

Adaptive Synchronized Tracking Control for Multiple Robotic Manipulators with Uncertain Kinematics and Dynamics

Dongya Zhao^{a, b*}, Shaoyuan Li^c, Quanmin Zhu^{a, d}

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

a. College of Chemical Engineering, China University of Petroleum, Qingdao, 266555, China.

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

c. Institute of Automation, Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai 200240, China.

d. Bristol Institute of Technology, University of the West of England, Frenchy Campus Coldharbour Lane, Bristol BS16 1QY, UK.

Abstract

In this study, a new adaptive synchronized tracking control approach is developed for the operation of multiple robotic manipulators in the presence of uncertain kinematics and dynamics. In terms of the system synchronization and adaptive control, the proposed approach can stabilize position tracking of each robotic manipulator while coordinating its motion with the other robotic manipulators. On the other hand, the developed approach can cope with kinematic and dynamic uncertainties. The corresponding stability analysis is presented to lay a foundation for theoretical understanding of the underlying issues as well as an assurance for safely operating real systems. Illustrative examples are bench tested to validate the effectiveness of the proposed approach. In addition, to face the challenging issues, this study provides an exemplary showcase with effectively to integrate several cross boundary theoretical results to formulate an interdisciplinary solution.

Keywords: Adaptive control, synchronized control, multiple robotic manipulators, kinematic uncertainty, dynamic uncertainty

1 Introduction

It has been increasingly important to employ multiple robotic manipulators to fulfil a common task

simultaneously in modern manufacturing systems such as assembling, transporting, painting and welding, and so on (Gueaieb et al. 2007; Gueaieb and Karray 2007; Nijmeijer and Rodriguez-Angeles 2003). Such multiple manipulators will have more functions in space and deep seas exploration. The aforementioned industrial applications require large maneuverability and manipulability, for which a single robotic manipulator cannot undertake easily. To effectively achieve these largely demanded task functionalities, an effective solution has been to use cooperative or coordinated multiple robotic manipulators systems (MRMS) (Gudino-Lau and Artagag 2005; Martinez-Rosas et al. 2006). Technically the corresponding algorithms are the most important key issues in MRMS (Kawasaki et al. 2006; Liu et al. 1999; Rocha et al. 2005; Zhang et al. 2008). Commonly kinematic and dynamic relationship of the manipulators must be coordinated during the motion process. It has been noticed that control of such systems still stands as one of the challenging issues in the field of robot control. It should be mentioned that most of the existing studies have focused on the control of single robotic manipulator, which cannot be used in MRMS directly. Cooperative control and master-slave control (Gueaieb et al 2007; Lee and Chung 1998) are conventional approaches in the MRMS. In common, these approaches require internal force measurement in controller design. It may be very difficult to measure or estimate internal force in practice (Sun and Mills 2002; Su 2003). Position synchronized control can coordinate MRMS without requiring internal force (Cheong et al. 2009; Chung and Slotine 2009). By virtue of efficient implementation, position synchronized control algorithms have attracted extensive attentions from academic research to industrial applications.

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. To use a structural way, the literatures about the position synchronized control are classified into four cases.

(1) Joint space synchronized control for MRMS

In the joint space synchronized control, the joints of MRMS will be synchronized or consensus, which means that all of the manipulators' joints position will be the same or keep a constant difference. In light of cross-coupling technique, an adaptive synchronized control algorithm has been designed for multi-robot assembly tasks (Sun and Mills 2002; Su 2003). A mutual synchronized control approach is studied with velocity observer (Rodriguez-Angeles and Nijmeijer 2004). By removing some restrictive assumptions, an adaptive position synchronized control algorithm is developed for multi-robots with flexible/rigid constraints by literature (Zhu 2005). A robust adaptive terminal sliding mode synchronized control scheme has been developed for MRMS, which can achieve finite-time stability (Zhao et al. 2009). The MRMS achieves synchronization in their joint space by using the above approaches, which deal with dynamic uncertainty by using adaptive control or robust

control. Though the joint position synchronized control can resolve some problems in the MRMS. However, the task space synchronized control algorithms of MRMS are more necessary in the practice. Compared with the existing approaches in joint space, the proposed approach is designed in task space and can deal with kinematic and dynamic uncertainties together.

