k] + d_1[k] && (5.54)\\
x_{21}[k+1] &= -x_{22}[k] + u_9[k] && (5.55)\\
x_{22}[k+1] &= 2x_{21}[k] + d_2[k] && (5.56)\\
y_2[k] &= x_{21}[k] - x_{22}[k] && (5.57)\\
x_{31}[k+1] &= -2x_{32}[k] + u_7[k] && (5.58)\\
x_{32}[k+1] &= x_{31}[k] - x_{32}[k] + x_4[k] && (5.59)\\
y_3[k] &= x_{31}[k] + x_{32}[k] && (5.60)\\
x_4[k+1] &= y_3[k] + x_1[k] + 2x_4[k] && (5.61)\\
x_5[k+1] &= x_1[k] + 2x_6[k] && (5.62)\\
x_6[k+1] &= 2y_2[k] + x_5[k] + x_6[k]. && (5.63)
\end{aligned}$$

First, we verify that the conditions in Step 1 in Figure 5.8 are met. There are two control agents, X_8 and X_9, connected to the plant agents O_1 and O_2 where the two disturbance agents D_1 and D_2 are connected. This satisfies Lemma 1. It is also easy to verify that the system meets the conditions in Theorem 3, as the control agents and respective targets are close to one another, and there are no nodes in common between different fastest path from control agents to their target agents. For Theorem 4, the structure of this system shows that both of the conditions hold, since each fastest path from X_i to T_i (for $7 \leq i \leq 9$) is unique, and each plant agent lies on some fastest path. Full targeting and growing analyses confirm that targeting and growing succeed. For Step 2, ignoring the disturbances, we force

$$\begin{aligned} y_4[k+2] &= v_7[k] \\ y_5[k+2] &= v_8[k] \\ y_6[k+2] &= v_9[k], \end{aligned}$$

as the propagation time to the target agents O_4, O_5 and O_6 are

$$\begin{aligned} \Delta(X_7, O_4) &= 2 \\ \Delta(X_8, O_5) &= 2 \\ \Delta(X_9, O_6) &= 2. \end{aligned}$$

This yields the control laws

$$\begin{aligned} u_7[k] &= v_7[k] - 3x_1[k] - 3x_{31}[k] + x_{32}[k] \\ &\quad -6x_4[k] + 2x_5[k] - u_8[k] \qquad (5.64) \\ u_8[k] &= v_8[k] - x_1[k] - 4x_{21}[k] + 4x_{22}[k] \\ &\quad -x_4[k] - 4x_5[k] - 2x_6[k] \qquad (5.65) \\ u_9[k] &= (v_9[k] - x_1[k] + 2x_{21}[k] + 4x_{22}[k] \\ &\quad -x_5[k] - 3x_6[k])/2. \qquad (5.66) \end{aligned}$$

For Step 3, we construct the augmented system (5.23)–(5.24):

$$x^*[k+1] = A_1^* x^*[k] + B^*(v[k] + \tilde{d}[k]) \qquad (5.67)$$

$$y^{tar}[k] = C^* x^*[k], \qquad (5.68)$$

where

$$\begin{aligned} x^* &= \begin{bmatrix} \bar{x}[k] \\ w[k] \end{bmatrix}, \\ A_1^* &= \begin{bmatrix} \bar{A} & O \\ \bar{C} & I \end{bmatrix} \end{aligned}$$

$$
\begin{aligned}
&= \begin{bmatrix} 0&1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&1&0&0\\ 0&0&1&0&0&0&0&1&0\\ 0&0&0&0&1&0&0&0&1 \end{bmatrix}, \\
B^* &= \begin{bmatrix} \bar{B} \\ O \end{bmatrix} \\
&= \begin{bmatrix} 0&0&0\\ 1&0&0\\ 0&0&0\\ 0&1&0\\ 0&0&0\\ 0&0&1\\ 0&0&0\\ 0&0&0\\ 0&0&0 \end{bmatrix}, \\
C^* &= \begin{bmatrix} \bar{C} & O \end{bmatrix} \\
&= \begin{bmatrix} 1&0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0 \end{bmatrix},
\end{aligned}
$$

with

$$\bar{A} = \text{block diag}\{A'_{\Delta(X_7,O_3)}, A'_{\Delta(X_8,O_5)}, A'_{\Delta(X_9,O_6)}\} \tag{5.69}$$

$$\bar{B} = \text{block diag}\{B'_{\Delta(X_7,O_3)}, B'_{\Delta(X_8,O_5)}, B'_{\Delta(X_9,O_6)}\} \tag{5.70}$$

$$\bar{C} = \text{block diag}\{C'_{\Delta(X_7,O_3)}, C'_{\Delta(X_8,O_5)}, C'_{\Delta(X_9,O_6)}\}, \tag{5.71}$$

where $\Delta(X_7,O_3) = \Delta(X_8,O_5) = \Delta(X_9,O_6) = 2$ and

$$A'_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \ B'_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \ C'_2 = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

We next determine K_1 and K_2 based on (5.26) and (5.27) to place all eigenvalues of $A_1^* - B^*K$ at the origin:

$$K_1 = \begin{bmatrix} 1&1&0&0&0&0\\ 0&0&1&1&0&0\\ 0&0&0&0&1&1 \end{bmatrix}, \ K_2 = \begin{bmatrix} 1&0&0\\ 0&1&0\\ 0&0&1 \end{bmatrix}.$$

Thus, the control law $v[k]$ in (5.25) is fully specified. For Step 4, substitute $v[k]$ into (5.64)–(5.66) to complete the controller design as

$$
\begin{aligned}
u_7[k] &= v_7[k] - 3x_1[k] - 3x_{31}[k] + x_{32}[k] \\
&\quad -6x_4[k] + 2x_5[k] - u_8[k] \\
&= -\bar{x_1}[k] - \bar{x_2}[k] - w_1[k] - 3x_1[k] - 3x_{31}[k] + x_{32}[k] \\
&\quad -6x_4[k] + 2x_5[k] - u_8[k] \qquad (5.72) \\
u_8[k] &= v_8[k] - x_1[k] - 4x_{21}[k] + 4x_{22}[k] \\
&\quad -x_4[k] - 4x_5[k] - 2x_6[k] \\
&= -\bar{x_3}[k] - \bar{x_4}[k] - w_2[k] - x_1[k] - 4x_{21}[k] + 4x_{22}[k] \\
&\quad -x_4[k] - 4x_5[k] - 2x_6[k] \qquad (5.73) \\
u_9[k] &= (v_9[k] - x_1[k] + 2x_{21}[k] + 4x_{22}[k] \\
&\quad -x_5[k] - 3x_6[k])/2 \\
&= (-\bar{x_5}[k] - \bar{x_6}[k] - w_3[k] - x_1[k] + 2x_{21}[k] + 4x_{22}[k] \\
&\quad -x_5[k] - 3x_6[k])/2, \qquad (5.74)
\end{aligned}
$$

where

$$\bar{x}[k] = [v_7[k-2], v_7[k-1], v_8[k-2], v_8[k-1], v_9[k-2], v_9[k-1]]^T.$$

Using the initial conditions and disturbances

$$
\begin{aligned}
x[0] &= [1 \;\; x_{20} \;\; x_{30} \;\; 6 \;\; 7 \;\; 8]^T, \text{ where } x_{20} \triangleq [2 \;\; 3],\; x_{30} \triangleq [4 \;\; 5] \\
\bar{x}[0] &= [0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0]^T \\
w[0] &= [0 \;\; 0 \;\; 0]^T \\
d_1[k] &= d_2[k] = 1 \text{ (for } k \geq 0), \qquad (5.75)
\end{aligned}
$$

simulations (see Figure 5.10) show that the control laws successfully drive the outputs of all agents to zero. Matlab code for the simulation is given in Appendix D.

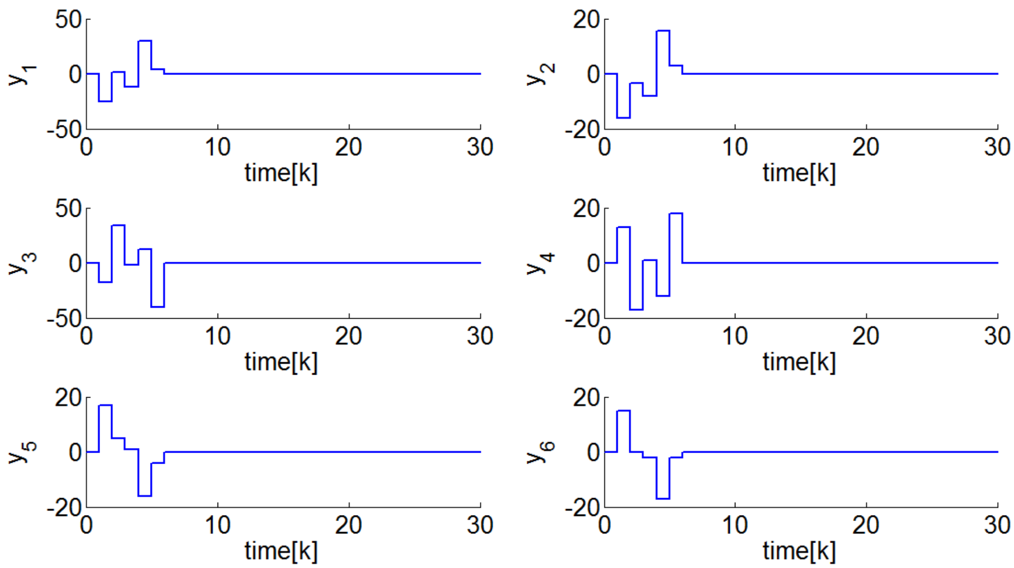

Figure 5.10: Simulation results for the six-agent queue system.

Chapter 6

Examples of Various Linear Multi-agent Systems

In the last chapter we demonstrated, using a queue and some grid systems, the effectiveness of the proposed control strategy. In this chapter, several additional examples of linear multi-agent systems with different structures are given to further illustrate the design algorithm in Figure 5.8. We consider networks with the following graph structures: a ring, a spider, a grid, a wheel, complete graphs, and null graphs. For simplicity, we assume that all the links between plant agents are bidirectional, and that the propagation time from each agent to any neighbour agent is one time step, i.e, $\delta_{ij} = 1$ and $\delta_i^d = 1$. For the dynamical equations in this chapter, we also assume for simplicity that $y_i[k] = x_i[k]$ (for $1 \leq i \leq n$). We present the control algorithm for each example with the minimum number of control agents. Notice that in this chapter, we violate our earlier notational requirement that D_i needs to be connected to plant agent O_i; the control algorithm remains the same with obvious notational changes. By the end of this chapter, we will have provided substantial evidence that the proposed control algorithm works well for the disturbance rejection problem for linear multi-agent systems.

6.1 Ring Structure

Consider the ring-structured system in Figure 6.1. This ring system has six plant agents and two disturbance agents. The dynamics of the ring system are as follows:

$$x_1[k+1] = x_2[k] + x_6[k] + u_7[k] + d_1[k] \quad (6.1)$$

$$x_2[k+1] = x_1[k] \quad (6.2)$$

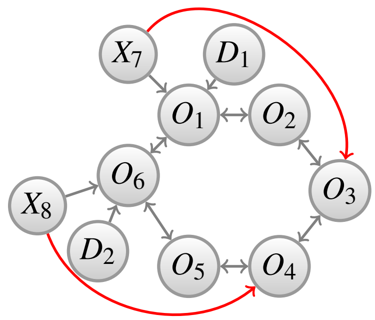

Figure 6.1: The ring system with two disturbance agents.

$$
\begin{aligned}
x_3[k+1] &= x_2[k]+x_4[k] && (6.3)\\
x_4[k+1] &= x_3[k]+x_5[k] && (6.4)\\
x_5[k+1] &= x_6[k] && (6.5)\\
x_6[k+1] &= x_1[k]+x_5[k]+u_8[k]+d_2[k]. && (6.6)
\end{aligned}
$$

According to the algorithm in Figure 5.8, the first step is to locate the control agents. Based on Lemma 1, two control agents X_1 and X_2 are put right beside agents D_1 and D_2, connected to plant agents O_1 and O_6. We set O_3 as the target of X_7 and O_4 as the target of X_8. The reader can verify that both targeting and growing succeed.

For Step 2, we ignore the disturbances and derive formulas for the control laws. From the network structure, we know that

$$
\begin{aligned}
\Delta(X_7,O_3) &= 3\\
\Delta(X_8,O_4) &= 3.
\end{aligned}
$$

Then we get

$$
\begin{aligned}
x_3[k+3] &= 2x_2[k]+x_4[k]+2x_6[k]+u_7[k]\\
&= v_7[k] && (6.7)\\
x_4[k+3] &= 2x_1[k]+2x_5[k]+x_3[k]+u_8[k]\\
&= v_8[k]. && (6.8)
\end{aligned}
$$

From (6.7) and (6.8), the control laws are

$$
\begin{aligned}
u_7[k] &= v_7[k]-2x_2[k]-x_4[k]-2x_6[k] && (6.9)\\
u_8[k] &= v_8[k]-2x_1[k]-2x_5[k]-x_3[k]. && (6.10)
\end{aligned}
$$

In Step 3 we find matrix $K = [K_1 \;\; K_2]$ so that all the eigenvalues of $A_1^* - B^*K$ are at the origin. Using

$$\begin{aligned}\bar{A} &= \text{block diag}\{A'_{\Delta(X_7,T_7)}, A'_{\Delta(X_8,T_8)}\}\\ \bar{B} &= \text{block diag}\{B'_{\Delta(X_7,T_7)}, B'_{\Delta(X_8,T_8)}\},\end{aligned}$$

we determine

$$\begin{aligned}A_1^* &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \in R^{r\times r},\\ B^* &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \in R^{r\times 1}.\end{aligned}$$

Hence, using(5.26) and (5.27), we set

$$K_1 = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix},\; K_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The control law $v[k] = -K_1\bar{v[k]} - K_2 w[k]$ is now fully specified. Substitute $v[k]$ into (6.9) and (6.10) to obtain

$$\begin{aligned}u_7[k] &= v_7[k] - 2x_2[k] - x_4[k] - 2x_6[k]\\ &= -\bar{x_1}[k] - \bar{x_2}[k] - \bar{x_3}[k] - w_1[k] - 2x_2[k] - x_4[k] - 2x_6[k]\\ u_8[k] &= v_8[k] - 2x_1[k] - 2x_5[k] - x_3[k]\\ &= -\bar{x_4}[k] - \bar{x_5}[k] - \bar{x_6}[k] - w_2[k] - 2x_1[k] - 2x_5[k] - x_3[k],\end{aligned}$$

where

$$\bar{x}[k] = [v_7[k-3], v_7[k-2], v_7[k-1], v_8[k-3], v_8[k-2], v_8[k-1]]^T.$$

Set the initial conditions and disturbances as follows:

$$\begin{aligned}x[k] &= [1\;\;2\;\;3\;\;4\;\;5\;\;6]^T\\ \bar{x}[k] &= [0\;\;0\;\;0\;\;0\;\;0\;\;0]^T\\ w[k] &= [0\;\;0]^T\\ d_1[k] &= 1 \text{ (for } k \geq 0)\\ d_2[k] &= 1 \text{ (for } k \geq 0).\end{aligned}$$

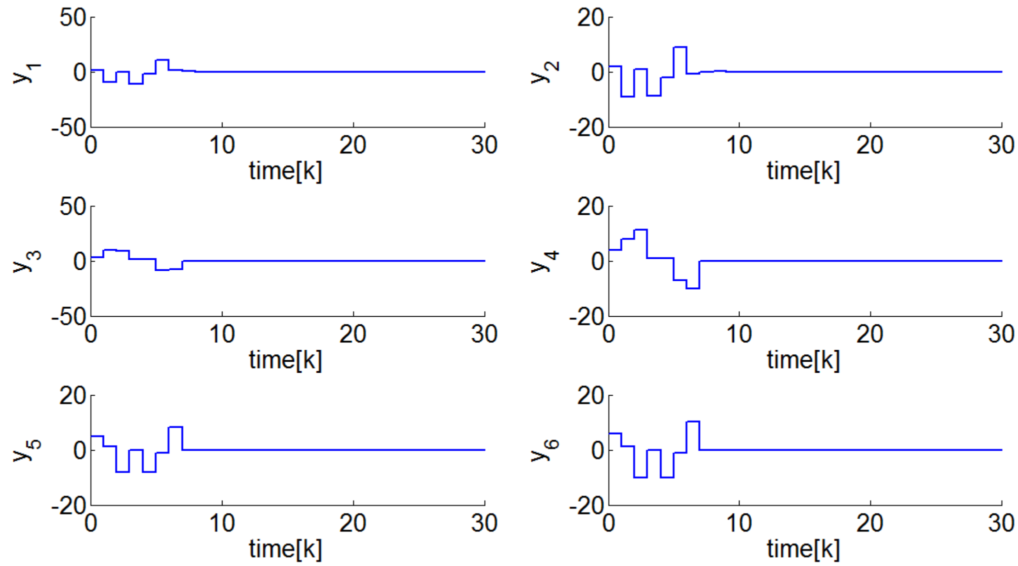

Figure 6.2: Simulation result for the ring structure system example.

The simulation results in Figure 6.2 show that the algorithm works. It rejects all the disturbances and zeros all the output signals of the plant agents in seven time steps.

6.2 Spider Structure

In this section, we study two spider structure systems. Consider the first spider structure system in Figure 6.3. This system has six plant agents and two disturbance agents. The dynamics of this spider system are as the follows:

$$x_1[k+1] = x_2[k]+x_4[k]+x_5[k]+x_6[k] \quad (6.11)$$
$$x_2[k+1] = x_1[k]+x_2[k]+x_3[k] \quad (6.12)$$
$$x_3[k+1] = 2x_2[k] \quad (6.13)$$
$$x_4[k+1] = x_1[k]+x_4[k]+u_9[k]+d_1[k] \quad (6.14)$$
$$x_5[k+1] = x_1[k]+x_5[k]+u_8[k] \quad (6.15)$$
$$x_6[k+1] = x_1[k]+x_6[k]+u_7[k]+d_2[k]. \quad (6.16)$$

From Lemma 1, we know that the system needs at least two control agents, attached to the plant agents to which the two disturbance agents D_1 and D_2 are attached. But for targeting and growing to succeed, additional control agents are needed. There are four branches in this system. Using

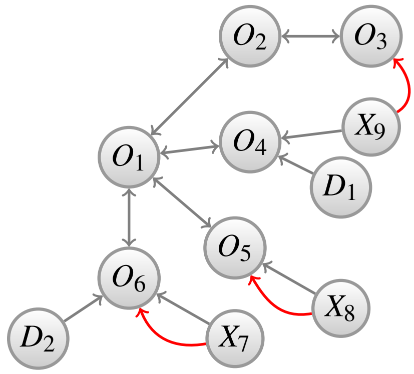

Figure 6.3: The spider system with six plant agents and two disturbance agents.

Theorem 3 and 4, we conclude that targeting and growing analyses succeed only if at least three control agents are introduced to the system. We set the control agents and assign their targets as shown in Figure 6.3. Targeting and growing analyses succeed for this arrangement.

