Synchronized control with neuro-agents for leader-follower based multiple robotic manipulators

Dongya Zhao[1,2*], Quanmin Zhu[1,3], Ning Li[4], Shaoyuan Li[4]

Corresponding author's email: dongyazhao@gmail.com; dyzhao@upc.edu.cn

1. College of Chemical Engineering, China University of Petroleum, Qingdao, China, 266580

2. State Key Laboratory of Heavy Oil Research, China University of Petroleum, Qingdao, China, 266580

3. Department of Engineering Design and Mathematics, University of the West of England, Coldharbour Lane, Bristol BS16 1QY, UK

4. Department of Automation, Shanghai Jiao Tong University, Shanghai, China, 200240

Abstract

In this paper, a new neural network enhanced synchronized control approach is proposed for multiple robotic manipulators systems (MRMS) based on leader-follower network communication topology. The justification of introducing two adaptive Radial Basis Function Neural Networks (RBF NN), also called neuro agents, is to facilitate the whole control system design and analysis. Otherwise such design is impossible with classical analytical procedure. The first agent is the neuro-compensator to accommodate uncertainty associated with the follower manipulators, and the second agent is the neuro-estimator to obtain acceleration of the leader manipulator. Correspondingly the stability analysis of the designed control system is formulated with Lyapunov method. Finally numerical bench tests under various critical conditions are conducted to validate the effectiveness of the proposed approach.

Keywords: Synchronized control, multiple robotic manipulators, leader-follower, neural networks, neuro-computing

1 Introduction

It has been increasingly important to employ multiple robotic manipulators to execute a commonly or interactively simultaneously shared task in modern manufacturing such as assembling, transporting, painting and welding, just to name a few [1-4]. Such multiple manipulators, if not yet, will have more functions in space and deep seas exploration. The aforementioned industrial applications require large maneuverability and manipulability, in which a single robotic manipulator cannot undertake easily or even impossibly. To effectively achieve these largely demanded task functionalities, an effective solution has been to use cooperative or coordinated MRMS [5, 6]. Conventional centralized and/or decentralized robot control algorithms have not addressed coordination/cooperation tasks [7, 8]. Note that most of existing coordinated control algorithms, such as cooperative control and master-slave control, require to measure internal force in implementation. It is known that internal force measurement is very difficult in practice. In the point of practical view, synchronization, coordination and cooperation are intimately linked subjects and have been used as synonyms to describe such system characteristics [9]. With the deepening of the research, it is found that position synchronized control can coordinate MRMS without measuring internal force [10-13]. It has been noticed that control of such systems still stands as one of the challenging issues in the field of robot control.

Due to the good executable ability, much attention has been attracted by mechanical systems synchronized control. To justify the motivation and necessity of the proposed study, there must make a critical survey on the existing representative work, which scrutinizes the achievement and potential hard nut issues. In light of cross-coupling technology, an adaptive synchronized control algorithm has been designed for multi-robot assembly tasks [13]. A mutual synchronization control approach has been studied with velocity observer [14]. By removing some restrictive assumptions, an adaptive position synchronized control has been developed for multi-robots with flexible/rigid constraints [15]. Sliding mode position synchronized control algorithm has been developed for MRMS, which has strong robustness [16, 17]. Passivity framework has been used in synchronization bilateral teleoperators, which can deal with bounded time delay [18]. It should be mentioned that most of the existing synchronized control approaches of MRMS use undirected communication networked topology graphs which assumed that at least each two neighboring manipulators can communicate with each other. However, the directed communication networked topology graphs are more appropriate in practice due to communicating cost and/or networked failure [19]. In this paper, a leader-flower based directed graph is used to describe the communicating networked topology. Note that leader-follower directed graph can achieve MRMS tracking synchronization which is more useful than that of leaderless synchronized control in industrial applications [3, 4]. In light of graph theory, a novel synchronization error is initially defined and used to design the synchronized controller for MRMS. By using the weighted matrix and Laplacian matrix in the proposed synchronization error design, the new synchronization error is different from the existing ones in nature.

System uncertainty of robotic manipulator may be induced by modeling error, backlash, friction, external disturbance and so forth, which deteriorates system control performance seriously. It should be accommodated in synchronized controller design. NN has strong learning ability and can approximate almost all of nonlinear function [20]. It has found that NN can estimate system uncertainty of robotic manipulator online effectively. Some NN based adaptive control algorithms are designed for single robotic manipulator [21-23]. Note that the mentioned NN based control algorithms are appropriate for the single robotic manipulators but cannot be used to the MRMS. Some indispensable extensions are required to design NN based synchronized controller according to the MRMS's kinematics and dynamics properties. It is not trivial to design the new RBF NN based adaptive law for MRMS because the leader-follower based synchronization error and graph weighted adjacency matrices should be embedded into it. The motivations of using RBF NN in MRMS are two folds: (1) It can compensate follower manipulators' system uncertainty online and then reduces the controller design complexity. (2) It can estimate the leader manipulator's acceleration online which is very difficult to measure in practice. Most of the leader-follower control algorithms assume that the bound of the acceleration should be given before the controller design [24] or complex observer should be used for the estimation [25]. In summary, using RBF NN in synchronized control of MRMS is novel, efficient, and challenging in designing such class of control systems and can simplify the controller design.

By using the leader-follower communicating topology and NN online learning technique, a new synchronized control algorithm with neuro-agents is developed for MRMS in this study. The proposed synchronized controller has the following characteristics: the leader-follower based synchronization error, RBF NN based follower manipulators' modeling error compensator and RBF NN based leader acceleration estimator. The neural network weighting parameters can be updated by an adaptive law online. The closed loop control is guaranteed to be stable by Lyapunov method. The robotic manipulators can track the leader's trajectory in a synchronous manner. In summary, the control algorithm is interesting and novel due to the use of new synchronization error, enhanced with the two neuro agents, RBF dynamic compensator and the estimator in MRMS. Especially, the RBF acceleration estimator of leader manipulator can relax the assumption that the leader acceleration bound should be known during the controller design. This assumption has been used in most of leader-follower multi-agent control systems.

In general, from the study, such control law design procedure can be summarized with the following 5 steps:

Step 1: Define a leader-follower type synchronization error which is more in line with industrial practice [18].

Step 2: Design RBF based online compensator to deal with system uncertainty and RBF based online estimator to deal with the leader's acceleration. Note that RBF is a universal approximate function which can estimate almost all nonlinear function, which can simplify the controller design greatly [26-28].

Step 3: Reform the MRMS dynamic model after the above two step operations. Note that this will be simplified as a second order system with small modeling errors.

Step 4: Design an adaptive synchronized control law by using Lyapunov methods [29].

Step 5: Stability analysis should be given to lay a foundation for the safety use of the proposed approach.

This study integrates several principles cross modeling, neuro computing, system and control domains for MRMS, such as leader-follower directed graph, RBF NN compensation and estimation, synchronized control and so on. Fairly speaking, there are many approaches enable to design the synchronized controllers for MRMS. However the proposed approach is more practical, simple and systematical. In the point view of the authors, this study may present an alternative but more effective solution for MRMS with new insight and application incentive.

The rest of this study is organized as follows. In section 2, synchronization error is defined in the concept of directed graph and leader-follower topology. In section 3, RBF NN scheme is elaborated to compensate the modeling error of robotic manipulator and to estimate the joint acceleration of leader manipulator. In section 4, the new synchronized control algorithm is developed with stability analysis. In section 5, illustrative examples are presented to validate the performance of the designed scheme. Finally, in section 6, some concluding remarks are given to complete the study.

2 Leader-follower based synchronization error

In this section, some basic concepts on algebraic graph theory are introduced to lay a foundation for MRMS synchronized control. The leader-follower based synchronization error is defined in light of these concepts.

A. Concepts on graph theory and leader-follower system

Consider a leader-follower system consisting of one leader and n followers. Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a directed graph, in which $\mathcal{V} = \{0,1,2,\cdots,n\}$ is the set of nodes. Node i denotes the ith robotic manipulator. $\mathcal{E}$ is the set of edges. An edge of $\mathcal{G}$ is represented by an ordered pair(i,j). $(i,j) \in \mathcal{E}$ if and only if the ith manipulator can send information to the jth manipulator directly, but not necessarily vice versa. Unlike the directed graph, the pairs of nodes on an undirected graph are unordered, in which the edge (i,j) means that manipulator i and j can obtain information from each other. Hence, the undirected graph is a special case of a directed graph. A directed tree is a directed graph, where every node has an exact parent except for the root, and the root has a directed path to every node. A directed spanning tree of $\mathcal{G}$ is a directed tree that contains all nodes of $\mathcal{G}$ [30].

Suppose the MRMS has $n+1$ m-link full actuated robotic manipulator including one leader and n followers. Let $A = (a_{ij}) \in R^{(n+1)m \times (n+1)m}$ be the weighted adjacency matrix of $\mathcal{G}$ with nonnegative elements, where $a_{ij} \in R^{m\times m} \geq 0$ with $a_{ij} > 0$ if there is an edge between manipulator i and manipulator j. Let $D = \operatorname{diag}\{d_0, d_1, \cdots, d_n\} \in R^{(n+1)m\times(n+1)m}$ be a block diagonal matrix, where $d_i = \sum_{j=0}^{n} a_{ij}$ for $i = 0,1,\cdots,n$. Then, the Laplacian of the weighted graph can be defined as

$$L = D - A \in R^{(n+1)m\times(n+1)m} \tag{1}$$

The connection weight between the follower manipulator and the leader manipulator is denoted by $b_i \in R^{m\times m}$ with $b_i > 0$ if there is an edge between them. Two theorems summarize the existing results on Laplacian matrix and graph theory.

Theorem 1. [31] The directed graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ has a directed spanning tree if and only if $\{\mathcal{V}, \mathcal{E}\}$ has at least one node with a directed path to all other nodes.

Theorem 2. The Laplacian matrix L of a directed graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ has at least m zero eigenvalue and all of the nonzero eigenvalues are in the open right-half plane. In addition, L has exactly m zero eigenvalue if and only if $\mathcal{G}$ has a directed spanning tree. Furthermore, $\operatorname{Rank}(L) = nm$ if and only if L has m simple zero eigenvalues.

Proof: The theorem can be proved easily along the method in [32].

B. Dynamic equation of robotic manipulators

Consider the m-link full actuated robotic manipulator. Its dynamic equation can be given as [33]:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \tag{2}$$

where $q, \dot{q}, \ddot{q} \in R^m$ are the joint position, velocity and acceleration, respectively. $M(q) \in R^{m\times m}$ is symmetric positive definite inertia matrix, $C(q,\dot{q})\dot{q} \in R^m$ is Coriolis and centripetal force vector, $G(q) \in R^m$ is gravitational force vector, $\tau \in R^m$ is joint torque vector.

Suppose leader manipulator dynamic equation is expressed as:

$$M_l(q_l)\ddot{q}_l + C_l(\dot{q}_l, q_l)\dot{q}_l + G_l(q_l) = \tau_l \tag{3}$$

The dynamics of ith follower manipulator is expressed as:

$$M_{0i}(q_i)\ddot{q}_i + C_{0i}(\dot{q}_i, q_i)\dot{q}_i + G_{0i}(q_i) = \tau_i + f_i(\dot{q}_i, q_i),\ i = 1,\cdots,n \tag{4}$$

where $M_{0i}(q_i)$, $C_{0i}(\dot{q}_i, q_i)$ and $G_{0i}(q_i)$ are nominal part of robotic manipulator dynamics, $f_i(\dot{q}_i, q_i) = -\Delta M_i(q_i)\ddot{q}_i - \Delta C_i(\dot{q}_i, q_i)\dot{q}_i - \Delta G_i(q_i)$ is the system uncertainty.

C. Leader-follower based synchronization error

The MRMS has $n+1$ robotic manipulators, in which leader manipulator indexed by 0 and follower manipulators indexed by $1, \cdots, n$. The topology relationships among the leader and followers are expressed by a directed graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ with $\mathcal{V} = \{0,1,2,\cdots,n\}$ and the adjacent matrix:

$$A = \begin{bmatrix} 0_m & 0_m & \cdots & 0_m \\ a_{10} & a_{11} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \in R^{(n+1)m\times(n+1)m} \tag{5}$$

where $0_m \in R^{m\times m}$ is a zero matrix, $a_{ij} \in R^{m\times m} > 0$.

Let $\bar{\mathcal{G}} = \{\bar{\mathcal{V}}, \bar{\mathcal{E}}\}$ as the subgraph of $\mathcal{G}$, which is formed by the follower manipulators and let:

$$\bar{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \in R^{nm\times nm} \tag{6}$$

Let $\bar{D} = \mathrm{diag}\{\bar{d}_1, \cdots, \bar{d}_n\} \in R^{nm\times nm}$ be a block diagonal matrix with $\bar{d}_i = \sum_{j=1}^{n} a_{ij}$ for $i = 1, \cdots, n$. It is obvious that the Laplacian of the subgraph $\bar{\mathcal{G}}$ can be defined as:

$$\bar{L} = \bar{D} - \bar{A} \tag{7}$$

For simplicity, assume that:

$$a_{ij} = \begin{cases} I_m, & \text{if } (j,i) \in \mathcal{E} \\ 0_m, & \text{otherwise} \end{cases} \tag{8}$$

where $I_m \in R^{m\times m}$ is an identity matrix.

Let the connection weight between follower manipulator i and the leader is defined as:

$$\bar{B} = \mathrm{diag}\{b_1, b_2, \cdots, b_n\} \tag{9}$$

where b_i is defined as:

$$b_i = \begin{cases} I_m, & \text{if manipulator } i \text{ is connected to the leader} \\ 0_m, & \text{otherwise} \end{cases} \tag{10}$$

Assumption 1. The joint position and velocity of the leader manipulator are available to its neighbors only.