(2) Task space synchronized control for MRMS

To achieve synchronized control objective, the desired position trajectories of end-effector of manipulators must be planned in task space, such as Cartesian space (Chen et al. 2011). This is the basis for MRMS based synchronized control algorithms. A kernel assumption of existing synchronized algorithms is that kinematics and Jacobian matrix of the robotic manipulator can be accurately obtained from joint space to Cartesian space. However it is difficult to obtain these accurate kinematic parameters in practice. Kinematic uncertainty is a practical and challenging problem in robot control (Cheah et al. 2006; Liu et al. 2008). For example, the robotic end-effector may be often interchanged with the other end-effector tools with different lengths, the manipulator picks up a tool with unknown length and unknown grasping points, there may be joint offsets in manipulators (Dixon 2007). From the above analysis, it can be concluded that kinematic uncertainty is a separate problem from dynamic uncertainty (Braganza et al. 2008; Cheah 2008). Without including kinematic uncertainties in robot controller design, it may compromise control performance or even affect the system stability (Cheng et al. 2009; Liang et al. 2010; Wang and Xie 2009a; Wang and Xie 2009b).

There are some achievements in task space synchronized control of MRMS. An adaptive consensus control is designed in light of multi-agent control principle and considers kinematic uncertainties (Cheng et al. 2008a). By the adaptive consensus control, the robots' end-effectors can move towards the same configuration. However, this control algorithm does not consider the robot dynamics which have to be incorporated in practical applications. A task space synchronized control is proposed for MRMS (Liu and Chopra 2012). By exploiting passivity-based synchronization principle, an adaptive synchronized control is designed, which can deal with dynamic uncertainty and time varying communication delays. Note that the kinematic uncertainties have not considered in this paper. A passivity based synchronized control is proposed for MRMS and can deal with uncertain kinematics and dynamics (Wang 2013). Though uncertain kinematics and dynamics are considered in this paper all end effectors of the MRMS are required to track the common desired trajectory which is a limited situation. In practice, the end effectors of MRMS are required to track their own desired trajectories while keep synchronization with each other. In general, the desired trajectories are different from each other, such as several manipulators transfer a workpiece together. Compared with the existing task space synchronized control algorithms, the proposed approach not only

achieves synchronization with considering kinematic and dynamic uncertainty but also can guarantee each manipulator to track different desired trajectories.

(3) Synchronized control for parallel robot

Parallel robots can be treated as a group of serial manipulators holding a load together. Some synchronized control algorithms are designed to improve their performance, which are very good inspiration for the MRMS. A saturated PI synchronous control algorithm is designed for parallel manipulators (Su et al. 2006). By using cross-coupling technique, a synchronous tracking control strategy has been developed without using robot dynamic model explicitly (Sun et al. 2006). A fully adaptive feedforward feedback synchronized control algorithm is designed for Stewart Platform (Zhao et al. 2008). A convex synchronized control method has been developed for a planar parallel manipulator to achieve several control performance specifications simultaneously (Ren et al. 2008). A finite time position synchronized control strategy has been developed for a 6 DOF parallel robot (Zhao et al. 2009). By modeling a 2DOF parallel robot in joint space, a computed torque based synchronized control algorithm is studied in (Shang et al. 2009). By defining a Jacobian matrix based synchronization error, an adaptive synchronized control algorithm is designed for a planar parallel manipulator (Ren et al. 2006). The parallel robot can be considered as a special case of MRMS. From the existing achievements in parallel robots, it shows that the synchronized control can improve the performance indeed. However, the existing approaches have not considered the kinematic uncertainties.

(4) Synchronized control for some other mechanical systems

There are synchronized control approaches for some other different mechanical systems, such as, experimental helicopters (Shan and Nowotny 2005), spacecraft formation flying (Shan 2008) and high order multi-agent system (Cui et al. 2008), multiple mobile robots (Zhao and Zou 2012) and so on. These literatures show that the synchronized control have many potential applications. To study the synchronized control in MRMS is developmental in the future practice. Another application is the counter control. The contour tracking is a task, and this control task is about position instead of time. Therefore, it requires all the motors to synchronize with respect to one position (Ouyang et al. 2012).

In brief summary of the existing synchronized control approaches, most of them are designed in joint space. Though some of them are designed in task space only a few literatures considering kinematic uncertainty. In industrial applications, such as transporting manipulators and assembly manipulators, require each manipulator to track different task space trajectories while maintain synchronization with each other (Sun 2010). Though (Wang 2013) presented a synchronized control approach for MRMS and can deal with both kinematic and dynamic

uncertainties it requires all of the manipulators to track a common desired trajectory. The main difference of the proposed approach from the one in (Wang, 2013) is that it can make the manipulators to track different desired trajectories and maintain synchronization among them. Although difficulties in design, considering synchronization, dynamic uncertainty and kinematic uncertainty together is more practical and will provide new insight in enhancing MRMS performance.