Calculate the propagation time from X_i to T_i (for $7 \leq i \leq 9$):

$$\begin{aligned}\Delta(X_7, O_6) &= 1\\ \Delta(X_8, O_5) &= 1\\ \Delta(X_9, O_3) &= 4.\end{aligned}$$

Hence,

$$\begin{aligned}x_6[k+1] &= x_1[k] + x_6[k] + u_7[k]\\ &= v_7[k] \qquad (6.17)\\ x_5[k+1] &= x_1[k] + x_5[k] + u_8[k]\\ &= v_8[k] \qquad (6.18)\\ x_3[k+4] &= 2x_2[k+3]\\ &= 2x_1[k+2] + 2x_2[k+2] + 2x_3[k+2]\\ &= 2x_1[k+1] + 8x_2[k+1] + 2x_3[k+1] + 2x_4[k+1] + 2x_5[k+1] + 2x_6[k+1]\\ &= 14x_1[k] + 14x_2[k] + 8x_3[k] + 4x_4[k] + 4x_5[k]\\ &\quad +4x_6[k] + 2u_7[k] + 2u_8[k] + 2u_9[k]\\ &= v_9[k]. \qquad (6.19)\end{aligned}$$

The formulas of the control laws are derived from (6.17)–(6.19) as

$$u_7[k] = v_7[k] - x_1[k] - x_6[k] \qquad (6.20)$$

$$u_8[k] = v_8[k] - x_1[k] - x_5[k] \tag{6.21}$$

$$\begin{aligned} u_9[k] &= 0.5v_9[k] - 7x_1[k] - 7x_2[k] - 4x_3[k] - 2x_4[k] \\ &\quad -2x_5[k] - 2x_6[k] - u_7[k] - u_8[k]. \end{aligned} \tag{6.22}$$

The next step is to find the K matrix and the control laws for $v[k]$. As we know the propagation time from all the control agents to their targets, we determine the K_1 and K_2 matrices to be

$$K_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}, \; K_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

and the $v[k]$ control laws to be

$$v_7[k] = -\bar{x}_1[k] - w_1[k] \tag{6.23}$$

$$v_8[k] = -\bar{x}_2[k] - w_2[k] \tag{6.24}$$

$$v_9[k] = -\bar{x}_3[k] - \bar{x}_4[k] - \bar{x}_5[k] - \bar{x}_6[k] - w_3[k]. \tag{6.25}$$

The final control laws of $u[k]$ are

$$\begin{aligned} u_7[k] &= v_7[k] - x_1[k] - x_6[k] \\ &= -\bar{x}_1[k] - w_1[k] - x_1[k] - x_6[k] \\ u_8[k] &= v_8[k] - x_1[k] - x_5[k] \\ &= -\bar{x}_2[k] - w_2[k] - x_1[k] - x_5[k] \\ u_9[k] &= 0.5v_9[k] - 7x_1[k] - 7x_2[k] - 4x_3[k] - 2x_4[k] - 2x_5[k] - 2x_6[k] - u_7[k] - u_8[k] \\ &= 0.5(-\bar{x}_3[k] - \bar{x}_4[k] - \bar{x}_5[k] - \bar{x}_6[k] - w_3[k]) \\ &\quad -7x_1[k] - 7x_2[k] - 4x_3[k] - 2x_4[k] - 2x_5[k] - 2x_6[k] - u_7[k] - u_8[k], \end{aligned}$$

where

$$\bar{x}[k] = [v_7[k-1], v_8[k-1], v_9[k-4], v_9[k-3], v_9[k-2], v_9[k-1]]^T.$$

To run a simulation, we set the initial conditions and disturbances as

$$\begin{aligned} x[k] &= [1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6]^T \\ \bar{x}[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\ w[k] &= [0\ \ 0\ \ 0]^T \\ d_1[k] &= 1 \text{ (for } k \geq 0) \\ d_2[k] &= 1 \text{ (for } k \geq 0). \end{aligned}$$

From the simulation result in Figure 6.4, we conclude that the proposed algorithm works perfectly in rejecting the disturbances and zeroing all plant agents.

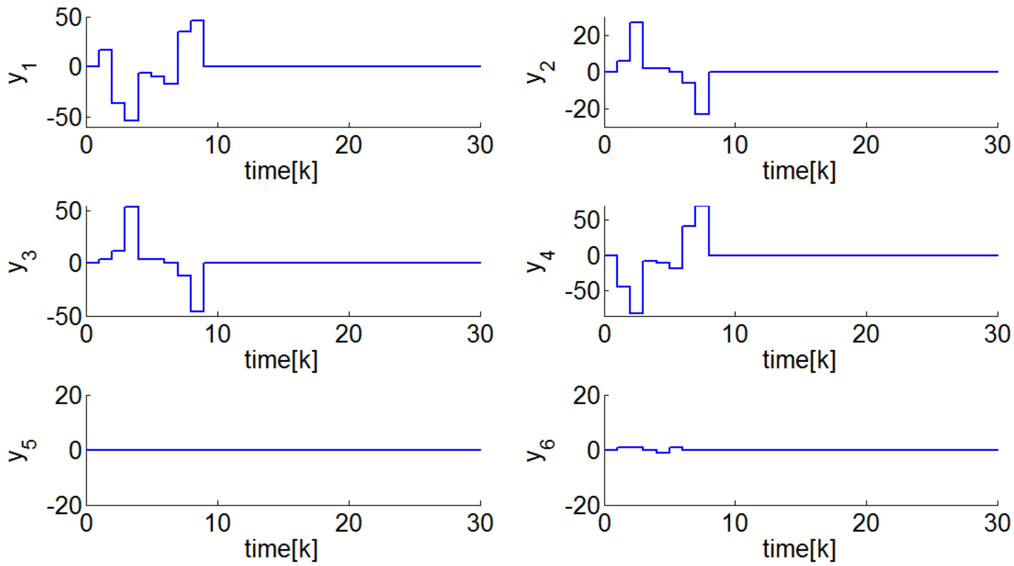

Figure 6.4: Simulation result for the first spider structure system example.

Now consider the second spider structure system in Figure 6.5. The structure of this system looks like the one in Figure 6.3. They both have two disturbances, three control agents and four branches. Even the way we assign targets for the control agents is similar. The only difference is that this system has one more plant agent.

The dynamics of the second spider system are

$$x_1[k+1] = x_2[k] + x_4[k] + x_6[k] + x_7[k] \tag{6.26}$$
$$x_2[k+1] = x_1[k] + 0.5x_3[k] \tag{6.27}$$
$$x_3[k+1] = x_2[k] + x_3[k] \tag{6.28}$$
$$x_4[k+1] = x_1[k] + 0.5x_4[k] + x_5[k] \tag{6.29}$$
$$x_5[k+1] = 0.5x_4[k] + u_{10}[k] + d_2[k] \tag{6.30}$$
$$x_6[k+1] = x_1[k] + 0.5x_6[k] + u_9[k] \tag{6.31}$$
$$x_7[k+1] = 0.5x_1[k] + u_8[k] + d_1[k]. \tag{6.32}$$

The propagation time from the three control agents to their respective target agents are

$$\begin{aligned} \Delta(X_8, O_7) &= 1 \\ \Delta(X_9, O_6) &= 1 \\ \Delta(X_{10}, O_3) &= 5, \end{aligned}$$

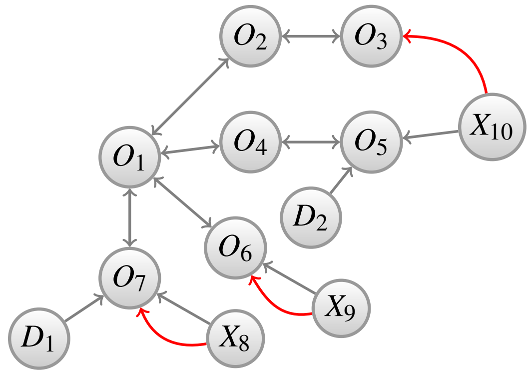

Figure 6.5: The spider system with seven plant agents and two disturbance agents.

so we determine that

$$
\begin{aligned}
x_6[k+1] &= 0.5x_1[k] + u_8[k] \\
&= v_8[k] \qquad (6.33) \\
x_5[k+1] &= x_1[k] + 0.5x_6[k] + u_9[k] \\
&= v_9[k] \qquad (6.34) \\
x_3[k+5] &= x_2[k+4] + x_3[k+4] \\
&= x_1[k+3] + x_2[k+3] + 1.5x_3[k+3] \\
&= x_1[k+2] + 2.5x_2[k+2] + 2x_3[k+2] + x_4[k+2] + x_6[k+2] + x_7[k+2] \\
&= 5x_1[k+1] + 3x_2[k+1] + 3.25x_3[k+1] + 1.5x_4[k+1] + x_5[k+1] \\
&\quad +1.5x_6[k+1] + x_7[k+1] + u_8[k+1] + u_9[k+1] \qquad (6.35) \\
&= 6x_1[k] + 6.75x_2[k] + 4.75x_3[k] + 4.75x_4[k] + 1.5x_5[k] \\
&\quad +4x_6[k] + 3.5x_7[k] + u_8[k] + u_9[k] + u_{10}[k] \\
&= v_{10}[k]. \qquad (6.36)
\end{aligned}
$$

Thus the formulas of the control laws can be derived from (6.33)–(6.36) as

$$
\begin{aligned}
u_8[k] &= v_8[k] - 0.5x_1[k] \qquad (6.37) \\
u_9[k] &= v_9[k] - x_1[k] - 0.5x_6[k] \qquad (6.38) \\
u_{10}[k] &= v_{10}[k] - 6x_1[k] - 6.75x_2[k] - 4.75x_3[k] - 4.75x_4[k] - 1.5x_5[k] \\
&\quad -4x_6[k] - 3.5x_7[k] - u_8[k] - u_9[k]. \qquad (6.39)
\end{aligned}
$$

Notice that terms $u_8[k+1]$ and $u_9[k+1]$ appear in (6.35). However, we want $y^{tar}[k]$ to be expressed as (for $n+1 \le i \le n+m$)

$$y_i^{tar}[k+\Delta_i] = \sum_{q \neq i} (\alpha_u^{qi} u_q[k] + \alpha_d^{qi} d_q[k]) + \alpha_1^i x_1[k] + \alpha_2^i x_2[k] + \cdots + \alpha_n^i x_n[k] + \alpha_u^i u_i[k] + \alpha_d^i d_i[k].$$

To this end, we substitute the expressions

$$u_8[k+1] = -0.5x_1[k+1] \tag{6.40}$$

$$u_9[k+1] = -x_1[k+1] - 0.5x_6[k+1], \tag{6.41}$$

into (6.35) at time step $k+1$, where we set $v_8[k+1] = 0$ and $v_9[k+1] = 0$ in (6.37) and (6.38). By substituting (6.40) and (6.41) into (6.35), we get the formula of $y_3^{tar}[k+5]$.

Next, determine the K_1 and K_2 matrices

$$K_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \; K_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

and the $v[k]$ control laws

$$v_8[k] = -\bar{x}_1[k] - w_1[k] \tag{6.42}$$

$$v_9[k] = -\bar{x}_2[k] - w_2[k] \tag{6.43}$$

$$v_{10}[k] = -\bar{x}_3[k] - \bar{x}_4[k] - \bar{x}_5[k] - \bar{x}_6[k] - \bar{x}_7[k] - w_3[k]. \tag{6.44}$$

The final control laws for $u[k]$ are

$$\begin{aligned} u_8[k] &= v_8[k] - 0.5x_1[k] \\ &= -\bar{x}_1[k] - w_1[k] - 0.5x_1[k] \\ u_9[k] &= v_9[k] - x_1[k] - 0.5x_6[k] \\ &= -\bar{x}_2[k] - w_2[k] - x_1[k] - 0.5x_6[k] \\ u_{10}[k] &= v_{10}[k] - 6x_1[k] - 6.75x_2[k] - 4.75x_3[k] - 4.75x_4[k] - 1.5x_5[k] \\ &\quad -4x_6[k] - 3.5x_7[k] - u_8[k] - u_9[k] \\ &= -\bar{x}_3[k] - \bar{x}_4[k] - \bar{x}_5[k] - \bar{x}_6[k] - \bar{x}_7[k] - w_3[k] \\ &\quad -6x_1[k] - 6.75x_2[k] - 4.75x_3[k] - 4.75x_4[k] - 1.5x_5[k] \\ &\quad -4x_6[k] - 3.5x_7[k] - u_8[k] - u_9[k], \end{aligned}$$

where

$$\bar{x}[k] = [v_8[k-1], v_9[k-1], v_{10}[k-4], v_{10}[k-3], v_{10}[k-2], v_{10}[k-1]]^T.$$

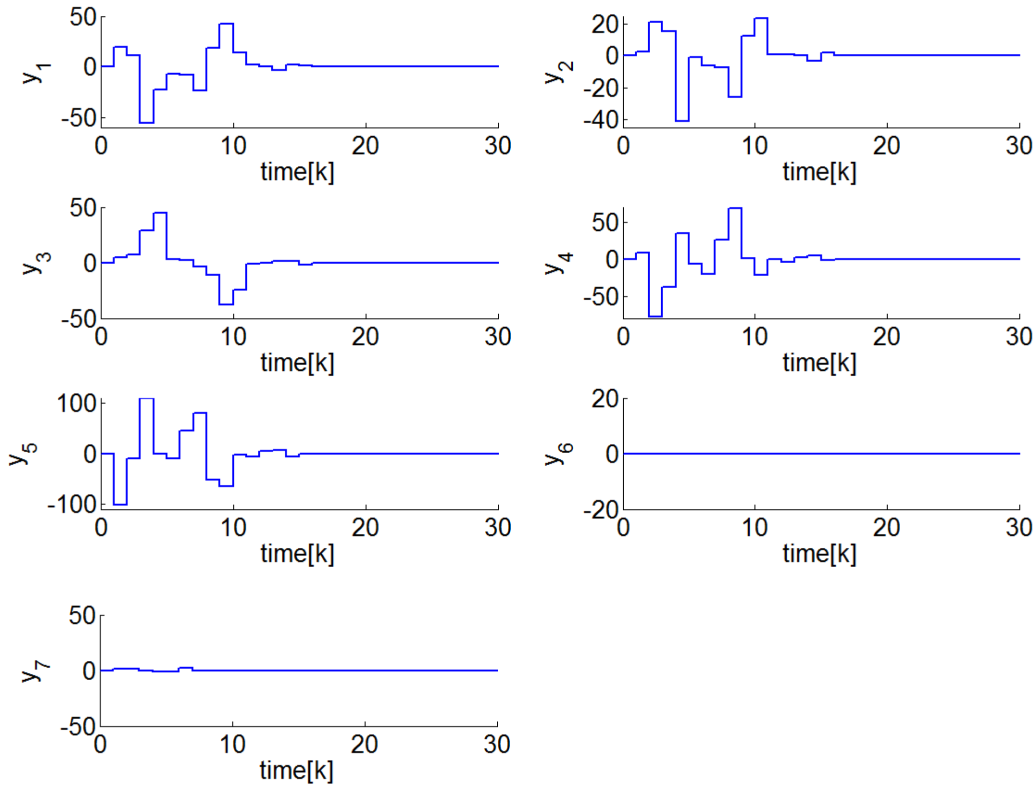

Figure 6.6: Simulation result for the second spider structure system example.

Set the initial conditions and disturbances as

$$
\begin{aligned}
x[k] &= [1 \;\; 2 \;\; 3 \;\; 4 \;\; 5 \;\; 6 \;\; 7]^T \\
\bar{x}[k] &= [0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0]^T \\
w[k] &= [0 \;\; 0 \;\; 0]^T \\
d_1[k] &= 1 \text{ (for } k \geq 0) \\
d_2[k] &= 1 \text{ (for } k \geq 0).
\end{aligned}
$$

The simulation result in Figure 6.6 shows that disturbance rejection is obtained again, although derivation of the control laws for the second spider example is trickier than for the first spider example.

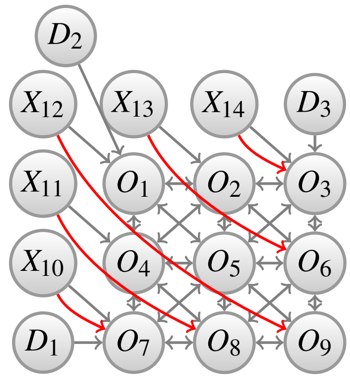

Figure 6.7: The grid structure system with nine plant agents and three disturbance agents.

6.3 Grid Structure

In this section, we consider an example of a grid structure with nine agents where additional diagonal edges are included between plant agents. As shown in Figure 6.7, there are three disturbances connected to the grid structure system. The dynamics are taken to be

$$x_1[k+1] = x_2[k]+x_4[k]+x_5[k]+u_{12}[k]+d_2[k] \quad (6.45)$$
$$x_2[k+1] = x_1[k]+x_3[k]+x_4[k]+x_5[k]+x_6[k]+u_{13}[k] \quad (6.46)$$
$$x_3[k+1] = x_2[k]+x_5[k]+x_6[k]+u_{14}[k]+d_3[k] \quad (6.47)$$
$$x_4[k+1] = x_1[k]+x_2[k]+x_5[k]+x_7[k]+x_8[k]+u_{11}[k] \quad (6.48)$$
$$x_5[k+1] = x_1[k]+x_2[k]+x_3[k]+x_4[k]+x_6[k]+x_7[k]+x_8[k]+x_9[k] \quad (6.49)$$
$$x_6[k+1] = x_2[k]+x_3[k]+x_5[k]+x_8[k]+x_9[k] \quad (6.50)$$
$$x_7[k+1] = x_4[k]+x_5[k]+x_8[k]+u_{10}[k]+d_1[k] \quad (6.51)$$
$$x_8[k+1] = x_4[k]+x_5[k]+x_6[k]+x_7[k]+x_9[k] \quad (6.52)$$
$$x_9[k+1] = x_5[k]+x_6[k]+x_8[k]. \quad (6.53)$$

Based on Lemma 1, a control agent must be placed next to each of the three disturbance agents, so we place X_{10} connected to O_7, X_{12} connected to O_1, and X_{14} connected to O_3. However, using just these three control agents, targets cannot be chosen such that each of the nine plant agents lies on one of the fastest paths from some control agent to its target. To satisfy the conditions of Theorems 3 and 4, we add two more control agents, X_{11} and X_{13}, as indicated in Figure 6.7. For this arrangement, targeting and growing analyses succeed.

The propagation time from the control agents to their target agents are

$$\Delta(X_{10}, O_7) = 1$$

$$\begin{aligned}\Delta(X_{11},O_8) &= 2\\ \Delta(X_{12},O_9) &= 3\\ \Delta(X_{13},O_6) &= 2\\ \Delta(X_{14},O_3) &= 1.\end{aligned}$$

To derive the expressions for the control laws $u[k]$, we first determine

$$\begin{aligned}x_7[k+1] &= x_4[k]+x_5[k]+x_8[k]+u_{10}[k]\\ &= v_{10}[k] \qquad (6.54)\\ x_8[k+2] &= x_4[k+1]+x_5[k+1]+x_6[k+1]+x_7[k+1]+x_9[k+1]\\ &= 2x_1[k]+3x_2[k]+2x_3[k]+2x_4[k]+4x_5[k]+2x_6[k]\\ &\quad +2x_7[k]+5x_8[k]+2x_9[k]+u_{10}[k]+u_{11}[k]\\ &= v_{11}[k] \qquad (6.55)\\ x_9[k+3] &= x_5[k+2]+x_6[k+2]+x_8[k+2]\\ &= x_1[k+1]+2x_2[k+1]+2x_3[k+1]+2x_4[k+1]+2x_5[k+1]\\ &\quad +2x_6[k+1]+2x_7[k+1]+2x_8[k+1]+3x_9[k+1]\\ &= 6x_1[k]+9x_2[k]+6x_3[k]+9x_4[k]+16x_5[k]+11x_6[k]+6x_7[k]\\ &\quad +11x_8[k]+6x_9[k]+2u_{10}[k]+2u_{11}[k]+u_{12}[k]+2u_{13}[k]+2u_{14}[k]\\ &= v_{12}[k] \qquad (6.56)\\ x_6[k+2] &= x_2[k+1]+x_3[k+1]+x_5[k+1]+x_8[k+1]+x_9[k+1]\\ &= 2x_1[k]+2x_2[k]+2x_3[k]+3x_4[k]+4x_5[k]+5x_6[k]\\ &\quad +2x_7[k]+2x_8[k]+2x_9[k]+u_{13}[k]+u_{14}[k]\\ &= v_{13}[k] \qquad (6.57)\\ x_3[k+1] &= x_2[k]+x_5[k]+x_6[k]+u_{14}[k]\\ &= v_{14}[k], \qquad (6.58)\end{aligned}$$

from which we derive that

$$\begin{aligned}u_{10}[k] &= v_{10}[k]-x_4[k]-x_5[k]-x_8[k] \qquad (6.59)\\ u_{11}[k] &= v_{11}[k]-2x_1[k]-3x_2[k]-2x_3[k]-2x_4[k]-4x_5[k]\\ &\quad -2x_6[k]-2x_7[k]-5x_8[k]-2x_9[k]-u_{10}[k] \qquad (6.60)\\ u_{12}[k] &= v_{12}[k]-6x_1[k]-9x_2[k]-6x_3[k]-9x_4[k]-16x_5[k]-11x_6[k]\\ &\quad -6x_7[k]-11x_8[k]-6x_9[k]-2u_{10}[k]-2u_{11}[k]-2u_{13}[k]-2u_{14}[k] \qquad (6.61)\\ u_{13}[k] &= v_{13}[k]-2x_1[k]-2x_2[k]-2x_3[k]-3x_4[k]-4x_5[k]-5x_6[k]\\ &\quad -2x_7[k]-2x_8[k]-2x_9[k]-u_{14}[k] \qquad (6.62)\\ u_{14}[k] &= v_{14}[k]-x_2[k]-x_5[k]-x_6[k]. \qquad (6.63)\end{aligned}$$

Using the propagation times for the fastest paths from X_i to T_i (for $10 \le i \le 14$), we can determine the matrices K_1 and K_2, following (5.26) and (5.27), to be

$$K_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \; K_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

