

Tracking Control via Variable-gain Integrator and Lookahead Simulation: Application to Leader-follower Multiagent Networks

Y. Wardi* C. Seatzu** M. Egerstedt*

* *School of Electrical and Computer Engineering, Georgia Institute of
Technology, Atlanta, GA 30332, USA.*

e-mail: ywardi@ece.gatech.edu, magnus@ece.gatech.edu.

** *Department of Electrical and Electronic Engineering, University of
Cagliari, Italy.*

e-mail: seatzu@diee.unica.it.

Abstract: This paper concerns an output-tracking technique based on a standalone integrator with variable gain. The control algorithm, recently proposed by the authors, appears to have a wide scope in linear and nonlinear systems while aiming at simple and efficient computations in the loop. For a class of memoryless systems it resembles a Newton-Raphson flow for solving the loop equation, but it is applicable to a broader class of dynamical systems. Furthermore, the technique is suitable for tracking constant as well as time-dependent reference signals, and its convergence performance is robust with respect to computational errors in the loop. The objective of this paper is to test the technique on a control problem arising in multi-agent systems. Specifically, we are motivated by regulating trajectories of follower agents by a lead agent in a platoon or swarm of multi-agent networks connected by the graph Laplacian. We study a particular example, which is challenging from the standpoint of control, with the aim of identifying the limits of the technique and investigating their possible extensions.

1. INTRODUCTION

Integrator is a common part of a controller for tracking applications of dynamical systems. Controllers rarely are comprised of a standalone integrator since typically the corresponding closed-loop system would either be unstable or have narrow stability margins. For this reason, tracking controllers often have proportional and derivative elements thereby forming the PID control (Franklin et al. (2015)). Nevertheless, authors of this paper recently considered a standalone integral control for regulating hardware and software processes in multicore processors, like power and throughput (see Chen et al. (2018) and references therein). Initially the inclusion of a proportional element was thought to be impractical on technical grounds, and the aforementioned stability issues were addressed by endowing the integrator with a variable gain which is computed in real-time according to the control law (Almoosa et al. (2012)). The tracking-control technique was extended from computer-related applications to a general class of times Discrete Event Dynamic Systems (DEDS), and its convergence was derived in that abstract setting (Wardi et al. (2016)).

Lately we made an initial attempt to extend the tracking controller from DEDS to continuous-time dynamical systems, both linear and nonlinear (Wardi et al. (2017)).

The motivation came from problems of controlling and regulating trajectories of platoons and swarms of multi-agent mobile systems. Ref. Wardi et al. (2017) presents the control technique that we developed, and tests it on several second-order systems and an eight-agent platoon. The results were deemed encouraging and stimulated a further investigation including applications to general classes of larger, more complicated systems. In particular, we wish to explore the limitations of the technique in networked mobile robots with coordinated motion (e.g., Desai et al. (2001); Zhang and Leonard (2006)) in terms of size, complexity, and manipulability as defined in Kawashima and Egerstedt (2014). The material in this paper provides a first step in this direction.

If the plant subsystem is nonlinear, the controller is nonlinear as well. Existing nonlinear regulation techniques, such as the Byrnes-Isidori regulator (Isidori and Byrnes (1990)) and Khalil's high-gain observers for output regulation (Khalil (1998)), are more general and perhaps more powerful than the technique described in Wardi et al. (2017). However, their effectiveness is due to significant computational sophistication, like nonlinear inversions and the appropriate nonlinear normal form. On the other hand, one of the objectives stated in Wardi et al. (2017) is to explore a computationally simple technique which is based on a variable-gain integrator. Despite its low

computational requirements, the regulation technique in Wardi et al. (2017) appears to work well for a number of simple test problems, and the purpose of this paper is to try it on more challenging control problems arising in the navigation of multi-agent systems.

The regulation technique described below is based on a lookahead simulation of the system which serves as both a predictor and an observer. Its convergence performance is robust to errors in the loop-computations, but sensitive to prediction errors. The accuracy of the predictor depends on how far in the future the simulator predicts the system's response, and therefore it appears to be advantageous to choose a small prediction horizon. On the other hand, analyses of two-dimensional systems, including the example considered in Wardi et al. (2017), indicate that the closed-loop systems are unstable if their predictor horizons are too short. To get around this conundrum we first choose a small prediction horizon without regard to stability considerations, then stabilize the system by speeding up the controller's response. The latter part does not increase the controller's gain, which could destabilize the closed-loop system, but increases its stiffness in a sense defined below; this seems not only to stabilize the system but also to improve its tracking when the target signal is time-dependent rather than a constant. All of the will be described in detail in the sequel.