It also should be mentioned that there are some bilateral control for robotic manipulators (Chopra et al. 2009; Liu et al. 2010). Note that, teleoperation systems only have two robotic manipulators, in which the two robotic manipulators track each other. Subjected to the adaptive bilateral control, the two robot's end-effectors converge to the same configuration. The synchronized control for MRMS considers n ($n \geq 2$) robots, which can guarantee each robotic manipulator track its own reference while synchronizing motions between each other (Sun et al. 2007; Sun and Tong 2009). For example, robot i can track its own reference while maintains the same distance between robot $i-1$ and robot $i+1$. To achieve this purpose, cross-coupling technique is used in the proposed approach while the aforementioned two papers do not use this method. In summary, the two papers mainly focus on the consensus without taking the each robotic manipulator's reference tracking into consideration. Though they consider robot consensus control with kinematic uncertainty they cannot be used to achieve synchronized control of MRMS directly. In contrast, the proposed approach in this study is distinguished from them in synchronization error definition, control algorithm design and control objective. To the best of the author's knowledge, there has been no paper to study on the synchronized control of MRMS against both kinematic and dynamic uncertainties.

With above justification, this study proposes a new adaptive synchronized tracking control (ASTC) MRMS with capacity in dealing with both kinematic and dynamic uncertainties explicitly. It should be mentioned that the presented ASTC is different from existing adaptive Jacobian tracking control (AJTC) (Cheah et al 2006; Liu et al 2008) and position adaptive synchronized control (PASC) (Sun and Mills 2002; Sun 2003) in controller design and stability analysis. Compared with AJTC, ASTC considers and coordinates complex kinematic relationship of MRMS. However, AJTC is designed for individual robotic manipulator without considering synchronized control issues. Compared with PASC, ASTC considers kinematic uncertainties explicitly and can estimate kinematic parameters online. However, PASC assumes that kinematic parameters can be known exactly before the controller design. Hence, the proposed approach extends AJTC in the case of synchronized control and extends PASC in the case of kinematic uncertainties. It also should be mentioned there two cases to deal with uncertain parameters in the multiple robotic manipulators' control, that is, satisfying linearity-in-parameters (Cheng et al. 2008b; 2008c)

and un-satisfying linearity-in-parameters (Cheng et al. 2010; Hou et al. 2009; Hou, Cheng et al. 2009). The proposed approach will use linearity-in-parameters approach to design kinematic and dynamic adaptive law.

2 Kinematic and dynamic models of MRMS

Assume that a MRMS is composed of n robotic manipulators. Joint space dynamic equation of the i th-robotic manipulator can be described:

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

where $q_i(t) \in R^m$ denotes the joint angular position, $M_i(q_i) \in R^{m \times m}$ is inertia matrix, $G_i(q_i) \in R^m$ is the gravitational force vector, $\tau_i \in R^m$ denotes the input torque vector, $S_i(q_i, \dot{q}_i)$ is given as follows:

$$S_i(q_i, \dot{q}_i) = \frac{1}{2} \dot{M}_i(q_i, \dot{q}_i) - \frac{1}{2} [C_i(q_i, \dot{q}_i)]^T \quad (i=1, \dots, n)$$

Property 1: Inertia matrix $M_i(q_i)$ is symmetric and uniformly positive definite for all $q_i(t) \in R^m$.

Property 2: Matrix $S_i(q_i, \dot{q}_i)$ is skew-symmetric so that $y^T S_i(q_i, \dot{q}_i) = 0$ for all $y \in R^m$.

Property 3: Dynamic equation (1) is linear in a set of physical parameters $\theta_i^d \in R^l$:

$$M_i(q_i) \ddot{q}_i + Y_i^d(q_i, \dot{q}_i, \ddot{q}_i) \theta_i^d = \tau_i$$

where $Y_i^d(\cdot) \in R^{m \times l}$ is dynamic regressor matrix.

For the synchronized control of MRMS, desired trajectories are specified in task space, such as Cartesian space.

Let $x_i \in R^m$ is the real task space trajectory of i th manipulator, the following equation is satisfied:

$$x_i(t) = h_i(q_i) \quad (2)$$

where $h_i(\cdot) \in R^m \rightarrow R^m$ denotes transforming function from joint space to task space of i th manipulator. Task space velocity \dot{x}_i is related to joint space velocity \dot{q}_i as:

$$\dot{x}_i = J_i(q_i) \dot{q}_i \quad (3)$$

where $J_i(\cdot) \in R^{m \times m}$ is Jacobian matrix from joint space to task space.

