

Adaptive Backstepping Synchronization for Networked Lagrangian Systems

Yassine Bouteraa

Jawhar Ghommam

G rard Poisson

ABSTRACT

A decentralized control law using a backstepping scheme is proposed to deal with a leader follower multiple robots structure. Based on graph theory and Laplacian, the coordination strategy combines the leader follower control with the decentralized control. In fact, in the proposed approach, each follower robot only needs the information exchange with its connected neighbors and does not assume the existence of direct communication channel with the leader, which mitigate the implementation cost, mainly when the cooperative robots number increases. The objective of the designed control law is to synchronize positions and velocities of multiple followers interconnected via the neighbor-based rule with respect to the leader's state. The proposed strategy is further extended to adaptive neural network coordination. The performance of synchronization control system and the nonlinear stability are derived by Lyapunov method. Simulations were performed to demonstrate the effectiveness of the proposed synchronization control approach.

General Terms

Synchronization and cooperative robot control.

Keywords

Cooperative control, adaptive mutual synchronization, robots coordination, graph theory, backstepping control.

1. INTRODUCTION

Recently, coordination control and consensus problems have been extensively studied in the area of distributed and decentralized networked systems. As one of the research topics, the coordination control of multiple robots has attracted much attention with applications in mobile robots [2], [21], robot manipulators [27], [6], [28], unmanned aerial vehicles [23], autonomous underwater vehicles [9], aircraft [30], Lagrangian systems in general [3], [1]. Multi-agent systems based on first-order consensus algorithms have attracted intensive attention in the literature [11], [17]. However, due to the broad class of agents should be described by second-order dynamic, the consensus problem modeled by double integrator is more challenging [15], [10]. The coordination of multi-robot systems and consensus problems are related with the synchronization problem which uses a graph theory [26]. According to Rodriguez [27], based on the type of interconnections in the system, different kinds of synchronization can be defined: mutual synchronization, when all agents occur on equal term, cooperative systems case [27], and external synchronization, when one object is more powerful than the others and its motion can be considered as independent of the motion of the other objects, master-slave systems case [31]. It is noted that there have been numerous investigations in multi-agent networks

with leader-follower structure [29], [18], [14], [25], [19], [11], [20]. An event-based motion reference is designed to drive the multi-robot system to achieve the best possible coordination [34]. However, the used approach in [34] is classified as centralized control. Most of earlier works in leader-follower multi-agent coordination approach require each follower to have the information access to the leader [31], [32], [33]. As a result, the network implementation becomes more complicated: costly and less robust. Notable studies demonstrate that the nearest neighbor can cause all agents to eventually move in the same direction despite the absence of centralized coordination [29], [7]. Recent researches propose approaches which each agent follower is solely based in the information of connected neighbors [4], [8]. To combine the leader-follower control with the decentralized control is the most practical model that benefits in the same time of the simplicity, efficiency and robustness.

This paper focuses on designing decentralized law to control a team of fully actuated manipulators in order to synchronize their movements while following a common desired trajectory, under a leader-follower structure. The proposed approach, based on combination of backstepping technique and graph theory control, works to synchronize positions and velocities of multiple followers interconnected via the neighbor-based rule with respect to the leader's state (see Fig.1).

In contrast of most previous works which deal with the first simple dynamic models such as linear systems and first-order second-order dynamics without non-linearly inertia matrices, the proposed strategy deals with highly nonlinear systems such as lagrangian systems.

Likewise, earlier works in the field of consensus and multi-agent cooperation using graph theory and laplacien [29], [15], [20] do not address the multi-agent systems where there is a desired path to follow and they have been just to settle for reach the weighted average of initial conditions.

A simple comparative with [27], [31] and [5], we note that the proposed contribution managed to eliminate the all-to-all coupling and consequently decrease the number of variables to manage in network.

Compared with classical leader follower structure [31], [22] our contributed work benefits of the neighbor-based rule to design a decentralized leader-follower structure such as the network do not requires direct communication channels between the leader and all followers. Also, the proposed topology permits a follower-follower data exchange. As result, network design less costly and more robust.

To meet the practical constraints, the case of the uncertainty parameters is studied. Compared with all the cited references that deal with the standard adaptive

synchronization [28], [3] the proposed control law uses the adaptive neural network that essentially avoids the computation of regression matrices and the tedium of the analysis.

This paper is organized as follows: In Section 2, the modeling multi-robot network is presented an overview of multi-agent communication and the problem formulation are discussed. In section 3, non-adaptive algorithm for trajectory tracking and synchronization control based on backstepping technique and graph theory is presented. Then, in order to deal with the presence of parametric modeling uncertainties, the control law has been extended to adaptive controller based on neural network algorithm. Finally, the simulation result section and the conclusion section follow, respectively.

2. PROBLEM STATEMENT AND PRELIMINARIES

2.1 Robot Modeling

This section describes a dynamic model of fully actuated Lagrangian systems that represents a typical class of robotic systems, such as ground vehicles, aircrafts, and robot arms. Following [16], the n degree-of-freedom robot manipulator composed of rigid bodies is expressed based on Newton's and Euler's equations as follows:

$$M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) = \tau_i \quad (1)$$

Where, $q_i \in \mathbb{R}^n$, denotes the joint angles of the i th manipulator, $\dot{q}_i \in \mathbb{R}^n$ and $\ddot{q}_i \in \mathbb{R}^n$ are the vectors of joint velocity and joint acceleration, respectively.

$M_i(q_i) \in \mathbb{R}^{n \times n}$ represents inertia matrix which is symmetric uniformly bounded and positive definite $C_i(q_i, \dot{q}_i)\dot{q}_i \in \mathbb{R}^{n \times n}$ is a vector function containing coriolis and centrifugal forces, $g_i(q_i) \in \mathbb{R}^{n \times n}$ is a vector function consisting of gravitational forces and τ_i is the vector function consisting of applied generalized torques.

