Modeling and Control of Cyber-Physical Systems

Project II - Report

September 21, 2023

Carena Simone, s309521
Carmone Francesco Paolo, s308126
Mazzucco Ludovica, s315093
Pulinas Federico, s319881

Contents

Introduction **2**
Brief description of the problem 2
Network Topologies 2
Distribuited Algorithms and Parameters choice 3

1 Distributed regulator based on a distributed neighborhood observer **4**
Results obtained using different references 4

2 Distributed regulator based on local observers **8**
Managing noise through matrix F 8
Effect of noise in the two scenarios 9
Results obtained using different references 9

List of Figures

1 Topology 1 2
2 Topology 2 2
3 Topology 3 2
4 Emphatizing R 3
5 Equally Weighted 3
6 Emphatizing Q 3
7 Distribuited Observer, Q1 R1, Topology 2 6
8 Distribuited Observer, Q1 R1, Topology 1 7
9 Distribuited Observer, Q1 R1, Topology 3 7
10 Step noise 9
11 Random noise 9
12 No noise 9
13 Cooperative observer 9
14 Local observer 9

List of Tables

1 Convergence Times with a constant reference (Distributed Observer) 5
2 Convergence Times with a sinusoidal reference (Distributed Observer) 5
3 Different Convergence Times with a Ramp Reference (Distributed Observer) . . . 6
4 Average Performance for each topology (Distributed Observer) 6
5 Different convergence times with constant reference (Local Observer) 10
6 Different convergence times with sinusoidal reference (Local Observer) 10
7 Different convergence times with a ramp reference (Local Observer) 11
8 Average performance for each topology (Local Observer) 11

Introduction

Brief description of the problem

The project involves the study of a Cyber Physical System made up of seven magnetic levitators, that can be modeled as nonlinear systems where the motion of the suspended mass is governed mainly by gravitational and magnetic forces. The main equation of the system are:
Actuator:

$$I_{\mathrm{m}} = K_{\mathrm{ampli}}(V_{\mathrm{con}} + V_{\mathrm{offsetc}})$$

Electromagnet:

$$F_{\mathrm{m}}(t) = K_{\mathrm{m}}\frac{I_{\mathrm{m}}(t)^2}{z(t)^2}$$

Ball dynamics:

$$m_{\mathrm{s}}\ddot{z}(t) = m_{\mathrm{s}}g - F_{\mathrm{m}}(t)$$

Position transducer:

$$V_{\mathrm{trasdp}}(t) = K_{\mathrm{trasdp}}[z(t) - z_0]$$

The cyber-physical network has to be modeled as a multi-agent system where just one of the nodes acts as *leader* node (S_0), while the remaining ones act as *follower* nodes (S_i). Each agent is described by the following state-space model, obtained by linearizing the previous equations:

$$\begin{cases} \dot{x}_{\mathrm{i}} = Ax_{\mathrm{i}} + Bu_{\mathrm{i}} \\ y_{\mathrm{i}} = Cx_{\mathrm{i}} \end{cases} \qquad \text{(Eq. 1)}$$

where:

$$A = \begin{bmatrix} 0 & 1 \\ 880.87 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ -9.9453 \end{bmatrix}, \quad C = \begin{bmatrix} 708.27 & 0 \end{bmatrix}$$

The aim of this project is to develop a distributed control protocol that specifically examines the impact of employing a cooperative observer or local observer on each agent. The objective is to enable the followers to track the leader's output. The subsequent discussion will explore how the behavior of the entire system is influenced by various factors such as the choice of the network structure, the type of reference signal the leader node imposes (constant, ramp, sinusoidal signal), the effects resulting from the selection of parameters R, Q and c, the effect of possible noise affecting the output measurements.

