Multi-Pendulum Synchronization Using Constrained Agreement Protocols

Rahul Chipalkatty[1], Magnus Egerstedt[1], and Shun-Ichi Azuma[2]
gtg500s@gatech.edu,magnus@ece.gatech.edu,sazuma@i.kyoto-u.ac.jp
[1]Georgia Institute of Technology, Atlanta, GA 30332, USA
[2]Graduate School of Informatics, Kyoto University, Uji, Kyoto 611-0011, Japan

***Abstract*— This paper considers the problem of coordinating multiple pendula attached to mobile bases. In particular, the pendula should move in such a way that their motion is synchronized, which calls for two problems to be solved simultaneously, namely a constrained optimal control problem for each pendulum, and a constrained agreement problem across the network of pendula. A novel way of manipulating the initial conditions in the consensus equation is presented that will solve the latter of these problems, and simulation results are presented that support the viability of the proposed approach.**

I. INTRODUCTION

In this paper we consider the problem of controlling a collection of pendula in a coordinated fashion. In particular, their mobile bases are to be controlled in such a way that, at some specified terminal time, they move in unison (with the same frequency and phase) at a fixed inter-pendula distance. This should be achieved using only local information, i.e. the control actions are only to be controlled based on information from neighboring pendula.

The motivation behind this work comes from recent efforts to develop robotic marionettes [1], [2], as shown in Figure 1. There, the ambition is to have the marionette execute sequences of motions in such a way that the transitions between different modes of operation are graceful and, at the same time, reflect the original play script. For this to work, it is paramount that the different limbs on an individual puppet are coordinated. If multiple puppets are acting together, this coordination issue becomes even more important. However, to reduce the computational burden, it is important that the coordination can be achieved while only taking into account the relevant, local information [2]. This paper is a manifestation of that idea and it provides the first basic building block needed to pull off this endeavor.

In this paper, we will let each pendulum solve an initial optimal control problem based on local information. The output of the optimal control problem is a desired final state, as well as a final state associated with neighboring pendula. The problem then becomes that of ensuring that, after an agreement on the final states over *all* pendula has been reached, the agreed upon states satisfy the constraints. And, it turns out that even though running the standard consensus equation (e.g. [3], [4]) – or versions of the gossip algorithm [5], [6] – will result in an agreement, the agreed upon states do not satisfy the constraints. However, what will be shown is that one can manipulate the initial conditions in order to satisfy the constraints.

The agreement protocol (or consensus equation) has by now emerged as a standard way in which to achieve agreement among agents in a distributed network. It can be utilized for anything from agreement in embedded physical systems like mobile robots or UAVs, to distributed computer networks, e.g. [7], [8], [9]. And, as the final value of agreement is dependent on the agents' initial conditions, it is not overly surprising that this dependency can be employed such that the final agreement state is guaranteed to satisfy certain constraints.

Relevant work on agreement for systems with constraints or oscillating dynamics include [10], where the stability of the Kuromoto model of coupled nonlinear oscillators was investigated, and [11], [12], where constrained consensus was considered. However, in the former case, no constraints were present, and in the latter case, the proposed solution required that the consensus update law be modified over time. In contrast, this paper will present a simple, static update law for achieving agreement while satisfying the constraints.

The outline of this paper is as follows: In Section 2, we introduce the networked pendulum-cart system, followed by a discussion in Section 3 about the optimal control design needed to synchronize the system if complete information (about all pendula) is available to each individual pendulum. In that section we also introduce our solution to the constrained agreement problem, followed by the simulation results, in Section 4.

II. PENDULUM DYNAMICS

A. Dynamics

The dynamics of a single cart-pendulum system (referred to as an agent) can be derived using Lagrange's Equations (e.g. [13])

$$\frac{d}{dt}(\frac{\partial \mathfrak{L}}{\partial \dot{q}}) - \frac{\partial \mathfrak{L}}{\partial q} = Q,\ \mathfrak{L} = T - P,$$

where T is the kinetic energy of the system, P is the potential energy of the pendulum, Q is the parameterized forces acting on the system, and $\mathfrak{L}$ is the Lagrangian.