Definition 1. Suppose the communication topology of the MRMS is a directed spanning tree. The synchronization error is defined as

$$\begin{cases} e_i^p \triangleq \sum_{j=1}^{n} a_{ji}(q_i - q_j) + b_i(q_i - q_l) \\ e_i^v \triangleq \sum_{j=1}^{n} a_{ji}(\dot{q}_i - \dot{q}_j) + b_i(\dot{q}_i - \dot{q}_l) \end{cases} \tag{11}$$

Remark 1. Partly illumed by multi-agent consensus error [24], synchronization error is defined by (11). According to (11), the synchronization here means that each follower manipulator can track the leader manipulator's position while synchronizes its motion with the neighbors. Though the leader's position and velocity are only available to its neighbors its effects can be transferred to other followers indirectly due to the directed spanning tree. This synchronization error is fully different from the existing ones [13, 14, 16, 17]. The directed graph based synchronized error is more practical than the existing undirected graph based ones.

The synchronization error dynamics can be written as:

$$\begin{cases} \dot{e}_i^p \triangleq e_i^v \\ \dot{e}_i^v \triangleq \sum_{j=1}^n a_{ji}\left(\ddot{q}_i - \ddot{q}_j\right) + b_i(\ddot{q}_i - \ddot{q}_l) \end{cases} \tag{12}$$

Define some vectors and matrices as follows:

$Q_F = [q_1^T, \cdots, q_n^T]^T$, $\dot{Q}_F = [\dot{q}_1^T, \cdots, \dot{q}_n^T]^T$, $T_F = [\tau_1^T, \cdots, \tau_n^T]^T$, $F_F = [f_1^T, \cdots, f_n^T]^T$, $E_p = \left[\left(e_1^p\right)^T, \cdots, \left(e_n^p\right)^T\right]^T$, $E_v = [(e_1^v)^T, \cdots, (e_n^v)^T]^T$, $M_F = \begin{bmatrix} M_{01}(q_1) & & 0 \\ & \ddots & \\ 0 & & M_{0n}(q_n) \end{bmatrix}$, $C_F = \begin{bmatrix} C_{01}(\dot{q}_1, q_1) & & 0 \\ & \ddots & \\ 0 & & C_{0n}(\dot{q}_n, q_n) \end{bmatrix}$, $G_F = [G_{01}^T(q_1), \cdots, G_{0n}^T(q_n)]^T$.

Then, (12) can be written in the matrix form:

$$\begin{cases} \dot{E}_p = E_v \\ \dot{E}_v = (\bar{L} + \bar{B})M_F^{-1}\left(-C_F\dot{Q}_F - G_F + T_F + F_F\right) - \bar{B}\boldsymbol{I}M_l^{-1}(-C_l(\dot{q}_l, q_l)\dot{q}_l - G_l(q_l) + \tau_l) \end{cases} \tag{13}$$

where $\boldsymbol{I} = [I_m, \cdots, I_m]^T \in R^{nm \times m}$.

Theorem 3. Consider MRMS (3) and (4) with a directed graph $\mathcal{G}$ communication topology, if $\mathcal{G}$ has a directed spanning tree and $E_p = 0$ and $E_v = 0$, then

$$[q_1^T, \cdots, q_n^T]^T = \boldsymbol{I}q_l \tag{14}$$

$$[\dot{q}_1^T, \cdots, \dot{q}_n^T]^T = \boldsymbol{I}\dot{q}_l \tag{15}$$

Proof: By using the similar proof procedure in [24], the result can be proved easily.

3 RBF neural network compensator and estimator for robotic manipulators

In this section, some RBF NN concepts will be introduced. Then two neuro-agents, that is, RBF based dynamic compensator and the leader joint acceleration estimator will be designed for MRMS.

A. Concepts on RBF neural networks

RBF NN has some desirable features such as local adjustment of weights and mathematical tractability, which attracted larger numbers of attentions in researches and applications. RBF NN with adaptive weights is addressed in [26]. RBF NN can be used in adaptive control of nonlinear system, in which RBF NN can adaptively compensate for the nonlinear dynamics [27]. As feedforward networks, RBF NN can mapping an input vector x to an output vector y. A RBF NN can be expressed by [20]:

$$\phi_i = g\left(\|x - c_i\|^2/\sigma_i^2\right),\ \mathrm{i} = 1,2, \cdots, n^*$$

$$y = W\phi(x)$$

where $x \in R^{m_r}$ is the input, $\phi = [\phi_1, \phi_2, \cdots, \phi_{n^*1}]^T \in R^{n^*}$ is the output of the hidden layer, $y \in R^{n_r}$ is the output of the network., $W \in R^{n_r \times n^*}$ is the weight matrix, $c_i \in R^{m_r}$ and $\sigma_i > 0$ are the center and width of the

ith kernel unit respectively. In RBF networks, $\|\cdot\|$ usually denotes the Euclidean norm. The continuous function $g: [0, \infty) \to R$ is the activation function which is often chosen to be the Gaussian function $g(\alpha) = \exp(-\alpha)$. It can be seen that each kernel node in the RBF NN computes an output that depends on a radially symmetric function, and usually the strongest output is obtained when the input is near the centroid of the node.

Remark 2. Note that under some mild assumptions RBF NN has a universal approximate ability to approximate almost all continuous functions over a compact set to any degree of accuracy [27]. Accordingly the RBF networks can be used to approximate the follower robotic manipulators' dynamic uncertainty $f_i(\dot{q}_i, q_i)$, $i = 1, \cdots, m$ and the leader's joint acceleration $\ddot{q}_l$.

Based on the existing results on the RBF neural networks, the following assumptions are made before the controller design [28].

Assumption 2. Given a positive number ε_0 and a continuous function $f(x): \mathscr{b} \to \mathcal{R}$, $\mathscr{b} \in R^{m_r}$ is a compact set, there is a weight matrix θ and a positive integer n^* such that the output $\hat{f}(x, \theta)$ of the neural networks with n^* nodes satisfies

$$\max_{x \in \mathscr{b}} \left\| \hat{f}(x, \theta) - f(x) \right\| \le \varepsilon_0$$

where n^* depends on ε_0 and $f(x)$.

Assumption 3. The output $\hat{f}(x, \theta)$ of the neural networks is continuous with respect to its arguments for all finite (x, θ).

B. RBF neural networks based dynamic compensator for follower manipulators

Design feedforward compensating controller:

$$\tau_{0i} = C_{0i}(\dot{q}_i, q_i)\dot{q}_i + G_{0i}(q_i) \tag{16}$$

Let $\tau_i = \tau_{0i} + M_{0i}\tau_{1i}$ and substitute τ_i into (3):

$$\ddot{q}_i = \tau_{1i} + M_{0i}^{-1}(q_i) f_i(q_i, \dot{q}_i), \ i = 1, \cdots, n \tag{17}$$

Let $h_i(q_i, \dot{q}_i) = M_{0i}^{-1}(q_i) f_i(q_i, \dot{q}_i)$ and substitute it into (17):

$$\ddot{q}_i = \tau_{1i} + h_i(q_i, \dot{q}_i), \ i = 1, \cdots, n \tag{18}$$

Define $x_i = \left[\left(e_i^p\right)^T, (e_i^v)^T \right]^T$, it is obvious that $h_i(q_i, \dot{q}_i)$ is a function of x_i. According to the RBF neural networks results, the nonlinear function $h_i(x_i)$ can be approximated by a static RBF neural network with output $\hat{h}_i(x_i, \theta_i)$, in which $\theta_i \in R^{n^*}$. Suppose θ_i^* is the optimal weight values to approximate $h_i(x_i)$ for x_i belong to a compact set $\mathscr{b}(N_{x_i}) \subset R^{2m}$ which is defined as $\mathscr{b}(N_{x_i}) \triangleq \{x_i: \|x_i\| \le N_{x_i}\}$.

Notation 1. For a matrix R, Frobenius matrix norm is defined as $\|R\|_F^2 \triangleq \sum_{ij} \left| r_{ij} \right|^2 = \mathrm{tr}(R^T R) = \mathrm{tr}(R R^T)$.

Assumption 4. All the weights belong to a large compact set $\mathcal{B}(M_{\theta_i}) \triangleq \{\theta_i : \|\theta_i\|_F \le M_{\theta_i}\}$, $M_{\theta_i} > 0$ is a positive number.

The optimal weight θ_i^* is defined as the element in $\mathcal{B}(M_{\theta_i})$ that can minimize the function $\|\hat{h}_i(x_i, \theta_i) - h_i(x_i)\|$ for $x_i \in \mathscr{b}(N_{x_i})$, that is:

$$\theta_i^* \triangleq \arg\min_{\theta_i \in \mathcal{B}(M_{\theta_i})} \left\{ \sup_{x_i \in \mathscr{b}(N_{x_i})} \left\| \hat{h}_i(x_i, \theta_i) - h_i(x_i) \right\| \right\} \tag{19}$$

Then (18) can be written as:

$$\ddot{q}_i = \tau_{1i} + \hat{h}_i(x_i, \theta_i^*) + \left(h_i(x_i) - \hat{h}_i(x_i, \theta_i^*) \right), \ i = 1, \cdots, n \tag{20}$$

Remark 3. In the adaptive law design, the estimation of θ_i^* can be restricted in the compact set $\mathcal{B}(M_{\theta_i})$ by using projection approach, which will be specified in the next section.

Define modeling error caused by RBF NN as:

$$\eta_i \triangleq h_i(x_i) - \hat{h}_i(x_i, \theta_i^*) \tag{21}$$

It is bounded by a finite positive constant:

$$\eta_{0i} \triangleq \sup_{t \ge 0} \left\| h_i(x_i) - \hat{h}_i(x_i, \theta_i^*) \right\| \tag{22}$$

By using the RBF NN properties, $\hat{h}_i(x_i, \theta_i^*)$ can be expressed in the following form:

$$\hat{h}_i(x_i, \theta_i^*) = \theta_i^* \phi_i(x_i) \tag{23}$$

where $\theta_i^* \in R^{m \times n^*}$ is optimal weight values matrix and $\theta_i^* \le M_{\theta_i}$, $\phi_i(x_i) \in R^{n^*}$ is output of the hidden layer of RBF NN, here it is the regressor.

By using (21) and (23), (20) can be written as:

$$\ddot{q}_i = \tau_{1i} + \theta_i^* \phi_i(x_i) + \eta_i, \ i = 1, \cdots, n \tag{24}$$

Let $\hat{\theta}_i^*$ be the estimation of θ_i^*, then design a RBF neural networks based compensator as:

$$\tau_{2i} = -\hat{\theta}_i^* \phi_i(x_i) \tag{25}$$

Let $\tau_{1i} = \tau_{3i} + \tau_{2i}$, substitute τ_{1i} into (24):

$$\ddot{q}_i = \tau_{3i} + \theta_i^* \phi_i(x_i) - \hat{\theta}_i^* \phi_i(x_i) + \eta_i, \ i = 1, \cdots, n \tag{26}$$

Define the estimation error of θ_i^* as:

$$\tilde{\theta}_i^* = \hat{\theta}_i^* - \theta_i^* \tag{27}$$

Then (26) can be expressed as:

$$\ddot{q}_i = \tau_{3i} - \tilde{\theta}_i^* \phi_i(x_i) + \eta_i, \ i = 1, \cdots, n \tag{28}$$

According to the expressions of τ_{0i}, τ_{1i} and τ_{2i}, the control input τ_i is written as:

$$\tau_i = C_{0i}(\dot{q}_i, q_i)\dot{q}_i + G_{0i}(q_i) - M_{0i}(q_i)\hat{\theta}_i^* \phi_i(x_i) + M_{0i}\tau_{3i}, \ i = 1, \cdots, n \tag{29}$$

Remark 4. τ_{0i} and τ_{2i} are feedforward compensators. By using them, follower manipulator's dynamic equation can be simplified as (28). The term $\hat{\theta}_i^*$ can be updated online by an adaptive law which will be designed in the next section.

C. RBF neural networks based acceleration estimator for leader manipulator

Substitute (29) into (12):

$$\begin{cases} \dot{e}_i^p = e_i^v \\ \dot{e}_i^v = \left(\sum_{j=1}^m a_{ij} + b_i\right)\left[\tau_{3i} - \tilde{\theta}_i^* \phi_i(x_i) + \eta_i\right] - \sum_{j=1}^m \left[a_{ij}\tau_{3j} - a_{ij}\tilde{\theta}_j^* \phi_j(x_j) + a_{ij}\eta_j\right] - b_i \ddot{q}_l \end{cases} \tag{30}$$

Then, (30) can be expressed in the matrix form:

$$\begin{cases} \dot{E}_p = E_v \\ \dot{E}_v \triangleq (\bar{L} + \bar{B})T + (\bar{L} + \bar{B})\Theta - (\bar{L} + \bar{B})\widetilde{H} - \bar{B}\boldsymbol{I}\ddot{q}_l \end{cases} \tag{31}$$

where $\mathrm{T} = [\tau_{31}, \cdots \tau_{3n}]^{\mathrm{T}}$, $\Theta = [\eta_1, \cdots \eta_n]^{\mathrm{T}}$, $\widetilde{H} = \left[\phi_1^T(x_1)\left(\tilde{\theta}_1^*\right)^T, \cdots, \phi_n^T(x_n)\left(\tilde{\theta}_n^*\right)^T\right]^T$.