The rest of the paper is organized as follows. Section 2 recounts the description of the regulation technique defined in Ref. Wardi et al. (2017). It also derives a tracking result for systems whose plants are memoryless nonlinearities, while deferring the general dynamic case, which is more complicated, to a forthcoming publication. Section 3, containing the main contribution of the paper, presents simulation experiments and analyzes them in an attempt to identify limitation of the technique. Finally, Section 4 concludes the paper.

2. FRAMEWORK FOR TRACKING AND REGULATION

Consider the system diagram shown in Figure 1, where t denotes time, and for a given integer $k \geq 1$, $r(t) \in R^k$ is the reference input-signal, $u(t) \in R^k$ is the input to the plant, $y(t) \in R^k$ is the output, and $e(t) := r(t) - y(t)$ is the error signal. Note that the input and output have the same dimension, k . The purpose of the controller is to ensure output tracking of the reference signal. If $r(t) \equiv r$ for a constant $r \in R^k$, and the plant is a time-invariant system, then 'tracking' means that

$$\lim_{t \rightarrow \infty} (r - y(t)) = 0. \quad (1)$$

However, in the event that $r(t)$ is a nonconstant function of time or the plant subsystem is time varying, Eq. (1) may be impossible to obtain and instead we seek an inequality of the form

$$\limsup_{t \rightarrow \infty} \|r(t) - y(t)\| < \epsilon \quad (2)$$

for a small $\epsilon > 0$. This case will be discussed in the sequel, and a technique for reducing ϵ by a parameter of the controller will be pointed out.

According to the general setting of this paper, the plant system is a dynamical system modelled by a differential

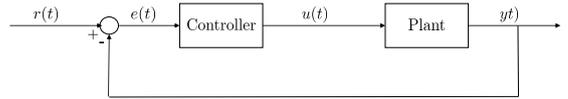


Fig. 1. Basic control system

equation, and $r(t)$ is time-dependent. However, in order to explain the salient features of the controller, we first discuss the case where the plant is a memoryless nonlinearity. This exposition serves in part as a survey of background material presented in Wardi et al. (2017).

2.1 The plant as a Memoryless Nonlinearity

Consider the case where the reference input $r(t)$ is a function of time, and the plant is a memoryless nonlinearity of the form

$$y(t) = g(u(t)) \quad (3)$$

for a continuously-differentiable function $g : R^k \rightarrow R^k$. Assume that the Jacobian $\frac{\partial g}{\partial u}(u(t))$ is nonsingular for every $u(t)$ encountered in the following discussion. The material described in the rest of this section appeared in Wardi et al. (2017) and its expanded version in the arxiv.

We define the controller by the equation

$$\dot{u}(t) = \left(\frac{\partial g}{\partial u}(u(t)) \right)^{-1} (r(t) - y(t)), \quad (4)$$

and notice that, in the event that $r(t) \equiv r$ for a constant $r \in R^k$, it represents the Newton-Raphson flow for solving the equation $r - g(u) = 0$.

Ref. Wardi et al. (2017) proves the following result.

Proposition 1. Suppose that $r(\cdot)$ is continuously differentiable in t , and let $\gamma > \|\dot{r}(t)\|$ for every $t \geq 0$. Then,

$$\limsup_{t \rightarrow \infty} \|r(t) - y(t)\| \leq \gamma. \quad (5)$$

Note that in the case where $r(t) \equiv r$ (a constant), Eq. (1) follows from Proposition 1 by taking $\gamma = 0$.

In order to reduce the Right-Hand Side (RHS) of Eq. (5) we can speed up the action of the controller without increasing its gain. This can be done by multiplying the RHS of (4) by a constant $\alpha > 1$, thereby modifying the definition of the controller from (4) to the following equation,

$$\dot{u}(t) = \alpha \left(\frac{\partial g}{\partial u}(u(t)) \right)^{-1} (r(t) - y(t)). \quad (6)$$

To see the effect of this on the e -to- u relation (see Figure 1), consider the example where that relation has the transfer function $\frac{1}{s+1}$, i.e., $U(s) = \frac{1}{s+1} E(s)$. Then replacing (4) by (6) results in the e -to- u transfer function of $\frac{\alpha}{s+\alpha}$, which does not change the DC gain, but shifts the pole from $s = -1$ to $s = -\alpha$ thereby reducing the settling time. For this reason we say that this action increases the stiffness of the controller, and label α as the *stiffness parameter*. Proposition 1 is modified as follows.