We can define the kinetic and potential energy as well as the parameterized forces acting on the system. Based on Figure 2, the only parameterized force, Q, on the system is the force, F, applied in the P_x direction. This force will be the control input, u, to the system. No damping force is considered in this model as pendula can be approximated as

(a) Puppet in initial configuration. (b) Puppet in wave motion. (c) Puppet starting a walk. (d) The final step in the walk mode.

Fig. 1. An image sequence of the a robotic marionette executing a *wave* followed by a *walk* mode.

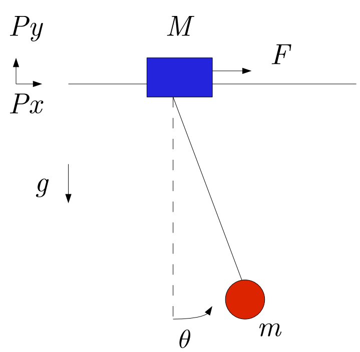


Fig. 2. Pendulum Diagram

zero damping systems. The resulting equations of motion are

$$\ddot{\theta} = -\frac{\ddot{P}_x}{l} cos(\theta) - \frac{g}{l} sin(\theta) \quad (1)$$

$$\ddot{P}_x = -\frac{ml\ddot{\theta}}{M+m} cos(\theta) + \frac{ml\dot{\theta}^2}{M+m} sin(\theta) + \frac{u}{M+m} \quad (2)$$

B. Linearization

These dynamics can be linearized about the $\theta = 0$, $\dot{\theta} = 0$, $\dot{P}_x = 0$ equilibrium point, giving the single pendulum system as $\dot{x}_i(t) = A_i x_i(t) + B_i u_i(t)$, where:

$$x_i = \begin{bmatrix} P_{x,i} \\ \dot{P}_{x,i} \\ \theta_i \\ \dot{\theta}_i \end{bmatrix}, A_i = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-(M+m)g}{Ml} & 0 \end{bmatrix}, B_i = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{-1}{Ml} \end{bmatrix}.$$

Note that this pair, (A_i, B_i), is completely controllable.

For a N planar pendulum system, the system can be written as $\dot{x}(t) = Ax(t) + Bu(t)$, where

$$x' = \begin{bmatrix} x'_1, & \dots, & x'_N \end{bmatrix}, u = \begin{bmatrix} u_1 & \dots & u_N \end{bmatrix}'$$

$$A = \begin{bmatrix} A_1 & & 0 \\ & \ddots & \\ 0 & & A_N \end{bmatrix}, B = \begin{bmatrix} B_1 & & 0 \\ & \ddots & \\ 0 & & B_N \end{bmatrix}.$$

Note that in this paper, $A_i = A_j$ and $B_i = B_j$ for $i, j = 1, \dots, N$, since the pendula are assumed to be homogeneous.

C. Assumptions

Throughout this paper, some assumptions will be made and here we gather the assumptions for the sake of easy reference. We first assume that each cart-pendulum system can measure its own cart position, cart velocity, pendulum angle, and pendulum angular velocity. It is also assumed that pendulum angles and angular velocities are small enough so that the linearized dynamics can be used to model the system behavior. Damping is also assumed to be small enough to approximate it as exerting zero forces on the system.

The network topology used in this paper is restricted to a static line graph topology. This assumption follows from the mechanical set-up of the robotic marionette. As such, it is also assumed that each agent can measure the state values of adjacent agents only. Adjacent agents can also communicate state estimates with each other.