Because τ_l is the function of q_l and $\dot{q}_l$, $\ddot{q}_l$, it is also the function of q_l and $\dot{q}_l$. Define $x_l = [q_l^T, \dot{q}_l^T]^T$, the nonlinear function $\ddot{q}_l(x_l)$ can be approximated by a static RBF NN with output $\hat{\ddot{q}}_l(x_l, \theta_l)$, in which $\theta_l \in R^{n^{**}}$.

$$\hat{\ddot{q}}_l(x_l, \theta_l) = \theta_l \phi_l(x_l) \tag{32}$$

Suppose θ_l^* is the optimal weight values to approximate $\ddot{q}_l(x_l)$ for x_l belong to a compact set $\mathscr{b}\left(N_{x_l}\right) \subset R^{2m}$ which is defined as $\mathscr{b}\left(N_{x_l}\right) \triangleq \left\{x_l: \|x_l\| \leq N_{x_l}\right\}$.

Assumption 5. All the weights belong to a large compact set $\mathcal{B}\left(M_{\theta_l}\right) \triangleq \left\{\theta_l: \|\theta_l\|_F \leq M_{\theta_l}\right\}$, $M_{\theta_l} > 0$ is a positive number.

The optimal weight θ_l^* is defined as the element in $\mathcal{B}\left(M_{\theta_l}\right)$ that can minimize the function $\left\|\hat{\ddot{\mathrm{q}}}_l(x_l, \theta_l) - \ddot{q}_l(x_l)\right\|$ for $x_l \in \mathscr{b}\left(N_{x_l}\right)$, that is:

$$\theta_l^* \triangleq \arg\min_{\theta_l \in \mathcal{B}\left(M_{\theta_l}\right)} \left\{\sup_{x_l \in \mathscr{b}\left(M_{x_l}\right)} \left\|\hat{h}_l(x_l, \theta_l) - h_l(x_l)\right\|\right\} \tag{33}$$

Define modeling error caused by RBF NN as:

$$\eta_l \triangleq \ddot{q}_l(x_l) - \hat{\ddot{\mathrm{q}}}_l(x_l, \theta_l^*) \tag{34}$$

It is bounded by a finite positive constant:

$$\eta_{0l} \triangleq \sup_{t \geq 0} \left\|\ddot{q}_l(x_l) - \hat{\ddot{\mathrm{q}}}_l(x_l, \theta_l^*)\right\| \tag{35}$$

By using (34), (31) can be written as:

$$\begin{cases} \dot{E}_p = E_v \\ \dot{E}_v \triangleq (\bar{L} + \bar{B})T + (\bar{L} + \bar{B})\Theta - (\bar{L} + \bar{B})\widetilde{H} - \bar{B}\boldsymbol{I}(\theta_l^* \phi_l(\mathrm{x}_l) + \eta_l) \end{cases} \tag{36}$$

The RBF NN based leader manipulator joint acceleration estimator can be designed as follows:

$$T_1 = (\bar{L} + \bar{B})^{-1}\bar{B}\boldsymbol{I}\hat{\theta}_l^* \phi_l(\mathrm{x}_l) \tag{37}$$

Control law T will be designed as:

$$T = T_1 + T_2 \tag{38}$$

Remark 5. T_1 is the leader joint acceleration estimator, its weight parameters $\hat{\theta}_l^*$ can be updated by an adaptive law online which will be specified in the next section.

4 Leader-follower based synchronized controller design

In this section the main results of this study will be summarized with stability analysis. For (36), in light of RBF neural network approximate ability, a leader-follower based adaptive synchronized control law can be designed for MRMS:

$$T_2 = (\bar{L} + \bar{B})^{-1}\left(-K_p E_p - K_v E_v\right) \tag{39}$$

Substitute (38) into (36), it yields:

$$\begin{cases} \dot{E}_p = E_v \\ \dot{E}_v \triangleq -K_p E_p - K_v E_v - (\bar{L} + \bar{B})\widetilde{H} + \bar{B}\boldsymbol{I}\tilde{\theta}_l^* \phi_l(x_l) + (\bar{L} + \bar{B})\Theta - \bar{B}\boldsymbol{I}\eta_l \end{cases} \tag{40}$$

Let $\mathbb{E} = \left[E_p^T, E_v^T\right]$, $\mathbb{A} = \begin{bmatrix} \mathbf{0} & \mathbb{I} \\ -K_p & -K_v \end{bmatrix}$, $\mathbb{B}_1 = \begin{bmatrix} \mathbf{0} \\ -(\bar{L} + \bar{B}) \end{bmatrix}$, $\mathbb{B}_2 = \begin{bmatrix} \mathbf{0} \\ \bar{B}\boldsymbol{I} \end{bmatrix}$, where $\mathbf{0}$ and $\mathbb{I}$ are zero matrix and identity matrix with appropriate dimensions. Then (40) can be rewritten as:

$$\dot{\mathbb{E}} = \mathbb{A}\mathbb{E} + \mathbb{B}_1\left(\widetilde{H} - \Theta\right) + \mathbb{B}_2\left(\tilde{\theta}_l^* \phi_l(x_l) - \eta_l\right) \tag{41}$$

RBF neural networks based adaptive laws are designed as:

$$\dot{\hat{\Xi}} = -\frac{1}{\Gamma_f}\mathbb{B}_1^T P \mathbb{E}[\phi_1^T(x_1), \cdots, \phi_n^T(x_n)] - \frac{c_f}{\Gamma_f}\frac{\widehat{H}^T \mathbb{B}_1^T P \mathbb{E}}{M_{\theta_I}^2}\hat{\Xi} \tag{42}$$

$$c_f = \begin{cases} 1, \text{if } \left\|\hat{\Xi}\right\| = M_{\theta_I} \text{ and } \widehat{H}^T \mathbb{B}_1^T P \mathbb{E} > 0 \\ 0, \text{otherwise} \end{cases} \tag{43}$$

$$\dot{\hat{\theta}}_l^* = -\frac{1}{\Gamma_l}\mathbb{B}_2^T P \mathbb{E}\phi_l^T(x_l) - \frac{c_l}{\Gamma_l}\frac{\phi_l^T(x_l)\left(\hat{\theta}_l^*\right)^T \mathbb{B}_2^T P \mathbb{E}}{M_{\theta_l}^2}\hat{\theta}_l^* \tag{44}$$

$$c_l = \begin{cases} 1, \text{if } \left\|\hat{\Xi}\right\| = M_{\theta_l} \text{ and } \phi_l^T(x_l)\left(\hat{\theta}_l^*\right)^T \mathbb{B}_2^T P \mathbb{E} > 0 \\ 0, \text{otherwise} \end{cases} \tag{45}$$

where $K_p, K_v \in R^{mn \times mn}$ are positive definite diagonal matrices, $\hat{\Xi} = \text{diag}\left\{\left(\hat{\theta}_1^*\right), \cdots, \hat{\theta}_n^*\right\} \in R^{mn \times n^* n}$, $\Xi = \text{diag}\{\theta_1^*, \cdots, \theta_n^*\} \in R^{mn \times n^* n}$, $\widehat{H} = \left[\hat{h}_1^T, \cdots, \hat{h}_n^T\right]^T$, $\Gamma_f, \Gamma_l > 0$ are positive constant, $M_{\theta_I} = \max_{i=1,\cdots n}\left\{M_{\theta_i}\right\}$, P is symmetric and positive definite matrix and satisfies Lyapunov equation $P\mathbb{A} + \mathbb{A}^T P = -Q$, $Q \geq 0$.

Define the following equations:

$$\mathbb{B}_1 = \begin{bmatrix} \mathbb{b}_{11}^1 & \cdots & \mathbb{b}_{1n}^1 \\ \vdots & \ddots & \vdots \\ \mathbb{b}_{2n1}^1 & \cdots & \mathbb{b}_{2nn}^1 \end{bmatrix}, \; P = \begin{bmatrix} p_{11} & \cdots & p_{12n} \\ \vdots & \ddots & \vdots \\ p_{2n1} & \cdots & p_{2n2n} \end{bmatrix}, \; \mathbb{E} = \begin{bmatrix} \mathbb{e}_1 \\ \vdots \\ \mathbb{e}_{2n} \end{bmatrix}, \; \mathbb{B}_1 = \begin{bmatrix} \mathbb{b}_1^2 \\ \vdots \\ \mathbb{b}_{2n}^2 \end{bmatrix}$$

where $\mathbb{b}_{ik}^1 \in R^{m \times m}$, $p_{ij} \in R^{m \times m}$, $\mathbb{e}_i \in R^{m \times m}$, $\mathbb{b}_i^2 \in R^{m \times m}$, $i = 1, \cdots, 2n$, $k = 1, \cdots, n$ $j = 1, \cdots, 2n$.

The distributed form of (42)-(45) can be written as:

$$\tau_{3i} = \left(\sum_{j=1}^n \left(a_{ij} + b_i\right)\right)^{-1}\left(\sum_{j=1}^n a_{ij}\tau_{3j} + b_i \hat{\theta}_l^* \phi_l(x_l) - k_{pi}e_i^p - k_{vi}e_i^v\right) \tag{46}$$

$$\dot{\hat{\theta}}_i^* = \frac{1}{\Gamma_f}\sum_{i=1}^{2n}\left(\left(\mathbb{b}_{ji}^1\right)^T p_{ij}\mathbb{e}_i\right)\phi_j^T(x_j) - \frac{c_f}{\Gamma_f}\frac{\phi_i^T(x_i)\left(\left(\mathbb{b}_{ji}^1\right)^T p_{ij}\mathbb{e}_i\right)\phi_i(x_i)}{M_{\theta_I}^2} \tag{47}$$

$$c_f = \begin{cases} 1, \text{if } \left\|\hat{\theta}_1^*\right\|_F = M_{\theta_i} \text{ and } \phi_i^T(x_i)\left(\left(\mathbb{b}_{ji}^1\right)^T p_{ij}\mathbb{e}_i\right)\phi_i(x_i) > 0 \\ 0, \text{otherwise} \end{cases} \tag{48}$$

$$\dot{\hat{\theta}}_l^* = -\frac{1}{\Gamma_l}\mathbb{B}_2^T P\mathbb{E}\phi_l^T(x_l) - \frac{c_l}{\Gamma_l}\frac{\phi_l^T(x_l)\left(\hat{\theta}_l^*\right)^T\mathbb{B}_2^T P\mathbb{E}}{M_{\theta_l}^2}\hat{\theta}_l^* \tag{49}$$

$$c_l = \begin{cases} 1, \text{if } \left\|\hat{\Xi}\right\| = M_{\theta_l} \text{ and } \phi_l^T(x_l)\left(\hat{\theta}_l^*\right)^T\mathbb{B}_2^T P\mathbb{E} > 0 \\ 0, \text{otherwise} \end{cases} \tag{50}$$

Remark 6. With the adaptive operation (42)-(45), the weight matrices can be updated online. Therefore training data sets are not required in the proposed approach. This is different from the conventional Off-Line RBF NN modeling approaches which need the data sets for training in advance. The converging property can be guaranteed by the Lyapunov method, the details can be found in the following context.

Theorem 4. If the directed graph $\mathcal{G}$ has a directed spanning tree, then leader-follower based adaptive synchronized control law (29), (38), (42)-(45) can make the closed loop (41) to be stable under the Assumptions 1-5, that is, the synchronization error E_p and E_v converge to a small residual set.

Proof: Chose a Lyapunov function candidate:

$$V = \frac{1}{2}\mathbb{E}^T P\mathbb{E} + \frac{1}{2}\Gamma_f\left\|\tilde{\Xi}\right\|_F^2 + \frac{1}{2}\Gamma_l\left\|\tilde{\theta}_l^*\right\|_F^2 \tag{51}$$

where $\tilde{\Xi}$ and $\tilde{\theta}_l^*$ are defined as:

$$\tilde{\Xi} = \hat{\Xi} - \Xi,\ \dot{\tilde{\Xi}} = \dot{\hat{\Xi}} \tag{52}$$

$$\tilde{\theta}_l^* = \hat{\theta}_l^* - \theta_l^*,\ \dot{\tilde{\theta}}_l^* = \dot{\hat{\theta}}_l^* \tag{53}$$

Differentiating V with time along closed loop (41):

$$\dot{V} = \frac{1}{2}\mathbb{E}^T(\mathbb{A}^T P + P\mathbb{A})\mathbb{E} + \left(\tilde{H}^T - \Theta^T\right)\mathbb{B}_1^T P\mathbb{E} + \left(\phi_l^T(x_l)\left(\tilde{\theta}_l^*\right)^T - \eta_l^T\right)\mathbb{B}_2^T P\mathbb{E} + \Gamma_f \mathrm{tr}\left(\dot{\tilde{\Xi}}\tilde{\Xi}^T\right) + \Gamma_l \mathrm{tr}\left(\dot{\tilde{\theta}}_l^*\left(\tilde{\theta}_l^*\right)^T\right) \tag{54}$$

Note that, $\mathbb{A}^T P + P\mathbb{A} = -Q$, $\tilde{H}^T\mathbb{B}_1^T P\mathbb{E} = \mathrm{tr}\left(\mathbb{B}_1^T P\mathbb{E}\tilde{H}^T\right)$, $\phi_l^T(x_l)\left(\tilde{\theta}_l^*\right)^T\mathbb{B}_2^T P\mathbb{E} = \mathrm{tr}\left(\mathbb{B}_2^T P\mathbb{E}\phi_l^T(x_l)\left(\tilde{\theta}_l^*\right)^T\right)$, then (54) can be written as:

$$\begin{aligned}\dot{V} = & -\frac{1}{2}\mathbb{E}^T Q\mathbb{E} + \Gamma_f \mathrm{tr}\left(\dot{\tilde{\Xi}}\tilde{\Xi}^T + \frac{1}{\Gamma_f}\mathbb{B}_1^T P\mathbb{E}\tilde{H}^T\right) \\ & +\Gamma_l \mathrm{tr}\left(\dot{\tilde{\theta}}_l^*\left(\tilde{\theta}_l^*\right)^T + \frac{1}{\Gamma_l}\mathbb{B}_2^T P\mathbb{E}\phi_l^T(x_l)\left(\tilde{\theta}_l^*\right)^T\right) - \Theta^T\mathbb{B}_1^T P\mathbb{E} - \eta_l^T\mathbb{B}_2^T P\mathbb{E}\end{aligned} \tag{55}$$