2.2 Problem formulation

Each robot manipulator hence forth called agent has a complete knowledge of its state and the states of some other neighbors through sensors communication. In this regard and in order to simplify the intercommunication process, we consider each robot as a node and the communication between all nodes is represented by an indirected graph. Let $G = (V, E)$ is a digraph with N nodes, the set nodes $V = 1, 2, \dots, m$ and edges $E \subseteq V \times V$. Each node is labeled by $v_i \in V$ and each edge is denoted by $e_{ij} = (v_i, v_j)$. The neighbors of agent v_i are denoted by $N = \{v_j \in V / (v_i, v_j) \in E\}$. The adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{m \times m}$ of a weighted digraph is defined as:

$$\begin{cases} a_{ij} > 0 & \text{if } (v_i, v_j) \in E \\ a_{ij} = 0 & \text{elsewhere} \end{cases}$$

Agent i communicates with agent j if j is a neighbor of i or if $a_{ij} = 0$. Note that an edge e_{ij} in a directed graph means that robot j can reach information from robot i , but not necessarily vice versa. In contrast, in an undirected graph, pairs of node are unordered and an edge e_{ij} implies that robots i and j can get information from one another. The adjacency matrix of an undirected graph has the same meaning as that of the directed graph except that $a_{ij} = a_{ji}$. The degree matrix of the

digraph $G = (V, E)$ is a diagonal matrix defined as:

$$d_{ij} = \begin{cases} \text{deg}_{\text{out}}(v_i) & \text{if } (i, j) \in E \\ 0 & \text{otherwise} \end{cases}$$

where $\text{deg}_{\text{out}}(v_i) = \sum_{j=1}^n a_{ij}$

The graph laplacian of G is defined as: $L = D - A$, where $D = [d_{ii}]$ the degree matrix of G . In the undirected graph case, L is symmetric positive semidefinite.

In the proposed strategy, we consider a multi-robot system formed by $m + 1$ rigid joint robot manipulators, such that the motion of one of the manipulators is independent of the other ones. This dominant agent will be referred to as the leader manipulator. The leader will be indexed by 0 and the follower set will be indexed by $1, 2, \dots, m$. In contrast of the leader independent motion, the follower manipulator motion is influenced by the leader and the other followers. To study the leader follower cooperation we define a graph topology G consisting of m robot manipulators as followers and one robot manipulator as a network leader. We define b_i ($i = 1, 2, \dots, m$) as positive coefficients that give insight into the communication between the i th follower and the leader. In the case where we have not an active data exchange between the i th follower and the network leader, a zero will be assigned to the value of b_i . In the present topology, the edge represents bidirectional communication links. This consists on a group of m manipulators interchanging information that can be viewed as an undirected graph (see Fig.1 and Fig.2).

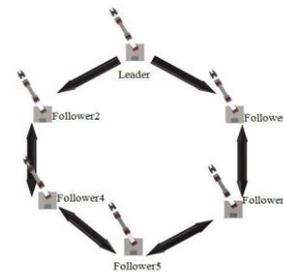


Figure 1: Leader-follower network topology.

3. CONTROL DESIGN

3.1 Non-adaptive backstepping approach

Starting with the state-space representation of the i th robot dynamic system (1):

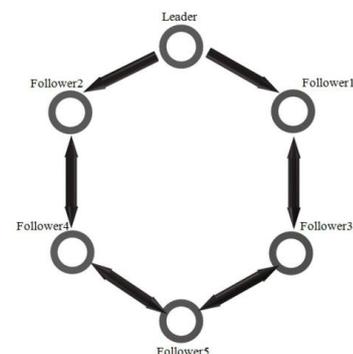


Figure 2: Leader-follower network graph.

$$\begin{cases} \dot{q}_{1i} = q_{2i} \\ M\dot{q}_{2i} = \tau - C(q_1, q_2)q_2 + g(q_1) \end{cases}$$

The philosophy in the backstepping method is to set q_{2i} as a virtual controller, which is responsible to synchronize the state q_{1i} with the leader state q_0 . Based in the multi-agent theory we define the following form:

$$u_i = \sum_{j \neq i} A_{ij}(q_{1j} - q_{1i}) + B_i(q_0 - q_{1i}) \quad (2)$$

where $A_{ij} = \text{diag}(a_{ij})_{m \times m}$; a_{ij} defines the i th row and the j th column element in the graph adjacency matrix A ; $B_i = \text{diag}(b_i)_{m \times m}$. Suppose that q_{2i} is the control to synchronize the state q_{1i} with the leader and k is positive scalar, we choose

$$q_{2i} = k u_i + \dot{q}_0 \quad (3)$$

Considering equation 6, equation 7 can be written in the compact form as: (3) considering equation 6, equation 7 can be written in the compact form as:

$$\dot{q}_1 - \dot{q}_0 \mathbf{1} = -k[(L + B) \otimes I_n](q_1 - q_0 \mathbf{1}) \quad (4)$$

Or

$$\dot{q}_1 - \dot{q}_0 \mathbf{1} = -k[(L + B) \otimes I_n](q_1 - q_0 \mathbf{1}) \quad (5)$$

Where $\dot{q}_1 = (\dot{q}_{11}^T, \dots, \dot{q}_{1i}^T, \dots, \dot{q}_{1m}^T)^T \in \mathbb{R}^{m \times n}$; $q_2 = (q_{21}^T, \dots, q_{2i}^T, \dots, q_{2m}^T)^T \in \mathbb{R}^{m \times n}$; $\mathbf{1} = (1, 1, \dots, 1)^T \in \mathbb{R}^n$; \otimes denotes the Kronecker product and I_n denotes the $n \times n$ identity matrix.