III. Coordinated Synchronization Control

A. Two Pendula Control

Given a system comprised of two planar pendula, a control law is sought to drive these two pendula in such a way that they achieve identical angles, angular velocities, and cart velocities while maintaining a set distance, d, between the carts. Therefore, a control law is desired to enforce the terminal constraints $P_{x,1}(T) - P_{x,2}(T) = d$, $\dot{P}_{x,1}(T) - \dot{P}_{x,2}(T) = 0$, $\theta_1(T) - \theta_2(T) = 0$, $\dot{\theta}_1(T) - \dot{\theta}_2(T) = 0$, i.e. $Cx(T) = k$, where

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix},\ k = \begin{bmatrix} d \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

The associated minimum energy, point-to-point transfer control problem becomes

$$\min_u J(u(t)) = \int_0^T ||u(t)||^2 dt \tag{3}$$

such that

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{4}$$
$$x(0) = x_0 \tag{5}$$
$$x(T) = x_T, \tag{6}$$

where we assume that x_T satisfies the constraints, i.e. $Cx_T = k$. The solution to this optimal control problem is

$$u_{opt}(x_T) = B' e^{A'(T-t)} W^{-1}(x_T - e^{AT} x_0), \tag{7}$$

where the Grammiam, W, is invertible and positive definite due to the controllability of the system. Plugging $u_{opt}(x_T)$ back into the cost gives

$J(u_{opt}(x_T)) =$

$$= (e^{AT}x_0 - x_T)' e^{-A'T} W^{-1} e^{-AT} (e^{AT} x_0 - x_T). \tag{8}$$

Since x_T is not unique, the goal now is to find the x_T that minimizes (8), which can be formulated as a quadratic programming problem

$$\min_{x_T} \frac{1}{2} x_T' Q x_T + R x_T \tag{9}$$

such that $Cx_T = k$, where

$$Q = 2e^{-A'T} W^{-1} e^{-AT} \tag{10}$$
$$R = -2x_0' W^{-1} e^{-AT}. \tag{11}$$

The unique solution,

$$x_{T_{opt}} = Q^{-1}(-R' + C'(CQ^{-1}C')^{-1}(k + CQ^{-1}R'), \tag{12}$$

gives the control,

$$u_{opt}(x_{T_{opt}}) = B' e^{A'(T-t)} W^{-1}(x_{T_{opt}} - e^{AT} x_0). \tag{13}$$

It should be noted that the pendula converge to a set distance d apart and have equal angles and angular velocities. They also have equal cart velocities; however, these velocities are not guaranteed to be zero. The following proposition states that they converge to an average of the initial velocities.

Proposition 1: The final velocity of a linearized two pendulum-cart system (A, B), using point-to-point transfer optimal control with a terminal linear constraint $Cx_T = k$, is the average of the initial velocities.

Proof: Recall that for a two pendulum system,

$$A = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}, B = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix},$$

which implies that

$$e^{AT} = \begin{bmatrix} e^{A_1 T} & 0 \\ 0 & e^{A_2 T} \end{bmatrix} \tag{14}$$

is block diagonal. In fact,

$$e^{A_i T} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & cos(\omega t) & \frac{1}{\omega} sin(\omega t) \\ 0 & 0 & -\omega sin(\omega t) & cos(\omega t) \end{bmatrix}, \tag{15}$$

where ω is the natural frequency, $\sqrt{\frac{(M+m)g}{Ml}}$.

In fact, one can show that the controllability Grammiam, W, as well as Q, are block diagonal as well. From (12), we can write $x_{T_{opt}}$ as

$$x_{Topt} = (-Q^{-1} + Q^{-1}C'(CQ^{-1}C')^{-1}CQ^{-1}) \cdot$$
$$\cdot(-2e^{-A'T}W^{-1})x_0 + Q^{-1}C'(CQ^{-1}C')^{-1}k \tag{16}$$

by plugging in R.