Property 4: The right hand of equation (3) is linear in a set of constant kinematic parameters $\theta_i^k \in R^p$, such as link length and link twist angles. Equation (3) can be written as:

$$Y_i^k(\cdot) \theta_i^k = \ddot{x}_i \tag{4}$$

where $Y_i^k(\cdot) \in R^{n \times p}$ is the kinematic regressor matrix.

Dynamic uncertainty of a robotic manipulator denotes to the dynamic parameter uncertainty here. For example, θ_i^d relates to dynamic equation (1) directly. If it cannot be estimated exactly, that is, the estimated errors is called dynamic uncertainty. Kinematic uncertainty of a robotic manipulator means the kinematic parameter uncertainty. In general, Jacobian matrix can describe the kinematic issue of a manipulator. Hence, the parameters of Jacobian matrix can be considered as kinematic parameters. If the parameters of Jacobian matrix cannot be known exactly, then they have kinematic uncertainty. For more details about dynamic and kinematic uncertainty, the readers can refer to the literatures (Dixon 2007) and reference therein.

3 Synchronization of MRMS

Consider n robotic manipulators. Their end-effectors move in task space in a synchronous manner. Task space position error vector of i th manipulator is defined as:

$$\Delta x_i(t) = x_i(t) - x_i^d(t) \tag{5}$$

where $x_i^d \in R^m$ is desired trajectory of i th manipulator.

The synchronization of multiple manipulators means that the difference of task space coordinate among the robotic manipulators should be a constant during their movement. This can be described by a synchronization function:

$$f(x_1, x_2, \dots, x_n) = A \tag{6}$$

where A is a constant vector.

Equation (6) is valid for the desired trajectory, that is:

$$f(x_1^d, x_2^d, \dots, x_n^d) = A \tag{7}$$

Equation (6) minus (7), there will be:

$$\Delta x_1(t) = \Delta x_2(t) = \dots = \Delta x_n(t) \tag{8}$$

Equation (8) gives the synchronized objective of multiple robotic manipulator systems. For more details of the synchronization definition of multiple robotic manipulator, the readers can refer to the literatures (Sun 2010) and references therein.

In summary, the synchronized control is to design a control algorithm which can guarantee each manipulator to track its desired trajectory while keep the difference among their task space coordinates to be a constant.

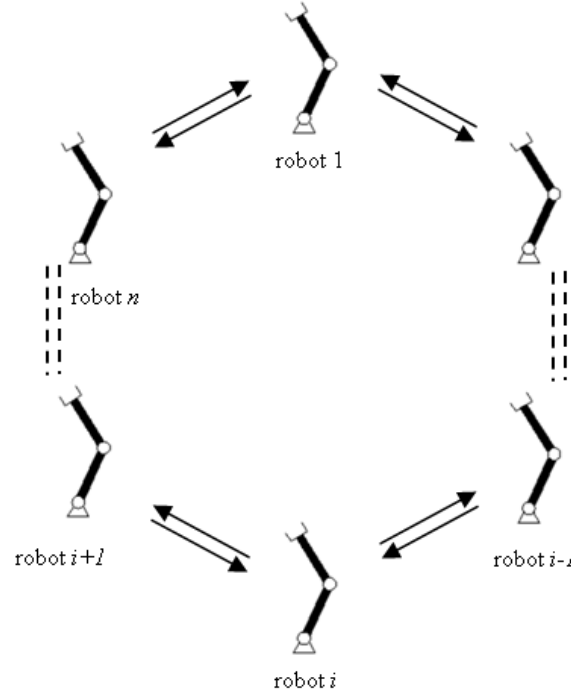


Figure 1 Information topology of multiple MRMS

Assume cyclic information topology is used in the MRMS. Figure 1 shows the topological graph. According to the synchronized objective and Figure 1, synchronization error and cross-coupling error vectors are defined as follows:

$$\begin{cases} \varepsilon_1(t) = \Delta x_1(t) - \Delta x_2(t) \\ \varepsilon_2(t) = \Delta x_2(t) - \Delta x_3(t) \\ \vdots \\ \varepsilon_{n-1}(t) = \Delta x_{n-1}(t) - \Delta x_n(t) \\ \varepsilon_n(t) = \Delta x_n(t) - \Delta x_1(t) \end{cases} \quad (9)$$

where $\varepsilon_i(t) \in R^m$ is synchronization error. If $\varepsilon_i(t) = 0$ for all $i = 1, \dots, n$, the synchronized objective can be achieved.