The objective is to design a torque controller τ_i which derives q_{1i} to q_0 for $1 \leq i \leq m$, i.e, $\lim_{t \rightarrow \infty} q_{1i} \rightarrow q_0$ for $t \rightarrow \infty$. To this end, we define the error signal for the i th robot as:

$$e_i = q_{1i} - q_0 \quad (6)$$

Since q_{2i} is not the real control for q_{1i} , we consider it as a virtual control and we propose the following error variable as:

$$z_i = q_{2i} - \alpha_i \quad (7)$$

with α_i being the virtual control for q_{1i} such that

$$\alpha_i = k u_i + \dot{q}_0 \quad (8)$$

Design the torque inputs for the i th robot manipulator in network comprised of m robots ($m \geq 2$).

$$\begin{aligned} \tau_i = & C_i(q_{1i}, q_{2i})q_{2i} + g(q_{1i}) - \frac{1}{2}kz_i - p e_i \\ & + M_i \left(\sum_{j \neq i} \frac{\partial \alpha_i}{\partial q_{ij}} (q_{2i} - q_{2j}) + \frac{\partial \alpha_i}{\partial q_{1i}} q_{2i} + \frac{\partial \alpha_i}{\partial q_0} \dot{q}_0 \right) \end{aligned} \quad (9)$$

where k and p are positive scalars.

Remark 1: It is noted that the proposed controller needs the velocity \dot{q}_0 and acceleration \ddot{q}_0 of the leader manipulator. Therefore, this information can be accessible to each follower manipulator in the controller design phase. It is also a common requirement in the existing literature [13] and [29].

Remark 2: With quick look in the controller expression, we can distinguish that the proposed control law only uses the information from those neighbors with which is connected. This aspect that reflects the decentralized control limits the communication flow. As result, an efficient and optimized network with a low cost.

Theorem 3.1 (Synchronized Trajectory Tracking Law) Consider a multi-robot system formed by $(m + 1)$ rigid joint

robot manipulators coupled under leader-followers topology (Fig.1). The proposed nonlinear control law (9) synchronizes all the follower manipulators' joints to the joint trajectory of the leader manipulator.

Proof Starting with equation 4, form [13] $(L + B)$ is a nonsingular M -matrix, then, for $P > 0$ and $Q > 0$ we have

$$Q = P(L + B) + (L + B)^T P \quad (10)$$

To show that for each $1 \leq i \leq m$, $q_i \rightarrow q_0$, we consider the global system and we propose the following lyapunov function as:

$$V = e^T (P \otimes I_n) e \quad (11)$$

with $e = q_1 - q_0 \mathbf{1}$ is the error signal in the compact form. The time derivative of V with respect to time gives

$$\dot{V} = [e^T (P \otimes I_n) \dot{e} + \dot{e}^T (P \otimes I_n) e] \quad (12)$$

Taking into account equation 5, then equation 12 can be written

$$\begin{aligned} \dot{V} = & -e^T (P \otimes I_n) [(L + B) \otimes I_n] e + -e^T [(L + B) \otimes I_n]^T (P \otimes I_n) e \\ = & -e^T [(P \otimes I_n) [(L + B) \otimes I_n] + [(L + B) \otimes I_n]^T (P \otimes I_n)] e \end{aligned} \quad (13)$$

This yields that

$$\dot{V} = -e^T (Q \otimes I_n) e \quad (14)$$

This shows that V is negative semi definite and therefore the origin $e = 0$ is asymptotically stable and $e \rightarrow 0$ as $t \rightarrow \infty$ i.e, $q_i \rightarrow q_0 \forall i \in [1, m]$. Since q_2 is not the real control for q_1 , we consider it as a virtual control and we propose the following error variable z as defined in (7). In a compact form the equation (7) can be written as:

$$z = q_2 - \alpha \quad (15)$$

with $z = [z_1^T, \dots, z_m^T]^T$, $q_2 = [q_{21}^T, \dots, q_{2m}^T]^T$, $\alpha = [\alpha_1^T, \dots, \alpha_m^T]^T$, therefore representation in the compact form of (8) such as

$$\alpha = k u + \dot{q}_0 \mathbf{1} \quad (16)$$

To find the real control input, we proceed by two steps of backstepping design procedure:

First step: Let the first control lyapunov function be:

$$V_1 = e^T P \quad (17)$$

The time derivative of V_1 becomes

$$\dot{V}_1 = 2e^T P z + [e^T P (\alpha - \dot{q}_0) + (\alpha - \dot{q}_0)^T P e] \quad (18)$$

A straightforward computation gives us:

$$\dot{V}_1 = 2e^T P z - e^T k Q \otimes I_n e \quad (19)$$

The first term of V_1 will be dealt in the next step of the backstepping procedure.

Second step: Differentiating z with respect to time gives

$$\dot{z} = \dot{q}_2 - \dot{\alpha} \quad (20)$$

Post multiplying by M both sides gives

$$M\dot{z} = M\dot{q}_2 - M\dot{\alpha}$$

Taking into account of the representation state of the i th robot dynamic system, the previous equation can be written as

$$M \dot{z} = \tau - C(q_1, q_2)q_2 - g(q_1) - M \left(\sum_{j \neq i} \frac{\partial \alpha}{\partial q_{1ij}} \dot{q}_{1ij} + \frac{\partial \alpha}{\partial q_{1i}} \dot{q}_{1i} + \frac{\partial \alpha}{\partial q_0} \dot{q}_0 \right) \quad (21)$$

Further calculation, will result in

$$M \dot{z} = \tau - C(q_1, q_2)q_2 - g(q_1) - M \left(\sum_{j \neq i} \frac{\partial \alpha}{\partial q_{1ij}} (q_{2i} - q_{2j}) + \frac{\partial \alpha}{\partial q_{1i}} q_2 + \frac{\partial \alpha}{\partial q_0} \dot{q}_0 \right) \quad (22)$$

Select the control for τ as equation (9). Let the second control lyapunov function

$$V_2 = V_1 + z^T M z \quad (23)$$

The time derivative of V_2 with respect to time gives

$$\begin{aligned} \dot{V}_2 &= -e^T Q e + 2e^T P z + 2z^T M \dot{z} \\ &= -e^T Q e - z^T k z + 2e^T P z - 2z^T p e \\ &= -e^T Q e - z^T k z \end{aligned} \quad (24)$$

This proves that V_2 is negative definite. Consequently we show the asymptotic convergence to zero of both errors e as z .