Now, since $A_1 = A_2$ and $B_1 = B_2$, block diagonality implies (after some calculations) that

$$x_{T_{opt}} = \frac{1}{2} \begin{bmatrix} e^{A_1 T} & e^{A_1 T} \\ e^{A_1 T} & e^{A_1 T} \end{bmatrix} x_0 + \frac{1}{2} C' k. \tag{17}$$

Substituting the value for $e^{A_1 T}$ and k into (17) now gives that the cart velocity for each pendulum is equal to the average of the initial velocities, i.e.

$$\dot{P}_{x,i}(T) = \frac{1}{2}(\dot{P}_{x,1}(0) + \dot{P}_{x,2}(0)),\ i = 1, 2. \tag{18}$$

■

Now, this approach can of course be extended to the N pendula case as well. This direct extension, however, requires complete initial state information for the computation of the optimal terminal state. In the next section, we will see how to overcome this problem.

B. N Pendula Control with Consensus

As a consequence of the previous section, $x_{T_{opt}}$ needs to be specified in order to be able to compute the control law. But, what if this (global) terminal state is only partially known to the agents through a local estimate? The challenge, then, is agreeing on a global terminal state using these local estimates in a way such that the constraints are satisfied. To begin with, we will let each agent solve the constrained point-to-point transfer problem defined only over the adjacent agents. As a result, agent i will have obtained what it believes to be the best final states for itself as well as for its neighbors ($i-1$ and $i+1$ if $i = (2, \ldots, N-1)$, 2 if $i = 1$, and $N-1$ if $i = N$). In fact, we will denote by $X_{i,j}$, the terminal state value that agent i thinks the agent j should have, as the outcome of the optimal control problem. And, for the two pendula case, we observe that we can rewrite the constraint matrix C as $C = \begin{bmatrix} I & -I \end{bmatrix}$, which implies that $X_{i,j}$ has to satisfy

$$\begin{bmatrix} I & -I \end{bmatrix} \begin{bmatrix} X_{1,1} \\ X_{1,2} \end{bmatrix} = k, \quad (19)$$

$$\begin{bmatrix} I & -I \end{bmatrix} \begin{bmatrix} X_{N,N-1} \\ X_{N,N} \end{bmatrix} = k, \quad (20)$$

for $i = 2, \ldots, N$,

$$\begin{bmatrix} I & -I & 0 \\ 0 & I & -I \end{bmatrix} \begin{bmatrix} X_{i,i-1} \\ X_{i,i} \\ X_{i,i+1} \end{bmatrix} = \begin{bmatrix} k \\ k \end{bmatrix}. \quad (21)$$

Note that these constraints are only enforced locally, i.e. there is no reason to believe that $X_{i,j} = X_{j,i}$ or that they are globally satisfied, e.g through

$$M \begin{bmatrix} X_{1,1} \\ \vdots \\ X_{N,N} \end{bmatrix} = b, \quad (22)$$

$$M = \begin{bmatrix} I & -I & 0 & \ldots & \ldots & 0 \\ 0 & I & -I & 0 & \ldots & 0 \\ \vdots & & & \ddots & & \vdots \\ 0 & \ldots & \ldots & 0 & I & -I \end{bmatrix}, \; b = \begin{bmatrix} k \\ \vdots \\ k \end{bmatrix}.$$

The enabling observation now is that we can use the consensus equation to update each pendulum's estimates of the global final state. This way, a final state may be found that satisfies the constraint for the entire network. We use $\eta_{i,j}(t)$ to denote agent i's estimate of what agent j's terminal state should be, at time t. If we let N_i represent the set of neighbors to agent i, the consensus equation becomes

$$\dot{\eta}_{ij}(t) = -\sum_{k \varepsilon N_i} (\eta_{ij}(t) - \eta_{kj}(t)) \;\; for\ i = 1, \ldots, N. \quad (23)$$

The question now is what initial conditions to use? Since agent i only has access to $X_{i,i-1}$,$X_{i,i}$, and $X_{i,i+1}$ (provided $i = 2, \ldots, N-1$), it seems clear (but not actually correct as we will see) that $\eta_{i,j}(0) = X_{i,j}$ for $j = i-1, i, i+1$. But what about the other values? In fact, what we propose is to let the initial conditions be given by