Network Topologies

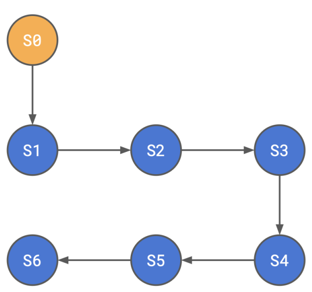

Figure 1: Topology 1

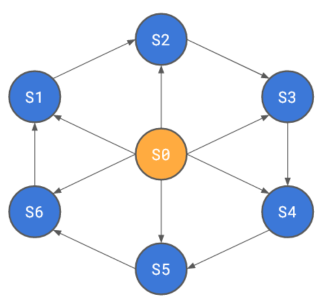

Figure 2: Topology 2

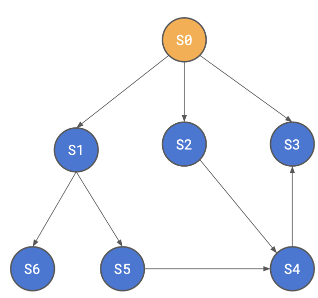

Figure 3: Topology 3

S_0 is the leader node, while S_i are the follower nodes ($i = 1 \dots 6$). Each link has a unitary weight.

A topology is identified by

- **Ad**, adjacency matrix. A matrix that describes the communication links between the nodes. Each component Ad_{ij} indicates a link directed from the node j to the node i.
- **G**, pinning matrix. A diagonal matrix whose $i-th$ diagonal value describe a communication link between the leader node and the $i-th$ follower node.
- **D**, Degree matrix. A diagonal matrix whose $i-th$ diagonal value indicates the number of incoming links for the node i. It can be computed as

$$D_{ii} = \sum_{\forall j \neq i} Ad_{ij}$$

The modeling of such topologies must take into account the reality that the leader node can communicate with a limited number of follower nodes, and typically, these nodes can only communicate with their own neighbors.

Distribuited Algorithms and Parameters choice

The previously defined communication network is used to distribuite information among all the nodes, and work on that data to better chase the reference signal. The state of each magnetic levitator use isn't directly measurable, since the output matrix C in Eq. 1 is not the identity matrix. To solve that issue, an observer has to be employed. Moreover, the protocols we are going to analyze share some parameters to calculate the input u, such as the coupling gain c, and the matrices Q, R. We decided to fix the coupling c, used to calculate the input u of each follower node, as varying it produced negligeable changes. By choosing c as

$$c \geq \frac{1}{2 \min_{i \in \mathcal{N}} \Re(\lambda_i)}$$

the Algebraic Riccati Equation has a unique positive definite solution P.

$$AP + PA^T + Q - PC^T R^{-1} CP = 0 \qquad \text{(Eq. 2)}$$

which results in nullifying the global disagreement error, i.e. $\lim_{t \to \infty} \delta(t) = 0$

The choices of the matrices Q and the scalar R are more interesting as they serve as the weights in this cost function minimization problem,

$$u(t) = \arg\min_u \frac{1}{2} \int_0^\infty x^T(t) Q x(t) + u^T(t) R u(t) dt \qquad \text{(Eq. 3)}$$

In this context the term *cost* refers to the energy associated to the state variable x and command effort u. Larger weights imply an higher cost of such variable, and thus something to minimize. When choosing Q and R what matters the most is their *ratio*[0]. As a general result one can increase Q to obtain a faster convergence speed, and increase R to reduce the command activity. The previous statement can be experimentally verified in the following figures:

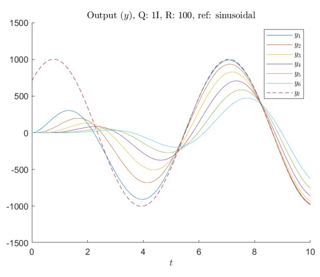

Figure 4: Emphatizing R

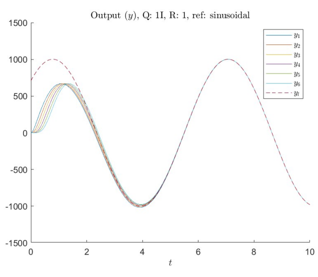

Figure 5: Equally Weighted

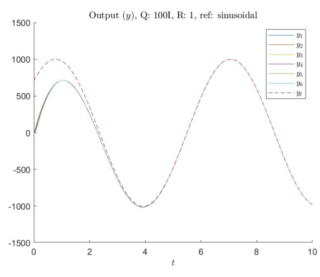

Figure 6: Emphatizing Q

[0] In our scenario Q is a matrix and R is a scalar, and since they're two different mathematical objects one can't define a ratio. Instead, we consider the infinity norm of Q, and define the ratio as $\doteq \|(Q)\|_\infty / R$