3.2 Adaptive neural network approach

In the first proposed study, the aforementioned schemes are based on the exact knowledge of the system, i.e., $M(q)$, $C(q, \dot{q})$ and $g(q)$ are supposed to be known. However for practical applications the robot network system are confronted with parameter uncertainties. Thus, if these matrices were unknown the control input (9) wouldn't be implementable. Therefore, in this section, we discuss the challenge of uncertainty in physical parameters of robots. To deal with this hypothesis, we can proceed in two ways. The first approach is to use the standard (conventional) adaptive control. While the second approach is to use a Radial Basis Function Neural Networks (RBFNN). What is motivating for the second approach is that it prevents the tedium of the analysis and computation of regression matrices in which their complexities depend mainly on the complexity of the system studied. Thus, in the proposed work, the RBFNN are proposed to model the unknown nonlinear dynamics of the networked system.

3.2.1 Neural Network background

Consider the multilayer neural network architecture presented in Fig.3. The output of the multilayer perception is as follows:

$$Y_i = \sum_{j=1}^{N_h} \left[w_{ij} \sigma \left(\sum_{k=1}^{N_i} v_{ik} x_k + \theta v_j \right) + \theta_{wi} \right] \quad (25)$$

Where $\sigma(\cdot)$ the activation is function; N_h is the number of hidden-layer neurons. N_i and N_0 are regarded as the number of input-layer and output-layer neurons, successively. v_{ik} represents the first-to-second-layer interconnections weights, w_{ij} represents the second-to-third-layer interconnection weights and x_k is the k th input to the neural network. θv_j and θ_{wi} denote the threshold offsets. Collecting all the neural network v_{ik} and w_{ij} into matrices V^T and W^T .

Then equation (25) can be written in terms of vector as:

$$Y = W^T \sigma(V^T x) \quad (26)$$

$Y(x)$ is considered a general smooth function from \mathbb{R}^{N_i} to \mathbb{R}^{N_0} that represent the unstructured uncertainties and it can be written as:

$$Y = W^T \phi(x) + \epsilon \quad (27)$$

where ϕ is the basis functions and ϵ is a reconstruction error vector such that

$\|\epsilon\| < \epsilon_M$ for all x

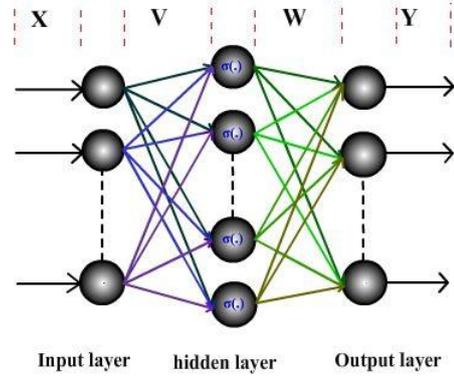


Figure 3: Schematic diagram of RBF Neural Network.

3.2.2 Neural Network Controller Design

The objective is to develop an adaptive neural network trajectory-tracking control law to synchronize multiple robot manipulators. Thus, assume that:

$$H = C(q_1, q_2)q_2 - g(q_1) + M \left(\sum_{j \neq i} \frac{\partial \alpha}{\partial q_{1ij}} \dot{q}_{1ij} + \frac{\partial \alpha}{\partial q_{1i}} \dot{q}_{1i} + \frac{\partial \alpha}{\partial q_0} \dot{q}_0 \right) \quad (28)$$

is unknown. According to the general approximation property of the RBFNNs, the global node nonlinearity for H can be written as:

$$H = \theta^T \varphi + \epsilon \quad (29)$$

where $\theta \in \mathbb{R}^{N_h \times N_0}$ is a vector of the output layer weights as a collection of NN

weights; $\varphi \in \mathbb{R}^{N_h}$ is the vector of output of hidden layer nodes called Radial

Basis Functions (RBF); ϵ is the neural network approximation error and N_h is the number of nodes in the hidden layer. The estimate of this nonlinear function is obtained as the output of the RBFNN and given by:

$$\hat{H} = \hat{\theta}^T \varphi(q_1, q_2, q_0, \dot{q}_{ij}) \quad (30)$$

where $\hat{\theta}$ is a current estimate of the neural network weights for which we propose to determine an update law.

Therefore, taking into account (28) – (29) – (30), the developed controller (9) takes the following adaptive version:

$$\tau = \hat{\theta}^T \varphi - P e - \frac{K_z}{2} z \quad (31)$$

Theorem 3.2 (Adaptive Synchronized Trajectory Tracking Law)

The developed adaptive control law (31) solves position synchronization to trajectory tracking of multiple robot-manipulators under leader-followers topology in presence of parametric uncertainties.