(Boundary Pendulum: end of line topology)

$$\eta_{1,1}(0) = \gamma X_{1,1}, \quad (24)$$
$$\eta_{1,2}(0) = \gamma X_{1,2}, \quad (25)$$
$$\eta_{1,l}(0) = \eta_{1,l-1}(0) \text{ for } l = 3, \ldots, N, \quad (26)$$

$$\eta_{N,N}(0) = \gamma X_{N,N}, \quad (27)$$
$$\eta_{N,N-1}(0) = \gamma X_{N,N-1}, \quad (28)$$
$$\eta_{N,l}(0) = \eta_{N,l+1}(0) \text{ for } l = 1, \ldots, N-2, \quad (29)$$

(Non-Boundary Pendulum) for $i = 2, \ldots, N-1$

$$\eta_{i,i-1}(0) = \gamma X_{i,i-1}, \quad (30)$$
$$\eta_{i,i}(0) = \gamma X_{i,i}, \quad (31)$$
$$\eta_{i,i+1}(0) = \gamma X_{i,i+1}, \quad (32)$$
$$\eta_{i,l_1}(0) = \eta_{i,l_1+1}(0), \text{ for } l_1 = 1, \ldots, i-2, \quad (33)$$
$$\eta_{i,l_2}(0) = \eta_{i,l_2-1}(0), \text{ for } l_2 = i+2, \ldots, N. \quad (34)$$

Here, $\gamma \in \mathbb{R}$ is a constant that needs to be determined.

In other words, the final state estimates for agents not adjacent to a boundary pendulum are initialized to the adjacent agent's final state estimate. Final state estimates for agents not adjacent to non-boundary agents are initialized to the same final state estimate as the adjacent agent closest to it. The following notation will be used to denote all agents estimate of agent i.

$$\eta_i(t) = \begin{bmatrix} \eta_{1,i}(t) \\ \vdots \\ \eta_{N,i}(t) \end{bmatrix}. \quad (35)$$

It is well known that (23) is globally asymptotically stable for connected graphs (e.g. [14]) and that the agreement state, η_c is

$$\eta_c = \frac{1}{N} \begin{bmatrix} \mathbf{1}'\eta_1(0) \\ \vdots \\ \mathbf{1}'\eta_N(0) \end{bmatrix}, \quad (36)$$

where $\mathbf{1}$ represents a vector with 1's in each position. Since $\eta_c \in \mathbb{R}^N$, the plan now is to use it as the terminal state in the point-to-point transfer problem.

To verify that η_c does in fact satisfy the global constraint and to determine γ, plug η_c into the constraint (22) for $X_{i,j}$,

$$M\eta_c =$$
$$= \frac{1}{N} M \begin{bmatrix} \mathbf{1}'\eta_1(0) \\ \vdots \\ \mathbf{1}'\eta_N(0) \end{bmatrix}$$
$$= \frac{1}{N} \begin{bmatrix} \begin{bmatrix} I & -I \end{bmatrix} \begin{bmatrix} \mathbf{1}'\eta_1(0) \\ \mathbf{1}'\eta_2(0) \end{bmatrix} \\ \vdots \\ \begin{bmatrix} I & -I \end{bmatrix} \begin{bmatrix} \mathbf{1}'\eta_{N-1}(0) \\ \mathbf{1}'\eta_N(0) \end{bmatrix} \end{bmatrix}$$
$$= \frac{1}{N} \begin{bmatrix} (\mathbf{1}'\eta_1(0)) - (\mathbf{1}'\eta_2(0)) \\ \vdots \\ (\mathbf{1}'\eta_i(0)) - (\mathbf{1}'\eta_{i+1}(0)) \\ \vdots \\ (\mathbf{1}'\eta_{N-1}(0)) - (\mathbf{1}'\eta_N(0)) \end{bmatrix}$$
$$= \frac{1}{N} \begin{bmatrix} \gamma k + \gamma k + (\eta_{3,1}(0) - \eta_{3,2}(0)) + \\ \cdots + (\eta_{N,1}(0) - \eta_{N,2}(0)) \\ \vdots \\ (\eta_{1i}(0) - \eta_{1,i+1}(0)) + \ldots \\ +(\eta_{i-1,i}(0) - \eta_{i-1,i+1}(0)) + \\ +\gamma k + \gamma k + \\ +(\eta_{i+2,i}(0) - \eta_{i+2,i+1}(0)) + \\ \cdots + (\eta_{N,i}(0) - \eta_{N,i+1}(0)) \\ \vdots \\ (\eta_{1,N-1}(0) - \eta_{1,N}(0)) + \ldots \\ +(\eta_{N-2,N-1}(0) - \eta_{N-2,N}(0)) + \\ \gamma k + \gamma k \end{bmatrix}.$$