$$\begin{cases} e_1(t) = \Delta x_1(t) + \beta \int_0^t (\varepsilon_1(w) - \varepsilon_n(w)) dw \\ e_2(t) = \Delta x_2(t) + \beta \int_0^t (\varepsilon_2(w) - \varepsilon_1(w)) dw \\ \vdots \\ e_{n-1}(t) = \Delta x_{n-1}(t) + \beta \int_0^t (\varepsilon_{n-1}(w) - \varepsilon_{n-2}(w)) dw \\ e_n(t) = \Delta x_n(t) + \beta \int_0^t (\varepsilon_n(w) - \varepsilon_{n-1}(w)) dw \end{cases} \quad (10)$$

where $e_i(t) \in R^m$ is cross-coupling error, $\beta \in R^{m \times m}$ is diagonal positive definite matrix, which represents coupling parameters matrix.

Remark 1: Position error $\Delta x_i(t)$ and synchronization error $\varepsilon_i(t)$ are included in cross-coupling error $e_i(t)$.

If a synchronized controller can drive $e_i(t)$ and $\varepsilon_i(t)$ converged to zero asymptotically, it also can make $\Delta x_i(t)$ converged to zero asymptotically. Hence cross-coupling technique can simplify the synchronized controller design (Koren 1980). Note that $\varepsilon_i(t)$ appear in $e_i(t)$ and $e_{i+1}(t)$ with opposite sign. Then $e_i(t)$ and $e_{i+1}(t)$ are driven in opposite directions with respect to $\varepsilon_i(t)$. This is helpful to eliminate the synchronization error (Sun and Mills 2002; Sun 2003).

From cross-coupling error, the following task space command vectors can be defined as (Sun and Mills 2002; Sun 2003):

$$\begin{cases} u_1^x(t) = \begin{bmatrix} \vdots \\ \vdots \\ \varepsilon_1(t) - \varepsilon_n(t) - \Lambda e_1(t) \end{bmatrix} \\ u_2^x(t) = \begin{bmatrix} \vdots \\ \vdots \\ \varepsilon_2(t) - \varepsilon_1(t) - \Lambda e_2(t) \\ \vdots \end{bmatrix} \\ \vdots \\ u_{n-1}^x(t) = \begin{bmatrix} \vdots \\ \vdots \\ \varepsilon_{n-1}(t) - \varepsilon_{n-2}(t) - \Lambda e_{n-1}(t) \end{bmatrix} \\ u_n^x(t) = \begin{bmatrix} \vdots \\ \vdots \\ \varepsilon_n(t) - \varepsilon_{n-1}(t) - \Lambda e_n(t) \end{bmatrix} \end{cases} \quad (11)$$

where $u_i^x(t) \in R^m$ is the task space command vector, $\Lambda \in R^{m \times m}$ is diagonal positive definite matrix, which is a feedback gain matrix.

From equations (10) and (11), task space general error vectors can be defined as (Sun and Mills 2002; Sun 2003):

$$\begin{cases} r_1^x(t) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} \\ r_2^x(t) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} \\ \vdots \\ r_{n-1}^x(t) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} \\ r_n^x(t) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} \end{cases} \quad (12)$$

The control objective of this study can be summarized as: design an adaptive synchronized control law to be able to estimate kinematic and dynamic parameters to guarantee asymptotical stability of MRMS. This means that the proposed approach can make position error Δx_i and synchronization error ε_i to be converged to zero asymptotically and simultaneously.

4 Adaptive synchronized controller design and stability Analysis

Since kinematic parameters are not known exactly. Assume $\hat{\theta}_i^k$ is the estimated value vector of real θ_i^k . Then the estimated task space velocity vector can be defined as:

$$\hat{r}_i^x(t) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} \quad (13)$$

Estimated task space general error vector can be defined as:

$$\hat{r}_i^x(t) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} \quad (14)$$

where $\Delta\theta_i^k(t)$ is estimation error of kinematic parameters, which is defined as:

$$\Delta\theta_i^k(t) = \theta_i^k - \hat{\theta}_i^k(t) \quad (15)$$

Remark 2: In light of the definitions of task space command vectors and general error vectors developed in (Sun and Mills 2002; Sun 2003), the estimated task space general error vector is defined as (14) by using the estimated velocity which is computed from the Jacobian regressor matrix $Y_i^k(q_i, \dot{q}_i)$ and estimated kinematic parameters $\hat{\theta}_i^k(t)$. Note that estimation error $\Delta\theta_i^k(t)$ will be eliminated by a kinematic adaptive law in the following development.