Proof let the new Lyapunov function will be of the form:

$$V_3 = V_2 + \frac{1}{2} \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \tilde{\theta}\} \quad (32)$$

where $\tilde{\theta} = \theta - \hat{\theta}$; Γ is a diagonal positive-definite control gain. The time derivative of V_3 is:

$$\dot{V}_3 = -e^T Q e + 2e^T P z + 2z^T M \dot{z} + \text{trace}\{\tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}}\} \quad (33)$$

$$\begin{aligned}
&= 2z^T \left[\tau - C(q_1, q_2)q_2 - g(q_1) \right. \\
&\quad \left. - M \left(\sum_{j \neq i} \frac{\partial \alpha}{\partial q_{1ij}} (q_{2i} - q_{2j}) + \frac{\partial \alpha}{\partial q_{1i}} q_2 + \frac{\partial \alpha}{\partial q_0} \dot{q}_0 \right) \right] - e^T Q e \\
&\quad + 2e^T P z + \text{trace} \{ \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} \} \\
&= 2z^T (\tau - H) - e^T Q e + 2e^T P z + \text{trace} \{ \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} \} \\
&= 2z^T (\tau - \theta^T \varphi(q_1, q_2, q_0, q_{ij}) - \epsilon) - e^T Q e + 2e^T P z + \\
&\quad \text{trace} \{ \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} \} \quad (33)
\end{aligned}$$

Let

$$\tau = \hat{\theta}^T \varphi - P e - \frac{K_z}{2} z$$

Therefore

$$\begin{aligned}
\dot{v}_3 &= 2z^T (-\hat{\theta}^T \varphi - P e - \epsilon) - e^T Q e + 2e^T P z + \\
&\quad \text{trace} \{ \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} \} - z^T K_z z \\
&= -e^T Q e - 2z^T \epsilon - 2z^T \hat{\theta}^T \varphi \\
&\quad - z^T k_z z + \text{trace} \{ \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} \} \quad (35)
\end{aligned}$$

Then, we can written that

$$\begin{aligned}
\dot{v}_3 &\leq -e^T Q e - 2z^T \epsilon \\
&\quad - z^T k_z z + \text{trace} \{ \tilde{\theta}^T (\Gamma^{-1} \dot{\tilde{\theta}} + 2\varphi z^T) \} \quad (36)
\end{aligned}$$

Choose

$$\dot{\tilde{\theta}} = -\Gamma(2\varphi z^T - k\tilde{\theta}) \quad (37)$$

Results in

$$\begin{aligned}
\dot{v}_3 &\leq -e^T Q e - 2z^T \epsilon - z^T k_z z + \text{trace} \{ \tilde{\theta}^T (\Gamma^{-1} (-2\Gamma\varphi z^T + \\
&\quad k\tilde{\theta}) + 2\varphi z^T) \} \quad (36)
\end{aligned}$$

This yield

$$\dot{v}_3 \leq -e^T Q e - 2z^T \epsilon - z^T k_z z + k \text{trace} \{ \tilde{\theta}^T (\theta - \tilde{\theta}) \}$$

For $\epsilon > 0$, the Neural Network approximation error is bounded by $\|\epsilon\| \leq \epsilon_M$. This gives

$$\begin{aligned}
\dot{v}_3 &\leq -\lambda_{\min}(Q) \|e\|^2 - 2\|z\| \epsilon_M - k\theta_M \|\tilde{\theta}\|_F - k \|\tilde{\theta}\|_F^2 - \\
&\quad \lambda_{\min}(k_z) \|z\|^2 \quad (40)
\end{aligned}$$

$$\text{Where } \|\tilde{\theta}\|_F = \sqrt{\text{trace} \{ \tilde{\theta}^T \theta \}}.$$

The equation (40) can be written in a more compact form as :

$$\begin{aligned}
\dot{v}_3 &\leq \begin{pmatrix} \|e\| \\ \|\tilde{\theta}\|_F \end{pmatrix}^T \begin{pmatrix} \frac{\lambda_{\min}(Q)}{2} & -\frac{1}{2} k\theta_M \\ -\frac{1}{2} k\theta_M & \frac{\lambda_{\min}(Q)}{2} \end{pmatrix} \begin{pmatrix} \|e\| \\ \|\tilde{\theta}\|_F \end{pmatrix} - \\
&\quad \lambda_{\min}(k_z) \|z\|^2 - 2\|z\| \epsilon_M \quad (41)
\end{aligned}$$

This proves that V_3 is negative definite.

4. SIMULATION RESULTS

To show the feasibility and performance of the proposed adaptive neural network based on backstepping design, a simulation study has been performed for a group of two-link full actuated robot manipulators.

The dynamic equations of the two-link manipulator system, used extensively in the literature, are:

$$\begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \begin{pmatrix} \ddot{q}_{11} \\ \ddot{q}_{12} \end{pmatrix} + \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} \begin{pmatrix} \dot{q}_{11} \\ \dot{q}_{12} \end{pmatrix} + \begin{pmatrix} g_1 \\ g_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} \quad (42)$$

where

$$M_{11} = m_2 l_2^2 + m_2 l_1 l_2 \cos q_2$$

$$M_{12} = (m_1 + m_2) l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 \cos q_2$$

$$M_{21} = M_{12}$$

$$M_{22} = m_2 l_2^2$$

The robot physical parameters are chosen as follows:

$l_1 = 1m$, length of the first link; $l_2 = 0.8m$, length of the second link; $m_1 = 1kg$, mass of the first link; $m_2 = 0.8kg$, mass of the second link. In the simulation, we assume that the network is homogeneous. The aforementioned tracking control is proposed for a network of 4 robot manipulators interconnected under a leader-follower scheme as shown in fig.4.

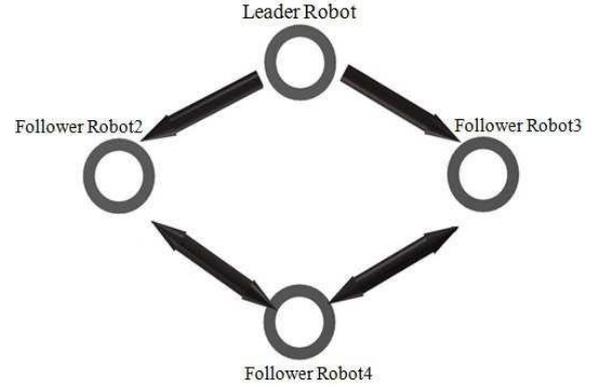


Figure 4: Leader-follower scheme.