The control laws $v[k]$ are

$$v_{10}[k] = -\bar{x}_1[k] - w_1[k] \tag{6.64}$$
$$v_{11}[k] = -\bar{x}_2[k] - \bar{x}_3[k] - w_2[k] \tag{6.65}$$
$$v_{12}[k] = -\bar{x}_4[k] - \bar{x}_5[k] - \bar{x}_6[k] - w_3[k] \tag{6.66}$$
$$v_{13}[k] = -\bar{x}_7[k] - \bar{x}_8[k] - w_4[k] \tag{6.67}$$
$$v_{14}[k] = -\bar{x}_9[k] - w_5[k]. \tag{6.68}$$

Substitute $v[k]$ into (6.59)–(6.63)

$$\begin{aligned}
u_{10}[k] &= v_{10}[k] - x_4[k] - x_5[k] - x_8[k] \\
&= -\bar{x}_1[k] - w_1[k] - x_4[k] - x_5[k] - x_8[k] \\
u_{11}[k] &= v_{11}[k] - 2x_1[k] - 3x_2[k] - 2x_3[k] - 2x_4[k] - 4x_5[k] \\
&\quad -2x_6[k] - 2x_7[k] - 5x_8[k] - 2x_9[k] - u_{10}[k] \\
&= -\bar{x}_2[k] - \bar{x}_3[k] - w_2[k] - 2x_1[k] - 3x_2[k] - 2x_3[k] \\
&\quad -2x_4[k] - 4x_5[k] - 2x_6[k] - 2x_7[k] - 5x_8[k] - 2x_9[k] - u_{10}[k] \\
u_{12}[k] &= v_{12}[k] - 6x_1[k] - 9x_2[k] - 6x_3[k] - 9x_4[k] - 16x_5[k] - 11x_6[k] \\
&\quad -6x_7[k] - 11x_8[k] - 6x_9[k] - 2u_{10}[k] - 2u_{11}[k] - 2u_{13}[k] - 2u_{14}[k] \\
&= -\bar{x}_4[k] - \bar{x}_5[k] - \bar{x}_6[k] - w_3[k] - 6x_1[k] - 9x_2[k] \\
&\quad -6x_3[k] - 9x_4[k] - 16x_5[k] - 11x_6[k] - 6x_7[k] - 11x_8[k] - 6x_9[k] \\
&\quad -2u_{10}[k] - 2u_{11}[k] - 2u_{13}[k] - 2u_{14}[k] \\
u_{13}[k] &= v_{13}[k] - 2x_1[k] - 2x_2[k] - 2x_3[k] - 3x_4[k] - 4x_5[k] - 5x_6[k] \\
&\quad -2x_7[k] - 2x_8[k] - 2x_9[k] - u_{14}[k] \\
&= -\bar{x}_7[k] - \bar{x}_8[k] - w_4[k] - 2x_1[k] - 2x_2[k] - 2x_3[k] - 3x_4[k] \\
&\quad -4x_5[k] - 5x_6[k] - 2x_7[k] - 2x_8[k] - 2x_9[k] - u_{14}[k] \\
u_{14}[k] &= v_{14}[k] - x_2[k] - x_5[k] - x_6[k] \\
&= -\bar{x}_9[k] - w_5[k] - x_2[k] - x_5[k] - x_6[k],
\end{aligned}$$

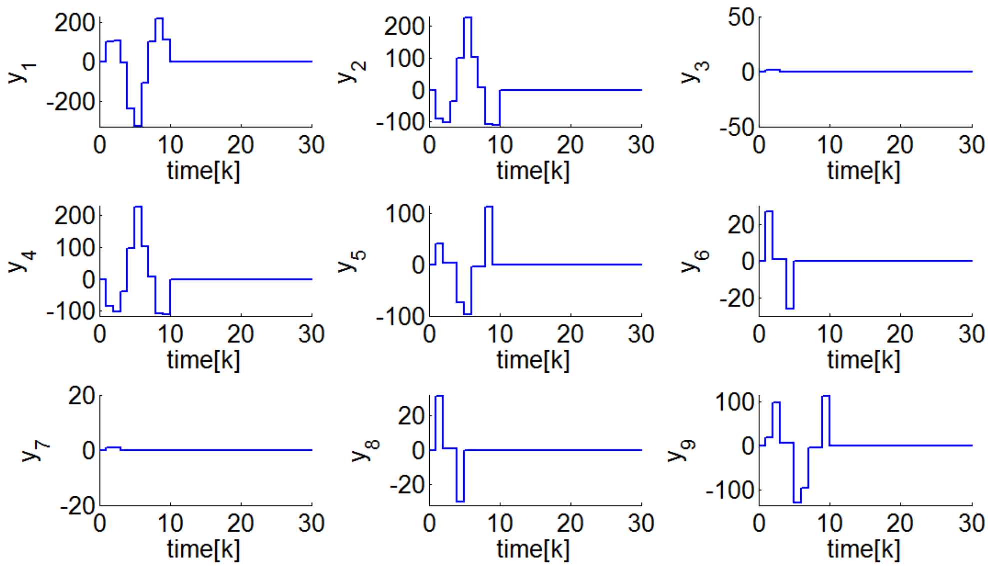

Figure 6.8: Simulation result for the grid structure system example.

where

$$\bar{x}[k] = [v_{10}[k-1], v_{11}[k-2], v_{11}[k-1], v_{12}[k-3], v_{12}[k-2], \\ v_{12}[k-1], v_{13}[k-2], v_{13}[k-1], v_{14}[k-1]]^T.$$

Set the initial conditions and disturbances as as

$$\begin{aligned} x[k] &= [1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7\ \ 8\ \ 9]^T \\ \bar{x}[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\ w[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\ d_1[k] &= 1\ (\text{for } k \geq 0) \\ d_2[k] &= 1\ (\text{for } k \geq 0). \end{aligned}$$

The simulation results in Figure 6.8 show that we successfully achieve deadbeat disturbance rejection.

6.4 Wheel Structure

We consider two wheel-structure systems in this section. A wheel structure is similar to a ring structure, but it has a center plant agent connected to all other plant agents, which greatly reduces

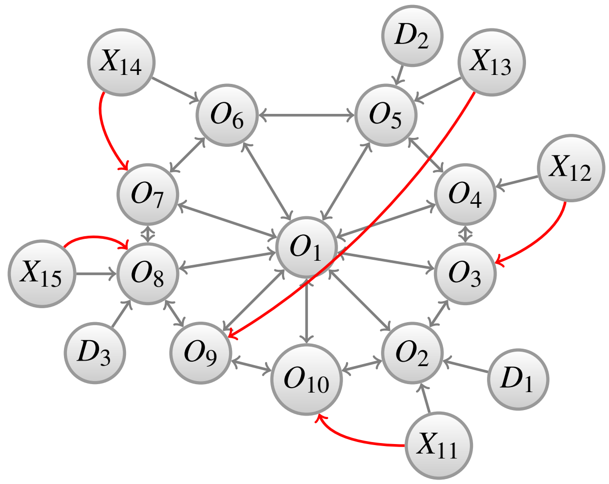

Figure 6.9: The first wheel structure system with ten plant agents and three disturbance agents.

some propagation times. The first example is a wheel structure system with ten plant agents and three disturbance agents, shown in Figure 6.9. The dynamics of this system are as follows:

$$
\begin{aligned}
x_1[k+1] &= x_2[k]+x_3[k]+x_4[k]+x_5[k]+x_6[k]+x_7[k]+x_8[k]+x_9[k]+x_{10}[k] && (6.69)\\
x_2[k+1] &= x_1[k]+x_3[k]+x_{10}[k]+u_{11}[k]+d_1[k] && (6.70)\\
x_3[k+1] &= x_1[k]+x_2[k]+x_4[k] && (6.71)\\
x_4[k+1] &= x_1[k]+x_3[k]+x_5[k]+u_{12}[k] && (6.72)\\
x_5[k+1] &= x_1[k]+x_4[k]+x_6[k]+u_{13}[k]+d_2[k] && (6.73)\\
x_6[k+1] &= x_1[k]+x_5[k]+x_7[k]+u_{14}[k] && (6.74)\\
x_7[k+1] &= x_1[k]+x_6[k]+x_8[k] && (6.75)\\
x_8[k+1] &= x_1[k]+x_7[k]+x_9[k]+u_{15}[k]+d_3[k] && (6.76)\\
x_9[k+1] &= x_1[k]+x_8[k]+x_{10}[k] && (6.77)\\
x_{10}[k+1] &= x_1[k]+x_2[k]+x_9[k]. && (6.78)
\end{aligned}
$$

According to Lemma 1, three control agents (X_{11}, X_{13}, and X_{15}) need to be placed beside the disturbance agents. To satisfy Theorem 3 and 4 in this case, at least two more control agents (X_{14} and X_{12}) are needed. We assign O_9 as the target of X_{13} so that the center agent O_1 lies on the fastest path from X_{13} to O_9. The reader can verify that both targeting and growing succeed for this setup.

The propagation time for each fastest path from X_i to T_i (for $11 \leq i \leq 15$) is

$$
\begin{aligned}
\Delta(X_{11},O_{10}) &= 2\\
\Delta(X_{12},O_3) &= 2
\end{aligned}
$$

$$
\begin{aligned}
\Delta(X_{13}, O_9) &= 3 \\
\Delta(X_{14}, O_7) &= 2 \\
\Delta(X_{15}, O_8) &= 1,
\end{aligned}
\tag{6.79}
$$

and therefore we have

$$
\begin{aligned}
x_8[k+1] &= x_1[k] + x_7[k] + x_9[k] + u_{15}[k] \\
&= v_{15}[k] \\
x_{10}[k+2] &= x_1[k+1] + x_9[k+1] + x_2[k+1] \\
&= 2x_1[k] + x_2[k] + 2x_3[k] + x_4[k] + x_5[k] + x_6[k] \\
&\quad + x_7[k] + 2x_8[k] + x_9[k] + 3x_{10}[k] + u_{11}[k] \\
&= v_{11}[k] \\
x_3[k+2] &= x_1[k+1] + x_2[k+1] + x_4[k+1] \\
&= 2x_1[k] + x_2[k] + 3x_3[k] + x_4[k] + 2x_5[k] + x_6[k] \\
&\quad + x_7[k] + x_8[k] + x_9[k] + 2x_{10}[k] + u_{11}[k] + u_{12}[k] \\
&= v_{12}[k] \\
x_7[k+2] &= x_1[k+1] + x_6[k+1] + x_8[k+1] \\
&= 2x_1[k] + x_2[k] + x_3[k] + x_4[k] + 2x_5[k] + x_6[k] \\
&\quad + 3x_7[k] + x_8[k] + 2x_9[k] + x_{10}[k] + u_{15}[k] + u_{14}[k] \\
&= v_{14}[k] \\
x_9[k+3] &= x_1[k+2] + x_8[k+2] + x_{10}[k+2] \\
&= x_1[k+1] + 2x_2[k+1] + x_3[k+1] + x_4[k+1] + x_5[k+1] + x_6[k+1] \\
&\quad + x_7[k+1] + x_8[k+1] + 2x_9[k+1] + x_{10}[k+1] \\
&= 11x_1[k] + 3x_2[k] + 4x_3[k] + 3x_4[k] + 3x_6[k] + 3x_7[k] + 4x_8[k] + 3x_9[k] \\
&\quad + 5x_{10}[k] + 2u_{11}[k] + u_{12}[k] + u_{13}[k] + u_{14}[k] + u_{15}[k] \\
&= v_{13}[k].
\end{aligned}
$$

Use these results to derive the control laws $u[k]$:

$$
u_{15}[k] = v_{15}[k] - x_1[k] - x_7[k] - x_9[k] \tag{6.80}
$$

$$
\begin{aligned}
u_{11}[k] &= v_{11}[k] - 2x_1[k] - x_2[k] - 2x_3[k] - x_4[k] - x_5[k] \\
&\quad - x_6[k] - x_7[k] - 2x_8[k] - x_9[k] - 3x_{10}[k]
\end{aligned}
\tag{6.81}
$$

$$
\begin{aligned}
u_{12}[k] &= v_{12}[k] - 2x_1[k] - x_2[k] - 3x_3[k] - x_4[k] - 2x_5[k] \\
&\quad - x_6[k] - x_7[k] - x_8[k] - x_9[k] - 2x_{10}[k] - u_{11}[k]
\end{aligned}
\tag{6.82}
$$

$$
u_{14}[k] = v_{14}[k] - 2x_1[k] - x_2[k] - x_3[k] - x_4[k] - 2x_5[k]
$$

$$-x_6[k]-3x_7[k]-x_8[k]-2x_9[k]-x_{10}[k]-u_{15}[k] \quad (6.83)$$

$$\begin{aligned} u_{13}[k] &= v_{13}[k]-11x_1[k]-3x_2[k]-4x_3[k]-3x_4[k]-3x_6[k]-3x_7[k] \\ &\quad -4x_8[k]-3x_9[k]-5x_{10}[k]-2u_{11}[k]-u_{12}[k]-u_{14}[k]-u_{15}[k]. \end{aligned} \quad (6.84)$$

From the propagation time $\Delta(X_i, T_i)$ (for $11 \le i \le 15$), we can find the matrices of K_1 and K_2:

$$K_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \; K_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Thus the control laws of $v[k]$ are

$$\begin{aligned} v_{11}[k] &= -\bar{x}_1[k]-\bar{x}_2[k]-w_1[k] \\ v_{12}[k] &= -\bar{x}_3[k]-\bar{x}_4[k]-w_2[k] \\ v_{13}[k] &= -\bar{x}_5[k]-\bar{x}_6[k]-\bar{x}_7[k]-w_3[k] \\ v_{14}[k] &= -\bar{x}_8[k]-\bar{x}_9[k]-w_4[k] \\ v_{15}[k] &= -\bar{x}_{10}[k]-w_5[k]. \end{aligned}$$

Substitute $v[k]$ into the expressions for $u[k]$ to obtain

$$\begin{aligned} u_{15}[k] &= v_{15}[k]-x_1[k]-x_7[k]-x_9[k] \\ &= -\bar{x}_{10}[k]-w_5[k]-x_1[k]-x_7[k]-x_9[k] \\ u_{11}[k] &= v_{11}[k]-2x_1[k]-x_2[k]-2x_3[k]-x_4[k]-x_5[k] \\ &\quad -x_6[k]-x_7[k]-2x_8[k]-x_9[k]-3x_{10}[k] \\ &= -\bar{x}_1[k]-\bar{x}_2[k]-w_1[k]-2x_1[k]-x_2[k]-2x_3[k] \\ &\quad -x_4[k]-x_5[k]-x_6[k]-x_7[k]-2x_8[k]-x_9[k]-3x_{10}[k] \\ u_{12}[k] &= v_{12}[k]-2x_1[k]-x_2[k]-3x_3[k]-x_4[k]-2x_5[k] \\ &\quad -x_6[k]-x_7[k]-x_8[k]-x_9[k]-2x_{10}[k]-u_{11}[k] \\ &= -\bar{x}_3[k]-\bar{x}_4[k]-w_2[k]-2x_1[k]-x_2[k]-3x_3[k]-x_4[k] \\ &\quad -2x_5[k]-x_6[k]-x_7[k]-x_8[k]-x_9[k]-2x_{10}[k]-u_{11}[k] \\ u_{14}[k] &= v_{14}[k]-2x_1[k]-x_2[k]-x_3[k]-x_4[k]-2x_5[k] \\ &\quad -x_6[k]-3x_7[k]-x_8[k]-2x_9[k]-x_{10}[k]-u_{15}[k] \\ &= -\bar{x}_8[k]-\bar{x}_9[k]-w_4[k]-2x_1[k]-x_2[k]-x_3[k]-x_4[k] \\ &\quad -2x_5[k]-x_6[k]-3x_7[k]-x_8[k]-2x_9[k]-x_{10}[k]-u_{15}[k] \\ u_{13}[k] &= v_{13}[k]-11x_1[k]-3x_2[k]-4x_3[k]-3x_4[k]-3x_6[k]-3x_7[k]-4x_8[k] \end{aligned}$$

$$
\begin{aligned}
& -3x_9[k]-5x_{10}[k]-2u_{11}[k]-u_{12}[k]-u_{14}[k]-u_{15}[k] \\
= \; & -\bar{x}_5[k]-\bar{x}_6[k]-\bar{x}_7[k]-w_3[k]-11x_1[k]-3x_2[k] \\
& -4x_3[k]-3x_4[k]-3x_6[k]-3x_7[k]-4x_8[k]-3x_9[k]-5x_{10}[k] \\
& -2u_{11}[k]-u_{12}[k]-u_{14}[k]-u_{15}[k],
\end{aligned}
$$

where

$$
\begin{aligned}
\bar{x}[k] = [&v_{11}[k-2], v_{11}[k-1], v_{12}[k-2], v_{12}[k-1], v_{13}[k-3], \\
&v_{13}[k-2], v_{13}[k-1], v_{14}[k-2], v_{14}[k-1], v_{15}[k-1]]^T.
\end{aligned}
$$

Set the initial conditions and disturbance signals to

$$
\begin{aligned}
x[k] &= [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10]^T \\
\bar{x}[k] &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T \\
w[k] &= [0\ 0\ 0\ 0\ 0]^T \\
d_1[k] &= 1 \text{ (for } k \geq 0) \\
d_2[k] &= 1 \text{ (for } k \geq 0).
\end{aligned}
$$

The simulation results in Figure 6.10 show that deadbeat disturbance rejection is successfully achieved.

The second wheel structure system has five plant agents and two disturbance agents. However, there is a disturbance agent connected to the center plant agent in this example. The dynamical equations of this system are

$$
\begin{aligned}
x_1[k+1] &= 2x_2[k]+x_4[k]+x_5[k] && (6.85) \\
x_2[k+1] &= x_1[k]+x_3[k]+x_5[k] && (6.86) \\
x_3[k+1] &= x_2[k]-x_4[k]+x_5[k]+u_7[k] && (6.87) \\
x_4[k+1] &= x_1[k]+3x_3[k]+x_5[k]+u_8[k]+d_1[k] && (6.88) \\
x_5[k+1] &= x_1[k]+x_2[k]+x_3[k]+x_4[k]+u_6[k]+d_2[k]. && (6.89)
\end{aligned}
$$

As required by Lemma 1, we place a control agent X_6 beside D_2, connected to O_5. Similarly, we connect another control agent X_8 to O_4 because of D_1. To satisfy Theorem 3 and 4, one more control agent X_7 is connected to O_3. For growing to succeed, we assign O_1 as T_8, O_2 as T_7 and O_5 as T_6. With this arrangement, both targeting and growing succeed.

The propagation times from X_i to T_i (for $6 \leq i \leq 8$) are

$$
\begin{aligned}
\Delta(X_6, O_5) &= 1 \\
\Delta(X_7, O_2) &= 2 \\
\Delta(X_8, O_1) &= 2,
\end{aligned}
$$

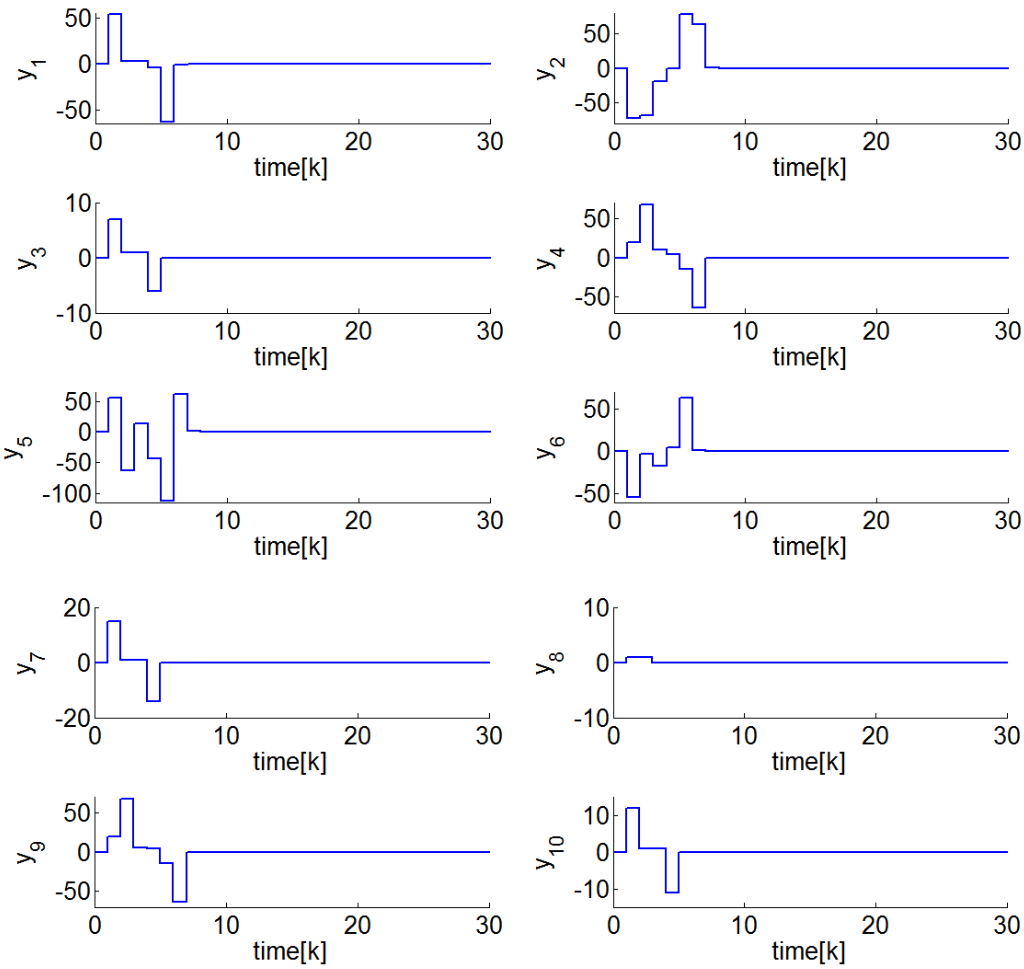

Figure 6.10: Simulation results for the first wheel structure system example.