Note that $\tilde{H}^T = [\phi_1^T(x_1), \cdots, \phi_n^T(x_n)]\tilde{\Xi}^T$, substitute adaptive law (42)-(45) into (55):

$$\begin{aligned}\dot{V} = & -\frac{1}{2}\mathbb{E}^T Q\mathbb{E} - \Theta^T\mathbb{B}_1^T P\mathbb{E} - \eta_l^T\mathbb{B}_2^T P\mathbb{E} \\ & -\mathrm{tr}\left(\frac{c_f}{\Gamma_f}\frac{\hat{H}^T\mathbb{B}_1^T P\mathbb{E}}{M_{\theta_I}^2}\hat{\Xi}\tilde{\Xi}^T\right) - \mathrm{tr}\left(\frac{c_l}{\Gamma_l}\frac{\phi_l^T(x_l)\left(\hat{\theta}_l^*\right)^T\mathbb{B}_2^T P\mathbb{E}}{M_{\theta_l}^2}\hat{\theta}_l^*\left(\tilde{\theta}_l^*\right)^T\right)\end{aligned} \tag{56}$$

Note that, the following inequality is always satisfied:

$$\mathrm{tr}\left(c_f\frac{\hat{H}^T\mathbb{B}_1^T P\mathbb{E}}{M_{\theta_I}^2}\hat{\Xi}\tilde{\Xi}^T\right) = c_f\frac{\hat{H}^T\mathbb{B}_1^T P\mathbb{E}}{M_{\theta_I}^2}\mathrm{tr}\left(\hat{\Xi}\hat{\Xi}^T - \hat{\Xi}\Xi\right) \geq 0 \tag{57}$$

$$\operatorname{tr}\left(c_l \frac{\phi_l^T(x_l)\left(\hat{\theta}_l^*\right)^T \mathbb{B}_2^T P\mathbb{E}}{M_{\theta_l}^2} \hat{\theta}_l^*\left(\tilde{\theta}_l^*\right)^T\right) = c_l \frac{\phi_l^T(x_l)\left(\hat{\theta}_l^*\right)^T \mathbb{B}_2^T P\mathbb{E}}{M_{\theta_l}^2} \operatorname{tr}\left(\hat{\theta}_l^*\left(\hat{\theta}_l^*\right)^T - \hat{\theta}_l^*(\theta_l^*)^T\right) \geq 0 \tag{58}$$

By using projections (43) and (45), it is very easy to obtain (57) and (58). Let $\chi = -\Theta^T \mathbb{B}_1^T - \eta_l^T \mathbb{B}_2^T$, then (56) will be:

$$\dot{V} \leq -\frac{1}{2}\mathbb{E}^T Q\mathbb{E} + \chi P\mathbb{E} \tag{59}$$

Let $\lambda_{\min}(Q)$ and $\lambda_{\max}(P)$ denote minimum eigenvalue of matrix Q and maximum eigenvalue of matrix P, respectively. Then, the following inequality will be:

$$\begin{aligned}\dot{V} &\leq -\frac{1}{2}\lambda_{\min}(Q)\|\mathbb{E}\|^2 + \|\chi\|\lambda_{\max}(P)\|\mathbb{E}\| \\ &= -\frac{1}{2}\|\mathbb{E}\|\left(\lambda_{\min}(Q)\|\mathbb{E}\| - 2\|\chi\|\lambda_{\max}(P)\right)\end{aligned} \tag{60}$$

Because $\|\eta_i\| \leq \eta_{0i}$ and $\|\eta_l\| \leq \eta_{0l}$, χ must be bounded and let:

$$\chi_0 \triangleq \sup_{t\geq 0}\|\chi\| \tag{61}$$

$$\dot{V} \leq -\frac{1}{2}\|\mathbb{E}\|\left(\lambda_{\min}(Q)\|\mathbb{E}\| - 2\chi_0\lambda_{\max}(P)\right) \tag{62}$$

From (62), one can see that $\mathbb{E}$ will converge to a residual set $\left\{\mathbb{E}: \|\mathbb{E}\| \leq 2\frac{\lambda_{\max}(P)}{\lambda_{\min}(Q)}\chi_0\right\}$. Hence x_i and x_l will be confined inside compacts $\mathscr{b}(N_{x_i})$ and $\mathscr{b}(N_{x_l})$, respectively. Then, all of the signals of the closed loop will be bounded. ■

Remark 7. The residual set is determined by $\lambda_{\max}(P)$ and $\lambda_{\min}(Q)$. Note that P is a solution of Lyapunov equation $P\mathbb{A} + \mathbb{A}^T P = -Q$. $\mathbb{A}$ is mainly composed by controller gain matrices K_p and K_v. If K_p, K_v and Q are given, P can be computed by using Matlab. The following example is used to show how K_p and K_v affect $\lambda_{\max}(P)$.

Example 1.

Choose $\mathbb{A} = \begin{bmatrix} 0 & 1 \\ -K_p & -K_v \end{bmatrix}$, $K_p = 0.1{:}\,0.1{:}\,10$, $K_v = 0.1{:}\,0.1{:}\,1$, $Q = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$. By using Matlab command $P = \text{lyap}(\mathbb{A}, Q)$, P can be resolved. The relationship of $\lambda_{\max}(P)$ with K_p and K_v is plotted in Figure 1.

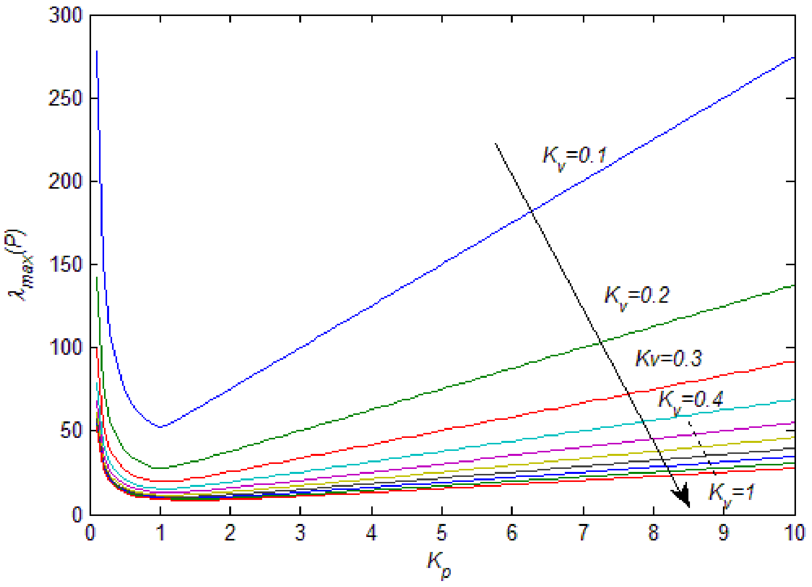

Figure 1. Relationship of $\lambda_{\max}(P)$ with K_p and K_v

Figure 1 shows that $\lambda_{max}(P)$ decreases first and then increases as K_p increases if Q and K_v are fixed. The maximum eigenvalue of the matrix P decreases as K_v increases, however it should be noted that the change rate of the maximum eigenvalue of the matrix P will not significantly increase as K_v is large enough.

Trial and error method can be used in controller parameters selection according to the relationship. First, select an appropriate Q according to the expected converging speed. Second, select a large enough K_v and then use the trial and error method to look for the best K_p. Finally, the previously tuned gains may need to be changed slightly by using a trial and error method.

The controller design procedure can be summarized as:

Step 1: Design RBF NN based compensator $\tau_i = C_{0i}(\dot{q}_i, q_i)\dot{q}_i + G_{0i}(q_i) - M_{0i}(q_i)\hat{\theta}_i^{*}\phi_i(x_i) + M_{0i}\tau_{3i}$, $i = 1, \cdots, n$ for follower manipulator i.

Step 2: Design RBF NN based acceleration estimator $T_1 = (\bar{L} + \bar{B})^{-1}\bar{B}\boldsymbol{I}\hat{\theta}_l^{*}\phi_l(\mathrm{x}_l)$ for leader manipulator.

Step 3: Design RBF NN adaptive synchronized controller as (39), (42)-(45), which can be written as distributed form (46)-(50).

Step 4: Stability analysis (51)-(62).

Remark 8. According to the design steps, the new synchronized control algorithm can be implemented with the enhanced functionality provided from the neuro-agents and leader-follower communicating topology. Its effectiveness can be validated by the stability analysis and the following illustrative examples.

5 Illustrative examples

In this section, 4 cases are presented to validate the performance of the proposed approach from various angles. Case 1 is the test of the proposed approach with 5 robotic manipulators. Case 2 is the test of the proposed approach with 9 robotic manipulators. Case 3 is the comparative test of a conventional feedback control method. Case 4 is the test of the proposed approach on the leader's desired trajectory at different frequencies.

Suppose that all of the leader and follower manipulators had same dynamics which was given as:

$$M_0(q)\ddot{q} + C_0(q,\dot{q})\dot{q} + G_0(q) = \tau + f$$

$$M_0(q) = \begin{bmatrix} J + m_1 + 2m_2\cos(q_2) & m_1 + m_2\cos(q_2) \\ m_1 + q_{02}\cos(q_2) & m_1 \end{bmatrix}$$

$$C_0(q,\dot{q}) = \begin{bmatrix} -m_2\dot{q}_2\sin(q_2) & -m_2(\dot{q}_1 + \dot{q}_2)\sin(q_2) \\ m_2\dot{q}_1\sin(q_2) & 0 \end{bmatrix}$$

$$G_0(q) = \begin{bmatrix} m_3 g\cos(q_1) + m_4 g\cos(q_1 + q_2) \\ m_4 g\cos(q_1 + q_2) \end{bmatrix}$$

$$f = 0.2\big(M_0(q)\ddot{q} + C_0(q,\dot{q})\dot{q} + G_0(q)\big)$$

where $J = 13.33$, $m_1 = 8.98$, $m_2 = 8.75$, $m_3 = 15$, $m_4 = 8.75$, $g = 9.8$. The leader manipulator's desired trajectory and velocity were specified as:

$$\begin{cases} q_1^d = 1 + 0.2\sin(0.5\pi t) \\ q_2^d = 1 - 0.2\cos(0.5\pi t) \end{cases}$$

$$\begin{cases} \dot{q}_1^d = 0.1\pi\cos(0.5\pi t) \\ \dot{q}_2^d = 0.1\pi\sin(0.5\pi t) \end{cases}$$

Case 1. The proposed approach for 5 robotic manipulators

A leader-follower based MRMS composed of five manipulators was considered, where the leader was indexed by 0, and the four followers were indexed by 1,2,3,4, respectively. The communication topology graph is shown in Figure 2. Note that none of the rest followers could directly receive information from the leader, except follower 3 and follower 4. In the topology, follower 4 had no directed path to the other followers and the leader had directed paths to the all followers.

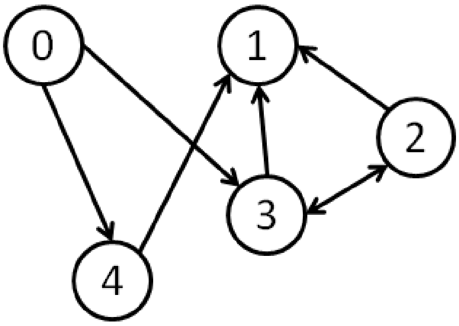

Figure 2. Directed graph of leader-follower system (5 robotic manipulators)

Consequently the adjacent matrix of the graph was set up as:

$$A=\begin{bmatrix}0_2 & 0_2 & 0_2 & 0_2 & 0_2\\ 0_2 & 0_2 & I_2 & I_2 & I_2\\ 0_2 & 0_2 & 0_2 & I_2 & 0_2\\ I_2 & 0_2 & I_2 & 0_2 & 0_2\\ I_2 & 0_2 & 0_2 & 0_2 & 0_2\end{bmatrix}$$

Laplacian of the followers was:

$$\bar{L}=\begin{bmatrix}3I_2 & -I_2 & -I_2 & -I_2\\ 0_2 & I_2 & -I_2 & 0_2\\ 0_2 & -I_2 & I_2 & 0_2\\ 0_2 & 0_2 & 0_2 & 0_2\end{bmatrix}$$

Interconnection relationship that is the block diagonal matrix between the leader and its followers was given as:

$$\bar{B}=\mathrm{diag}\{0_2,0_2,I_2,I_2\}$$

The controller parameters were selected as: $K_p=\mathrm{diag}(2,\cdots,2)\in R^{8\times 8}$, $K_v=\mathrm{diag}(10,\cdots,10)\in R^{8\times 8}$, $\Gamma_{\mathrm{f}}=20,\ \Gamma_{\mathrm{l}}=20$.

Figure 3 is the position synchronization performance of joint-1 of the MRMS, where dashed line is the joint-1 position of leader manipulator, others are the joint-1 position of follower manipulators. Figure 4 is the position synchronization performance of joint-2 position of the MRMS. Figure 5 is the velocity synchronization performance of joint-1 of the MRMS, where dashed line is the joint-1 velocity of leader manipulator, others are the joint-1 velocity of follower manipulators. Figure 6 is the velocity synchronization performance of joint-2 of the MRMS. Simulation results of Figure 3-6 show that the followers' joint position and velocity can converge to leader's joint position and velocity with acceptable small residual errors.