Proposition 2. Suppose that $r(\cdot)$ is continuously differentiable in t , and let $\gamma > \|\dot{r}(t)\|$ for every $t \geq 0$. Suppose also that the trajectory $\{u(t) : t \in [0, \infty)\}$ is contained in an open set $D \subset R^k$ where the Jacobian $\frac{\partial g}{\partial u}(u)$ is nonsingular. Then,

$$\limsup_{t \rightarrow \infty} \|r(t) - y(t)\| \leq \frac{\gamma}{\alpha}. \quad (7)$$

The proof is similar to that of Proposition 1 as derived in Wardi et al. (2017), hence relegated to the appendix.

2.2 The Plant as a Dynamical System

Suppose that the plant is a dynamical system with the input $u(t) \in R^k$, state $x(t) \in R^n$, and output $y(t) \in R^k$, for given positive integers k and n . The state equation has the form

$$\dot{x}(t) = f(x(t), u(t)) \quad (8)$$

for a function $f : R^n \times R^k \rightarrow R^n$, with the boundary condition $x(0) = x_0 \in R^n$. The output equation has the form

$$y(t) = h(x(t)) \quad (9)$$

for a function $h : R^n \rightarrow R^k$. Note that, as in the last subsection, the dimensions of $u(t)$ and $y(t)$ are identical.

The following assumption on the system is made:

Assumption 1. (i). The function $f(x, u)$ is continuously differentiable in x for every $u \in R^k$, and continuous in u for every $x \in R^n$.

(ii). For every compact sets $\Gamma_1 \subset R^n$ and $\Gamma_2 \subset R^k$, the functions $f(x, u)$ and $\frac{\partial f}{\partial x}(x, u)$ are Lipschitz continuous on $\Gamma_1 \times \Gamma_2$.

(iii). For every compact set $\Gamma_2 \subset R^k$, there exists $K > 0$ such that, for every $x \in R^n$ and $u \in \Gamma_2$,

$$\|f(x, u)\| \leq K(\|x\| + 1).$$

(iv). The function $h(x)$ is continuously differentiable.

This assumption implies that the differential equation (8) has a unique solution on $t \in [0, \infty)$ for every piecewise-continuous function $u(\cdot)$.

A key question is how to define a function like $g(u(t))$ which was used to define the controller in the last subsection via Eq. (4). In the current setting of dynamic plant-subsystems there is no natural way to express $y(t)$ as a function of $u(t)$ since the system's output at time t depends on its state evolution during a past, positive-length interval. One way to define $g(u(t))$ is to simulate the system for a positive amount of time, say T seconds for a given $T > 0$, from time t onwards. By "simulation" we mean numerical simulation. Thus, assuming that $x(t)$ and $u(t)$ can be observed at time t , we define $g(u(t))$ for a given $T > 0$ as follows: Solve (numerically) the differential equation

$$\dot{\tilde{x}}(\xi) = f(\tilde{x}(\xi), u(t)), \quad \xi \in [t, t+T] \quad (10)$$

with the initial condition $\tilde{x}(t) = x(t)$, then define $g(u(t)) = h(\tilde{x}(t+T))$. Note that $g(u(t))$ also depends on t , T , and $x(t)$ via the initial condition of (10), hence ought to be denoted by $g(t, x(t), u(t), T)$, but we use the shorthand notation $g(x(t), u(t))$. Thus, formally, we set

$$g(x(t), u(t)) := h(\tilde{x}(t+T)). \quad (11)$$

We mention that the computations of $g(x(t), u(t))$ and its derivative $\frac{\partial g}{\partial u}(x(t), u(t))$ can be performed by any computational technique, we have used the forward Euler method. The controller equation is defined, in analogy with Eq. (4), by the following equation,

$$\dot{u}(t) = \left(\frac{\partial g}{\partial u}(x(t), u(t)) \right)^{-1} (r(t+T) - g(x(t), u(t))). \quad (12)$$

Note the term $r(t+T)$, not $r(t)$, in (12); the rationale is that $g(x(t), u(t))$ attempts to estimate $x(t+T)$ and hence

it is compared to $r(t+T)$ in the controller's definition. Since $g(x(t), u(t))$ attempts to estimate $y(t+T)$ at time t , we call it a *predictor*. Observe that in Eq. (10) we use the input $u(t)$ throughout the interval $\xi \in [t, t+T]$; the reason is that $u(\xi)$ will change throughout that interval according to (12), but we do not know, at time t , its values at $\xi > t$. Thus, the predictor at time t is defined via (10) for a fixed input $u(t)$ throughout the interval $\xi \in [t, t+T]$, but the actual input $u(\xi)$, defined via (12), will not be a constant in that interval.