We now note that for agent j not adjacent to agent i, $\eta_{j,i}$ was chosen in (24)-(34) to equal $\eta_{j,i+1}$, so the terms $(\eta_{j,i} - \eta_{j,i+1})$ above equal zero. Also note that for agent j adjacent to agent i, the terms $(\eta_{j,i} - \eta_{j,i+1})$ above equal γk. Hence,

$$M\eta_c = \frac{1}{N}\begin{bmatrix}\gamma k + \gamma k\\ \vdots \\ \gamma k + \gamma k \\ \vdots \\ \gamma k + \gamma k\end{bmatrix} = \frac{2}{N}\begin{bmatrix}\gamma k\\ \vdots \\ \gamma k \\ \vdots \\ \gamma k\end{bmatrix} = \begin{bmatrix}k\\ \vdots \\ k \\ \vdots \\ k\end{bmatrix} = b, \qquad (37)$$

by setting $\gamma = \frac{N}{2}$.

Therefore, all the pendula meet the global terminal constraint for the entire network. The consensus equation updates the global terminal state estimate for each pendulum, which then can extract the terminal state estimates of itself and the adjacent agents, χ_i, by letting

(Boundary Pendulum: end of line topology)

$$\chi_1 = \begin{bmatrix}\eta_{11}(t)\\ \eta_{12}(t)\end{bmatrix}, \qquad (38)$$

$$\chi_N = \begin{bmatrix}\eta_{N,N-1}(t)\\ \eta_{N,N}(t)\end{bmatrix}, \qquad (39)$$

(Non-Boundary Pendulum) for $i = 2, \ldots, N-1$

$$\chi_i = \begin{bmatrix}\eta_{i,i-1}(t)\\ \eta_{i,i}(t)\\ \eta_{i,i+1}(t)\end{bmatrix}. \qquad (40)$$

Now, each agent n can calculate its control at time t, for $n = 1, \ldots, N$,

$$u_n(\chi_n(t)) = B_n' e^{A_n'(T-t)} W_n(t)^{-1} (\chi_n(t) - e^{A_n T} x_n(t)). \quad (41)$$

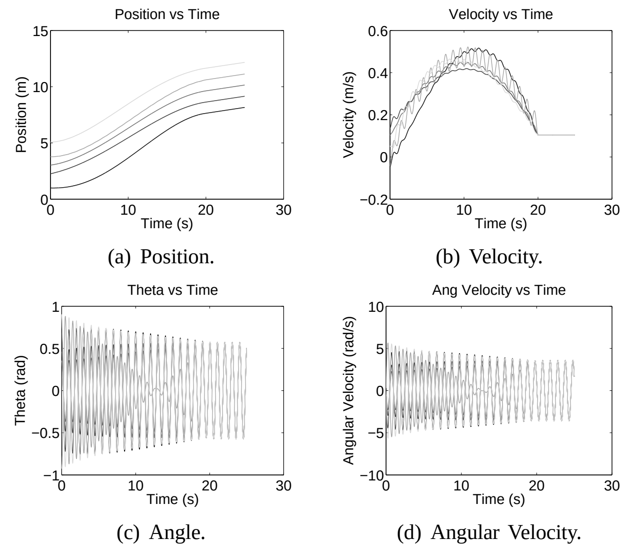
Here, A_n and B_n are the corresponding two or three pendula state space models, depending if it is a boundary pendulum or not. $W_n(t)$ is the Grammian for the pair (A_n, B_n) from time t to T. $x_n(t)$ is the current state of the agent n and its adjacent agents. We now have a control that drives the entire network to a terminal state that satisfies the terminal constraint.