One is the network leader, the other are three identical followers robots. Define the initial joint coordinations of follower robots as following: robot1: $(q_{11}, q_{12}) = (20, 20)$ degree, robot2: $(q_{21}, q_{22}) = (10, 10)$ degree, robot3: $(q_{31}, q_{32}) = (-10, -10)$ degree. Where (q_{11}, q_{12}) , (q_{31}, q_{32}) and (q_{31}, q_{32}) denote two joint coordinations of each robot. Consider that the common trajectory to be tracked is a periodic form like: $q_d = \sin(t)\sin(2\cos(t))$.

To specify to which robots each robot is connected, we propose the adjacency matrix

$$A = \begin{pmatrix} 0 & 0 & 0.8 \\ 0 & 0 & 0.7 \\ 0.8 & 0.7 & 0 \end{pmatrix} \quad (43)$$

The data exchange among network is reflected by the following Laplacian matrix

$$L = \begin{pmatrix} 1 & 0 & -0.8 \\ 0 & 1 & -0.7 \\ -0.8 & -0.7 & 2 \end{pmatrix} \quad (44)$$

4.1 Case 1: Non-adaptive Control

Let us first study the synchronized tracking controller when the neural network adaptation law is not activated. The control gains used for simulation are shown below:

Table 1: Control gains for adaptive simulation

Control Gains	Follower1	Follower2	Follower3
k	50	40	40
p	100	100	200
b	0.5	0.4	0

The movement of each follower robot is shown in Fig. 5 that illustrate the robots synchronization tracking a common trajectory. According to the simulation result, it can be obtained that all the follower robots can track the trajectory of

the leader. This proves that the tracking and synchronization objectives are attained by the proposed controller. Fig.6 show the synchronization of the robots velocities. Here it is interesting to note that the robots begin their actions from different starting points and thus achieve a speed of synchronization in the various robots in the network will be a requirement of robustness of the coordination task. The tracking error and synchronization error are depicted in Fig.7 and Fig8, respectively.

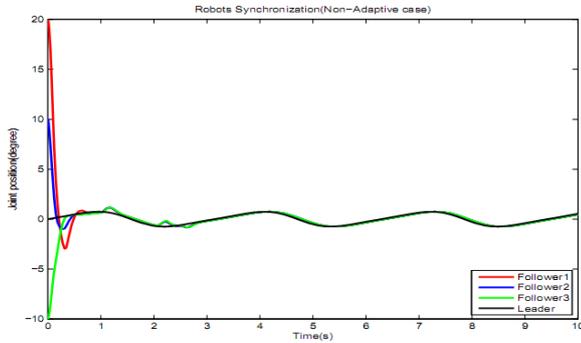


Figure 5: Trajectory Tracking and Synchronization

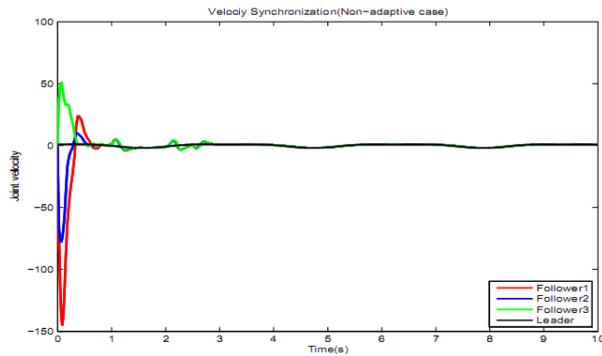


Figure 6: Velocity synchronization.

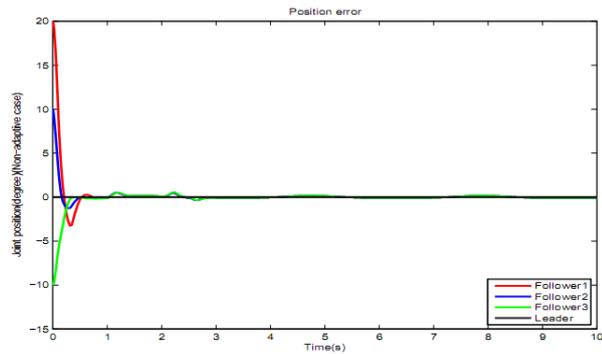


Figure 7: Trajectory Tracking error.

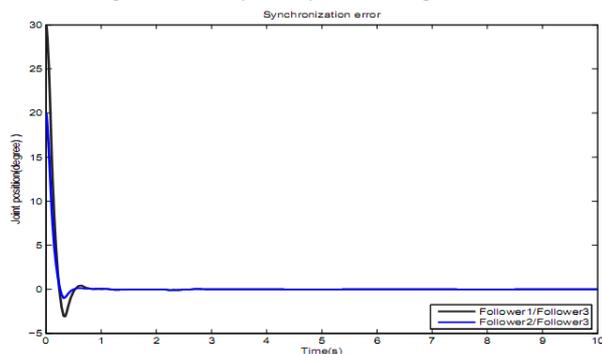


Figure 8: Synchronization error.

4.2 Case 2: Adaptive Control

In this case, we assume that the physical parameters of robots are unknown. Thus, a neural network adaptation is used to estimate on-line the uncertain parameters. The control gains used for simulation are shown below:

Table 2: Control gains for adaptive simulation

Control Gains	Follower1	Follower2	Follower3
k	50	40	40
p	200	200	250
b	0.5	0.4	0
Γ	0.002	0.002	0.002

The proposed network architecture uses two Gaussian function of the form:

$$\phi_i = \exp \left[\frac{-\|x - c_i\|^2}{2\gamma_i^2} \right]$$

where, $x = [q_i^T, \dot{q}_i^T, q_0^T]^T$ is the input vector of the RBF; $c_i \in \mathbb{R}^2$ is the center vector and $\gamma^2 \in \mathbb{R}$ is the variance. The value of c_i and γ^2 are fixed for the robots at $[100, 100, 100]^T$ and 50, respectively.