Due to its dependence on $x(t)$, Eq. (12) provides $u(t)$ in a feedback form. Furthermore, Eqs. (8) and (12) taken together define the closed-loop system in terms of the aggregate state variable $(x^\top, u^\top)^\top \in R^{n+k}$ via the following equation

$$\begin{pmatrix} \dot{x}(t) \\ \dot{u}(t) \end{pmatrix} = \begin{pmatrix} f(x(t), u(t)) \\ \left(\frac{\partial g}{\partial u}(x(t), u(t)) \right)^{-1} (r(t) - g(x(t), u(t))) \end{pmatrix}, \quad (13)$$

and we have to be concerned with the stability of this system. It was mentioned earlier that convergence of the regulation algorithm can be sensitive to measurement errors, meaning, in this case, prediction errors. One way to limit these errors is to choose a small lookahead horizon T . However, our analysis in Wardi et al. (2017) and subsequent experience with various examples have shown that for a small-enough T the closed-loop system is unstable. To get around this difficulty we increase the stiffness of the controller by a factor $\alpha > 1$ as in the last subsection; the closed-loop system becomes

$$\begin{pmatrix} \dot{x}(t) \\ \dot{u}(t) \end{pmatrix} = \begin{pmatrix} f(x(t), u(t)) \\ \alpha \left(\frac{\partial g}{\partial u}(x(t), u(t)) \right)^{-1} (r(t) - g(x(t), u(t))) \end{pmatrix}. \quad (14)$$

Therefore, a key theoretical question is whether the system has the following property: For every $T > 0$ there exists $\bar{\alpha} > 0$ such that, for every $\alpha > \bar{\alpha}$, the closed-loop system, defined by Eq. (14), is stable. It is easy to answer this question in the case of linear, time-invariant systems. Let $f(x, u) = Ax + Bu$, and $y = Cx$, for given matrices $A \in R^{n \times n}$, $B \in R^{n \times k}$ and $C \in R^{k \times n}$, and suppose that A is nonsingular. Then, for a given stiffness parameter $\alpha > 0$, the aggregate closed-loop system (14) is LTI as well, and was shown in Wardi et al. (2017) to have the following form:

$$\begin{pmatrix} \dot{x}(t) \\ \dot{u}(t) \end{pmatrix} = \Phi_{T,\alpha} \begin{pmatrix} x(t) \\ u(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \Psi_{T,\alpha} \end{pmatrix} r(t+T), \quad (15)$$

where the matrices $\Phi_{T,\alpha} \in R^{(n+k) \times (n+k)}$ and $\Psi_{T,\alpha} \in R^{k \times k}$ are defined by

$$\Phi_{T,\alpha} := \begin{pmatrix} A & B \\ -\alpha(CA^{-1}(e^{AT} - I_n)B)^{-1}Ce^{AT} & -\alpha I_k \end{pmatrix}, \quad (16)$$

and

$$\Psi_{T,\alpha} := \alpha(CA^{-1}(e^{AT} - I_n)B)^{-1}. \quad (17)$$

Stability of the closed-loop system can be ascertained by checking whether the matrix $\Phi_{T,\alpha}$ is Hurwitz. The next section examines such a system in detail.

3. EXAMPLE

We consider a multi-agent system arranged in a tandem configuration where the motion trajectories of the agents are coordinated by the graph Laplacian. The first agent is the leader and it is the only agent with an exogenous input. It is tasked with regulating the trajectory of the last agent to a given planar curve. Let N denote the number of agents, and denote the trajectory of the i th agent by $x_i(t)$, $i = 1, \dots, N$. Let D_i denote the neighborhood of agent i according to the Laplacian, then its motion is defined by the equation

$$\dot{x}_i(t) = \sum_{j \in D_i} (x_j(t) - x_i(t)) + \delta_{i,1} u(t), \quad (18)$$

where $\delta_{i,1}$ is the Kronecker delta, and $u(t)$ is the exogenous control which can be applied only to agent 1. The neighborhoods reflect the tandem nature of the network. Thus, for every $i = 2, \dots, N-1$, $D_i = \{i-1, i+1\}$, $D_1 = \{2\}$, and $D_N = \{N-1\}$. The objective of the control is to have x_N track a given curve in R^2 , denoted by $\{r(t)\}$. The tandem configuration of the network renders the control problem challenging because of the indirect connection between the lead agent and the last agent, especially for a large N . This corresponds to a weak manipulability of the network, and may require the application of large, fast-changing inputs $u(t)$ for effective trajectory control of the last agent. The objective of this section is to explore and test the efficacy and limitations of the proposed regulation technique.