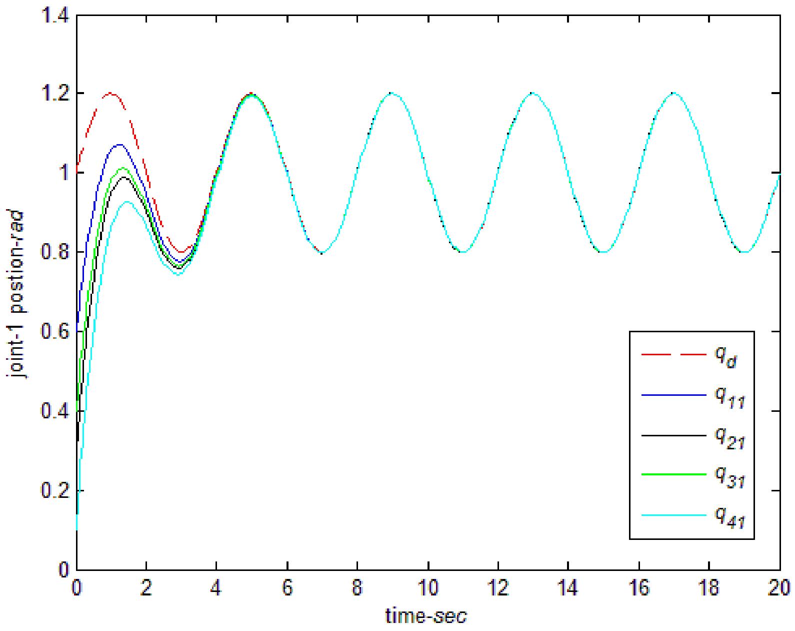

Figure 3 Joint-1 position (the proposed approach for 5 robotic manipulators)

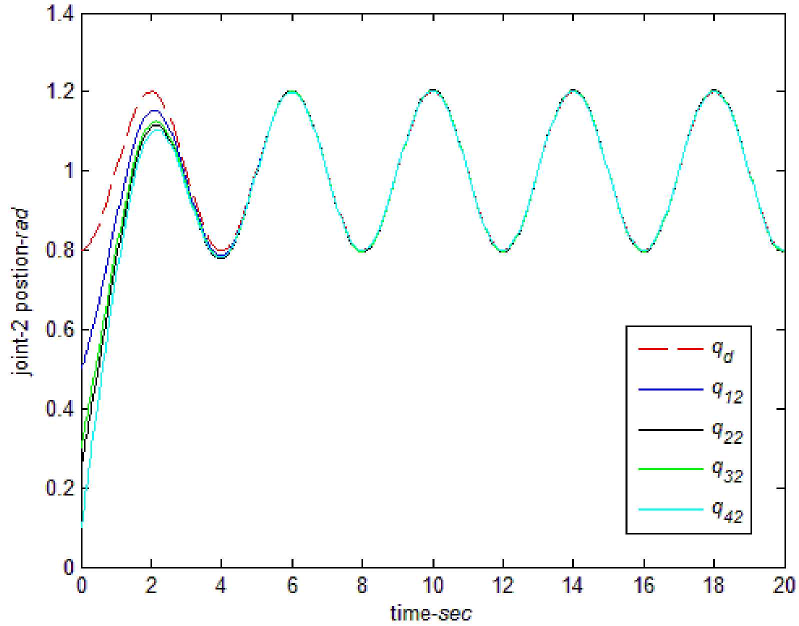

Figure 4 Joint-2 position (the proposed approach for 5 robotic manipulators)

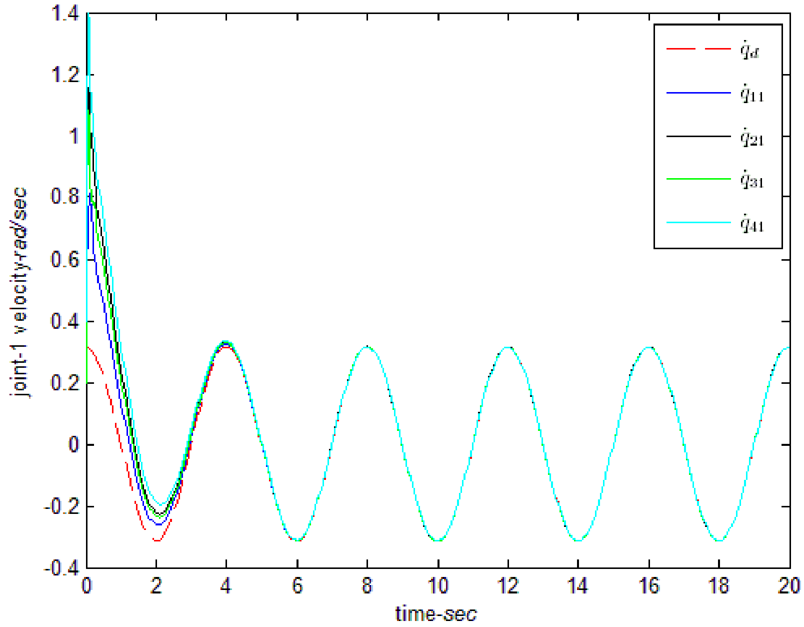

Figure 5 Joint-1 velocity (the proposed approach for 5 robotic manipulators)

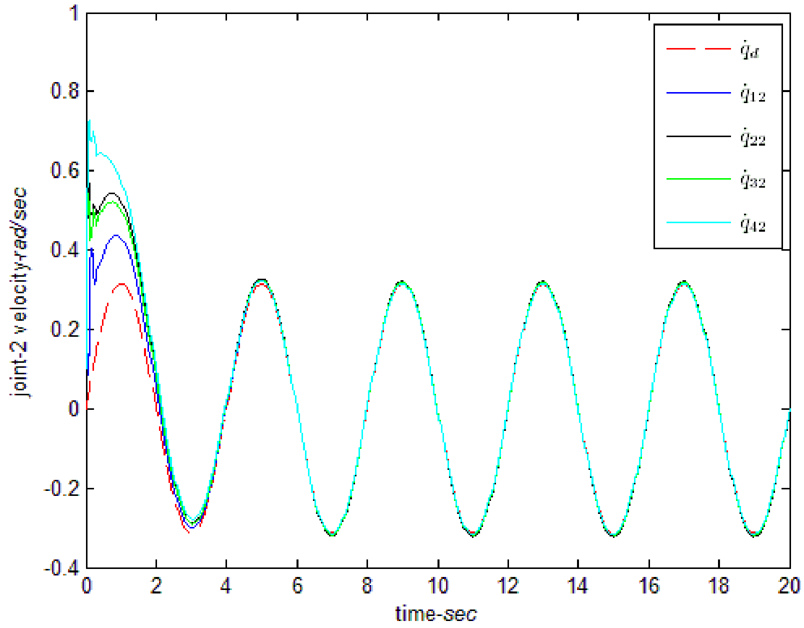

Figure 6 Joint-2 velocity (the proposed approach for 5 robotic manipulators)

To show the relationship of synchronization errors with controller parameter matrices K_p and K_v, the performances of different K_p of synchronization error of joint-1 are shown in Figure 7. The values of $\|\mathbb{E}\|$ under different K_p are given in Table 1. The simulation results show that the synchronization errors will decrease first and then increase as K_p increases. These results are consistent with the conclusion drawn from Example 1.

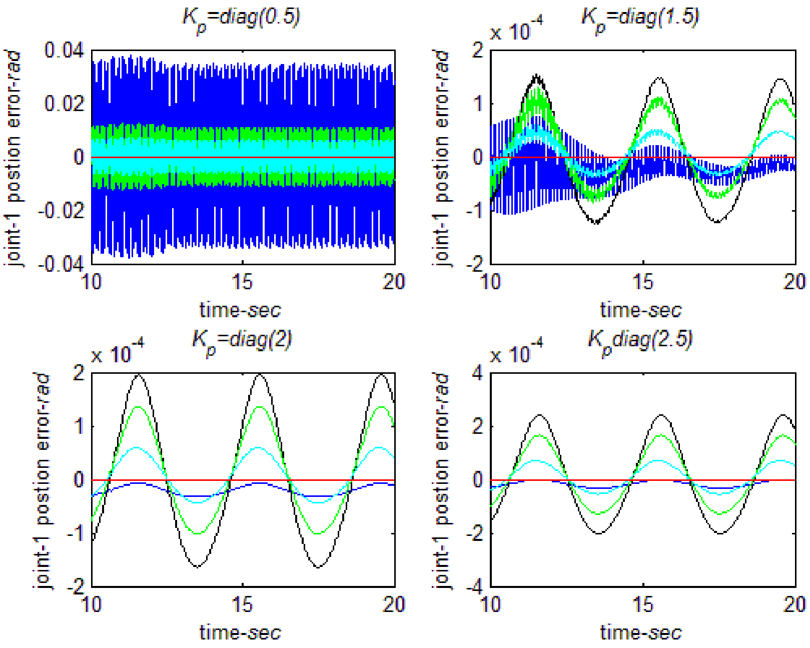

Figure 7 Residual synchronization errors of joint 1 with different K_p ($K_v = \text{diag}(10),\ 10 \le t \le 20\ sec$)

Remark 9. In classical theory, K_p is the proportion gain. In general, the synchronization error will decrease as K_p increase. If K_p is larger than critical value the system will be oscillated or event un-stable. Then, the synchronization error will increase after K_p is larger than that value.

Table 1 The norm of synchronization errors of all joints with respect to different K_p

Feedback gain	$\|\mathbb{E}\|$ $(K_v = \text{diag}(10),\ 10 \le t \le 20\ sec)$
$K_p = \text{diag}(0.5)$	97.0807
$K_p = \text{diag}(1)$	4.6594
$K_p = \text{diag}(1.5)$	0.0936
$K_p = \text{diag}(2)$	0.0314
$K_p = \text{diag}(2.5)$	0.0287
$K_p = \text{diag}(3)$	0.0283
$K_p = \text{diag}(3.5)$	0.0292
$K_p = \text{diag}(4)$	0.0305
$K_p = \text{diag}(4.5)$	0.0318
$K_p = \text{diag}(5)$	0.0333
$K_p = \text{diag}(5.5)$	0.0344
$K_p = \text{diag}(6)$	0.0354

Case 2. The proposed approach for 9 robotic manipulators

In this case, a leader-follower based MRMS composed of nine manipulators was tested, where the leader was indexed by 0, and the eight followers were indexed by $1,\cdots,8$ respectively. Figure 6 shows the communication topology graph with a directed spanning tree.

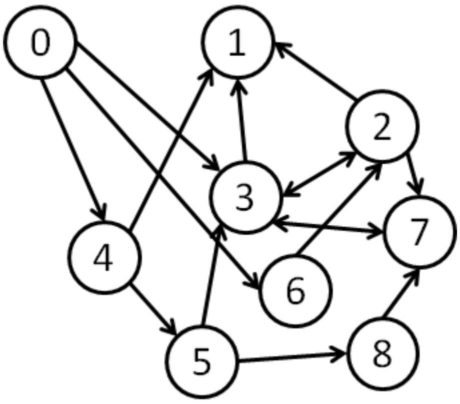

Figure 8. Directed graph of leader-follower system (9 robotic manipulators)

Consequently the adjacent matrix of the graph was set up as:

$$A = \begin{bmatrix} 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & I_2 & I_2 & I_2 & 0_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & 0_2 & I_2 & 0_2 & 0_2 & I_2 & 0_2 & 0_2 \\ I_2 & 0_2 & I_2 & 0_2 & 0_2 & I_2 & 0_2 & I_2 & 0_2 \\ I_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & 0_2 & 0_2 & I_2 & 0_2 & 0_2 & 0_2 & 0_2 \\ I_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & I_2 & I_2 & 0_2 & 0_2 & 0_2 & 0_2 & I_2 \\ 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & I_2 & 0_2 & 0_2 & 0_2 \end{bmatrix}$$

Laplacian of the followers was:

$$\bar{L} = \begin{bmatrix} 3I_2 & -I_2 & -I_2 & -I_2 & 0_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & 2I_2 & -I_2 & 0_2 & 0_2 & -I_2 & 0_2 & 0_2 \\ 0_2 & -I_2 & 3I_2 & 0_2 & -I_2 & 0_2 & -I_2 & 0_2 \\ 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & 0_2 & -I_2 & I_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & -I_2 & -I_2 & 0_2 & 0_2 & 0_2 & 3I_2 & -I_2 \\ 0_2 & 0_2 & 0_2 & 0_2 & -I_2 & 0_2 & 0_2 & I_2 \end{bmatrix}$$

Interconnection relationship that is the block diagonal matrix between the leader and its followers was given as:

$$\bar{B} = \mathrm{diag}\{0_2, 0_2, I_2, I_2, 0_2, I_2, 0_2, 0_2\}$$

The controller parameters were selected as: $K_p = \mathrm{diag}(2,\cdots,2) \in R^{16\times16}$, $K_v = \mathrm{diag}(10,\cdots,10) \in R^{16\times16}$, $\Gamma_{\mathrm{f}} = 20,\ \Gamma_{\mathrm{l}} = 20.$

Figures 9-12 show performance. It is obvious that the performances are satisfactory. In intuition the performance might be deteriorated greatly with the increase of robotic manipulators and/or the degree of freedom (DOF) of each robotic manipulator. However, the simulation results show the numbers of robotic manipulators almost do not affect the performances. This is because that the distributed control algorithm is used, in which each robotic manipulator computes the control law for itself. If the communication topology graph has a directed spanning tree, the proposed approach will make the closed loop to be stable. The numbers will not heavily influence the performance. In the case of increasing DOF, the performances will almost not be affected if each robotic manipulator has a controller with proper computing ability. Due to the space limitations, the simulation of high DOF case is omitted here. The readers can simulate it in MATLAB easily.