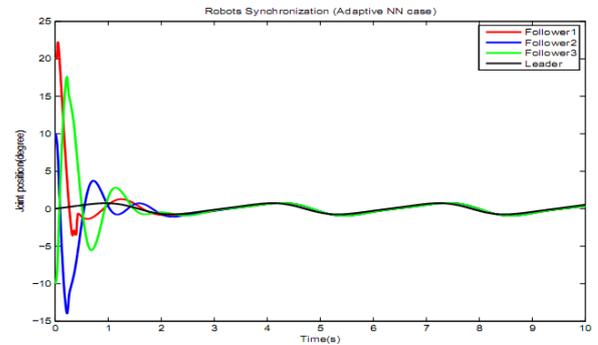


Figure 9: Synchronization of networked robots: adaptive case.

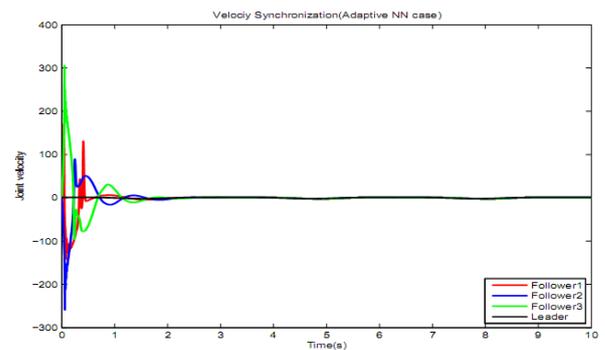


Figure 10: Velocity synchronization: adaptive case

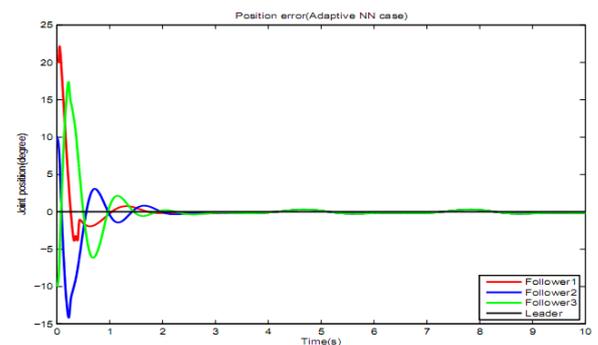


Figure 11: Trajectory Tracking error: adaptive case.

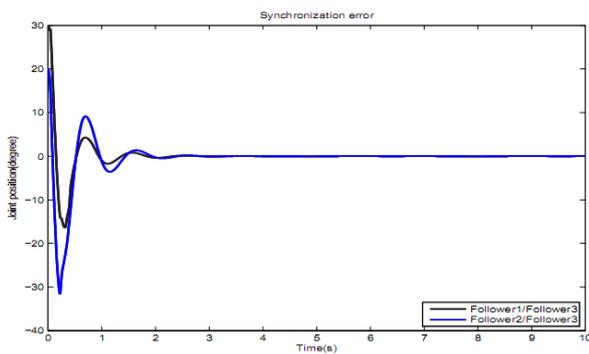


Figure 12: Synchronization error: adaptive case.

The behavior evolution of each robot in the proposed network is seen in Fig.9. An overview of this result clearly shows a synchronization of movement between robots and a perfect trajectory tracking who proves that the synchronization and the trajectory tracking were achieved. Since the robots have different starting points, moreover they can not face the same external disturbances, velocity synchronization remains a challenge to achieve. In this context, the velocity synchronization is shown in Fig.10. Tracking error and synchronization error are presented in Fig.11 and Fig.12, respectively.

5. CONCLUSIONS

A decentralized control law to trajectory tracking and motion synchronization of networked lagrangian systems has been developed. First, using the combination of backstepping technique and graph theory control, the proposed approach works to synchronize positions and velocities of multiple followers interconnected via the neighbor-based rule with respect to the leader's state. In contrast with previous work in the literature dealing with linear system and single or double integrator models with a constant inertia matrix, the proposed strategy permits highly nonlinear systems. Second, to deal with parameter uncertainty problems, specially in practical applications, the proposed approach is extended to an adaptive neural network control law. The control performance is analyzed by the lyapunov method and it's shown that the developed controller guarantees asymptotic convergence of both synchronization errors and position errors. Simulation results obtained from a multi-manipulators motion demonstrate the effectiveness of the leader-follower coordination strategy. In a prospective research work we are interested to extend this methodology in order to take into account communication delays.

REFERENCES

[1] J. Mei, W. Ren, and G. Ma, "Distributed coordinated tracking with a dynamic leader for multiple Euler-Lagrange systems," *IEEE Transactions on Automatic Control*, Issue:99, January 2011.
 [2] J. Ghommam, H. Mehrjerdi, M. Saad and F. Mnif, "Formation path following control of unicycle-type mobile robots, *Robotics and Autonomous Systems*, Volume 58, Issue 5, 31 May 2010, Pages 727-736 .
 [3] Soon-Jo Chung and Jean-Jacques E. Slotine "Cooperative Robot Control and Concurrent Synchronization of Lagrangian Systems". *IEEE*