IV. Simulation Results

The stated control laws are implemented in a MATLAB simulation of the presented pendulum dynamics. Simulations are run with the following parameters: $g = 9.8\ m/s,\ l = 0.30\ m,\ M = 1\ kg,\ m = 0.2\ kg$, and $d = 1.0\ m$ for the pendulum model. It should be noted that in order for this distributed control strategy to be effective, the consensus algorithm must converge to the agreement value before the specified final time in the optimal control law, which in this case is 20 seconds.

In Figure 3, the results are shown for a five pendula scenario using the optimal control law and the consensus equation. As a comparision, the same initial conditions are run for the centralized case, where full network state information is known to all agents, i.e. only the optimal control law is needed without the consensus equation. The results of this case are shown in Figure 4.

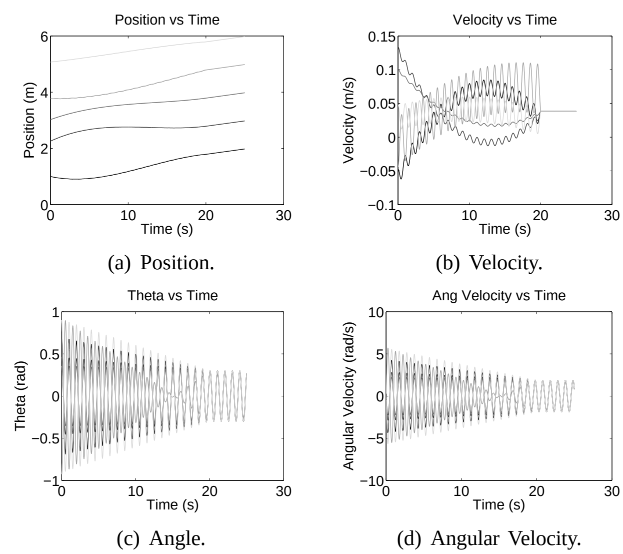
It can be seen for both cases that at 20 seconds, the distance between adjacent pendula is close to $1\ m$, as prescribed, while the velocities, angles, and angular velocities are identical for all the pendula. The animations of these scenarios are given in Figure 5.



(a) Position. (b) Velocity.

(c) Angle. (d) Angular Velocity.

Fig. 3. 5 Distributed Pendula Result.



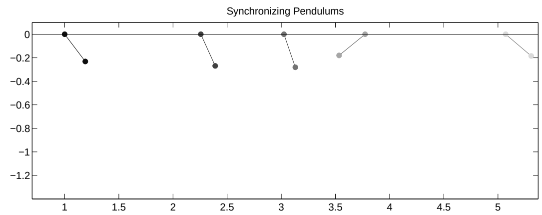
(a) Position. (b) Velocity.

(c) Angle. (d) Angular Velocity.

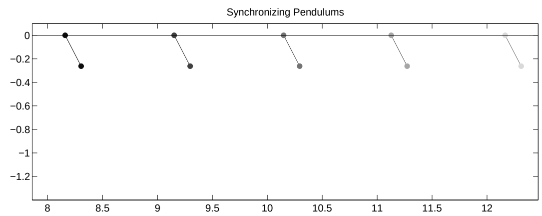
Fig. 4. 5 Centralized Pendula Result.