1 Distributed regulator based on a distributed neighborhood observer

We assume that the leader and followers are identical, meaning they share the same dynamic matrix A and output matrix C as described in equation Eq. 1. However the leader does not receive any input u since its behavior is not influenced by others, but it can still chase a reference signal by applying a local feedback linear controller with an appropriate gain, denoted as K_{reg}. This modification ensures that the closed-loop dynamics are stable and follows a desired reference. From now on, the dynamics matrix of all the agents will be rewritten as $A_{\text{new}} \doteq A - BK_{reg}$. Finally, we can adjust the eigenvalues of A_{new} and the initial state to control the dynamics of the leader. To be more precise we can obtain these kind of reference signals:

- **Constant reference**, by placing a null eigenvalue and a negative one (for stability reasons). The initial condition can be arbitraly chosen to change the amplitude of the step;
- **Sinusoidal reference**, by placing a couple of complex conjugates with null real part. We chose $[-i + i]$ to get a sin function with frequency 1Hz;
- **Ramp reference**, by placing a couple of null eigenvalues.

The follower agents' estimated dynamics will look like this, due to the contribution of the distribuited regulator, and the cooperative observer

$$S_{i\neq 0} \quad \doteq \begin{cases} \dot{\hat{x}} = (A - BK_{reg})\hat{x} + Bu - cF\xi \\ u = cK\varepsilon \end{cases} \tag{Eq. 4}$$

where c is the coupling gain, K is the State Variable Feedback (SVFB) control gain, F is the observer gain, ε is the neighborhood tracking error and ξ is the neighborhood output estimation error. The follower nodes compute the correction operand $cF\xi$ considering the contribution of the neighbors' estimation error.

Results obtained using different references

The following are the results obtained using the distributed state observer with the variuos topologies and different references. To measure the performance of each topology in combination with the various choises of the Q and R matrices, the convergence time of the output $t(y)$ has been chosen as evaluation metric. The convergence time $t(y_i)$ of the i-th follower node ($i = 1 \dots 6$) is defined as the instant in which the follower's output (y_i) differs of a quantity $\varepsilon_{\text{sett}} = 0.1$ from the leader's output (y_0). To avoid transient errors at the beginning the time is considered from $1.0\,s$ on[1]

$$t(y_i) = \arg\min_{t>1.0} \{\, |y_0 - y_i| < \varepsilon_{\text{sett}}\} \tag{Eq. 5}$$

The system's settling time $t(y)$ is computed as the greatest settling time among all of the nodes, that is, when the last node properly tacks the leader

$$t(y) = \max_{i=1\dots6} t(y_i) = \max_{i=1\dots6} \arg\min_{t>1.0} \{\, |y_0 - y_i| < \varepsilon_{\text{sett}}\} \tag{Eq. 6}$$

[1]Sometimes the convergence criterion does not follow this indicator, since this type of evaluation can yield imprecise results, especially in the case of the sinusoidal reference. In this case the analysis of the convergence time has been performed by directly reterving the values from the plots.

Topologies	$\mathbf{Q}$ and $\mathbf{R}$ Matrices	$t(y)$
Topology 1 (1)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$166.1\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$53.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$20.5\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$11.1\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$7.5\,s$
Topology 2 (2)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$69.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$22.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$9.4\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$6.0\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$4.6\,s$
Topology 3 (3)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$93.3\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$30.6\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$12.2\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$7.4\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$5.5\,s$

Table 1: Convergence Times with a constant reference (Distributed Observer)

Topologies	$\mathbf{Q}$ and $\mathbf{R}$ Matrices	$t(y)$
Topology 1 (1)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$23.6\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$10.3\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$8.9\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$8.9\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$8.8\,s$
Topology 2 (2)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$9.1\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$7.5\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$8.9\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$8.9\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$8.9\,s$
Topology 3 (3)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$11.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$9.6\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$8.9\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$8.9\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$8.9\,s$

Table 2: Convergence Times with a sinusoidal reference (Distributed Observer)

Topologies	$\mathbf{Q}$ and $\mathbf{R}$ Matrices	$t(y)$
Topology 1 (1)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$15.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$9.7\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$7.0\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$6.8\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$5.9\,s$
Topology 2 (2)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$6.3\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$7.7\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$6.7\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$5.4\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$4.3\,s$
Topology 3 (3)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$11.3\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$7.7\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$7.0\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$6.1\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$4.9\,s$

Table 3: Different Convergence Times with a Ramp Reference (Distributed Observer)