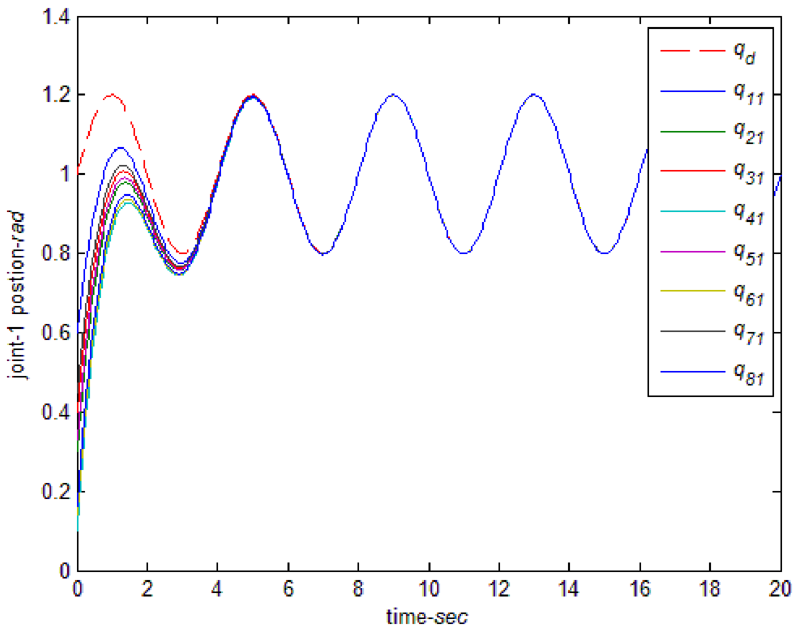

Figure 9 Joint-1 position (the proposed approach for 9 robotic manipulators)

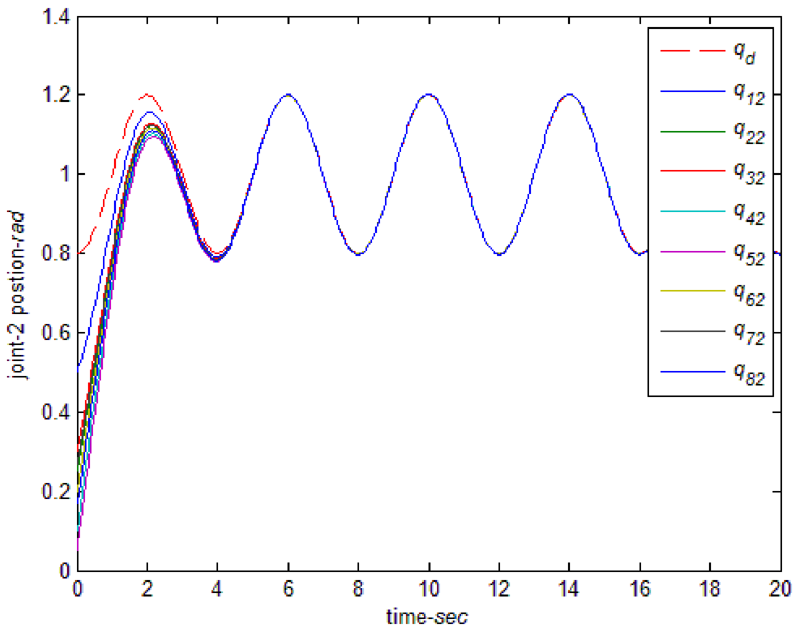

Figure 10 Joint-2 position (the proposed approach for 9 robotic manipulators)

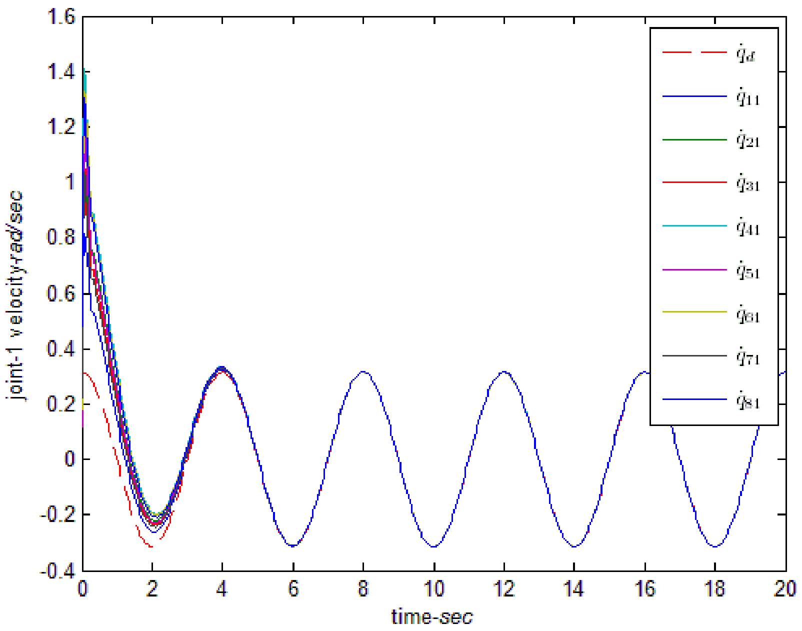

Figure 11 Joint-1 velocity (the proposed approach for 9 robotic manipulators)

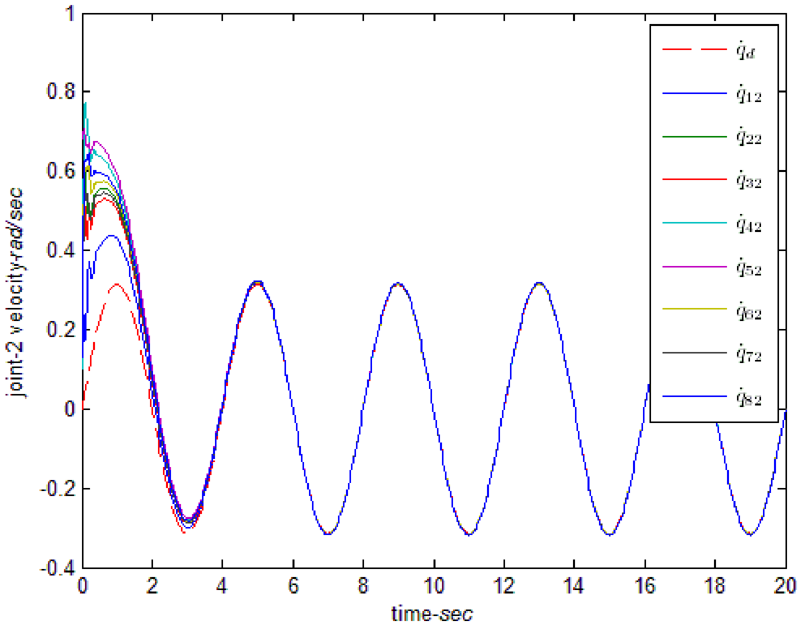

Figure 12 Joint-2 velocity (the proposed approach for 9 robotic manipulators)

Case 3. The conventional feedback control

In this case, a conventional feedback consensus control law was used to control the 5 robotic manipulators. The communication topology graph is shown in Figure 2. The control law was designed as:

$$\tau_i = C_{0i}(\dot{q}_i, q_i)\dot{q}_i + G_{0i}(q_i) - M_{0i}(q_i)\hat{h}_i + M_{0i}\tau_{3i}$$

$$\tau_{3i} = \left(\sum_{j=1}^{n}\left(a_{ij} + b_i\right)\right)^{-1}\left(\sum_{j=1}^{n} a_{ij}\tau_{3j} + b_i\hat{\ddot{q}}_l - k_{pi}e_i^p - k_{vi}e_i^v\right)$$

where $k_p = \text{diag}(2,2)$, $k_v = \text{diag}(10,10)$, $\hat{h}_i$ and $\hat{\ddot{q}}_l$ can be estimated by the designers' experience.

This control law is a common consensus algorithm. It could be designed according to many existing methods, such as [19, 24]. The dynamics uncertainty and leader's acceleration were estimated offline. Assumed the dynamic uncertainty $\Delta M_i(q_i) = 0.1M_{0i}$, $\Delta C_i(\dot{q}_i, q_i) = 0.1C_{0i}(\dot{q}_i, q_i)$, $\Delta G_i(q_i) = 0.1G_{0i}(q_i)$. The estimation of the system uncertainty $\hat{h}_i = 0.6h_i$ and the estimation of leader's acceleration $\hat{\ddot{q}}_l = 0.8\ddot{q}_l$.

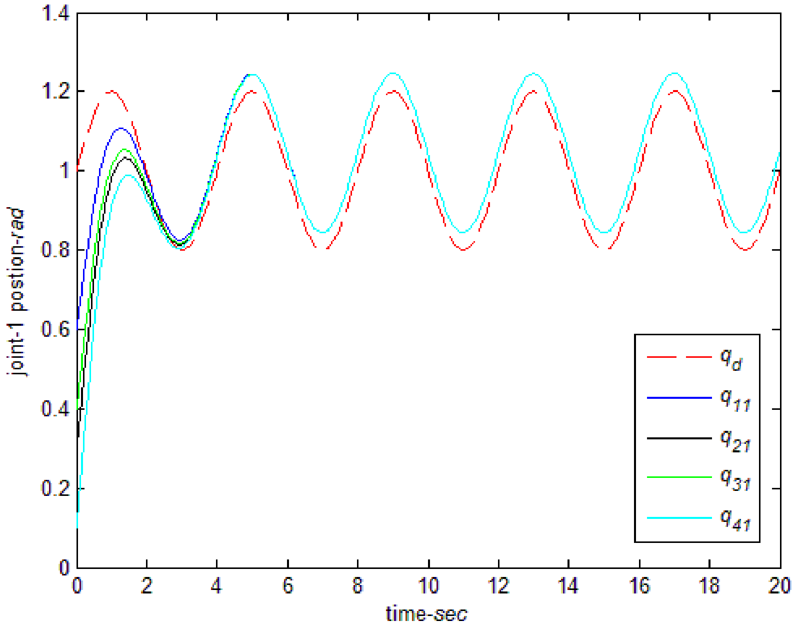

Figure 13 Joint-1 position (the conventional approach for 5 robotic manipulators)

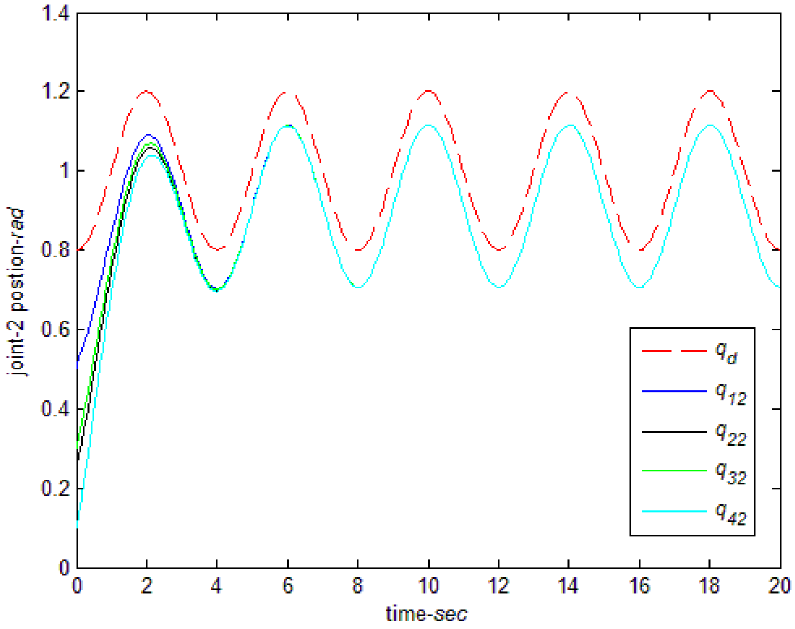

Figure 14 Joint-2 position (The conventional approach for 5 robotic manipulators)

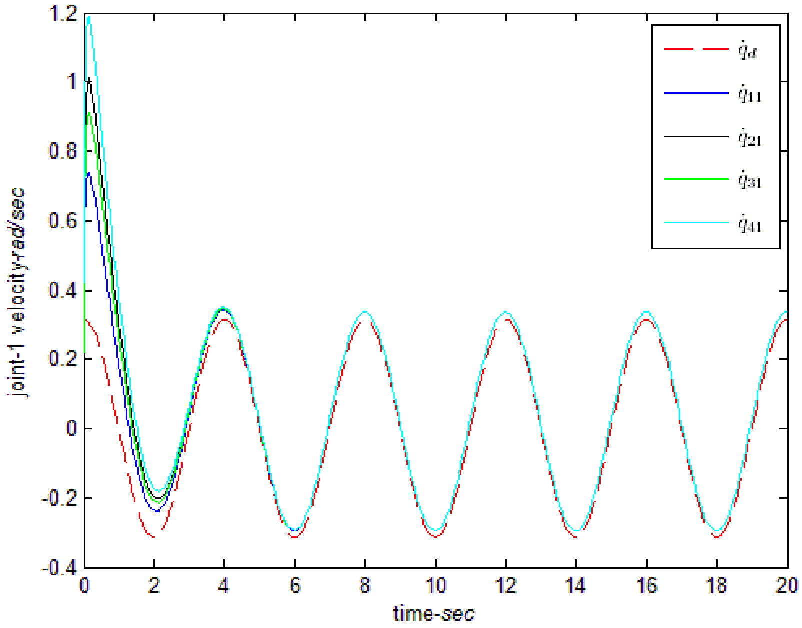

Figure 15 Joint-1 velocity (the conventional approach for 5 robotic manipulators)

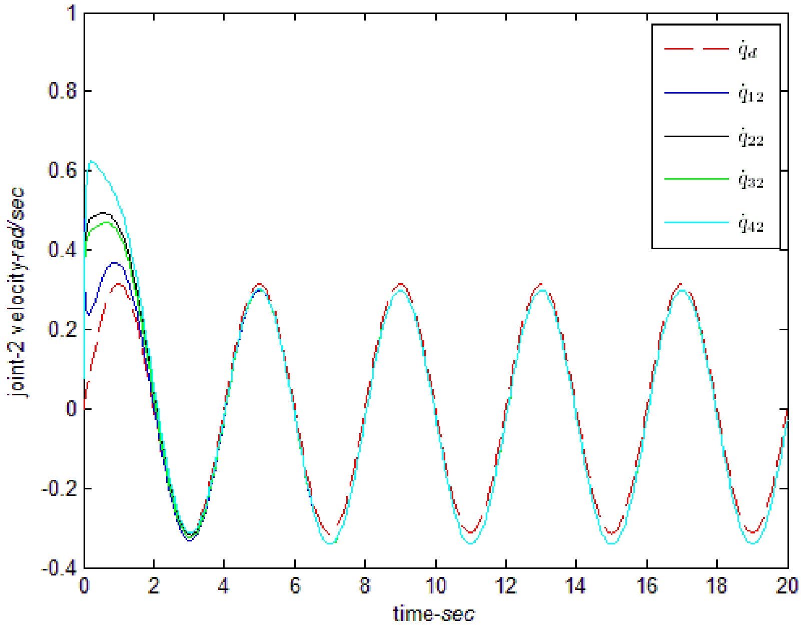

Figure 16 Joint-2 velocity (the conventional approach for 5 robotic manipulators)

Figures 13-16 are the performances obtained from the conventional consensus control. It is obvious that the performances are not good. There are residual consensus errors. This is because that the system uncertainty and leader' acceleration cannot be obtained accurately. This case also validates the necessity and effective of the proposed neuro-agents in estimating the system modeling error and leader's acceleration.