V. Conclusions

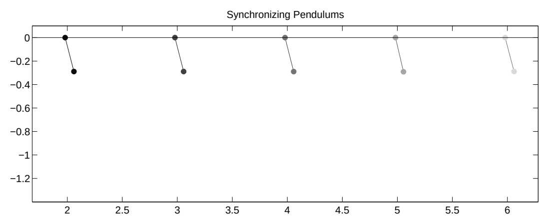
This paper demonstrates how the initial conditions in the agreement protocol can be manipulated to produce results that satisfy linear constraints. This technique was applied to the control of a distributed network of linearized pendula on a line graph topology. We found that this algorithm, together with point-to-point transfer optimal control, was able to drive these linear systems to a terminal manifold. This manifold



(a) Initial Condition.



(b) Distributed case After 25s.



(c) Centralized case After 25s.

Fig. 5. 5 Pendula Animation.

synchronizes the pendula oscillations and maintains a desired cart formation.

Acknowledgments

This work was supported by the US National Science Foundation through Grant # 0820004.

References

[1] M. Egerstedt, T.Murphey, and J. Ludwig, “Motion programs for puppet choreography and control,” in *Hybrid Systems: Computation and Control*. Pisa, Italy: Springer-Verlag, April 2007, pp. 190–202.

[2] P. Martin and M. Egerstedt, “Switched systems with applications to robotic marionettes,” in *Workshop on Discrete Event Systems*, Gothenburg, Sweden, May 2008.

[3] R. Olfati-Saber and R. M. Murray, “Agreement problems in networks with directed graphs and switching toplogy,” in *Proc. 42nd IEEE Conf. Decision Control*, vol. 4, Maui, HI, Dec 2003, pp. 4126–4132.

[4] A. Jadbabaie, J. Lin, and A. S. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, June 2003.

[5] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, “Gossip algorithms: Design, analysis, and applications,” in *Proceedings IEEE Infocomm*, vol. 3, Miami, March 2005, pp. 1653–1664.

[6] L. Xiao, S. Boyd, and S. Lall., “A scheme for robust distributed sensor fusion based on average consensus,” in *International Conference on Information Processing in Sensor Networks*, Los Angeles, 2005, pp. 63–70.

[7] H. Tanner, A. Jadbabaie, and G. Pappas, “Coordination of multiple autonomous vehicles,” in *Proceedings of the 11th IEEE Mediterranean Conference on Control and Automation*, Rhodes, Greece, June 2000.

[8] W. Ren and R. Beard, “Consensus of information under dynamically changing interaction topologies,” in *Proc. Am. Control Conf.*, vol. 6, Jun. 30-Jul. 2, 2004, pp. 4939–4944.

[9] D. Dimarogonas and K. Kyriakopoulos, “Decentralized navigation functions for multiple robotic agents with limited sensing capabilities,” *Journal of Intelligent and Robotic Systems*, vol. 48, no. 3, pp. 411–433, March 2007.

[10] A. Jadbabaie, N. Motee, and M. Barahona, “On the stability of the kuramoto model of coupled nonlinear oscillators,” in *Proc. American Control Conference*, vol. 5, 2004, pp. 4296–4301 vol.5.

[11] K. L. Moore and D. Lucarelli, “Forced and constrained consensus among cooperating agents,” in *Proc. IEEE Networking, Sensing and Control*, 19–22 March 2005, pp. 449–454.

[12] A. Nedic, A. Ozdaglar, and P. A. Parrilo, “Constrained consensus,” LIDS, Tech. Rep. 2779, Feb. 2008.

[13] M. W. Spong, *Robot Dynamics and Control*. New York, NY, USA: John Wiley & Sons, Inc., 1989.

[14] E. A. Wei Ren, “Distributed multi-vehicle coordinated control via local information exchange,” *International Journal of Robust and Nonlinear Control*, vol. 17, pp. 1002–1033, 2007.