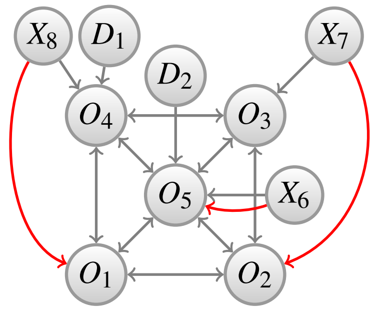

Figure 6.11: The second wheel structure system with five plant agents and two disturbance agents.

and we have

$$\begin{aligned} x_5[k+1] &= x_1[k]+x_2[k]+x_3[k]+x_4[k]+u_6[k] \\ &= v_6[k] & (6.90) \\ x_2[k+2] &= x_1[k+1]+x_3[k+1]+x_5[k+1] \\ &= x_1[k]+4x_2[k]+x_3[k]+x_4[k]+2x_5[k]+u_6[k]+u_7[k] \\ &= v_7[k] & (6.91) \\ x_1[k+2] &= 2x_2[k+1]+x_4[k+1]+x_5[k+1] \\ &= 4x_1[k]+x_2[k]+6x_3[k]+x_4[k]+3x_5[k]+u_6[k]+u_8[k] \\ &= v_8[k]. & (6.92) \end{aligned}$$

Hence, the control laws $u[k]$ are

$$\begin{aligned} u_6[k] &= v_6[k]-x_1[k]-x_2[k]-x_3[k]-x_4[k] & (6.93) \\ u_7[k] &= v_7[k]-x_1[k]-4x_2[k]-x_3[k]-x_4[k]-2x_5[k]-u_6[k] & (6.94) \\ u_8[k] &= v_8[k]-4x_1[k]-x_2[k]-6x_3[k]-x_4[k]-3x_5[k]-u_6[k]. & (6.95) \end{aligned}$$

The controller gain matrices K_1 and K_2 are

$$K_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}, \; K_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

while the control laws for $v[k]$ are

$$\begin{aligned} v_6[k] &= -\bar{x}_1[k]-w_1[k] \\ v_7[k] &= -\bar{x}_2[k]-\bar{x}_3[k]-w_2[k] \\ v_8[k] &= -\bar{x}_4[k]-\bar{x}_5[k]-w_3[k]. \end{aligned}$$

Substitute $v[k]$ into the formula for $u[k]$ in (6.93)–(6.95)

$$
\begin{aligned}
u_6[k] &= v_6[k] - x_1[k] - x_2[k] - x_3[k] - x_4[k] \\
&= -\bar{x}_1[k] - w_1[k] - x_1[k] - x_2[k] - x_3[k] - x_4[k] \\
u_7[k] &= v_7[k] - x_1[k] - 4x_2[k] - x_3[k] - x_4[k] - 2x_5[k] - u_6[k] \\
&= -\bar{x}_2[k] - \bar{x}_3[k] - w_2[k] - x_1[k] - 4x_2[k] \\
&\quad -x_3[k] - x_4[k] - 2x_5[k] - u_6[k] \\
u_8[k] &= v_8[k] - 4x_1[k] - x_2[k] - 6x_3[k] - x_4[k] - 3x_5[k] - u_6[k] \\
&= -\bar{x}_4[k] - \bar{x}_5[k] - w_3[k] - 4x_1[k] - x_2[k] \\
&\quad -6x_3[k] - x_4[k] - 3x_5[k] - u_6[k],
\end{aligned}
$$

where

$$\bar{x}[k] = [v_6[k-1], v_7[k-2], v_7[k-1], v_8[k-2], v_8[k-1]]^T.$$

Finally, set the initial conditions as

$$
\begin{aligned}
x[k] &= [1\ \ 2\ \ 3\ \ 4\ \ 5]^T \\
\bar{x}[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\
w[k] &= [0\ \ 0\ \ 0]^T \\
d_1[k] &= 1 \text{ (for } k \geq 0) \\
d_2[k] &= 1 \text{ (for } k \geq 0).
\end{aligned}
$$

Simulation results in Figure 6.12 show, once again, that we successfully achieve deadbeat disturbance rejection.

6.5 Complete Graph Structure

Consider the system in Figure 6.13, a plant that has a complete graph structure where each plant agent is connected to all the other plant agents. This system has six plant agents and three disturbance agents. The dynamics are

$$x_1[k+1] = x_1[k] + x_2[k] + x_3[k] + x_4[k] + x_5[k] + x_6[k] \tag{6.96}$$

$$x_2[k+1] = x_1[k] + x_3[k] + x_4[k] + x_5[k] + x_6[k] + u_7[k] + d_1[k] \tag{6.97}$$

$$x_3[k+1] = x_1[k] + x_2[k] + x_4[k] + x_5[k] + x_6[k] + u_8[k] \tag{6.98}$$

$$x_4[k+1] = x_1[k] + x_2[k] + x_3[k] + x_5[k] + x_6[k] + u_9[k] + d_2[k] \tag{6.99}$$

$$x_5[k+1] = x_1[k] + x_2[k] + x_3[k] + x_4[k] + x_6[k] + u_{10}[k] + d_3[k] \tag{6.100}$$

$$x_6[k+1] = x_1[k] + x_2[k] + x_3[k] + x_4[k] + x_5[k] + u_{11}[k]. \tag{6.101}$$

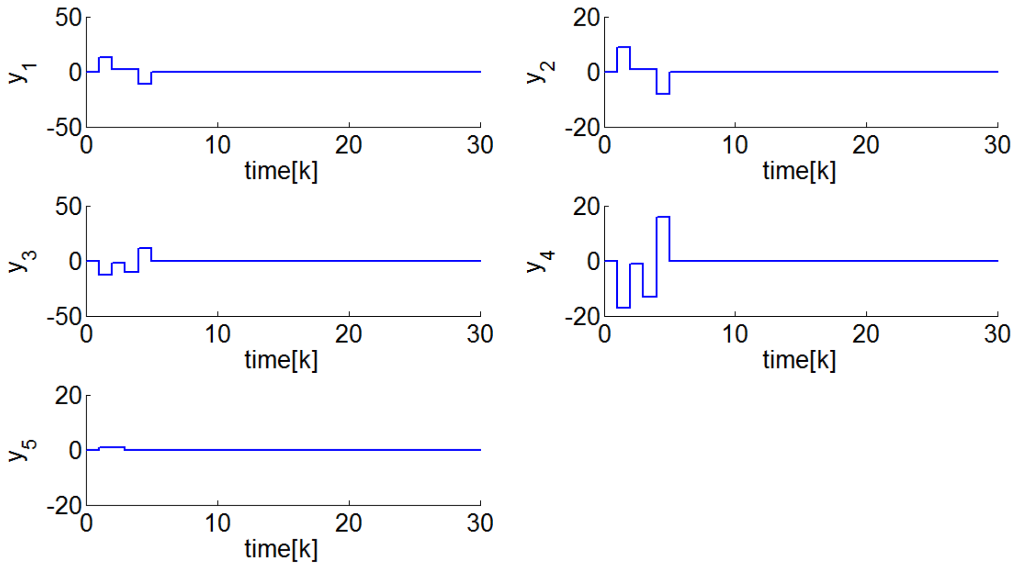

Figure 6.12: Simulation result for the second wheel structure system example.

We assign five control agents and their targets to satisfy Lemma 1, Theorem 3, and Theorem 4. Targeting and growing both succeed.

We determine the propagation time from the control agents to their targets to be

$$\begin{aligned}\Delta(X_7, O_1) &= 2\\ \Delta(X_8, O_3) &= 1\\ \Delta(X_9, O_4) &= 1\\ \Delta(X_{10}, O_5) &= 1\\ \Delta(X_{11}, O_6) &= 1,\end{aligned}$$

and therefore

$$\begin{aligned}x_3[k+1] &= x_1[k]+x_2[k]+x_4[k]+x_5[k]+x_6[k]+u_8[k]\\ &= v_8[k] \qquad (6.102)\\ x_4[k+1] &= x_1[k]+x_2[k]+x_3[k]+x_5[k]+x_6[k]+u_9[k]\\ &= v_9[k] \qquad (6.103)\\ x_5[k+1] &= x_1[k]+x_2[k]+x_3[k]+x_4[k]+x_6[k]+u_{10}[k]\\ &= v_{10}[k] \qquad (6.104)\\ x_6[k+1] &= x_1[k]+x_2[k]+x_3[k]+x_4[k]+x_5[k]+u_{11}[k]\end{aligned}$$

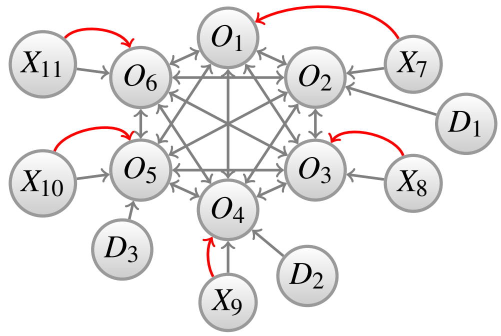

Figure 6.13: The complete graph structure system with six plant agents and three disturbance agents.

$$
\begin{aligned}
&= v_{11}[k] && (6.105)\\
x_1[k+2] &= x_1[k+1]+x_2[k+1]+x_3[k+1]+x_4[k+1]+x_5[k+1]+x_6[k+1] &&\\
&= 6x_1[k]+5x_2[k]+5x_3[k]+5x_4[k]+5x_5[k]+5x_6[k] &&\\
&\quad +u_7[k]+u_8[k]+u_9[k]+u_{10}[k] &&\\
&= v_7[k]. && (6.106)
\end{aligned}
$$

Next, derive the control laws $u[k]$

$$
\begin{aligned}
u_8[k] &= v_8[k]-x_1[k]-x_2[k]-x_4[k]-x_5[k]-x_6[k] && (6.107)\\
u_9[k] &= v_9[k]-x_1[k]-x_2[k]-x_3[k]-x_5[k]-x_6[k] && (6.108)\\
u_{10}[k] &= v_{10}[k]-x_1[k]-x_2[k]-x_3[k]-x_4[k]-x_6[k] && (6.109)\\
u_{11}[k] &= v_{11}[k]-x_1[k]-x_2[k]-x_3[k]-x_4[k]-x_5[k] && (6.110)\\
u_7[k] &= v_7[k]-6x_1[k]-5x_2[k]-5x_3[k]-5x_4[k]-5x_5[k] &&\\
&\quad -5x_6[k]-u_8[k]-u_9[k]-u_{10}[k], && (6.111)
\end{aligned}
$$

and the matrices of K_1 and K_2:

$$
K_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \; K_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.
$$

The control laws $v[k]$ are

$$
\begin{aligned}
v_7[k] &= -\bar{x}_1[k]-\bar{x}_2[k]-w_1[k]\\
v_8[k] &= -\bar{x}_3[k]-w_2[k]
\end{aligned}
$$

$$
\begin{aligned}
v_9[k] &= -\bar{x}_4[k] - w_3[k] \\
v_{10}[k] &= -\bar{x}_5[k] - w_4[k] \\
v_{11}[k] &= -\bar{x}_6[k] - w_5[k].
\end{aligned}
$$

Substitute $v[k]$ into the expressions for $u[k]$ to obtain

$$
\begin{aligned}
u_8[k] &= v_8[k] - x_1[k] - x_2[k] - x_4[k] - x_5[k] - x_6[k] \\
&= -\bar{x}_3[k] - w_2[k] - x_1[k] - x_2[k] - x_4[k] - x_5[k] - x_6[k] \\
u_9[k] &= v_9[k] - x_1[k] - x_2[k] - x_3[k] - x_5[k] - x_6[k] \\
&= -\bar{x}_4[k] - w_3[k] - x_1[k] - x_2[k] - x_3[k] - x_5[k] - x_6[k] \\
u_{10}[k] &= v_{10}[k] - x_1[k] - x_2[k] - x_3[k] - x_4[k] - x_6[k] \\
&= -\bar{x}_5[k] - w_4[k] - x_1[k] - x_2[k] - x_3[k] - x_4[k] - x_6[k] \\
u_{11}[k] &= v_{11}[k] - x_1[k] - x_2[k] - x_3[k] - x_4[k] - x_5[k] \\
&= -\bar{x}_6[k] - w_5[k] - x_1[k] - x_2[k] - x_3[k] - x_4[k] - x_5[k] \\
u_7[k] &= v_7[k] - 6x_1[k] - 5x_2[k] - 5x_3[k] - 5x_4[k] - 5x_5[k] \\
&\quad -5x_6[k] - u_8[k] - u_9[k] - u_{10}[k] \\
&= -\bar{x}_1[k] - \bar{x}_2[k] - w_1[k] - 6x_1[k] - 5x_2[k] - 5x_3[k] \\
&\quad -5x_4[k] - 5x_5[k] - 5x_6[k] - u_8[k] - u_9[k] - u_{10}[k],
\end{aligned}
$$

where

$$
\bar{x}[k] = [v_7[k-2], v_7[k-1], v_8[k-1], v_9[k-1], v_{10}[k-1], v_{11}[k-1]]^T.
$$

For simulation purposes, set the initial conditions and disturbances as

$$
\begin{aligned}
x[k] &= [1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6]^T \\
\bar{x}[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\
w[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\
d_1[k] &= 1 \text{ (for } k \geq 0) \\
d_2[k] &= 1 \text{ (for } k \geq 0).
\end{aligned}
$$

From the simulation results in Figure 6.14, we see that the algorithm works in the complete graph structure system.

6.6 Null Graph Structure

Consider the other extreme case, namely a null graph structure where the plant agents have no connections to one another. An example of a null graph structure system with six plant agents

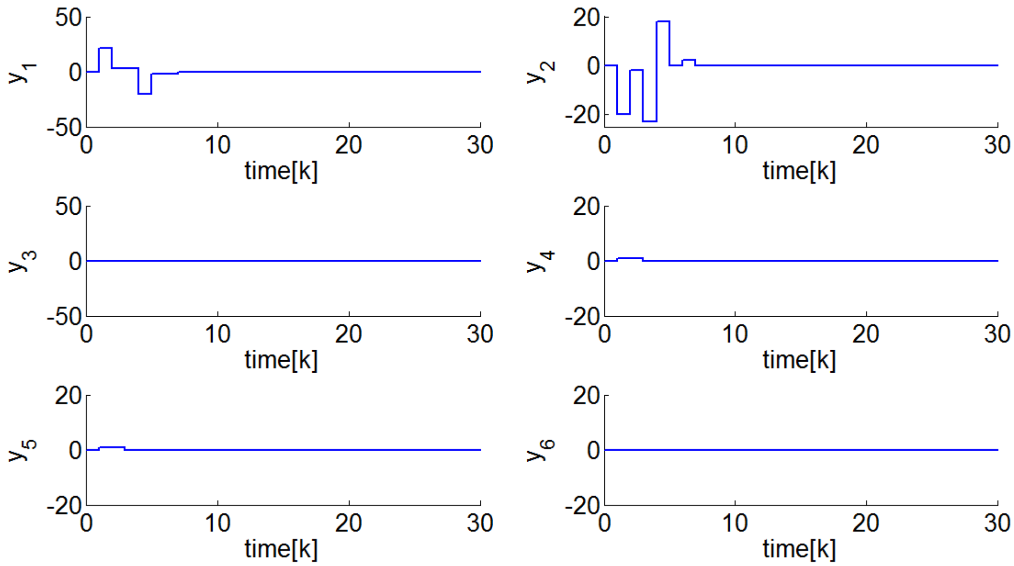

Figure 6.14: Simulation result for the complete graph structure system example.

and six disturbance agents is shown in Figure 6.15. The dynamics are taken to be

$$x_1[k+1] = 2x_1[k] + u_7[k] + d_1[k] \tag{6.112}$$
$$x_2[k+1] = x_2[k] + u_8[k] + d_2[k] \tag{6.113}$$
$$x_3[k+1] = x_3[k] + u_9[k] + d_3[k] \tag{6.114}$$
$$x_4[k+1] = x_4[k] + u_{10}[k] + d_4[k] \tag{6.115}$$
$$x_5[k+1] = x_5[k] + u_{11}[k] + d_5[k] \tag{6.116}$$
$$x_6[k+1] = x_6[k] + u_{12}[k] + d_6[k]. \tag{6.117}$$

As all the propagation times from control agents to respective targets are one, we can directly derive the control laws of $u[k]$ based on (6.112)–(6.117):

$$u_7[k] = v_7[k] - 2x_1[k] \tag{6.118}$$
$$u_8[k] = v_8[k] - x_2[k] \tag{6.119}$$
$$u_9[k] = v_9[k] - x_3[k] \tag{6.120}$$
$$u_{10}[k] = v_{10}[k] - x_4[k] \tag{6.121}$$
$$u_{11}[k] = v_{11}[k] - x_5[k] \tag{6.122}$$
$$u_{12}[k] = v_{12}[k] - x_6[k]. \tag{6.123}$$

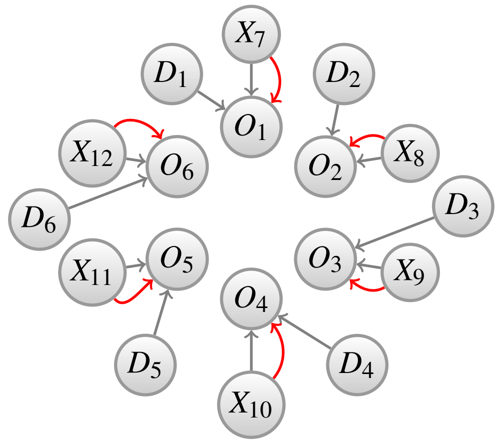

Figure 6.15: The null graph structure system with six plant agents and six disturbance agents.

Find the matrices of K_1 and K_2

$$K_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \; K_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

and the control laws of $v[k]$:

$$\begin{aligned} v_7[k] &= -\bar{x}_1[k] - w_1[k] \\ v_8[k] &= -\bar{x}_2[k] - w_2[k] \\ v_9[k] &= -\bar{x}_3[k] - w_3[k] \\ v_{10}[k] &= -\bar{x}_4[k] - w_4[k] \\ v_{11}[k] &= -\bar{x}_5[k] - w_5[k] \\ v_{12}[k] &= -\bar{x}_6[k] - w_6[k]. \end{aligned}$$

Substitute $v[k]$ into the expressions for $u[k]$

$$\begin{aligned} u_7[k] &= v_7[k] - 2x_1[k] \\ &= -\bar{x}_1[k] - w_1[k] - 2x_1[k] \\ u_8[k] &= v_8[k] - x_2[k] \\ &= -\bar{x}_2[k] - w_2[k] - x_2[k] \\ u_9[k] &= v_9[k] - x_3[k] \end{aligned}$$

$$
\begin{aligned}
&= -\bar{x}_3[k] - w_3[k] - x_3[k] \\
u_{10}[k] &= v_{10}[k] - x_4[k] \\
&= -\bar{x}_4[k] - w_4[k] - x_4[k] \\
u_{11}[k] &= v_{11}[k] - x_5[k] \\
&= -\bar{x}_5[k] - w_5[k] - x_5[k] \\
u_{12}[k] &= v_{12}[k] - x_6[k] \\
&= -\bar{x}_6[k] - w_6[k] - x_6[k],
\end{aligned}
$$

where

$$\bar{x}[k] = [v_7[k-1], v_8[k-1], v_9[k-1], v_{10}[k-1], v_{11}[k-1], v_{12}[k-1]]^T.$$

Set the initial conditions and disturbance signals as

$$
\begin{aligned}
x[k] &= [1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6]^T \\
\bar{x}[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\
w[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\
d_1[k] &= 1 \text{ (for } k \geq 0) \\
d_2[k] &= 1 \text{ (for } k \geq 0).
\end{aligned}
$$

From the simulation results in Figure 6.16, we see that deadbeat disturbance rejection is obtained.