The closed-loop system defined by Eq. (15) is linear, time invariant, and has dimension $2N + 2$. Its forward loop is described by the equations $\dot{x}(t) = Ax(t) + Bu(t)$ and $y(t) = Cx(t)$, and the feedback law is defined by (12). Consequently the system is described by Eqs. (15) with (16) and (17). The system matrices A , B , and C are

$$A = \begin{pmatrix} -I_2 & I_2 & 0_2 & 0_2 & \dots \\ I_2 & -2I_2 & I_2 & 0_2 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & 0_2 & 0_2 & I_2 & -I_2 \end{pmatrix} \in R^{2N \times 2N},$$

where I_2 and 0_2 are the 2×2 identity matrix and zero matrix, respectively;

$$B = \begin{pmatrix} I_2 \\ 0_2 \\ \dots \\ \dots \\ 0_2 \end{pmatrix} \in R^{2N \times 2},$$

and

$$C = (0_2 \dots 0_2 I_2) \in R^{2 \times 2N}.$$

We chose $N = 10$ and experiment with two targets: a point (constant), and a circle around a given point. The results are described in the following paragraphs.

In all of the experiments we solved the differential equations by the forward Euler method. For the agents' trajectories and feedback law (Eq. (17)) we use the stepsize of $dt = 0.001$, and for the lookahead simulations (Eq. (10)) we chose the stepsize to be $\Delta t := 0.01T$. The final time is $t_f = 20$ seconds, and the stiffness parameter is set to $\alpha = 40$. In all cases we show the graph of the first coordinate of agent 10, $\{x_{10,1}(t)\}$, but not of the second coordinate since it displays a comparable behavior

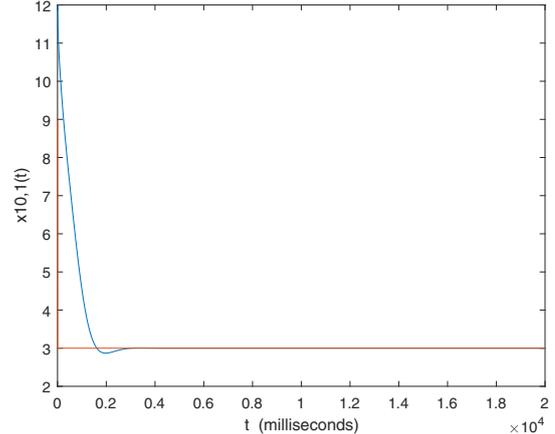


Fig. 2. $x_{10,1}(t)$. Target signal: $r(t) \equiv (3, 1)^\top$; $T = 1.0$

regarding convergence to the target curve. We also show the graph of the first coordinate of the control, $\{u_1(t)\}$, for some of the experiments.

Experiment 1. The target is the point, $r(t) \equiv (3, -1)^\top \in R^2$. We first set $T = 0.1$, but despite the large value of α , the closed-loop system is unstable and $x_{10,1}(t)$ oscillates unboundedly. We then increased T to 1.0, but the system is still unstable. Next, we increased the stiffness of the Laplacian.¹ In accordance with the linear structure of the network we define the stiffness parameter of agent i , denoted by β_i , as an affine function of i ; we chose $\beta_i = 2i + 3$. Consequently the graph-Laplacian equation becomes

$$\dot{x}_i(t) = \beta_i \sum_{j \in D_i} (x_j(t) - x_i(t)) + \delta_{i,1} u(t). \quad (19)$$

The corresponding change in the system-matrix A is that its i th 2×2 -block row is multiplied by β_i , while the matrices B and C remain unchanged. The simulation results are shown in Figure 2 and Figure 3 for $x_{10,1}(t)$ and $u_1(t)$, respectively, and they clearly indicate a tracking convergence.

We note in Figure 3 that the control variable $u(t)$ can have large values at the initial stage of the algorithm. How to keep its magnitude within given upper bounds will be seen in the later discussion.