Case 4. The proposed approach under different frequencies for the leader's desired trajectory

This further test was used to tracking different frequencies for the leader's desired trajectory. The communication topology graph is shown in Figure 2. The controller parameters were selected as those in Case 1.

The leaders' trajectories used in Figures 15-18 were given as:

$$\begin{cases} q_1^d = 1 + 0.2\sin(0.1\pi t) \\ q_2^d = 1 - 0.2\cos(0.1\pi t) \end{cases}$$

$$\begin{cases} \dot{q}_1^d = 0.02\pi\cos(0.1\pi t) \\ \dot{q}_2^d = 0.02\pi\sin(0.1\pi t) \end{cases}$$

The leaders' trajectories used in Figures 19-22 were given as:

$$\begin{cases} q_1^d = 1 + 0.2\sin(3\pi t) \\ q_2^d = 1 - 0.2\cos(3\pi t) \end{cases}$$

$$\begin{cases} \dot{q}_1^d = 0.6\pi\cos(3\pi t) \\ \dot{q}_2^d = 0.6\pi\sin(3\pi t) \end{cases}$$

From Figures 17-24, it can be seen that the performances of the proposed approach are good enough under different frequencies for the leader's desired trajectory. Again, the simulation results validate the synchronized capability of the proposed approach.

Remark 10. In this paper, different frequencies of the leader's desired trajectory are tested. The performances of the proposed approach are good and accepted. It is more interesting to find the explicit relationship between the frequency of the leader's desired trajectory with the synchronization performance. However, it is not an easy job. Because most of existing synchronized control approaches are designed and analyzed in the time domain, which cannot give the explicit relationship between the frequency and the performance. The main purposed of this paper is to give a stable synchronized control based on the neural networks. The authors will consider synchronized controller design in the frequency domain in their following works.

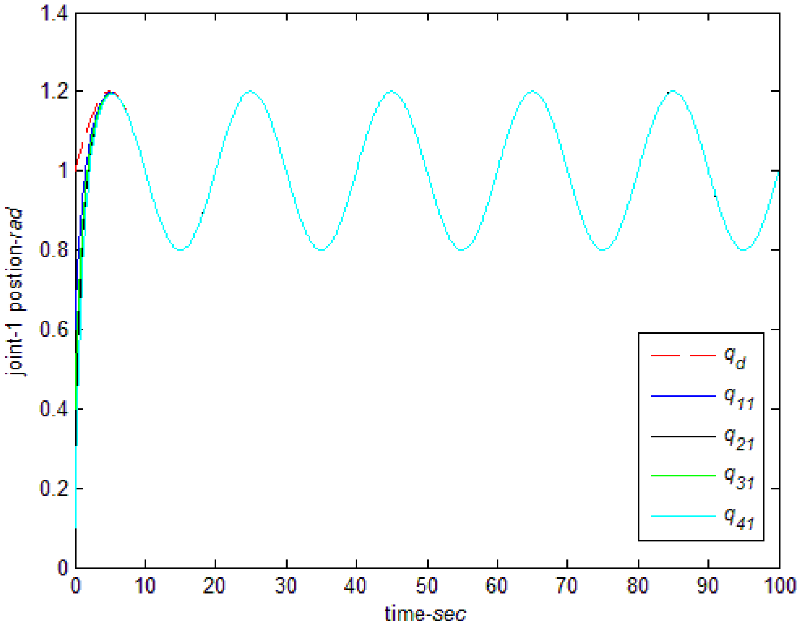

Figure 17 Joint-1 position (the proposed approach for lower frequency, $f = 0.05$Hz)

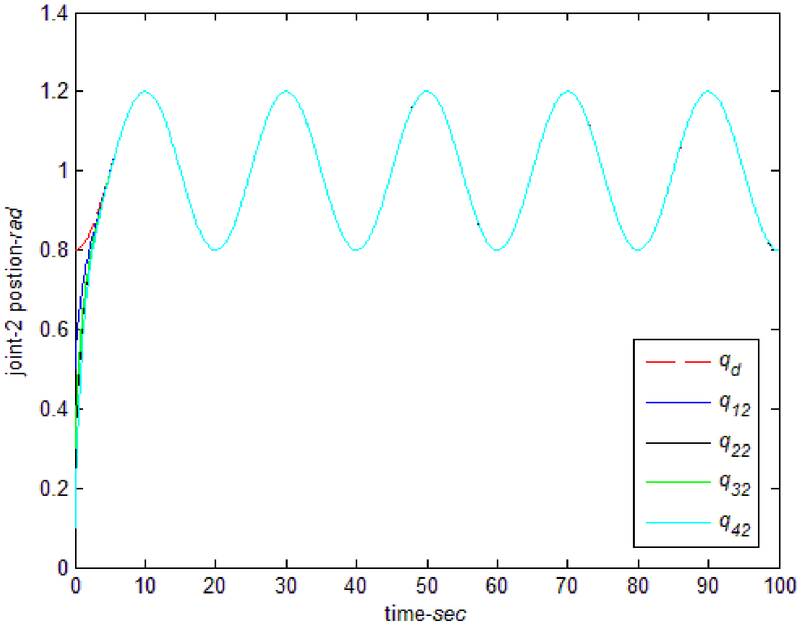

Figure 18 Joint-2 position (the proposed approach for lower frequency, $f = 0.05$Hz)

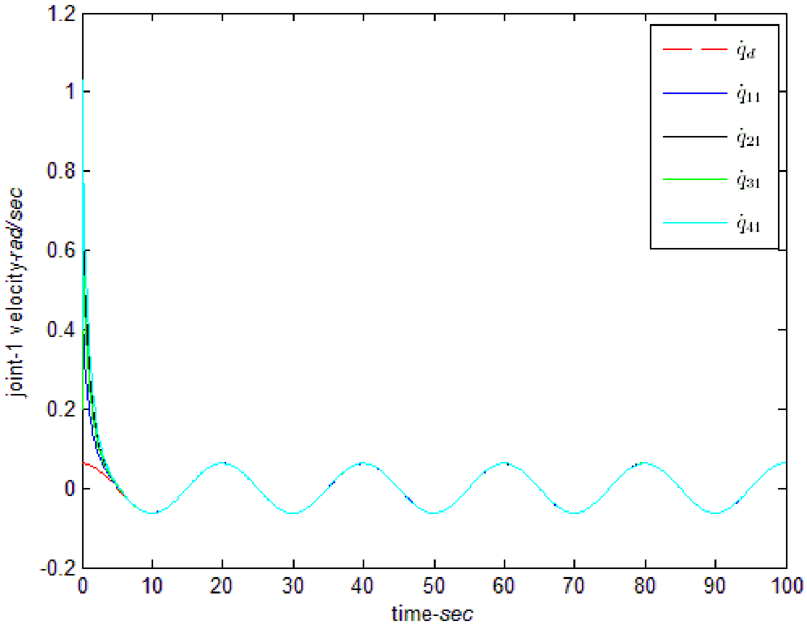

Figure 19 Joint-1 velocity (the proposed approach for lower frequency, $f = 0.05$Hz)

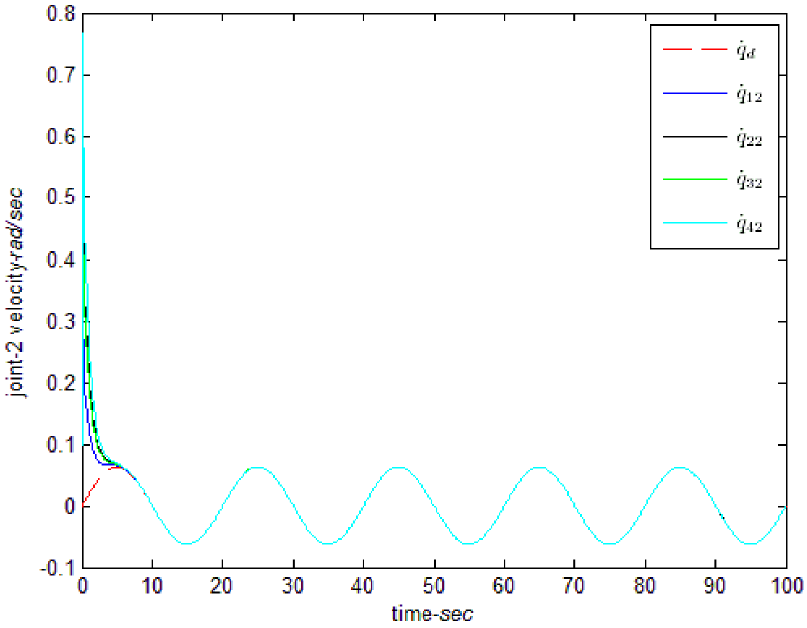

Figure 20 Joint-2 velocity (the proposed approach for lower frequency, $f = 0.05$Hz)

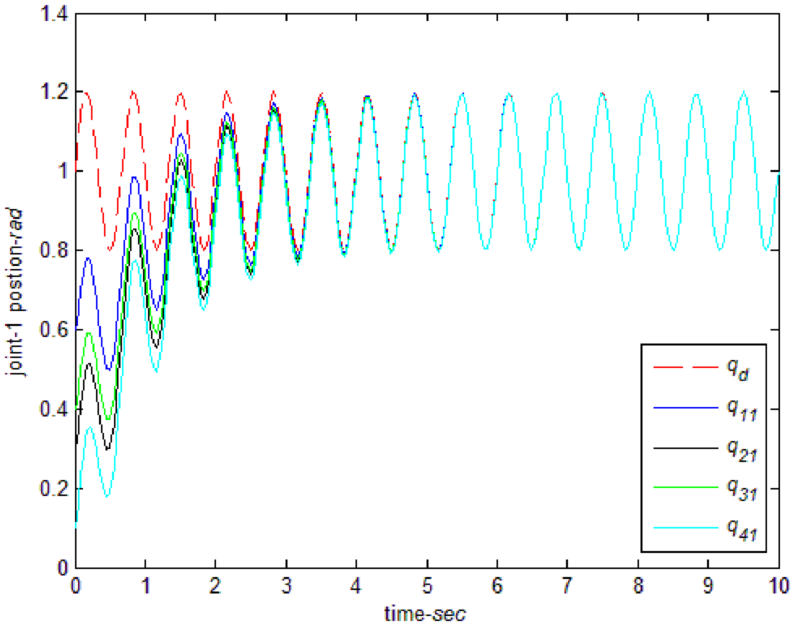

Figure 21 Joint-1 position (the proposed approach for higher frequency, $f = 1.5$Hz)

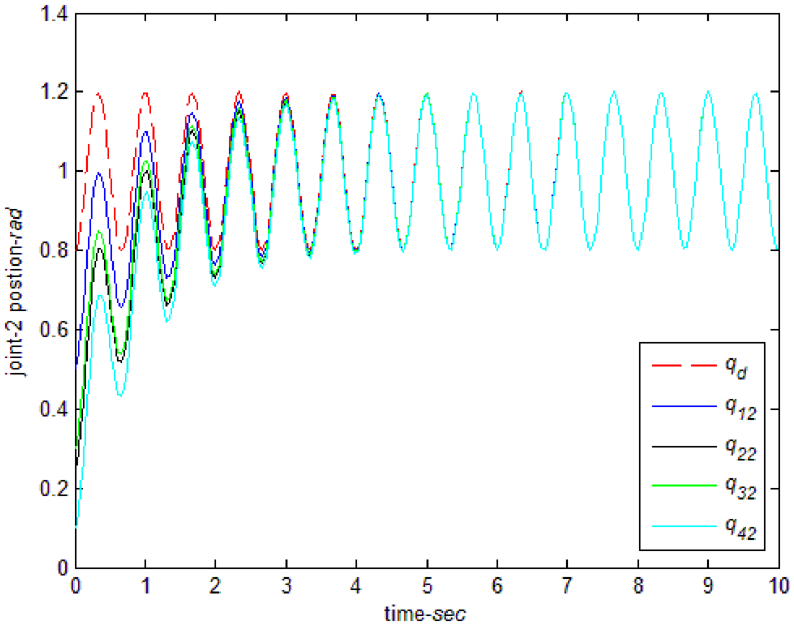

Figure 22 Joint-2 position (the proposed approach for higher frequency, $f = 1.5$Hz)