A detailed comparison of the topologies is given below, where as comparison metric is considered the average convergence time $t_{\text{avg}}(y)$, for each topology with respect to each different reference, which consists of the mean of each convergence time for such topology and reference

Topologies	$t_{\text{avg}}(y)$ Constant	$t_{\text{avg}}(y)$ Sinusoidal	$t_{\text{avg}}(y)$ Ramp
Topology 1 (1)	$51.8\,s$	$12.1\,s$	$9.1\,s$
Topology 2 (2)	$22.6\,s$	$8.7\,s$	$6.1\,s$
Topology 3 (3)	$29.8\,s$	$9.6\,s$	$7.4\,s$

Table 4: Average Performance for each topology (Distributed Observer)

From the table above, it's evident that the best results are yield by the second topology (2), where the leader node is connected to each one of the followers. Although this topology does not accurately represent the real world for the reasons explained earlier, it can serve as a useful benchmark for comparing the results obtained with the other topologies.

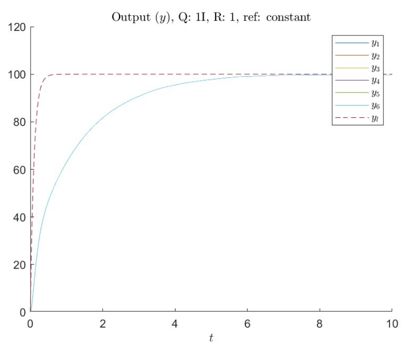

Figure 7: Distribuited Observer, Q1 R1, Topology 2

Contrary to that, the first topology (1), is the worst in terms of convergence time. In this topology, only one node is connected to the leader, and each node receives information from a single other follower. This limited connectivity slows down the distribution of information compared to a more interconnected topology. However, this topology has the advantage of having fewer links, which reduces the cost of the communication infrastructure.

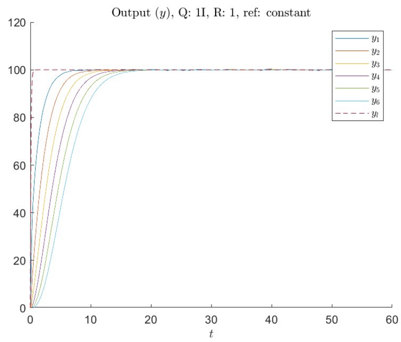

Figure 8: Distribuited Observer, Q1 R1, Topology 1

The last topology (3) falls somewhere between the other two: the leader pins three nodes of six nodes, resulting in a reasonably interconnected network. Although this topology is not as connected as the first one, it still yields comparable results. This demonstrates that this network configuration can achieve satisfactory convergence time without requiring excessive connections, striking a balance between performance and cost.

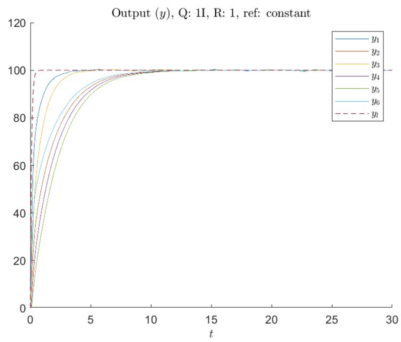

Figure 9: Distribuited Observer, Q1 R1, Topology 3

One last consideration is to be made about how the choice of the R parametr impacts the convergence time. It is possible to see that by choosing an R parameter greater with respect to the Q matrix yield a greater convergence time; this is due to the fact that the R parameter directly impacts the command activity (the greater R, the lower the command activity): having an higher value of the paramter leads to an higher convergence time.

2 Distributed regulator based on local observers

This is a variant of the previous scenario. Instead of using a distributed observer, the single agents exploit local information on the output estimation error, and only take into account their own $\tilde{y}$. As a result, the estimated dynamics formula will be:

$$\begin{cases} \dot{\hat{x}} = (A - BK_{reg})\hat{x} + Bu - cF\tilde{y} \\ u = cK\hat{\varepsilon} \end{cases}$$

The leader node does not change with respect to the distributed case.