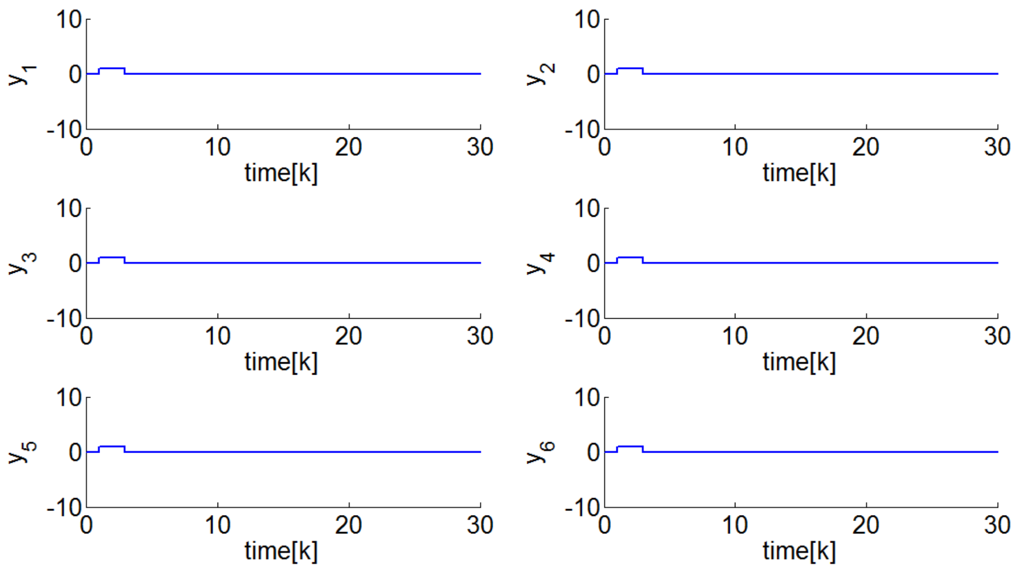

Figure 6.16: Simulation result for the null graph structure system example.

Chapter 7

Deadbeat Disturbance Rejection for a Class of Nonlinear Systems

In the last two chapters, the deadbeat disturbance rejection problem for linear multi-agent systems was discussed. Several examples in Chapter 6 show that the proposed control approach works well in the linear case, which leaves us the question as to whether or not this approach works for nonlinear systems. In this chapter, we focus on a certain class of nonlinear multi-agent system, where we propose a conjecture that the proposed control approach still works.

7.1 Three Constraints

In this section, we introduce three constraints on the nonlinear dynamics given in Chapter 1 (in addition to the existing Assumptions A_1–A_8) such that the control algorithm from Chapters 5-6 still manages to achieve deadbeat disturbance rejection. These results are preliminary, and the main result is presented later as a conjecture.

The three constraints we require the nonlinear dynamics to satisfy are summarized here, then discussed below:

Constraint C_1*:* The dynamics of the system have the form

$$x_i[k+1] = f_i(x[k],y[k]) + B_i^u u[k] + B_i^d d_i[k] \quad (7.1)$$
$$y_i[k] = C_i x_i[k], \quad (7.2)$$

where $f_i(x[k],y[k])$ is a nonlinear combination of the plant agent states $x[k] = [x_1^T, \cdots, x_n^T]^T$ and output $y[k] = [y_1, \cdots, y_n]^T$, and where B_i^u and B_i^d are scalars. (If there is no control agent or disturbance connected to plant agent O_i, for $1 \le i \le n$, then B_i^u or B_i^d is set to zero.)

Constraint C_2: The output of the targets has the form

$$y_i^{tar}[k+\Delta_i] = g_i(x[k], y[k], u_{j\neq i}[k], d_{j\neq i}[k]) + P_i^u u_i[k] + P_i^d d_i[k], \tag{7.3}$$

where $g_i(x[k], y[k], u_{j\neq i}[k], d_{j\neq i}[k])$ is a nonlinear expression and where P_i^u and P_i^d are scalars.

We can re-package (7.3) in various ways, as follows:

$$\begin{aligned} y_i^{tar}[k+\Delta_i] &= g_i(x[k], y[k], u_{j\neq i}[k], d_{j\neq i}[k]) + P_i^u u_i[k] + P_i^d d_i[k] \\ &\triangleq g_i'(x[k], y[k], u_{j\neq i}[k]) + P_i^u u_i[k] + \tilde{d}[k] && (7.4) \\ &\triangleq v_i[k] + \tilde{d}[k] && (7.5) \\ &\triangleq \tilde{v}_i[k]. && (7.6) \end{aligned}$$

The term $\tilde{d}[k]$ in (7.4) is the sum of $P_i^d d_i[k]$ and all the terms in $g_i(x[k], y[k], u_{j\neq i}[k], d_{j\neq i}[k])$ that include disturbances; if all disturbance signals are zero, then $\tilde{d}$ is zero. Note that the term $\tilde{d}[k]$ may be a function of states of plant agents and control signals, and the term $g_i'(x[k], y[k], u_{j\neq i}[k])$ might be zero. We define $v_i[k]$ in (7.5) to be $g_i'(x[k], y[k], u_{j\neq i}[k]) + P_i^u u_i[k]$, and we define $\tilde{v}_i[k]$ in (7.6) to be $v_i[k] + \tilde{d}[k]$.

We now follow the same process used in Chapter 5 for the linear case. From (7.4) and (7.5), we can determine the control law $u_i[k]$ as

$$u_i[k] = (v_i[k] - g_i'(x[k], y[k], u_{j\neq i}[k]))/P_i^u. \tag{7.7}$$

By substituting the control law $u[k]$ back into (7.4), we find an expression for $y_i^{tar}[k+\Delta_i]$ that does not contain any $u[k]$ terms. In fact, we want this expression to also not be affected by any plant-agent state. Thus, we introduce the last constraint:

Constraint C_3: The final expression of the target outputs, in terms of the $v[k]$ signals, has the form

$$y_i^{tar}[k+\Delta_i] = h_i(v_{j\neq i}[k], d_{j\neq i}[k]) + v_i[k] + \beta_i^d d_i[k], \tag{7.8}$$

where $h_i(v_{j\neq i}[k], d_{j\neq i}[k])$ is a nonlinear expression independent of $x[k]$ and where β_i^d is a scalar. Note, in particular, that no plant state terms enter into (7.8).

Constraints C_1–C_3 arose from considering various examples. Take, for instance, the queue plant structure in Figure 7.1. Set $y_i[k] = x_i[k]$, for $1 \le i \le 4$. The propagation time from one agent to any neighbour is one, which implies that

$$\begin{aligned} \Delta(X_5, O_4) &= 2 && (7.9) \\ \Delta(X_6, O_2) &= 2. && (7.10) \end{aligned}$$

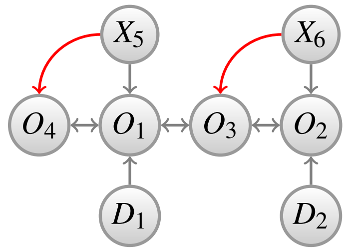

Figure 7.1: A queue structure system with four plant agent and two disturbances.

Let's suppose the queue has the following nonlinear dynamics which does *not* have the form (7.1):

$$\begin{aligned} x_1[k+1] &= (x_1[k]+x_4[k]+2u_5[k]+x_3^2[k])d_1[k] && (7.11)\\ x_2[k+1] &= (-x_3[k]+2u_6[k])d_2[k] && (7.12)\\ x_3[k+1] &= x_1^2[k]+4x_2[k] && (7.13)\\ x_4[k+1] &= 3x_1[k]. && (7.14) \end{aligned}$$

Disregarding the disturbances, we have

$$\begin{aligned} x_4[k+2] &= 3x_1[k+1]\\ &= 3(x_1[k]+x_4[k]+2u_5[k]+x_3^2[k])\\ &= v_5[k]\\ x_3[k+2] &= x_1^2[k+1]+4x_2[k+1]\\ &= (x_1[k]+x_4[k]+2u_5[k]+x_3^2[k])^2+4(-x_3[k]+2u_6[k])\\ &= v_6[k], \end{aligned}$$

which leads to

$$\begin{aligned} u_5[k] &= (v_5[k]-3x_1[k]-3x_4[k]-3x_3^2[k])/6 && (7.15)\\ u_6[k] &= (v_6[k]-(x_1[k]+x_4[k]+2u_5[k]+x_3^2[k])^2+4x_3[k])/8. && (7.16) \end{aligned}$$

Substitute the control laws back to the expressions for the outputs of the targets:

$$\begin{aligned} y_4^{tar}[k+2] &= 3x_1[k+1]\\ &= 3(x_1[k]+x_4[k]+2u_5[k]+x_3^2[k])d_1[k]\\ &= v_5[k]d_1[k] && (7.17)\\ y_3^{tar}[k+2] &= x_1^2[k+1]+4x_2[k+1]\\ &= (x_1[k]+x_4[k]+2u_5[k]+x_3^2[k])^2d_1^2[k]+4(-x_3[k]+2u_6[k])d_2[k]\\ &= (v_5^2[k]d_1^2[k])/9+(v_6[k]-v_5^2[k]/9)d_2^2[k]. && (7.18) \end{aligned}$$

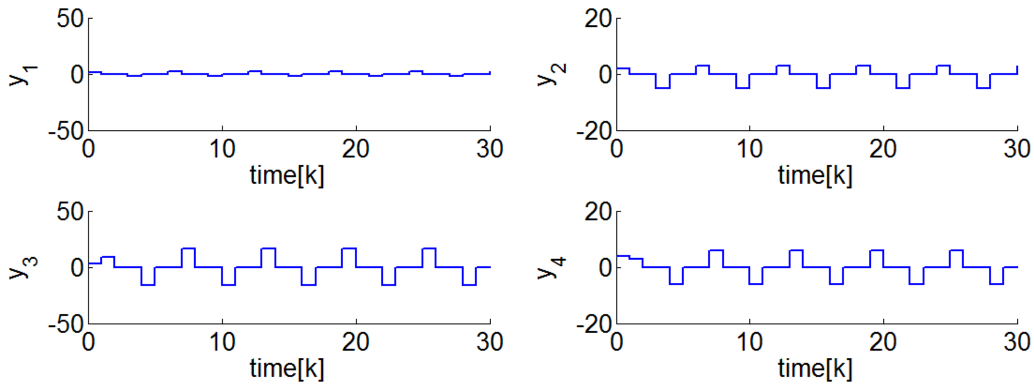

Figure 7.2: Simulation result for the queue plant with dynamics in (7.11)–(7.14).

Although the target output (7.17) and (7.18) have the form of (7.8), the dynamics (7.11) and (7.12) do not have the form of (7.1) due to how the disturbances enter the equations. In this example, if both the disturbance signals are equal to one, deadbeat disturbance rejection is (obviously) achieved. But if we set the disturbances to $d_1[k] = d_2[k] = 2$, then the simulation results in Figure 7.2 show that the algorithm does not reject the disturbance. (Moreover, if the disturbances are zero, Assumption A_6 is violated, but that is not relevant for the point being made here.) Our conclusion is that, to avoid problems of this nature, the safest approach is to constrain the dynamics such that the disturbance enters the equations in an additive manner, which motivated (7.1) in Constraint C_1.

The next example again has the structure in Figure 7.1, but we set the disturbance $d_2[k] = 0$ in this case, i.e., there is no disturbance D_2. The dynamics are as follows:

$$x_1[k+1] = x_1[k] + x_3^2[k] + x_4[k] + 2u_5[k] + d_1[k] \tag{7.19}$$
$$x_2[k+1] = -x_3[k] + 2u_6[k] \tag{7.20}$$
$$x_3[k+1] = x_1[k] + x_2[k]x_3[k] \tag{7.21}$$
$$x_4[k+1] = 3x_1[k]. \tag{7.22}$$

As before, we can determine control law $u_5[k]$:

$$u_5[k] = (v_5[k] - 3x_1[k] - 3x_4[k]x_3^2[k])/6. \tag{7.23}$$

For control law $u_6[k]$, we have

$$\begin{aligned} x_3[k+2] &= x_1[k+1] + x_2[k+1]x_3[k+1] \\ &= x_1[k] + x_4[k] + x_3^2[k] + 2u_5[k] - x_1[k]x_3[k] - x_3^2[k]x_2[k] \end{aligned}$$

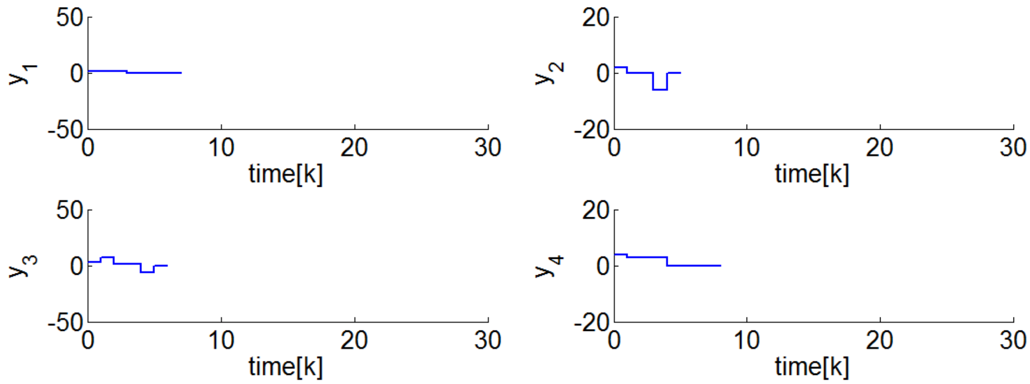

Figure 7.3: Simulation result for the queue plant with dynamics in (7.19)–(7.22). The simulation was aborted when $u_6[k]$ could not be computed.

$$
\begin{aligned}
&+(2x_1[k]+2x_3[k]x_2[k])u_6[k] \qquad (7.24)\\
=\;& v_6[k],
\end{aligned}
$$

which implies

$$u_6[k] \;=\; (v_6[k]-v_5[k]/3+x_1[k]x_3[k]+x_2[k]x_3^2[k])/(2x_1[k]+2x_2[k]x_3[k]). \qquad (7.25)$$

From (7.24), we see that the target output does not have the form of (7.3) due to the state-dependent term $(2x_1[k]+2x_3[k]x_2[k])u_6[k]$. This term causes problems in $u_6[k]$ if the denominator $2x_1[k]+2x_2[k]x_3[k]$ is ever zero (or even very small). Indeed, because of this term the whole algorithm can fail, as shown in Figure 7.3. To avoid such negative effects, we impose the form of (7.3) in Constraint C_2.

Finally, consider a third example (again with the structure in Figure 7.1):

$$
\begin{aligned}
x_1[k+1] &= x_1[k]+x_4[k]+x_3^2[k]+2u_5[k]+d_1[k] \qquad (7.26)\\
x_2[k+1] &= -x_3[k]+u_6[k]+d_2[k] \qquad (7.27)\\
x_3[k+1] &= x_1^2[k]x_3[k]+x_2[k] \qquad (7.28)\\
x_4[k+1] &= 3x_1[k]. \qquad (7.29)
\end{aligned}
$$

To find the control laws, we have

$$
\begin{aligned}
x_4[k+2] &= 3x_1[k+1]\\
&= 3x_1[k]+3x_4[k]+3x_3^2[k]+6u_5[k]\\
&= v_5[k]
\end{aligned}
$$

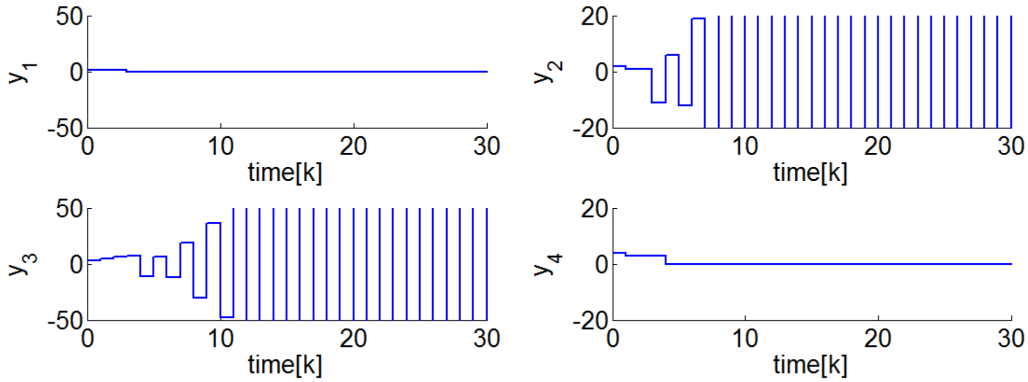

Figure 7.4: Simulation result for the queue plant with dynamics in (7.26)–(7.29).

$$\begin{aligned} x_3[k+2] &= x_1^2[k+1]x_3[k+1]+x_2[k+1] \\ &= (x_1^2[k]x_3[k]+x_2[k])(x_1[k]+x_4[k]+x_3^2[k]+2u_5[k])^2 \\ &\quad -x_3[k]+u_6[k] \\ &= v_6[k]. \end{aligned}$$

Deduct the control laws

$$u_5[k] = (v_5[k]-3x_1[k]-3x_4[k]-3x_3^2[k])/6 \tag{7.30}$$

$$u_6[k] = (v_6[k]+x_3[k]-(x_1^2[k]x_3[k]+x_2[k])(x_1[k]+x_4[k]+x_3^2[k]+2u_5[k])^2), \tag{7.31}$$

and substitute into the target output expressions:

$$y_4^{tar}[k+2] = v_5[k]+3d_1[k] \tag{7.32}$$

$$y_3^{tar}[k+2] = (x_1^2[k]x_3[k]+x_2[k])(2v_5[k]d_1[k]/3+d_1^2[k])+v_6[k]+d_2[k]. \tag{7.33}$$

Note that (7.33) violates the form in (7.8) because of the term of $x_1^2[k]x_3[k]+x_2[k]$. This term makes $y_3^{tar}[\cdot]$ change with the state value of $x_1[k]$, $x_2[k]$ and $x_3[k]$, which can have serious (destabilizing) effects. See Figure 7.4. This example motivates why we impose the form (7.8) in Constraint C_3.

7.2 A Conjecture

Under Constraints C_1–C_3, we suspect that the control strategy from Chapter 5 should work. Formally, we state the following conjecture:

Conjecture: *If a nonlinear multi-agent system with constant disturbances satisfies Constraints C_1–C_3, the control algorithm from Chapter 5 achieves deadbeat disturbance rejection.*

The conjecture has been verified to hold for many nonlinear examples that satisfy Constraints C_1–C_3. Here we present three such examples. For simplicity, the propagation time from any agent to any neighbour in the three examples is one. We also set the output of every plant agent as $y_i[k] = x_i[k]$.