Transactions on Robotics, vol. 25, Issue:3, pp. 686 - 700 , June 2009.
 [4] Z. G. Hou, L. Cheng, and M. Tan, "Decentralized robust adaptive control for the multiagent system consensus problem using neural networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 3, pp. 636-647, Jun. 2009.
 [5] W. Ren, "Distributed leaderless consensus algorithms for networked EulerLagrange systems," *Int. J. Control*, vol. 82, no. 11, pp. 2137-2149, Nov. 2009.
 [6] Dong Sun, Can Wang, Wen Shang and Gang Feng, "A Synchronization Approach to Trajectory Tracking of Multiple Mobile Robots While Maintaining Time-Varying Formations," *IEEE Transactions on Robotics* , Vol. 25, No. 5, October, 2009.
 [7] Wei Ren, "Distributed leaderless consensus algorithms for networked EulerLagrange systems," *International Journal of Control*, Vol. 82, No. 11, November 2009, 2137-2149
 [8] Long Cheng Zeng-Guang Hou Min Tan "Decentralized adaptive leaderfollower control of multi-manipulator system with uncertain dynamics," *the 34th Annual Conference of IEEE Industrial Electronics*, Nov. 2008.
 [9] Sotzing, C.C.; Evans, J.; Lane, D.M.; "A Multi-Agent Architecture to Increase Coordination Efficiency in Multi-AUV Operations," *IEEE Conference OCEANS 2007-Europe, Aberdeen*, June 2007.
 [10] Lee, D., and Spong, M.W. , "Stable Flocking of Multiple Inertial Agents on Balanced Graphs", *IEEE Transactions on Automatic Control*, 52, 1469-1475, 2007.
 [11] Reza Olfati-Saber "Consensus and Cooperation in Networked Multi-Agent Systems" invited paper, Vol. 95, No. 1, *Proceedings of the IEEE*, January 2007.
 [12] J. Hu and Y. Hong, "Leader following coordination of multi agent systems with coupling time delays", *Physica A*, vol. 374, no. 2, pp.853-863, 2007.
 [13] J. Hu and Y. Hong, "Leader-following coordination of multi-agent systems with coupling time delays", *Physica A*, vol. 374, no. 2, pp. 853-863, 2007.
 [14] Y. Hong, J. Hu, and L. Gao, "Tracking control for multi-agent consensus with an active leader and variable topology", *Automatica*, vol. 42, no. 7, pp. 1177-1182, 2006.
 [15] Olfati-Saber, R. "Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory", *IEEE Transactions on Automatic Control*, 51, 401-420, 2006.
 [16] M. W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley and Sons, Inc., New York, 2006.
 [17] W. Ren and R. W. Beard. "Consensus seeking in multi-agent systems using dynamically changing interaction topologies." *IEEE Transactions on Automatic Control*, 5(5):655-661, May 2005.
 [18] Z. Lin, B. Francis, and M. Maggiore, "Necessary and sufficient graphical conditions for formation control of unicycles", *IEEE Transactions on Automatic Control*, vol. 50, no. 1, pp. 121-127, 2005.
 [19] W. Wang and J.J.E. Slotine, "On partial contraction analysis for coupled nonlinear oscillators", *Biological Cybernetics*, vol. 92, no. 1, pp. 38-53, 2005.

- [20] W. Ren, R.W. Beard, and E.M. Atkins, "survey of consensus problems in multi-agent coordination", Proceedings of American Control Conference, Portland, Oregon, pp. 1859-1864, 2005.
- [21] Lixin Gao, Daizhan Cheng, Comment on "Coordination of Groups of mobile autonomous agents using nearest neighbor Rules", IEEE transactions on Automatic control, pp. 1913-1916, novembre 2005.
- [22] Xiaohai LI and Jizhong XIAO, Robot Formation Control in LeaderFollower Motion Using Direct Lyapunov Method, International Journal of Intelligent Control and Systems vol. 10, no. 3, 244-250, september 2005.
- [23] De Sousa, J.B.; Girard, A.R.; Hedrick, J.K.; Elemental maneuvers and coordination structures for unmanned air vehicles, IEEE Conference on Decision and Control, Nassau, December 2004.
- [24] R. S. VARGA, Gerschgorin and His Circles, Springer-Verlag, Berlin, Germany, 2004.
- [25] J.A. Fax and R.M. Murray, "Information flow and cooperative control of vehicle formations", IEEE Transactions on Automatic Control, vol. 49, no. 9, pp. 1465-1476, 2004.
- [26] Olfati-Saber, R., and Murray, R.M., "Consensus Problems in Networks of Agents with Switching Topology and Time-Delays," in the special issue of the IEEE Transactions On Automatic Control on Networked Control Systems, Vol. 49, No. 9, Sept. 2004.
- [27] Alejandro Rodriguez-Angeles and Henk Nijmeijer "Mutual Synchronization of Robots via Estimated: State Feedback: A Cooperative Approach" IEEE Transactions On Control Systems Technology, Vol. 12, No. 4, July 2004.
- [28] Dong Sun "Position synchronization of multiple motion axes with adaptative coupling control", Automatica, vol. 39, no.6, pp. 997-1005, 2003.
- [29] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules", IEEE Transactions on Automatic Control, vol. 48, no. 6, pp. 988-1001, June 2003.
- [30] Ribichini, G.; Frazzoli, E.; Efficient coordination of multiple-aircraft systems, IEEE conference on Decision and Control Proceeding, pp. 1035-1040, December 2003.
- [31] A. Rodriguez-Angeles; H. Nijmeijer, Coordination of two robot manipulators based on position measurements only , International Journal of Control, 1366-5820, Volume 74, Issue 13, Pages 1311-1323, 2001.
- [32] WONG LEE, K., and KHALIL, H. K., Adaptive output feedback control of robot manipulators using high-gain observer. Int. J. of Control, 67, pp.869886, 1997.
- [33] LIU, Y.-H. , XU, Y., and BERGERMAN, M., Cooperation control of multiple manipulators with passive joints. IEEE Trans. Robotics Automation, 15, pp. 258-267, 1999.
- [34] J. Tarn, A.K. Bejczy, "Intelligent planning and control for multirobot coordination: An event-based approach", IEEE Transactions on Robotics and Automation, vol. 12, no. 3, pp. 439-452, 1996.