Managing noise through matrix F

What is really challenging in this task is to properly design the coefficients of the matrix F, given that in the former case it was calculated through the Riccati Equation. Now the only requirement we have to comply with is that the matrix $A + cFC$ must be Hurwitz. Nevertheless, we could try to rewrite the expression for $\hat{x}$ in order to better visualize the dependencies among the terms, especially when it comes to the presence of noise on measurements (η). After having switched to the Laplace domain, the equation we get to is:

$$[sI - (A + cFC)]\hat{X}(s) = -cFY(s) - cFH(s) + BU(s) \qquad \text{(Eq. 7)}$$

Now we can focus on the transfer function which links the estimation of $\hat{x}$ to the noise η, whose complete form has been obtained through Matlab and it is reported below:

$$\frac{\hat{X}(s)}{H(s)} = \frac{1}{-100s^2 + 70827F_1 \cdot cs + 70827F_2 \cdot c} \cdot \begin{bmatrix} 100c \cdot (F_2 + F_1 s) \\ 100F_2 \cdot cs \end{bmatrix} \qquad \text{(Eq. 8)}$$

F_1 and F_2 are the two entries of matrix F while c is the coupling gain.
As a prerequisite the roots of the denominator must be negative and this is assured by the Routh criterion i.e. the coefficients of the polynomial have the same sign (this is a necessary but not sufficient condition, nonetheless we can proceed with a trial-and-error strategy to prove the correctness of our assumptions), so we can decide to set both F_1 and F_2 to negative values. Moreover, if the "input" η is a step this will be reflected on the $\hat{x}$ as a constant signal with amplitude the previous pseudo-transfer function evaluated with $s = 0$ (solved by applying the Final Value theorem). The result will be:

$$\frac{\hat{x}(0)}{\eta(0)} = \frac{1}{ab} \cdot \begin{bmatrix} 100c \cdot F_2 \\ 0 \end{bmatrix} \qquad \text{(Eq. 9)}$$

where we assume that a and b are the roots of the previous polynomial for given (non positive) values of F_1 and F_2. As it can be highlighted, $F_2 = 0$ is the only value allowing to zero the fraction and, as a consequence, cancel the contribution of the noise. Hence, the complete form of F is $[-1,\ 0]$.
As stated above, this setting works in the particular case of a step noise representing the sensor measurement error, but we report here the plot of the estimated y ($\hat{y}$) subject to step noise compared with the one in which the noise is represented by a random signal with zero mean and given variance. For the sake of clarity we have stressed the amplitude of noise in both cases to the value 300. It is demonstrated that this nice feature vanishes in the presence of random noise. One more observation is that those considerations are valid for $\hat{y}$ which is directly linked to $\hat{x}$, the variable used to derive the transfer function, but when it comes to the real followers' outputs we have noticed that there's a fixed offset with respect to the real leader's output.

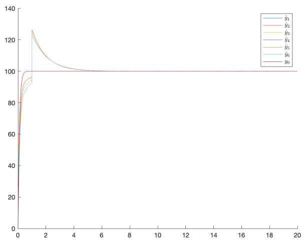

Figure 10: Step noise

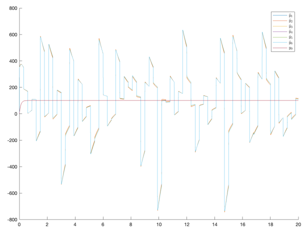

Figure 11: Random noise

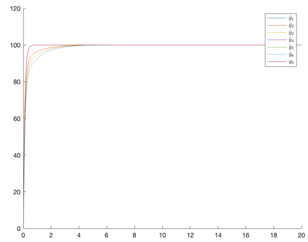

Figure 12: No noise

Effect of noise in the two scenarios

If instead of considering the previous simplified case with the step for the representation of noise we take into account a more general and realistic situation, we will notice that the local observer performance considerably degrade with respect to the cooperative one. This is due to the fact that local observers are more vulnerable to errors on sensors while in the other version the contribution of all the agents leverages the problem. This is clearly shown in the plots below.