The structure of the first system is the same shown in Figure 7.1. The dynamics satisfy Constraint C_1:

$$\begin{aligned} x_1[k+1] &= x_1[k] + x_4[k] + x_3^2[k] + 2u_5[k] + d_1[k] && (7.34) \\ x_2[k+1] &= -x_3[k] + 2u_6[k] + d_2[k] && (7.35) \\ x_3[k+1] &= x_1^2[k] + 4x_2[k] && (7.36) \\ x_4[k+1] &= 3x_1[k]. && (7.37) \end{aligned}$$

Disregarding the disturbances, we have

$$\begin{aligned} x_4[k+2] &= 3x_1[k+1] \\ &= 3x_1[k] + 3x_4[k] + 6u_5[k] + 3x_3^2[k] \\ &= v_5[k] \\ x_3[k+2] &= x_1^2[k+1] + 4x_2[k+1] \\ &= (x_1[k] + x_4[k] + x_3^2[k] + 2u_5[k])^2 - 4x_3[k] + 8u_6[k] \\ &= v_6[k], \end{aligned}$$

which means

$$\begin{aligned} u_5[k] &= (v_5[k] - 3x_1[k] - 3x_4[k] - 3x_3^2[k])/6 && (7.38) \\ u_6[k] &= (v_6[k] - (x_1[k] + x_4[k] + x_3^2[k] + 2u_5[k])^2 + 4x_3[k])/8. && (7.39) \end{aligned}$$

Substitute the two expressions of $u_5[k]$ and $u_6[k]$ into the output of the targets:

$$\begin{aligned} y_4^{tar}[k+2] &= 3x_1[k] + 3x_4[k] + 6u_5[k] + 3x_3^2[k] + 3d_1[k] && (7.40) \\ &= v_5[k] + 3d_1[k] && (7.41) \\ y_3^{tar}[k+2] &= (x_1[k] + x_4[k] + x_3^2[k] + 2u_5[k] + d_1[k])^2 - 4x_3[k] + 8u_6[k] + 4d_2[k] && (7.42) \\ &= v_6[k] + 2v_5[k]d_1[k]/3 + d_1^2[k] + 4d_2[k]. && (7.43) \end{aligned}$$

We see that (7.40) and (7.42) satisfy the form of (7.3), while (7.41) and (7.43) have the form of (7.8). So we expect that the proposed control algorithm should work for this system. Just like the linear case, form the matrices of K_1 and K_2 based on the fact that the propagation time from both control agents to their targets is two time steps:

$$K_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \; K_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Then the control laws $v[k]$ are

$$v_5[k] = -\bar{x}_1[k] - \bar{x}_2[k] - w_1[k] \tag{7.44}$$
$$v_6[k] = -\bar{x}_3[k] - \bar{x}_4[k] - w_2[k]. \tag{7.45}$$

The final (nonlinear) control laws of $u[k]$ are as the following

$$\begin{aligned}
u_5[k] &= (v_5[k] - 3x_1[k] - 3x_4[k] - 3x_3^2[k])/6 \\
&= (-\bar{x}_1[k] - \bar{x}_2[k] - w_1[k] - 3x_1[k] - 3x_4[k] - 3x_3^2[k])/6 \\
u_6[k] &= (v_6[k] - (x_1[k] + x_4[k] + x_3^2[k] + 2u_5[k])^2 + 4x_3[k])/8 \\
&= (-\bar{x}_3[k] - \bar{x}_4[k] - w_2[k] - (x_1[k] + x_4[k] \\
&\quad + x_3^2[k] + 2u_5[k])^2 + 4x_3[k])/8,
\end{aligned}$$

where

$$\bar{x}[k] = [v_5[k-2], v_5[k-1], v_6[k-2], v_6[k-1]]^T.$$

Set the initial conditions and disturbances as the following:

$$\begin{aligned}
x[k] &= [1\ \ 2\ \ 3\ \ 4]^T \\
\bar{x}[k] &= [0\ \ 0\ \ 0\ \ 0]^T \\
w[k] &= [0\ \ 0]^T \\
d_1[k] &= 1 \text{ (for } k \geq 0) \\
d_2[k] &= 1 \text{ (for } k \geq 0).
\end{aligned}$$

The simulation results in Figure 7.5 show that the algorithm works for this nonlinear system. It achieves deadbeat disturbance rejection in seven time steps.

The second example is the wheel-structured system shown in Figure 7.6. The dynamics of this system have the form of (7.1):

$$x_1[k+1] = x_1^2[k] + x_4[k] + x_2^2[k]x_3[k] + u_6[k] + d_1[k] \tag{7.46}$$
$$x_2[k+1] = x_1[k]x_3[k] + x_4[k]x_5[k] + x_2^2[k] + u_7[k] + d_2[k] \tag{7.47}$$
$$x_3[k+1] = \sin(x_1[k]) + x_2[k] + x_5[k] + u_8[k] \tag{7.48}$$
$$x_4[k+1] = x_1[k] + x_2^2[k] + x_4[k]x_5[k] \tag{7.49}$$
$$x_5[k+1] = x_2^2[k] + x_4[k] + x_3[k]. \tag{7.50}$$

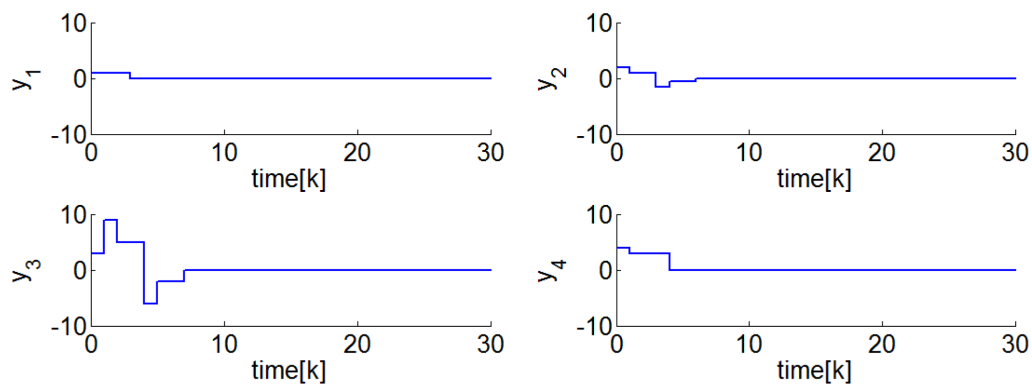

Figure 7.5: Simulation result for the queue plant with dynamics in (7.34)–(7.37).

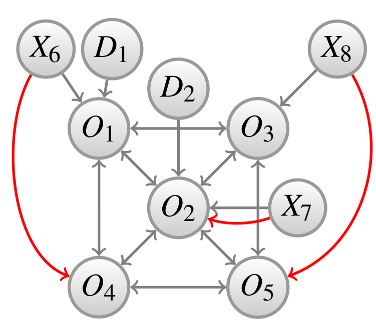

Figure 7.6: The wheel structured nonlinear system with five plant agents and two disturbance agents.

To determine the formula of the control laws of $u[k]$, we evaluate

$$\begin{aligned}
x_2[k+1] &= x_1[k]x_3[k]+x_4[k]x_5[k]+x_2^2[k]+u_7[k] \\
&= v_7[k] \\
x_5[k+2] &= x_2^2[k+1]+x_3[k+1]+x_4[k+1] \\
&= (x_1[k]x_3[k]+x_4[k]x_5[k]+x_2^2[k]+u_7[k])^2+\sin(x_1[k]) \\
&\quad +x_2[k]+x_5[k]+u_8[k]+x_1[k]+x_2^2[k]+x_4[k]x_5[k] \\
&= v_8[k] \\
x_4[k+2] &= x_1[k+1]+x_2^2[k+1]+x_4[k+1]x_5[k+1] \\
&= x_1^2[k]+x_4[k]+x_2^2[k]x_3[k]+u_6[k]+(x_1[k]x_3[k] \\
&\quad +x_4[k]x_5[k]+x_2^2[k]+u_7[k])^2+(x_1[k]+x_2^2[k]+x_4[k]x_5[k]) \\
&\quad (x_2^2[k]+x_4[k]+x_3[k]) \\
&= v_6[k],
\end{aligned}$$

which leads to

$$u_7[k] = v_7[k]-x_1[k]x_3[k]-x_4[k]x_5[k]-x_2^2[k] \tag{7.51}$$

$$\begin{aligned}
u_8[k] &= v_8[k]-(x_1[k]x_3[k]+x_4[k]x_5[k]+x_2^2[k]+u_7[k])^2-\sin(x_1[k]) \\
&\quad -x_2[k]-x_5[k]-x_1[k]-x_2^2[k]-x_4[k]x_5[k]
\end{aligned} \tag{7.52}$$

$$\begin{aligned}
u_6[k] &= v_6[k]-x_1^2[k]-x_4[k]-x_2^2[k]x_3[k]-(x_1[k]x_3[k] \\
&\quad +x_4[k]x_5[k]+x_2^2[k]+u_7[k])^2-(x_1[k]+x_2^2[k]+x_4[k]x_5[k]) \\
&\quad (x_2^2[k]+x_4[k]+x_3[k]).
\end{aligned} \tag{7.53}$$

Substitute the control laws back into the expressions for the target outputs:

$$y_2^{tar}[k+1] = x_1[k]x_3[k]+x_4[k]x_5[k]+x_2^2[k]+u_7[k]+d_2[k] \tag{7.54}$$

$$= v_7[k]+d_2[k] \tag{7.55}$$

$$\begin{aligned}
y_5^{tar}[k+2] &= (x_1[k]x_3[k]+x_4[k]x_5[k]+x_2^2[k]+u_7[k]+d_2[k])^2 \\
&\quad +\sin(x_1[k])+x_2[k]+x_5[k]+u_8[k]+x_1[k]+x_2^2[k]+x_4[k]x_5[k]
\end{aligned} \tag{7.56}$$

$$= 2v_7[k]d_2[k]+d_2^2[k]+v_8[k] \tag{7.57}$$

$$\begin{aligned}
y_4^{tar}[k+2] &= x_1^2[k]+x_4[k]+x_2^2[k]x_3[k]+u_6[k]+d_1[k]+(x_1[k]x_3[k] \\
&\quad +x_4[k]x_5[k]+x_2^2[k]+u_7[k]+d_2[k])^2+(x_1[k]+x_2^2[k] \\
&\quad +x_4[k]x_5[k])(x_2^2[k]+x_4[k]+x_3[k])
\end{aligned} \tag{7.58}$$

$$= v_6[k]+d_1[k]+2v_7[k]d_2[k]+d_2^2[k], \tag{7.59}$$

from which we confirm that the form of the target outputs are consistent with (7.3) and (7.8). As the system satisfies Constraints C_1–C_3, the proposed control algorithm should work for the deadbeat disturbance rejection problem.

As the propagation time from the control agents to their targets are

$$\begin{aligned}\Delta(X_6,O_4) &= 2\\ \Delta(X_7,O_2) &= 1\\ \Delta(X_8,O_5) &= 2,\end{aligned}$$

it is easy to form the matrices K_1 and K_2

$$K_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix},\ K_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

and the control laws of $v[k]$

$$\begin{aligned}v_6[k] &= -\bar{x}_1[k]-\bar{x}_2[k]-w_1[k]\\ v_7[k] &= -\bar{x}_3[k]-w_2[k]\\ v_8[k] &= -\bar{x}_4[k]-\bar{x}_5[k]-w_3[k].\end{aligned}$$

Thus, the final control laws $u[k]$ are

$$\begin{aligned}u_7[k] &= v_7[k]-x_1[k]x_3[k]-x_4[k]x_5[k]-x_2^2[k]\\ &= -\bar{x}_3[k]-w_2[k]-x_1[k]x_3[k]-x_4[k]x_5[k]-x_2^2[k]\\ u_8[k] &= v_8[k]-(x_1[k]x_3[k]+x_4[k]x_5[k]+x_2^2[k]+u_7[k])^2-\sin(x_1[k])\\ &\quad -x_2[k]-x_5[k]-x_1[k]-x_2^2[k]-x_4[k]x_5[k]\\ &= -\bar{x}_4[k]-\bar{x}_5[k]-w_3[k]-(x_1[k]x_3[k]+x_4[k]x_5[k]+x_2^2[k]\\ &\quad +u_7[k])^2-\sin(x_1[k])-x_2[k]-x_5[k]-x_1[k]-x_2^2[k]-x_4[k]x_5[k]\\ u_6[k] &= v_6[k]-x_1^2[k]-x_4[k]-x_2^2[k]x_3[k]-(x_1[k]x_3[k]\\ &\quad +x_4[k]x_5[k]+x_2^2[k]+u_7[k])^2-x_1[k]-x_2^2[k]-x_4[k]x_5[k]\\ &\quad -x_2^2[k]-x_4[k]-x_3[k]\\ &= -\bar{x}_1[k]-\bar{x}_2[k]-w_1[k]-x_1^2[k]-x_4[k]-x_2^2[k]x_3[k]\\ &\quad -(x_1[k]x_3[k]+x_4[k]x_5[k]+x_2^2[k]+u_7[k])^2-(x_1[k]+x_2^2[k]\\ &\quad +x_4[k]x_5[k])(x_2^2[k]+x_4[k]+x_3[k]),\end{aligned}$$

where

$$\bar{x}[k] = [v_6[k-2], v_6[k-1], v_7[k-1], v_8[k-2], v_8[k-1]]^T.$$

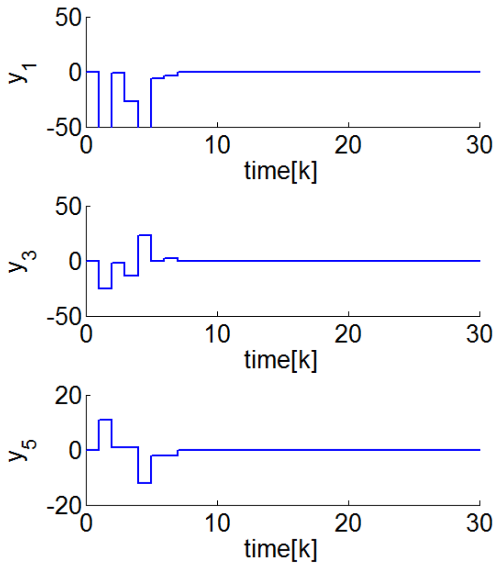

Figure 7.7: Simulation result for the wheel plant with dynamics in (7.46)–(7.49).

Set the initial conditions as the following:

$$
\begin{aligned}
x[k] &= [1\ \ 2\ \ 3\ \ 4\ \ 5]^T \\
\bar{x}[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\
w[k] &= [0\ \ 0\ \ 0]^T \\
d_1[k] &= 1 \text{ (for } k \geq 0) \\
d_2[k] &= 1 \text{ (for } k \geq 0).
\end{aligned}
$$

Deadbeat disturbance rejection is achieved, as shown in the simulation results in Figure 7.7.

The last example is the grid-structured nonlinear system shown in Figure 7.8. The dynamics of this system are

$$x_1[k+1] = x_1^2[k] + x_7[k] + \sin(x_4[k]) + x_4[k]x_5[k] + u_{12}[k] + d_1[k] \tag{7.60}$$
$$x_2[k+1] = (x_4[k] + x_5[k] + x_8[k])^3 + u_{10}[k] + d_2[k] \tag{7.61}$$
$$x_3[k+1] = \sin(x_7[k])\cos(x_5[k]) + x_3[k] + x_6^2[k] + u_{14}[k] + d_3[k] \tag{7.62}$$
$$x_4[k+1] = x_1[k] + x_7[k] + x_5[k] + x_4^2[k] + x_8^2[k] + x_2^2[k] + u_{11}[k] \tag{7.63}$$
$$x_5[k+1] = x_1[k] + x_2[k] + x_3[k] + x_4[k] + x_6[k] + x_7[k] + x_8[k] + x_9[k] \tag{7.64}$$
$$x_6[k+1] = x_3[k] + x_5[k] + x_7[k] + x_8[k] + x_9[k] \tag{7.65}$$
$$x_7[k+1] = x_1[k]x_3[k] + x_4^2[k] + x_5[k] + x_6[k] + u_{13}[k] \tag{7.66}$$

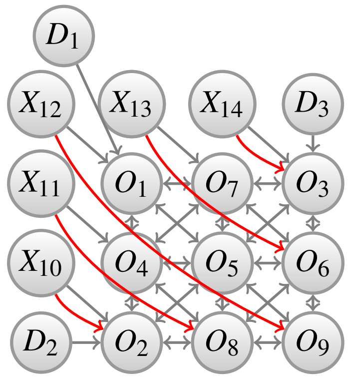

Figure 7.8: The grid structured nonlinear system with nine plant agents and three disturbance agents.

$$x_8[k+1] = x_2[k]+x_4[k]+x_5[k]+x_6[k]+x_9[k] \quad (7.67)$$
$$x_9[k+1] = x_5[k]+x_6[k]+x_8[k]. \quad (7.68)$$

Determine the control laws $u[k]$ by ignoring the disturbances,

$$\begin{aligned}
x_2[k+1] &= (x_4[k]+x_5[k]+x_8[k])^3+u_{10}[k] \\
&= v_{10}[k] \\
x_3[k+1] &= \sin(x_7[k])\cos(x_5[k])+x_3[k]+x_6^2[k]+u_{14}[k] \\
&= v_{14}[k] \\
x_8[k+2] &= x_2[k+1]+x_4[k+1]+x_5[k+1]+x_6[k+1]+x_9[k+1] \\
&= 2x_1[k]+3x_7[k]+2x_3[k]+x_4[k]+x_4^2[k]+3x_5[k]+2x_6[k] \\
&\quad +x_2[k]+x_2^2[k]+3x_8[k]+x_8^2[k]+2x_9[k]+(x_4[k]+x_5[k]+x_8[k])^3 \\
&\quad +u_{10}[k]+u_{11}[k] \\
&= v_{11}[k] \\
x_6[k+2] &= x_3[k+1]+x_5[k+1]+x_7[k+1]+x_8[k+1]+x_9[k+1] \\
&= x_1[k]+x_1[k]x_3[k]+x_7[k]+\sin(x_7[k])\cos(x_5[k])+2x_3[k]+2x_4[k] \\
&\quad +x_4^2[k]+3x_5[k]+4x_6[k]+x_6^2[k]+2x_2[k]+2x_8[k]+2x_9[k] \\
&\quad +u_{14}[k]+u_{13}[k] \\
x_9[k+3] &= x_5[k+2]+x_6[k+2]+x_8[k+2] \\
&= x_1[k+1]+2x_7[k+1]+2x_3[k+1]+2x_4[k+1]+2x_5[k+1]+2x_6[k+1] \\
&\quad +2x_2[k+1]+2x_8[k+1]+3x_9[k+1] \\
&= 4x_1[k]+x_1^2[k]+2x_1[k]x_3[k]+7x_7[k]+2\sin(x_7[k])\cos(x_5[k])+6x_3[k] \\
&\quad +4x_4[k]+4x_4^2[k]+\sin(x_4[k])+x_4[k]x_5[k]+2(x_4[k]+x_5[k]+x_8[k])^3
\end{aligned}$$

$$+11x_5[k]+9x_6[k]+2x_6^2[k]+4x_2[k]+2x_2^2[k]+7x_8[k]+6x_9[k]$$
$$+2u_{13}[k]+2u_{14}[k]+2u_{11}[k]+2u_{10}[k]+u_{12}[k],$$

and therefore deriving

$$u_{10}[k] = v_{10}[k]-(x_4[k]+x_5[k]+x_8[k])^3 \tag{7.69}$$
$$u_{14}[k] = v_{14}[k]-\sin(x_7[k])\cos(x_5[k])-x_3[k]-x_6^2[k] \tag{7.70}$$
$$\begin{aligned} u_{11}[k] = {} & v_{11}[k]-2x_1[k]-3x_7[k]-2x_3[k]-x_4[k]-x_4^2[k] \\ & -3x_5[k]-2x_6[k]-x_2[k]-x_2^2[k]-3x_8[k]-x_8^2[k]-2x_9[k] \\ & -(x_4[k]+x_5[k]+x_8[k])^3-u_{10}[k] \end{aligned} \tag{7.71}$$
$$\begin{aligned} u_{13}[k] = {} & v_{13}[k]-x_1[k]-x_1[k]x_3[k]-x_7[k]-\sin(x_7[k])\cos(x_5[k]) \\ & -2x_3[k]-2x_4[k]-x_4^2[k]-3x_5[k]-4x_6[k]-x_6^2[k] \\ & -x_2[k]-2x_8[k]-2x_9[k]-u_{14}[k] \end{aligned} \tag{7.72}$$
$$\begin{aligned} u_{12}[k] = {} & v_{12}[k]-4x_1[k]-x_1^2[k]-2x_1[k]x_3[k]-7x_7[k]-2\sin(x_7[k])\cos(x_5[k]) \\ & -6x_3[k]-4x_4[k]-4x_4^2[k]-\sin(x_4[k])-x_4[k]x_5[k]-11x_5[k] \\ & -2(x_4[k]+x_5[k]+x_8[k])^3-9x_6[k]-2x_6^2[k]-4x_2[k]-2x_2^2[k] \\ & -7x_8[k]-6x_9[k]-2u_{13}[k]-2u_{14}[k]-2u_{11}[k]-2u_{10}[k]. \end{aligned} \tag{7.73}$$