Experiment 2. The target signal is a circle, $r(t) = (10 + \cos(t), 10 + \sin(t))^\top$. To counter the delay's effects due to the fact that $r(t)$ is time dependent we chose $T = 0.5$ instead of $T = 1.0$, but kept the rest of the simulation parameters, including β_i , the same as for Experiment 1. The simulation results are depicted in Figure 4 and Figure 5. In Figure 4, the trajectory $x_{10,1}(t)$ is marked in blue, and the target value $r_1(t)$ is marked in red. The figure indicates convergence of $x_{10,1}(t)$ to $r_1(t)$ in about 5 secs. However, Figure 5 exhibits a considerable magnitude and large variations of the control signal $u(t)$ at the start of

¹ Strictly speaking, we described the application of the stiffness parameter to the controller equation (14) but not to the plant. The Laplacian is considered here as a part of the plant, but also can be viewed as a part of the controller and there is no reason not to increase its stiffness in the indicated manner.

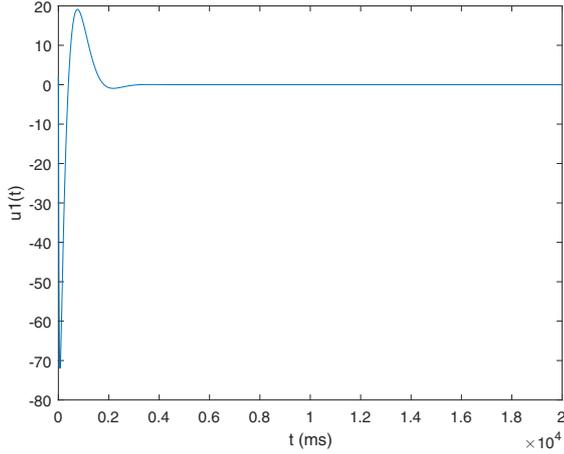


Fig. 3. $u_1(t)$. Target signal: $r(t) \equiv (3, 1)^\top$; $T = 1.0$

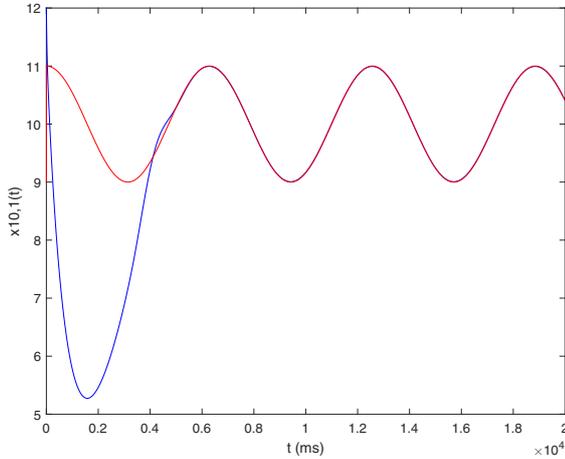


Fig. 4. $x_{10,1}(t)$. Target signal: $r(t) = (10 + \cos(t), 10 + \sin(t))^\top$; $T = 0.5$

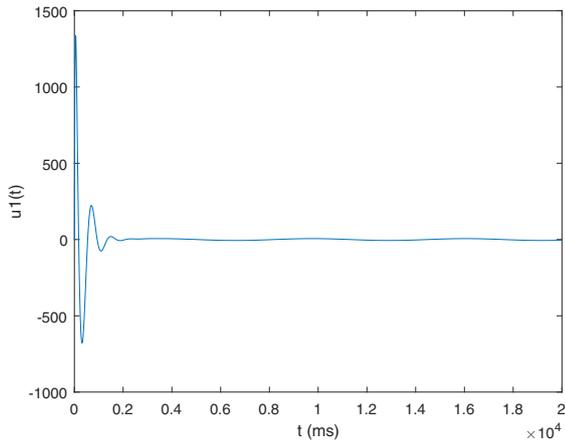


Fig. 5. $u_1(t)$. Target signal: $r(t) = (10 + \cos(t), 10 + \sin(t))^\top$; $T = 0.5$

the algorithm. This is not surprising in light of the fact that the controller essentially is a standalone integrator, but it can be problematic in applications.