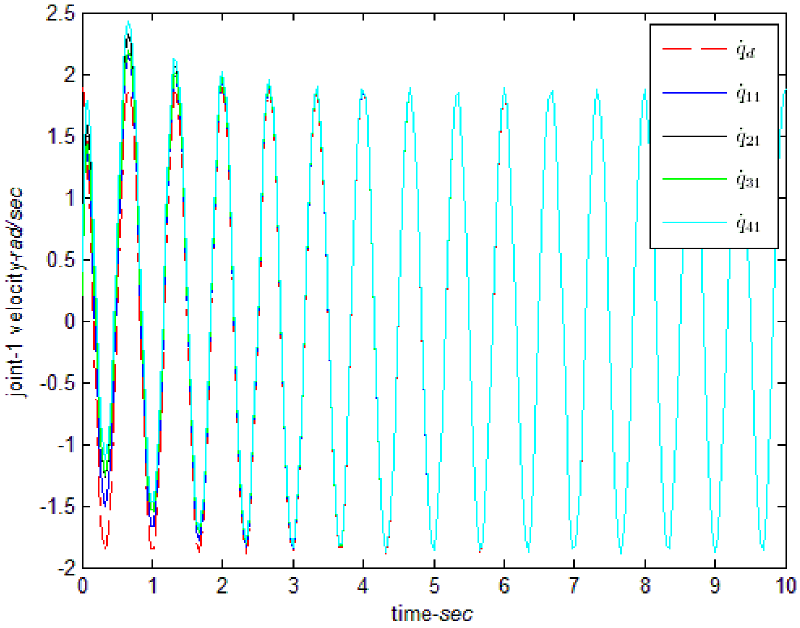

Figure 23 Joint-1 velocity (the proposed approach for higher frequency, $f = 1.5$Hz)

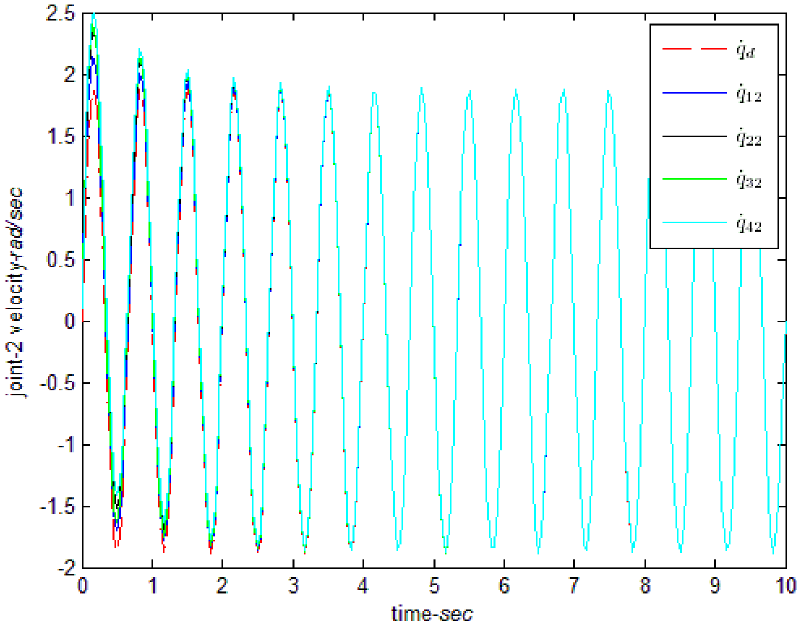

Figure 24 Joint-2 velocity (the proposed approach for higher frequency, $f = 1.5$Hz)

Remark 10. In this paper, different frequencies of the leader's desired trajectory are tested to show the acceptable performance. It is more interesting to find the explicit relationship between the frequency of the leader's desired trajectory with the synchronization performance. However, it is not an easy job. Because most of existing synchronized control approaches are designed and analyzed in time domain, which cannot give the explicit formulation to link the frequency and the performance. The main purposed of this paper is to give a stable synchronized control based on the neural networks. The authors will consider synchronized controller design in the frequency domain in their following work.

Remark 11. Although the inverse kinematics problem is very difficult such that the desired trajectory of most industrial serial robotic manipulators planned in their joint space, it is worthwhile pointing out that actually, this is not always the case. To plan trajectories in the joint space, usually, inverse kinematics problem needs to be resolved first, since most the time, the tasks carried out by the robotic manipulators are assigned in the task space except the robot manipulators are taught by operators via teach-pendants.

Remark 12. The proposed synchronized strategy can be applied to the coordinated control problem of multiple transport robotic manipulators only when the strict conditions that all the robotic manipulators are exactly the same and their bases are set up with the same orientation are satisfied. Under these two conditions, the relative positions of the robotic manipulators can be guaranteed to be constant after they can be synchronized, which is known to be a precondition to transport a common object. However, these two conditions cannot be met sometimes in real environment, which means that the proposed approach should be wisely used in applications.

6 Conclusions

By theoretical analysis and simulation demonstrations, a novel leader-follower based synchronized control framework has been initially constructed for MRMS. A directed graph based synchronization error is defined by using the leader-follower topology. In light of fully taking advantage of using the RBF NN and adaptive control principles, the proposed approach has well claimed capacity to compensate follower manipulators' uncertainty and estimate leader manipulator's acceleration in terms of convergence and stability. It is worth noting that the study has provided a good example to develop new solutions to the challenging and practically highly demanded issues encountered in MRMS. In addition this study provides an exemplary showcase with effectively to integrate several cross boundary theoretical results in the fields of control, parameter estimation, and neuro-computing, which reflects the philosophy of interdisciplinary study having been the tendency in emerging research. The immediate future work will be applying this new scheme to resolve some ad hoc problems (such as time delay and time varying information topology) commonly encountered in MRMS.

Acknowledgements

This work is partially supported by the National Nature Science Foundation of China under Grant 61004080, 61273188, Shandong Provincial Natural Science Foundation under Grant ZR2011FM003, China and the

Fundamental Research Funds for the Central Universities of China, Development of key technologies project of Qingdao Economic and Technological Development Zone under Grant 2011-2-52, Taishan Scholar Construction Engineering Special funding. Finally the authors are grateful to the editor and the anonymous reviewers for their helpful comments and constructive suggestions with regard to the revision of the paper.

Reference

[1] W. Gueaieb, S. Al-Sharhan, B. Miodrag, Robust computationally efficient control of cooperative closed-chain manipulators with uncertain dynamics, Automatica, **43**(5) (2007) 842-851

[2] W. Gueaieb, F. Karray, A robust hybrid intelligent position/force control scheme for cooperative manipulators, IEEE Transactions on Mechatronics, 12(2) (2007) 109-125

[3] H. Nijmeijer, A. Rodriguez-Angeles, Synchronization of mechanical systems, Singapore, World Scientific, 2003

[4] D. Sun, Synchronization and control of multiagent systems, CRC Press, Taylor & Francis Group, 2010

[5] J. C. Martinez-Rosas, M. A. Arteaga, A. M. Castillo-Sanchez, Decentralized control of cooperative robots without velocity-force measurements, Automatica, 42(2) (2006) 329-336

[6] J. Gudino-Lau, M. A. Artegag, Dynamic model and simulation of cooperative robots: a case study, Robotica, 23(5) (2005) 615-624

[7] Y.-H. Liu, Y. Xu, M. Bergerman, Cooperation control of multiple manipulators with passive joints, IEEE Transactions on Robotics and Automation, 15(2) (1999) 258-267

[8] H. Kawasaki, S. Ueki, S. Ito, Decentralized adaptive coordinated control of multiple robot arms without using a force sensor, Automatica, 42(3) (2006) 481-488

[9] R. Rocha, J. Dias, A. Carvalho, Cooperative multi-robot systems: A study of vision-based 3-D mapping using information theory, Robotics and Autonomous Systems, 53(3-4) (2005) 282-311

[10] D. Zhang, L. Wang, J. Yu, Geometric topology based cooperation for multiple robots in adversarial environments, Control Engineering Practice, 16(9) (2008) 1092-1100

[11] H.-K. Lee, M. J. Chung, Adaptive controller of a master-slave system for transparent teleoperation. Journal of Robotic Systems, 15(8) (1998) 465-475

[12] D. Sun, J. K. Mills, Adaptive synchronized control for coordination of multirobot assembly tasks, IEEE Transactions on Robotics and Automation, 18(4) (2002) 498-510

[13] D. Sun, Position synchronization of multiple motion axes with adaptive coupling control, Automatica, 39(6)

(2003) 997-1005

[14] A. Rodriguez-Angeles, H. Nijmeijer, Mutual synchronization of robots via estimated state feedback: a cooperative approach, IEEE Transactions on Control Systems Technology, 12(4) (2004) 542-554

[15] W.-H. Zhu, On adaptive synchronization control of coordinated multirobots with flexible/rigid constraints, IEEE Transactions on Robotics, 21(3) (2005) 520-525

[16] D. Zhao, S. Li, F. Gao, Q. Zhu, Robust adaptive terminal sliding mode-based synchronised position control for multiple motion axes systems, IET Control Theory and Applications, 3(1) (2009) 136-150

[17] D. Zhao, C Liu, Q. Zhu, Low-pass-filter-based position synchronization sliding mode control for multiple robotic manipulator systems, IMechE Part I: Journal of Systems and Control Engineering, 225(8) (2011) 1136-1148

[18] N. Chopra, M. W. Spong, R. Lozano, Synchronization of bilateral teleoperators with time delay, Automatica, 44(8) (2008) 2142-2148

[19] R. Olfati-Saber, R. M. Murray, Consensus problems in networks of agent with switching topology and time-delay, IEEE Automatic Control, 49(9) 2004 1520-1533

[20] L. P. J. Veelenturf, Analysis and Applications of Artificial Neural Networks, New York, Prentice Hall, 1995

[21] R.-J. Wai, Tracking control based on neural network strategy for robot manipulator, Neurocomputing， 51 (2003) 425-445.

[22] O. Mohareri, R. Dhaouadi, A. B. Rad, Indirect adaptive tracking control of a nonholomic mobile robot via neural networks, Neurocomputing, 88 (2012) 54-66.

[23] B. Daachi, T. Madani, A. Benallegue, Adaptive neural controller for redundant robot manipulators and collision avoidance with mobile obstacles, Neurocomputing, 79 (2012) 50-60.

[24] S. Khoo, L, Xie, Z. Man, Robust finite-time consensus tracking algorithm for multirobot systems, IEEE Transactions on Mechatronics, 14(2) (2009) 219-228

[25] Y. Hong, G. Chen, L. Bushnell, Distributed observers design for leader-following control of multi-agent networks, Automatica, 44(3) (2008) 846-850

[26] D. S. Broomhead, D. Lowe, Multivariable functional interpolation and adaptive networks, Complex Systems, 2, (1988) 321-355

[27]Q. Zhu, S. Fei, T. Zhang, T. Li, Adaptive RBF neural-networks control for a class of time-delay nonlinear systems, Neurocomputing, 71(16-18) (2008) 3617-3624

[28] G. Bugmann, Normalized Gaussian Radial Basis Function networks, Neurocomputing, 20(1-3), (1998), 97-110

[29] H. K. Khalil, Nonlinear Systems (3^{rd}), New Jersey, Prentice Hall, 2002

[30] W. Ren, Multi-vehicle consensus with a time-varying reference state, Systems & Control Letters, 56(7-8) (2007) 474-483

[31] W. Ren, R. W. Beard, Distributed Consensus in Multi-vehicle Cooperative Control, Springer-Verlag, New-York, 2007

[32] W. Ren, R. W. Beard, Consensus seeking in multiagent systems under dynamically changing interaction topologies consensus seeking in multiagent systems under dynamically changing interaction topologies, IEEE Transactions on Automatic Control, 50(5) (2005) 655-661

[33] M. W. Spong, S. Hutchinson, M. Vidyasagar, Robot Modeling and Control, New York, John Wiley and Sons, 2006

Dongya Zhao received BEng from Shandong University, Jinan, China, in 1998, MSc from Tianhua Institute of Chemical Machinery & Automation, Lanzhou, China, in 2002 and PhD from Shanghai Jiao Tong University, Shanghai, China, in 2009. He was a research fellow in Nanyang Technological University during 7/2011 to 7/2012. Since 2002, he has been with College of Chemical Engineering, China University of Petroleum, where he is currently an Associate Professor. His research interests include robot control, sliding mode control, process modeling and control, nonlinear system control and analysis.

Quanmin Zhu received his M.Sc. from Harbin Institute of Technology, China in 1983 and Ph.D. from University of Warwick, UK, in 1989. He is currently a Professor in control systems at Department of Engineering, Design and mathematics, University of the West of England, Bristol, UK. His main research interest is in the area of nonlinear system modelling, identification, and control. His other research interest is in investigating electrodynamics of acupuncture points and sensory stimulation effects in human body, modelling of human meridian systems, and building up electro-acupuncture instruments.

Ning Li was born in Shandong, China, in 1974. She received the B.S. and M.S. degrees from Qingdao University of Science and Technology, Qingdao, China, in 1996 and 1999, respectively, and the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, in 2002. She is currently an associate professor of the Department of Automation, Shanghai Jiao

Tong University, Shanghai, China. Her research interests include modeling and control of complex systems, predictive control, and fuzzy systems.

Shaoyuan Li was born in Hebei, China, in 1965. He received the B.S. and M.S. degrees in automation from Hebei University of Technology, Tianjin, China, in 1987 and 1992, respectively, and the Ph.D. degree from the Department of Computer and System Science, Nankai University, Tianjin, in 1997. He is currently a Professor with the Department of Automation, Shanghai Jiao Tong University, Shanghai, China. His research interests include fuzzy systems, model predictive control, dynamic system optimization, and system identification.