Substitute the control laws $u[k]$ into the target output expressions:

$$y_2^{tar}[k+1] = (x_4[k]+x_5[k]+x_8[k])^3+u_{10}[k]+d_2[k] \tag{7.74}$$
$$= v_{10}[k]+d_2[k] \tag{7.75}$$
$$y_3^{tar}[k+1] = \sin(x_7[k])\cos(x_5[k])+x_3[k]+x_6^2[k]+u_{14}[k]+d_3[k] \tag{7.76}$$
$$= v_{14}[k]+d_3[k] \tag{7.77}$$
$$\begin{aligned} y_8^{tar}[k+2] = {} & 2x_1[k]+3x_7[k]+2x_3[k]+x_4[k]+x_4^2[k]+3x_5[k]+2x_6[k]+d_2[k] \\ & +x_2[k]+x_2^2[k]+3x_8[k]+x_8^2[k]+2x_9[k]+(x_4[k]+x_5[k]+x_8[k])^3 \end{aligned} \tag{7.78}$$
$$= v_{11}[k]+d_2[k] \tag{7.79}$$
$$\begin{aligned} y_6^{tar}[k+2] = {} & x_1[k]+x_1[k]x_3[k]+x_7[k]+\sin(x_7[k])\cos(x_5[k])+2x_3[k]+2x_4[k] \\ & +x_4^2[k]+3x_5[k]+4x_6[k]+x_6^2[k]+2x_2[k]+2x_8[k]+2x_9[k] \\ & +u_{14}[k]+u_{13}[k]+d_3[k] \end{aligned} \tag{7.80}$$
$$= v_{13}[k]+d_3[k] \tag{7.81}$$
$$\begin{aligned} y_9^{tar}[k+3] = {} & 4x_1[k]+x_1^2[k]+2x_1[k]x_3[k]+7x_7[k]+2\sin(x_7[k])\cos(x_5[k]) \\ & +6x_3[k]+4x_4[k]+4x_4^2[k]+\sin(x_4[k])+x_4[k]x_5[k] \\ & +2(x_4[k]+x_5[k]+x_8[k])^3+11x_5[k]+9x_6[k]+2x_6^2[k] \\ & +4x_2[k]+2x_2^2[k]+7x_8[k]+6x_9[k]+2u_{13}[k]+2u_{14}[k] \\ & +2u_{11}[k]+2u_{10}[k]+u_{12}[k]+d_1[k]+2d_2[k]+2d_3[k] \end{aligned} \tag{7.82}$$
$$= v_{12}[k]+d_1[k]+2d_2[k]+2d_3[k], \tag{7.83}$$

These have the form of (7.3) and (7.8). Hence, the system meets Constraints C_1–C_3. Lastly, we determine the control laws $v[k]$. As the propagation time from control agents to their targets are

$$\begin{aligned} \Delta(X_{10}, O_2) &= 1 \\ \Delta(X_{11}, O_8) &= 2 \\ \Delta(X_{12}, O_9) &= 3 \\ \Delta(X_{13}, O_6) &= 2 \\ \Delta(X_{14}, O_3) &= 1, \end{aligned}$$

the matrices K_1 and K_2 are

$$K_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \ K_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

The control laws $v[k]$ are

$$v_{10}[k] = -\bar{x}_1[k] - w_1[k] \tag{7.84}$$
$$v_{11}[k] = -\bar{x}_2[k] - \bar{x}_3[k] - w_2[k] \tag{7.85}$$
$$v_{12}[k] = -\bar{x}_4[k] - \bar{x}_5[k] - \bar{x}_6[k] - w_3[k] \tag{7.86}$$
$$v_{13}[k] = -\bar{x}_7[k] - \bar{x}_8[k] - w_4[k] \tag{7.87}$$
$$v_{14}[k] = -\bar{x}_9[k] - w_5[k], \tag{7.88}$$

and therefore the final control laws $u[k]$ are

$$\begin{aligned} u_{10}[k] &= -\bar{x}_1[k] - w_1[k] - (x_4[k] + x_5[k] + x_8[k])^3 \\ u_{14}[k] &= -\bar{x}_9[k] - w_5[k] - \sin(x_7[k])\cos(x_5[k]) - x_3[k] - x_6^2[k] \\ u_{11}[k] &= -\bar{x}_2[k] - \bar{x}_3[k] - w_2[k] - 2x_1[k] - 3x_7[k] - 2x_3[k] \\ &\quad -x_4[k] - x_4^2[k] - 3x_5[k] - 2x_6[k] - x_2[k] - x_2^2[k] - 3x_8[k] \\ &\quad -x_8^2[k] - 2x_9[k] - (x_4[k] + x_5[k] + x_8[k])^3 - u_{10}[k] \\ u_{13}[k] &= -\bar{x}_7[k] - \bar{x}_8[k] - w_4[k] - x_1[k] - x_1[k]x_3[k] - x_7[k] \\ &\quad -\sin(x_7[k])\cos(x_5[k]) - 2x_3[k] - 2x_4[k] - x_4^2[k] - 3x_5[k] - 4x_6[k] \\ &\quad -x_6^2[k] - x_2[k] - 2x_8[k] - 2x_9[k] - u_{14}[k] \\ u_{12}[k] &= -\bar{x}_4[k] - \bar{x}_5[k] - \bar{x}_6[k] - w_3[k] - 4x_1[k] - x_1^2[k] - 2x_1[k]x_3[k] \\ &\quad -7x_7[k] - 2\sin(x_7[k])\cos(x_5[k]) - 6x_3[k] - 4x_4[k] - 4x_4^2[k] - \sin(x_4[k]) \\ &\quad -x_4[k]x_5[k] - 11x_5[k] - 2(x_4[k] + x_5[k] + x_8[k])^3 - 9x_6[k] - 2x_6^2[k] - 4x_2[k] \\ &\quad -2x_2^2[k] - 7x_8[k] - 6x_9[k] - 2u_{13}[k] - 2u_{14}[k] - 2u_{11}[k] - 2u_{10}[k], \end{aligned}$$

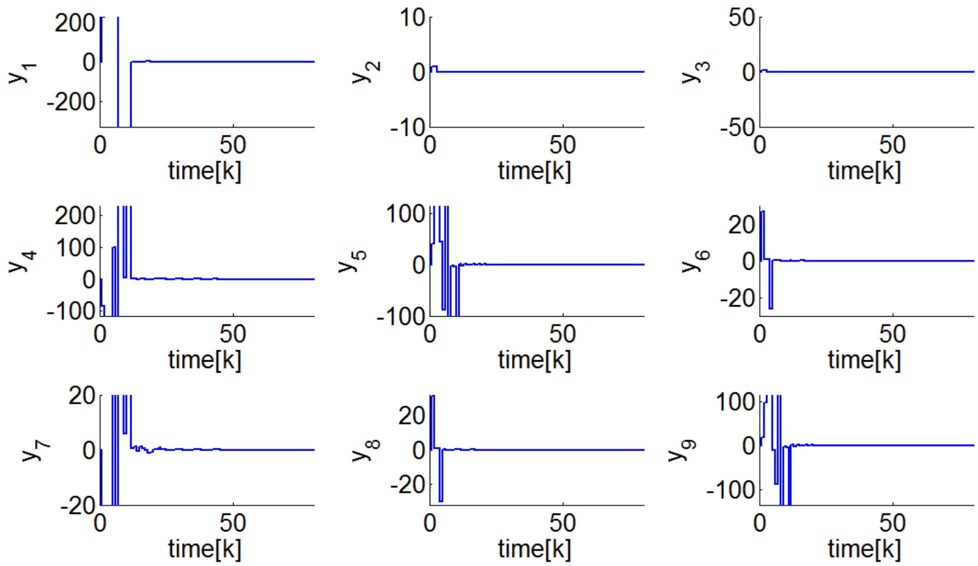

Figure 7.9: Simulation result for the wheel plant with dynamics in (7.60)–(7.68).

where

$$\begin{aligned}\bar{x}[k] = [&v_{10}[k-1], v_{11}[k-2], v_{11}[k-1], v_{12}[k-3], v_{12}[k-2],\\ &v_{12}[k-1], v_{13}[k-2], v_{13}[k-1], v_{14}[k-1]]^T.\end{aligned}$$

Set the initial conditions and disturbances as

$$\begin{aligned} x[k] &= [1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7\ \ 8\ \ 9]^T \\ \bar{x}[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\ w[k] &= [0\ \ 0\ \ 0\ \ 0\ \ 0]^T \\ d_1[k] &= 1\ (\text{for } k \geq 0) \\ d_2[k] &= 1\ (\text{for } k \geq 0). \end{aligned}$$

The simulation results in Figure 7.9 shows that deadbeat disturbance rejection is achieved.

The three examples verify the conjecture that the proposed control algorithm achieves deadbeat disturbance rejection for systems that satisfy Constraints C_1–C_3. A formal proof is left for future research work.

Chapter 8

Summary and Future Directions

This work was initially motivated by a psychological-control problem where some people, playing the role of control agents, attempt to control a crowd of people. Based on previous research that focused on decentralized output regulation of nonlinear discrete-time multi-agent systems, our goal was to achieve deadbeat disturbance rejection for unknown constant disturbances. Two specific problems were investigated: (1) for a given plant with known disturbance agent locations, determine how many control agents are needed, where their locations should be, and which plant agents should be chosen as their targets, in order to successfully obtain deadbeat disturbance rejection; (2) for a given plant with known disturbance agent locations and for a given set of control agent locations with known targets, find, if possible, control laws to successfully obtain deadbeat disturbance rejection.

Our overall approach is to extend research results developed for the regulation problem to the disturbance rejection problem. For Problem 1, a new necessary condition was developed for targeting and growing to succeed, namely that a control agent must be connected to each plant agent to which a disturbance agent is connected. This new condition, in addition to the four necessary conditions previously developed, puts a bound on the minimal number of control agents, and constrains the locations of control agents, needed for successful disturbance rejection. For Problem 2, a new control approach, essentially a multi-loop control scheme with an inner loop based on the regulation approach from the previous research, and an outer loop which is composed of integrators and state-feedback terms, was proposed. The approach was shown to work well for linear systems, and preliminary results show that it works for certain nonlinear systems. The proposed approach is a new way to deal with deadbeat disturbance rejection. A strength of the approach include the simple interpretation of the necessary condition in Lemma 1, making the result useful for large-scale systems. Although not fully studied yet, another likely strength is that the approach often results in less sensing workload per control agent compared to standard centralized methods for non-trivial examples with multiple control agents. On the other hand, the approach is significantly more complicated than the regulation results.

There are a few research directions that warrant further exploration. In the following, we have listed down some ideas for future work:

(a) Better characterize the class of nonlinear multi-agent systems for which the proposed approach works.

(b) Prove the conjecture proposed for the specified class of nonlinear system.

(c) Loosen up the requirements that disturbances be constants and that *deadbeat* disturbance rejection be obtained. In the linear case, it is predicted to be straightforward to loosen both of these requirements, but the nonlinear case is less obvious.

(d) Further investigate the improvement in sensing workload for the proposed technique compared to standard centralized methods.

(e) Accommodate delays and constraints on communications and sensing.

(f) Allow for unknown or time-varying locations of disturbance agents. A fault detection scheme may be useful to determine the locations of disturbance agents.

(g) Extend the work from discrete-time systems to a continuous-time framework or discrete-event framework, both which may be more natural in some applications.

Appendix A: Proof of Lemma 2

Before proving controllability of the augmented system (5.23)–(5.24), let us recall that the augmented plant is

$$\begin{aligned} x^*[k+1] &= A_1^* x^*[k] + B^*(v[k] + \tilde{d}[k]) \\ y^{tar}[k] &= C^* x^*[k], \end{aligned}$$

where

$$\begin{aligned} x^* &= \begin{bmatrix} \bar{x}[k] \\ w[k] \end{bmatrix}, \\ A_1^* &= \begin{bmatrix} \bar{A} & O \\ \bar{C} & I \end{bmatrix}, \\ B^* &= \begin{bmatrix} \bar{B} \\ O \end{bmatrix}, \\ C^* &= \begin{bmatrix} \bar{C} & O \end{bmatrix}, \end{aligned}$$

$$\begin{aligned} \bar{A} &= \text{block diag}\{A'_{\Delta(X_{n+1},T_{n+1})}, \dots, A'_{\Delta(X_{n+m},T_{n+m})}\} \\ \bar{B} &= \text{block diag}\{B'_{\Delta(X_{n+1},T_{n+1})}, \dots, B'_{\Delta(X_{n+m},T_{n+m})}\} \\ \bar{C} &= \text{block diag}\{C'_{\Delta(X_{n+1},T_{n+1})}, \dots, C'_{\Delta(X_{n+m},T_{n+m})}\}, \end{aligned}$$

and

$$A'_r = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \in R^{r \times r},$$

$$
\begin{aligned}
B'_r &= \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \in R^{r\times 1}, \\
C'_r &= \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \in R^{1\times r}, \\
D'_r &= 0.
\end{aligned}
$$

Proving that the augmented system (5.23)–(5.24) is controllable is equivalent to proving that matrix $[A_1^* - \lambda I \; B^*]$ is full rank, for each eigenvalue λ of matrix A_1^* [3]. The following is the proof that the matrix $[A_1^* - \lambda I \; B^*]$ is full rank for each eigenvalue of matrix A_1^*.

First, we show the complete form of the $(n'+m) \times (n'+m)$ matrix of A_1^*, where

$$n' = \Delta_1 + \Delta_2 + \cdots + \Delta_m,$$

as follows:

$$A_1^* = \left[\begin{array}{c:c:c:c|c}
\begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\0&0&0&\cdots&0\end{matrix} & & & & \\ \hdashline
 & \begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\0&0&0&\cdots&0\end{matrix} & & & \\ \hdashline
 & & \ddots & & O \\ \hdashline
 & & & \begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\0&0&0&\cdots&0\end{matrix} & \\ \hline
\underbrace{1\ 0\ \cdots\ \cdots\ 0}_{\Delta_1} & & & & \\ \hdashline
 & \underbrace{1\ 0\ \cdots\ \cdots\ 0}_{\Delta_2} & & & \underbrace{\begin{matrix}1&0&\cdots&0\\0&1&\ddots&\vdots\\\vdots&\ddots&\ddots&0\\0&\cdots&0&1\end{matrix}}_{m} \\ \hdashline
 & & \cdots\cdots & & \\ \hdashline
 & & & \underbrace{1\ 0\ \cdots\ \cdots\ 0}_{\Delta_m} &
\end{array}\right].$$

The empty blocks are all zero matrices.

To find the eigenvalues of A_1^*, we set

$$|\lambda I - A_1^*| = 0.$$

To this end, we have $\lambda I - A_1^* =$

$$
\left[\begin{array}{cccc|c}
\begin{matrix} \lambda & 1 & 0 & \cdots & 0 \\ 0 & \lambda & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda & 1 \\ 0 & 0 & 0 & \cdots & \lambda \end{matrix} & & & & \\
\hdashline
& \begin{matrix} \lambda & 1 & 0 & \cdots & 0 \\ 0 & \lambda & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda & 1 \\ 0 & 0 & 0 & \cdots & \lambda \end{matrix} & & & O \\
\hdashline
& & \ddots & & \\
\hdashline
& & & \begin{matrix} \lambda & 1 & 0 & \cdots & 0 \\ 0 & \lambda & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda & 1 \\ 0 & 0 & 0 & \cdots & \lambda \end{matrix} & \\
\hline
-1\ 0 \cdots \cdots 0 & & & & \\
\cdashline{1-4}
& -1\ 0 \cdots \cdots 0 & & & \begin{matrix} \lambda-1 & 0 & \cdots & 0 \\ 0 & \lambda-1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda-1 \end{matrix} \\
\cdashline{1-4}
& & \cdots\cdots & & \\
\cdashline{1-4}
& & & -1\ 0 \cdots \cdots 0 &
\end{array}\right],
$$

which implies that

$$|\lambda I - A_1^*| = \lambda^{n'} \cdot (\lambda - 1)^m = 0$$

and therefore

$$\lambda = 0 \text{ or } 1.$$

When $\lambda = 0$, we have $[A_1^* - 0 \cdot I \;\; B^*] =$

$$\left[\begin{array}{c:c:c:c|cccc|c:c:c:c}
\begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\0&0&0&\cdots&0\end{matrix} & & & & \multicolumn{4}{c|}{} & \begin{matrix}0\\0\\\vdots\\0\\1\end{matrix} & & & \\ \hdashline
& \begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\0&0&0&\cdots&0\end{matrix} & & & \multicolumn{4}{c|}{O} & & \begin{matrix}0\\0\\\vdots\\0\\1\end{matrix} & & \\ \hdashline
& & \ddots & & \multicolumn{4}{c|}{} & & & \vdots & \\ \hdashline
& & & \begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\0&0&0&\cdots&0\end{matrix} & \multicolumn{4}{c|}{} & & & & \begin{matrix}0\\0\\\vdots\\0\\1\end{matrix} \\ \hline
\mathbf{1}\;0\cdots\cdots 0 & & & & 1&0&\cdots&0 & \multicolumn{4}{c}{} \\ \hdashline
& \mathbf{1}\;0\cdots\cdots 0 & & & 0&1&\ddots&\vdots & \multicolumn{4}{c}{O} \\ \hdashline
& & \cdots\cdots & & \vdots&\ddots&\ddots&0 & \multicolumn{4}{c}{} \\ \hdashline
& & & \mathbf{1}\;0\cdots\cdots 0 & 0&\cdots&0&1 & \multicolumn{4}{c}{}
\end{array}\right].$$

After some elementary row/column operations, the bold 1's can be cancelled. The matrix becomes

$$
\left[\begin{array}{cccc|c|cccc}
\begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & & \\
& \begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & O & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & \\
& & \ddots & & & & & \ddots & \\
& & & \begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} \\
\hline
& & O & & \begin{matrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{matrix} & & O & &
\end{array}\right],
$$

which clearly has $n' + m$ independent rows, so the matrix $[A_1^* - 0 \cdot I \;\; B^*]$ is full rank.

When $\lambda = 1$, we have

$[A_1^* - I \;\; B^*] =$

$$
\left[\begin{array}{cccc|c|cccc}
\begin{matrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 1 \\ 0 & 0 & 0 & \cdots & \mathbf{-1} \end{matrix} & & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & & \\
& \begin{matrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 1 \\ 0 & 0 & 0 & \cdots & \mathbf{-1} \end{matrix} & & & O & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & \\
& & \ddots & & & & & \ddots & \\
& & & \begin{matrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 1 \\ 0 & 0 & 0 & \cdots & \mathbf{-1} \end{matrix} & & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} \\
\hline
1 \;\; 0 \;\; \cdots\cdots \;\; 0 & & & & & & & & \\
& 1 \;\; 0 \;\; \cdots\cdots \;\; 0 & & & O & & O & & \\
& & \cdots\cdots & & & & & & \\
& & & 1 \;\; 0 \;\; \cdots\cdots \;\; 0 & & & & &
\end{array}\right].
$$

Performing elementary row/column operations, the bold -1's can be cancelled, leaving the matrix

$$
\left[\begin{array}{cccc|c|cccc}
\begin{matrix} \mathbf{-1} & 1 & 0 & \cdots & 0 \\ 0 & \mathbf{-1} & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{-1} & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & & \\
& \begin{matrix} \mathbf{-1} & 1 & 0 & \cdots & 0 \\ 0 & \mathbf{-1} & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{-1} & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & O & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & \\
& & \ddots & & & & & \ddots & \\
& & & \begin{matrix} \mathbf{-1} & 1 & 0 & \cdots & 0 \\ 0 & \mathbf{-1} & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{-1} & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} \\
\hline
1 \;\; 0 \;\; \cdots\cdots \;\; 0 & & & & & & & & \\
& 1 \;\; 0 \;\; \cdots\cdots \;\; 0 & & & O & & O & & \\
& & \cdots\cdots & & & & & & \\
& & & 1 \;\; 0 \;\; \cdots\cdots \;\; 0 & & & & &
\end{array}\right].
$$

Further row/column operations yield the matrix

$$\left[\begin{array}{cccc|c|cccc}
\begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & & \\ \hdashline
& \begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & O & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & \\ \hdashline
& & \ddots & & & & & \ddots & \\ \hdashline
& & & \begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} \\ \hline
\begin{matrix} 1 & 0 & \cdots & \cdots & 0 \end{matrix} & & & & & & & & \\ \hdashline
& \begin{matrix} 1 & 0 & \cdots & \cdots & 0 \end{matrix} & & & O & & O & & \\ \hdashline
& & \cdots & & & & & & \\ \hdashline
& & & \begin{matrix} 1 & 0 & \cdots & \cdots & 0 \end{matrix} & & & & &
\end{array}\right],$$

in which again there are $n' + m$ independent rows, so the matrix $[A_1^* - I \;\; B^*]$ is full rank.

In conclusion, the matrix $[A_1^* - \lambda I \;\; B^*]$ is full rank for both $\lambda = 0$ or 1, so the augmented system is controllable.