To get around this problem we saturate the norm of $u(t)$, $\|u(t)\|$, by projecting $u(t)$, at each time $t \in [0, t_f]$, into a

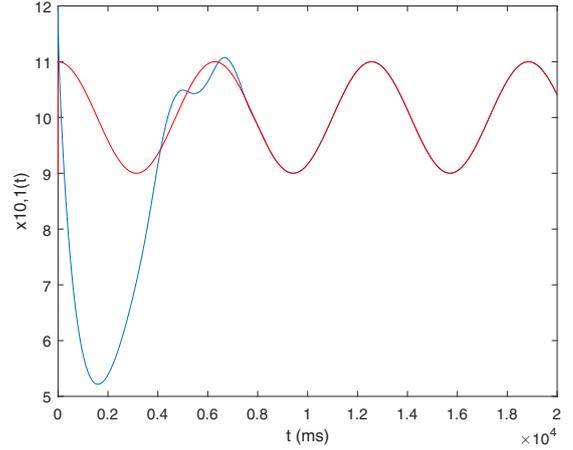


Fig. 6. $x_{10,1}(t)$. Target signal: $r(t) = (10 + \cos(t), 10 + \sin(t))^\top$; $T = 0.5$; $R = 20$

circle in R^2 with center at 0 and a given radius R , denoted by $B(0, R)$. Formally, denote by $\Pi_R(u)$ the projection function from R^2 into $B(0, R)$. Moreover, denote the RHS of (12) by $\zeta(x(t), u(t))$ so that (12) is of the form

$$\dot{u}(t) = \zeta(x(t), u(t)).$$

Now we replace (12) by the following one-sided differential equation,

$$u(t + dt) = \Pi_R(u(t) + \zeta(x(t), u(t))dt).$$

We note that if $\|u(t)\| < R$ or $\|u(t)\| = R$ and $\langle \zeta(t), u(t) \rangle < 0$ then Eq. (12) is unchanged, whereas if $\|u(t)\| = R$ and $\langle \zeta(t), u(t) \rangle \geq 0$ then $\|u(t + dt)\| = R$ as well. Assuming that at $\|u(0)\| \leq R$ by choice, this implies that $\|u(t)\| \leq R$ for every $t \geq 0$.

The practical effect of this is not only to limit $\|u(t)\|$ to R but also often to dampen its variation. However, this can come at the expense of longer settling times. We chose $R = 20$, and the results are depicted in Figure 6 and Figure 7. Comparing these to the unsaturated case shown in Figures 4 and 5, respectively, we obtain convergence of $x_{10,1}(t)$ to $r_1(t)$ in about 7 seconds as compared to 5 seconds for the unsaturated system. However, $\|u(t)\| \leq 20$ for all t , as compared to the unsaturated system where $\|u(t)\|$ reached magnitudes over 1,300.

Finally, we tried to lower the input saturation bound to $R = 5$. The results for $x_{10,1}(t)$ are depicted in Figure 8 while $u_1(t)$ is not shown. Figure 8 indicates that tracking is not obtained. The reason is that the control $u(t)$ does not have enough power to track the target-signal $r(t)$ because its frequency is too high. In fact, according to the simulation results (not shown) $\|u(t)\| = 5$ for every $t \in [0, t_f]$, namely the control variable is saturated throughout the algorithm's run, and its bound of 5 is insufficient to achieve the output tracking.

4. CONCLUSIONS

This paper concerns a technique for output regulation and tracking in continuous-time dynamical systems. The technique is based on an integral control with a variable gain, and a lookahead simulation to predict future outputs of the plant. Both stability and tracking performance are enhanced by adding stiffness to the controller. We

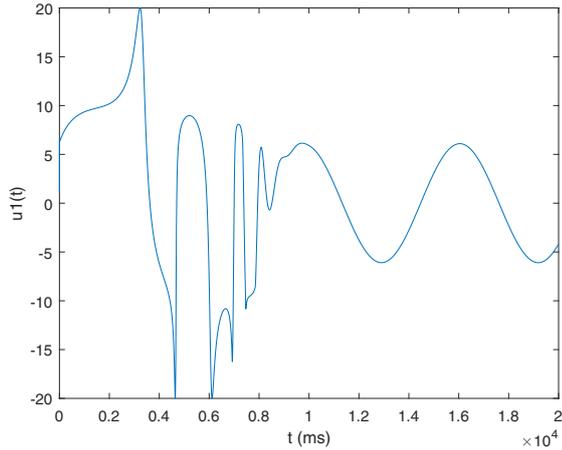


Fig. 7. $u_1(t)$. Target signal: $r(t) = (10 + \cos(t), 10 + \sin(t))^T$; $T = 0.5$; $R = 20$