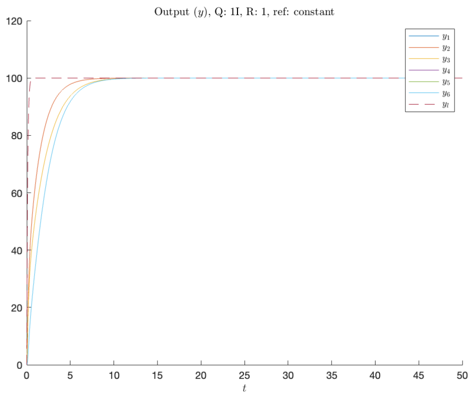

Figure 13: Cooperative observer

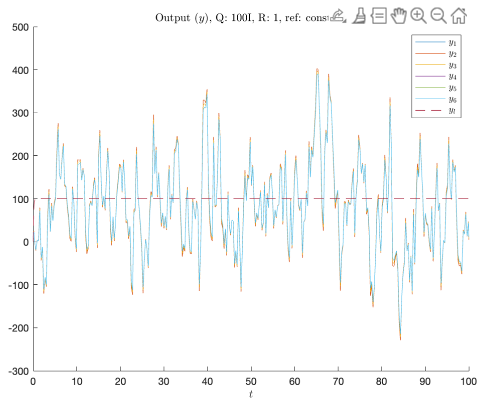

Figure 14: Local observer

Results obtained using different references

The following results are obtained by using the local state observer with the various topologies and different references. To measure the performance of each topology in combination with the various choises of the Q and R matrices, the same evaluation metric as the previous case has been chosen (see Eq. 5 and Eq. 6). The considered time starts from $1.0\,s$ to avoid transient errors[2].

[2]Sometimes the convergence criterion does not follow this indicator, since this type of evaluation can yield imprecise results, especially in the case of the sinusoidal reference. In this case the analysis of the convergence time has been performed by directly retreiving the values from the plots.

Topologies	$\mathbf{Q}$ and $\mathbf{R}$ Matrices	$t(y)$
Topology 1 (1)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$170.6\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$54.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$20.4\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$10.8\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$6.6\,s$
Topology 2 (2)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$65.1\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$23.6\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$10.1\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$5.3\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$4.8\,s$
Topology 3 (3)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$105.2\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$31.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$11.5\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$6.5\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$5.2\,s$

Table 5: Different convergence times with constant reference (Local Observer)

Topologies	$\mathbf{Q}$ and $\mathbf{R}$ Matrices	$t(y)$
Topology 1 (1)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$22.1\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$17.4\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$13.5\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$12.4\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$6.9\,s$
Topology 2 (2)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$8.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$7.7\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$6.9\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$6.8\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$6.8\,s$
Topology 3 (3)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$11.7\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$8.1\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$6.9\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$6.9\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$6.8\,s$

Table 6: Different convergence times with sinusoidal reference (Local Observer)

Topologies	$\mathbf{Q}$ and $\mathbf{R}$ Matrices	$t(y)$
Topology 1 (1)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$10.8\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$9.0\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$5.9\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$5.7\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$5.3\,s$
Topology 2 (2)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$6.9\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$6.8\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$5.6\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$5.2\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$4.7\,s$
Topology 3 (3)	$\mathbf{Q} = \mathbf{I_2},\ R = 100$	$7.2\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 10$	$6.2\,s$
	$\mathbf{Q} = \mathbf{I_2},\ R = 1$	$5.8\,s$
	$\mathbf{Q} = 10 \cdot \mathbf{I_2},\ R = 1$	$5.3\,s$
	$\mathbf{Q} = 100 \cdot \mathbf{I_2},\ R = 1$	$4.9\,s$

Table 7: Different convergence times with a ramp reference (Local Observer)

A detailed comparison of the topologies is given below, where as comparison metric is considered the average convergence time $t_{\text{avg}}(y)$, as in the previous analysis

Topologies	$t_{\text{avg}}(y)$ Constant	$t_{\text{avg}}(y)$ Sinusoidal	$t_{\text{avg}}(y)$ Ramp
Topology 1 (1)	$52.7\,s$	$14.5\,s$	$7.3\,s$
Topology 2 (2)	$21.8\,s$	$7.4\,s$	$5.8\,s$
Topology 3 (3)	$32.1\,s$	$8.1\,s$	$5.9\,s$

Table 8: Average performance for each topology (Local Observer)

As it is possible to see from the table above, using a local observer yields similar results to the previous case, where the observer was distributed instead (see 4).
The second topology is still the best performer in terms of convergence time, due to the great quantity of followers connected to the leader, and the first one still gives the slowest times, since the information takes longer to spread through the network. The third topology, as expected, provides good results in terms of convergence time, with the addition of being a likely real-case scenario, due to the number of followers connected to the leader.