□

Appendix B: Proof of Lemma 3

Using the matrices A_1^* and B^* from Appendix A and the K matrix from (5.26)–(5.27), we have $A_1^* - B^*K =$

$$
\left[\begin{array}{c:c:c:c|c:c:c:c}
\begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\-1&-1&-1&\cdots&-1\end{matrix} & & & & \begin{matrix}0\\0\\\vdots\\0\\-1\end{matrix} & & & \\ \hdashline
 & \begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\-1&-1&-1&\cdots&-1\end{matrix} & & & & \begin{matrix}0\\0\\\vdots\\0\\-1\end{matrix} & & \\ \hdashline
 & & \ddots & & & & \vdots & \\ \hdashline
 & & & \begin{matrix}0&1&0&\cdots&0\\0&0&1&&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\0&0&\cdots&0&1\\-1&-1&-1&\cdots&-1\end{matrix} & & & & \begin{matrix}0\\0\\\vdots\\0\\-1\end{matrix} \\ \hline
\underbrace{1\quad 0\quad \cdots\cdots\quad 0}_{\Delta_1} & & & & & & & \\ \hdashline
 & \underbrace{1\quad 0\quad \cdots\cdots\quad 0}_{\Delta_2} & & & \multicolumn{4}{c}{\underbrace{\begin{matrix}1&0&\cdots&0\\0&1&\ddots&\vdots\\\vdots&\ddots&\ddots&0\\0&\cdots&0&1\end{matrix}}_{m}} \\ \hdashline
 & & \cdots\cdots & & & & & \\ \hdashline
 & & & \underbrace{1\quad 0\quad \cdots\cdots\quad 0}_{\Delta_m} & & & &
\end{array}\right].
$$

To determine the eigenvalues of matrix $A_1^* - B^*K$, we need to find λ which ensures that

$$|\lambda I - (A_1^* - B^*K)| = 0.$$

The matrix $\lambda\, I - (A_1^* - B^*K)$ is

$$
\left[\begin{array}{c:c:c:c|c:c:c:c}
\begin{smallmatrix} \lambda & -1 & & & \\ & \lambda & -1 & & \\ & & \ddots & \ddots & \\ & & & \lambda & -1 \\ 1 & 1 & 1 & \cdots & \lambda+1 \end{smallmatrix} & & & & \begin{smallmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{smallmatrix} & & & \\
\hdashline
& \begin{smallmatrix} \lambda & -1 & & & \\ & \lambda & -1 & & \\ & & \ddots & \ddots & \\ & & & \lambda & -1 \\ 1 & 1 & 1 & \cdots & \lambda+1 \end{smallmatrix} & & & & \begin{smallmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{smallmatrix} & & \\
\hdashline
& & \ddots & & & & \vdots & \\
\hdashline
& & & \begin{smallmatrix} \lambda & -1 & & & \\ & \lambda & -1 & & \\ & & \ddots & \ddots & \\ & & & \lambda & -1 \\ 1 & 1 & 1 & \cdots & \lambda+1 \end{smallmatrix} & & & & \begin{smallmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{smallmatrix} \\
\hline
\underbrace{\begin{smallmatrix} -1 & 0 & \cdots & \cdots & 0 \end{smallmatrix}}_{\Delta_1} & & & & \multicolumn{4}{c}{\multirow{4}{*}{$\underbrace{\begin{matrix} \lambda-1 & 0 & \cdots & 0 \\ 0 & \lambda-1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda-1 \end{matrix}}_{m}$}} \\
\hdashline
& \underbrace{\begin{smallmatrix} -1 & 0 & \cdots & \cdots & 0 \end{smallmatrix}}_{\Delta_2} & & & & & & \\
\hdashline
& & \cdots & & & & & \\
\hdashline
& & & \underbrace{\begin{smallmatrix} -1 & 0 & \cdots & \cdots & 0 \end{smallmatrix}}_{\Delta_m} & & & &
\end{array}\right].
$$

After some elementary row/column operations, the matrix becomes

$$
\left[\begin{array}{cccc|cccc}
\begin{matrix} \lambda & -1 & & & \\ & \lambda & -1 & & \\ & & \ddots & \ddots & \\ & & & \lambda & -1 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{matrix} & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & & \\
\hline
& \begin{matrix} \lambda & -1 & & & \\ & \lambda & -1 & & \\ & & \ddots & \ddots & \\ & & & \lambda & -1 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{matrix} & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & \\
\hline
& & \ddots & & & & \ddots & \\
\hline
& & & \begin{matrix} \lambda & -1 & & & \\ & \lambda & -1 & & \\ & & \ddots & \ddots & \\ & & & \lambda & -1 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{matrix} & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} \\
\hline
\boldsymbol{-\lambda \; 1-\lambda \; \cdots \; 1-\lambda \; 1-\lambda^2} & & & & \lambda-1 & 0 & \cdots & 0 \\
\hline
& \boldsymbol{-\lambda \; 1-\lambda \; \cdots \; 1-\lambda \; 1-\lambda^2} & & & 0 & \lambda-1 & \ddots & \vdots \\
\hline
& & \cdots & & \vdots & \ddots & \ddots & 0 \\
\hline
& & & \boldsymbol{-\lambda \; 1-\lambda \; \cdots \; 1-\lambda \; 1-\lambda^2} & 0 & \cdots & 0 & \lambda-1
\end{array}\right].
$$

After some additional column operations for the two blocks on the left side, the matrix then becomes

$$
\left[\begin{array}{cccc|cccc}
\begin{array}{ccccc} \mathbf{0} & -1 & & & \\ & \mathbf{0} & -1 & & \\ & & \ddots & \ddots & \\ & & & \mathbf{0} & -1 \\ 0 & 0 & 0 & \cdots & 0 \end{array} & & & & \begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{array} & & & \\
& \begin{array}{ccccc} \mathbf{0} & -1 & & & \\ & \mathbf{0} & -1 & & \\ & & \ddots & \ddots & \\ & & & \mathbf{0} & -1 \\ 0 & 0 & 0 & \cdots & 0 \end{array} & & & & \begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{array} & & \\
& & \ddots & & & & \vdots & \\
& & & \begin{array}{ccccc} \mathbf{0} & -1 & & & \\ & \mathbf{0} & -1 & & \\ & & \ddots & \ddots & \\ & & & \mathbf{0} & -1 \\ 0 & 0 & 0 & \cdots & 0 \end{array} & & & & \begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{array} \\
\hline
-\lambda^{\Delta_1+1} \;\; 1-\lambda^{\Delta_1} \;\; \cdots \;\; 1-\lambda^3 \;\; 1-\lambda^2 & & & & \lambda-1 & 0 & \cdots & 0 \\
& -\lambda^{\Delta_2+1} \;\; 1-\lambda^{\Delta_2} \;\; \cdots \;\; 1-\lambda^3 \;\; 1-\lambda^2 & & & 0 & \lambda-1 & \ddots & \vdots \\
& & \cdots & & \vdots & \ddots & \ddots & 0 \\
& & & -\lambda^{\Delta_m+1} \;\; 1-\lambda^{\Delta_m} \;\; \cdots \;\; 1-\lambda^3 \;\; 1-\lambda^2 & 0 & \cdots & 0 & \lambda-1
\end{array}\right].
$$

Finally, after some elementary row operations, we have

$$
\left[\begin{array}{cccc|cccc}
\begin{matrix} 0 & -1 & & & \\ & 0 & -1 & & \\ & & \ddots & \ddots & \\ & & & 0 & -1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & & \\
& \begin{matrix} 0 & -1 & & & \\ & 0 & -1 & & \\ & & \ddots & \ddots & \\ & & & 0 & -1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} & & \\
& & \ddots & & & & \ddots & \\
& & & \begin{matrix} 0 & -1 & & & \\ & 0 & -1 & & \\ & & \ddots & \ddots & \\ & & & 0 & -1 \\ 0 & 0 & 0 & \cdots & 0 \end{matrix} & & & & \begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{matrix} \\
\hline
-\lambda^{\Delta_1+1}\ \mathbf{0} \cdots \mathbf{0}\ \mathbf{0} & & & & \mathbf{0} & 0 & \cdots & 0 \\
& -\lambda^{\Delta_2+1}\ \mathbf{0} \cdots \mathbf{0}\ \mathbf{0} & & & 0 & \mathbf{0} & \ddots & \vdots \\
& & \cdots & & \vdots & \ddots & \ddots & 0 \\
& & & -\lambda^{\Delta_m+1}\ \mathbf{0} \cdots \mathbf{0}\ \mathbf{0} & 0 & \cdots & 0 & \mathbf{0}
\end{array}\right].
$$

It is obvious that the matrix is full rank except when $\lambda = 0$. This implies that all eigenvalues of matrix $A_1^* - B^*K$ are zero. Hence, the choice of K in (5.26)–(5.27) ensures that the closed-loop eigenvalues of the augmented system 5.23–5.24 are all at the origin.

Appendix C: Matlab Code for the System in Figure 5.4

```
x = zeros(31,6);
x(1,:) = [1 2 3 4 5 6];
%x_1, x_2, x_3, x_4, x_5, x_6
x_bar = zeros(31,6);
x_bar(1,:) = [0 0 0 0 0 0];
z = zeros(31,3);  %The integrator states
v = zeros(31,3); %v_7, v_8, v_9
u = zeros(31,3); %u_7, u_8, u_9
y = zeros(31,3); %y_7, y_8, y_9
xy = zeros(31,6); %The outputs of each plant agents
yz = zeros(31,3); %w[k]
D1=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];
D2=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];

K1 = [1 1 0 0 0 0; 0 0 1 1 0 0; 0 0 0 0 1 1];
K2 = [1 0 0; 0 1 0; 0 0 1];
K = [K1 K2];

for k = 1:30
    if k<3
        x_bar(k,:) = [0 0 0 0 0 0];
    else
        x_bar(k,1) = v(k-2,1);
        x_bar(k,2) = v(k-1,1);
```

```
    x_bar(k,3) = v(k-2,2);
    x_bar(k,4) = v(k-1,2);
    x_bar(k,5) = v(k-2,3);
    x_bar(k,6) = v(k-1,3);
end
if k==1
    z(1,:)=[0 0 0];
    yz(1,:)=[0 0 0];
else
    z(k,:) = z(k-1,:) + y(k-1,:);
    yz(k,:) = z(k,:);
end

  v(k,:) = (-K1*x_bar(k,:)'-K2*yz(k,:)')';

 u(k,1) = v(k,1)-2*x(k,1)-4*x(k,3)-3*x(k,4)-x(k,5)-x(k,6); %u7
 u(k,2) = (v(k,2)-3*x(k,1)-5*x(k,2)-x(k,4)-4*x(k,5)-x(k,6))/2;  %u8
 u(k,3) = (v(k,3)-12*x(k,2)-5*x(k,3)-x(k,4)-x(k,5)-5*x(k,6))/3;  %u9

 x(k+1,1) = x(k,1)+2*x(k,3)+x(k,4)+u(k,1)+D1(1,k);
 xy(k+1,1) = x(k+1,1);

 x(k+1,2) = 3*x(k,2)+x(k,3)+x(k,6)+u(k,3)+D2(1,k);
 xy(k+1,2) = x(k+1,2);

 x(k+1,3) = x(k,1)+x(k,2)+x(k,5)+u(k,2);
 xy(k+1,3) = x(k+1,3);

 x(k+1,4) = x(k,1)+x(k,4)+x(k,5);
 xy(k+1,4) = x(k+1,4);

 x(k+1,5) = 2*x(k,3)+x(k,4)+x(k,6);
 xy(k+1,5) = x(k+1,5);

 x(k+1,6) = 3*x(k,2)+x(k,5)+x(k,6);
 xy(k+1,6) = x(k+1,6);


 y(k+1,:) = [xy(k+1,4) xy(k+1,5) xy(k+1,6)];
```

```
end

T = 0:1:30;
figure(1);
subplot(3,2,1),hold on, axis([0 30 -50 50]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_1'),
set(h,'fontsize',20),h=stairs(T,xy(:,1)),set(h,'linewidth',2.0);
subplot(3,2,2),hold on, axis([0 30 -20 20]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_2'),
set(h,'fontsize',20),h=stairs(T,xy(:,2)),set(h,'linewidth',2.0);
subplot(3,2,3),hold on, axis([0 30 -50 50]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_3'),
set(h,'fontsize',20),h=stairs(T,xy(:,3)),set(h,'linewidth',2.0);
subplot(3,2,4),hold on, axis([0 30 -20 20]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_4'),
set(h,'fontsize',20),h=stairs(T,xy(:,4)),set(h,'linewidth',2.0);
subplot(3,2,5),hold on, axis([0 30 -20 20]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_5'),
set(h,'fontsize',20),h=stairs(T,xy(:,5)),set(h,'linewidth',2.0);
subplot(3,2,6),hold on, axis([0 30 -20 20]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_6'),
set(h,'fontsize',20),h=stairs(T,xy(:,6)),set(h,'linewidth',2.0);
```

Appendix D: Matlab Code for the System in Figure 5.10

```
x = zeros(31,8);
x(1,:) = [1 2 3 4 5 6 7 8];
%x_1, x_{21}, x_{22}, x_{31}, x_{32}, x_4, x_5, x_6
x_bar = zeros(31,6);
x_bar(1,:) = [0 0 0 0 0 0];
z = zeros(31,3);  %The integrator states
v = zeros(31,3); %v_7, v_8, v_9
u = zeros(31,3); %u_7, u_8, u_9
y = zeros(31,3); %y_7, y_8, y_9
xy = zeros(31,6); %The outputs of each plant agents
yz = zeros(31,3); %w[k]
D1=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];
D2=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];

K1 = [1 1 0 0 0 0; 0 0 1 1 0 0; 0 0 0 0 1 1];
K2 = [1 0 0; 0 1 0; 0 0 1];
K = [K1 K2];


for k = 1:30
    if k<3
        x_bar(k,:) = [0 0 0 0 0 0];
    else
        x_bar(k,1) = v(k-2,1);
```

```
    x_bar(k,2) = v(k-1,1);
    x_bar(k,3) = v(k-2,2);
    x_bar(k,4) = v(k-1,2);
    x_bar(k,5) = v(k-2,3);
    x_bar(k,6) = v(k-1,3);
end
if k==1
    z(1,:)=[0 0 0];
    yz(1,:)=[0 0 0];
else
    z(k,:) = z(k-1,:) + y(k-1,:);
    yz(k,:) = z(k,:);
end

  v(k,:) = (-K1*x_bar(k,:)'-K2*yz(k,:)')';

  u(k,1) = v(k,1)-v(k,2)-3*x(k,4)+x(k,5)-5*x(k,6)-2*x(k,1)
  +2*x(k,7)+2*x(k,8)+4*x(k,2)-4*x(k,3); %u1
  u(k,2) = v(k,2)-x(k,6)-x(k,1)-4*x(k,7)-2*x(k,8)
  -4*x(k,2)+4*x(k,3);  %u3
  u(k,3) = (v(k,3)-x(k,1)-x(k,7)-3*x(k,8)
  +2*x(k,2)+4*x(k,3))/2;  %u6

  x(k+1,1) = x(k,6)+x(k,1)+2*x(k,7)+u(k,2)+D1(1,k);
  xy(k+1,1) = x(k+1,1);

  x(k+1,2) = -x(k,3)+u(k,3);
  x(k+1,3) = 2*x(k,2)+D2(1,k);
  xy(k+1,2) = x(k+1,2)-x(k+1,3);

  x(k+1,4) = -2*x(k,5)+u(k,1);
  x(k+1,5) = x(k,4)-x(k,5)+x(k,6);
  xy(k+1,3) = x(k+1,4)+x(k+1,5);

  x(k+1,6) = xy(k,3)+2*x(k,6)+ x(k,1);
  xy(k+1,4) = x(k+1,6);

  x(k+1,7) = x(k,1)+2*x(k,8);
  xy(k+1,5) = x(k+1,7);
```

```
    x(k+1,8) = x(k,7)+x(k,8)+2*xy(k,2);
    xy(k+1,6) = x(k+1,8);


    y(k+1,:) = [xy(k+1,4) xy(k+1,5) xy(k+1,6)];

end

T = 0:1:30;
figure(1);
subplot(3,2,1),hold on, axis([0 30 -50 50]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_1'),
set(h,'fontsize',20),h=stairs(T,xy(:,1)),set(h,'linewidth',2.0);
subplot(3,2,2),hold on, axis([0 30 -20 20]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_2'),
set(h,'fontsize',20),h=stairs(T,xy(:,2)),set(h,'linewidth',2.0);
subplot(3,2,3),hold on, axis([0 30 -50 50]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_3'),
set(h,'fontsize',20),h=stairs(T,xy(:,3)),set(h,'linewidth',2.0);
subplot(3,2,4),hold on, axis([0 30 -20 20]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_4'),
set(h,'fontsize',20),h=stairs(T,xy(:,4)),set(h,'linewidth',2.0);
subplot(3,2,5),hold on, axis([0 30 -20 20]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_5'),
set(h,'fontsize',20),h=stairs(T,xy(:,5)),set(h,'linewidth',2.0);
subplot(3,2,6),hold on, axis([0 30 -20 20]),
h=gca,set(h,'fontsize',20),h=xlabel('time[k]'),
set(h,'fontsize',20),h=ylabel('y_6'),
set(h,'fontsize',20),h=stairs(T,xy(:,6)),set(h,'linewidth',2.0);
```

Bibliography

[1] M. Aoki. On feedback stabilizability of decentralized dynamic systems. *Automatica*, 1972. 2

[2] K. Bergey, Kevin Spieser, and D. E. Davison. The psychological dynamics of students in a classroom: Modeling and control strategies based on suggestibility theory. In *Conference on Control Applications*, Singapore, October 2007. IEEE.

[3] Chi-Tsong Chen. *Linear System Theory and Design*. Oxford University Press, 2009. 89

[4] J. Corfmat and A. Morse. Decentralized control of linear multivariable systems. *Automatica*, 12, 1976. 2

[5] J. Shamma (editor). *Cooperative control of distributed multi-agent systems*. 2007. 2

[6] J. Feddema, C. Lewis, and D. Schoenwald. Decentralized control of cooperative robotic vehicles: Theory and application. *IEEE Transactions on Robotics and Automation*, 18(5), 2002. 2

[7] Z. Gao. Active disturbance rejection control: a paradigm shift in feedback control system design. *American Control Conference*, 2006. 2

[8] T. Gustavi, D. Dimarogonas, M. Egerstedt, and X. Hu. Sufficient conditions for connectivity maintenance and rendezvous in leader-follower networks. *Automatica*, 46(1), 2010. 2

[9] J.B. Hoagg. Deadbeat internal model control for command following and disturbance rejection in discrete-time systems. *American Control Conference*, 2006. 2

[10] W. S. Levine. *The Control Handbook*. CRC Press, 1996.

[11] R. Olfati-Saber, J. Fax, and R. Murray. Consensus and cooperation in networked multiagent systems. *Proceedings of the IEEE*, 95(1), 2007. 2

[12] Richard M. Phelan. *Automatic Control Systems*. Cornell University Press, 1977.

[13] Z. Qu. *Cooperative Control of Dynamical Systems: Applications to Autonomous Vehicles*. Springer, 2009. 2

[14] W. Ren and R. Beard. Distributed consensus in multi-vehicle cooperative control: Theory and applications. 2008. 2

[15] Nasim A. Shams. Decentralized regulation of nonlinear discrete-time multi-agent systems. MASc thesis, University of Waterloo, 2011. 3, 11, 12, 13, 15, 17

[16] Nasim A. Shams and Daniel E. Davison. A graph-theoretic condition necessary for the stabilization of crowds using cooperating control agents. In *IEEE Canadian Conference on Electrical and Computer Engineering*, Calgary, Canada, May 2010. 1

[17] Nasim A. Shams and Daniel E. Davison. A co-operative control approach to the regulation of nonlinear discrete-time structured systems. In *IEEE Canadian Conference on Electrical and Computer Engineering*, Niagara Falls, Canada, May 2011. vii, 1, 6, 7, 8, 11, 14, 15, 16, 17, 21, 22

[18] D.D. Siljak. *Decentralized Control of Complex Systems*. Academic Press, 1991. 2

[19] K. Spieser and D. E. Davison. Multi-agent stabilization of the psychological dynamics of one-dimensional crowds. *Automatica*, 45, 2009. 1

[20] Kevin Spieser. Stabilizing the psychological dynamics of people in a crowd. MASc thesis, University of Waterloo, 2008. 1

[21] Kevin Spieser and D. E. Davison. Stabilizing the psychological dynamics of people in a queue. In *Proceedings of the American Control Conference*, Seattle, WA, June 2008. AACC.

[22] S. Sundaram and C. N. Hadjicostis. Distributed function calculation via linear iterative strategies in the presence of malicious agents. *IEEE Transactions on Automatic Control*, 56(7), 2011. 2

[23] S. Wang and E.J. Davison. On the stabilization of decentralized control systems. *IEEE Transactions on Automatic Control*, AC-18, 1973. 2

[24] Xiaoli Wang. A distributed control approach to a robust output regulation problem for multi-agent linear systems. *IEEE Transactions on Automatic Control*, 55(12), 2010. 2

[25] D. Yua, J. Lam, and Z. Wang. Persistent disturbance rejection via state feedback for networked control systems. *Chaos, Solitons And Fractals*, 40(1), 2009. 2