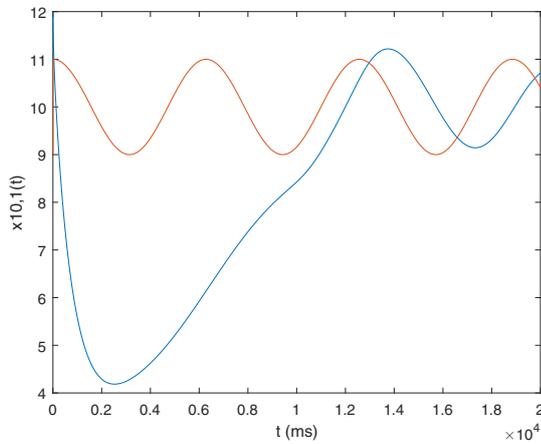


Fig. 8. $x_{10,1}(t)$. Target signal: $r(t) = (10 + \cos(t), 10 + \sin(t))^T$; $T = 0.5$; $R = 5$

tested the technique on a control problem arising in a multi-agent network where the motion trajectories of its agents are coordinated by the graph Laplacian. The results indicate that the regulation technique may have merit in applications to swarms of autonomous agents, and future research will test this hypothesis theoretically and in a laboratory setting.

5. APPENDIX

Proof of Proposition 2: Define the function $V(u, t)$ by

$$V(u, t) = \frac{1}{2} \|r(t) - y(t)\|^2. \quad (20)$$

Since $y(t) = g(u(t))$, and by Eq. (6), we have that

$$\dot{V}(u(t), t) = (r(t) - y(t))^T (\dot{r}(t) - \alpha(r(t) - y(t))). \quad (21)$$

By the fact that $\|\dot{r}(t)\| \leq \gamma$, it follows from (21) that

$$\dot{V}(u(t), t) \leq -\|r(t) - y(t)\|(\alpha\|r(t) - y(t)\| - \gamma). \quad (22)$$

Fix $\epsilon > 0$. If

$$\|r(t) - y(t)\| > (1 + \epsilon) \frac{\gamma}{\alpha}, \quad (23)$$

then, by (22),

$$\dot{V}(u(t), t) \leq -\|r(t) - y(t)\|\epsilon\gamma. \quad (24)$$

Applying (23) to (24), we obtain that

$$\dot{V}(u(t), t) \leq -(1 + \epsilon)\epsilon\gamma^2/\alpha. \quad (25)$$

By Lyapunov's direct method, this implies that

$$\limsup_{t \rightarrow \infty} \|r(t) - y(t)\| \leq (1 + \epsilon) \frac{\gamma}{\alpha}.$$

Since ϵ can be arbitrarily small, this implies Eq. (7) thereby completing the proof.

REFERENCES

- Almoosa, N., Song, W., Wardi, Y., and Yalamanchili, S. (2012). A power capping controller for multicore processors. In *American Control Conference (ACC)*, Montreal, Canada, June 27-29, 4709–4714.
- Chen, X., Wardi, Y., and Yalamanchili, S. (2018). Instruction-throughput regulation in computer processors with data-center applications. *Discrete Event Dynamic Systems: Theory and Applications*, 28, 127–158.
- Desai, J., Ostrowski, J., and Kumar, R. (2001). Modeling and control of formations of nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation*, 17(6), 905–908.
- Franklin, G., Powell, J., and Emami-Naeini, A. (2015). *Feedback Control of Dynamical Systems*. Prentice Hall.
- Isidori, A. and Byrnes, C. (1990). Output regulation of nonlinear systems. *IEEE Transactions on Automatic Control*, 35, 131–140.
- Kawashima, H. and Egerstedt, M. (2014). Manipulability of leader-follower networks under a rigid-link approximation. *Automatica*, 50(3), 695–796.
- Khalil, H. (1998). On the design of robust servomechanisms for minimum phase nonlinear systems. *Proc. 37th IEEE Conference on Decision and Control*, Tampa, FL, 3075–3080.
- Wardi, Y., Seatzu, C., Chen, X., and Yalamanchili, S. (2016). Performance regulation of event-driven dynamical systems using infinitesimal perturbation analysis. *Nonlinear Analysis: Hybrid Systems*, 22, 116–136.
- Wardi, Y., Seatzu, C., Egerstedt, M., and Buckley, I. (2017). Performance regulation and tracking via look-ahead simulation: Preliminary results and validation. In *56th IEEE Conf. on Decision and Control*, Melbourne, Australia, December 12-15.
- Zhang, F. and Leonard, N. (2006). Coordinated patterns of unit speed particles on a closed curve. *Systems and Control Letters*, doi: 10.1016/j.sysconle.2006.10.027.