Define joint space command vector as:

$$u_i(t) = \hat{J}_i^{-1}(q_i, \hat{\theta}_i^k) u_i^x(t) \quad (16)$$

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} u_i^x(t) \quad (17)$$

Define joint space general error vector as:

$$r(t) = \begin{bmatrix} \dot{q}_1 - \dot{q}_1^d \\ \vdots \\ \dot{q}_n - \dot{q}_n^d \end{bmatrix} = \hat{J}_i^{-1}(q_i, \hat{\theta}_i^k) \hat{r}_i^x(t) \quad (18)$$

$$\begin{bmatrix} \dot{q}_1 - \dot{q}_1^d \\ \vdots \\ \dot{q}_n - \dot{q}_n^d \end{bmatrix} \quad (19)$$

In light of (16)-(19), the i th robotic manipulator dynamic equation can be rewritten as:

$$M_i(q_i) \ddot{q}_i + \begin{bmatrix} 1 & \dots & 0 \\ 2 & \dots & \dots \end{bmatrix} \dot{q}_i + \begin{bmatrix} 1 & \dots & 0 \\ 2 & \dots & \dots \end{bmatrix} q_i = \tau_i \quad (20)$$

According to Property 3, equation (18) can be written as:

$$M_i(q_i) \ddot{q}_i + \begin{bmatrix} 1 & \dots & 0 \\ 2 & \dots & \dots \end{bmatrix} \dot{q}_i + \begin{bmatrix} 1 & \dots & 0 \\ 2 & \dots & \dots \end{bmatrix} q_i = \tau_i \quad (21)$$

where $Y_i^d(q_i, \dot{q}_i, \ddot{q}_i)$ is defined as:

$$Y_i^d(q_i, \dot{q}_i, \ddot{q}_i) = \begin{bmatrix} 1 & \dots & 0 \\ 2 & \dots & \dots \end{bmatrix} \ddot{q}_i + \begin{bmatrix} 1 & \dots & 0 \\ 2 & \dots & \dots \end{bmatrix} \dot{q}_i + \begin{bmatrix} 1 & \dots & 0 \\ 2 & \dots & \dots \end{bmatrix} q_i \quad (22)$$

Assumption 1: The desired joint position q_i^d and its derivatives $\dot{q}_i^d, \ddot{q}_i^d$ are all bounded and smooth.

Assumption 2: The task space position x_i and velocity \dot{x}_i , joint space position q_i and velocity \dot{q}_i are measurable.

Assumption 3: The robotic manipulators are working in a finite task space such that the Jacobian matrices are of full rank (Cheah et al 2006; Liu et al 2008).

Under Assumptions 1-3, the following distributed ASTC control law is designed to achieve synchronized control of MRMS in the presence of both kinematic and dynamic uncertainties:

$$\begin{cases} \tau_1 = Y_1^d \hat{\theta}_1^d - \hat{J}_1^T(q_1, \hat{\theta}_1^k) (K_v \dot{\varepsilon}_1 - \hat{J}_1^T(q_1, \hat{\theta}_1^k) K_\varepsilon (\varepsilon_1 - \varepsilon_n)) \\ \tau_2 = Y_2^d \hat{\theta}_2^d - \hat{J}_2^T(q_2, \hat{\theta}_2^k) (K_v \dot{\varepsilon}_2 - \hat{J}_2^T(q_2, \hat{\theta}_2^k) K_\varepsilon (\varepsilon_2 - \varepsilon_1)) \\ \vdots \\ \tau_{n-1} = Y_{n-1}^d \hat{\theta}_{n-1}^d - \hat{J}_{n-1}^T(q_{n-1}, \hat{\theta}_{n-1}^k) (K_v \dot{\varepsilon}_{n-1} - \hat{J}_{n-1}^T(q_{n-1}, \hat{\theta}_{n-1}^k) K_\varepsilon (\varepsilon_{n-1} - \varepsilon_{n-2})) \\ \tau_n = Y_n^d \hat{\theta}_n^d - \hat{J}_n^T(q_n, \hat{\theta}_n^k) (K_v \dot{\varepsilon}_n - \hat{J}_n^T(q_n, \hat{\theta}_n^k) K_\varepsilon (\varepsilon_n - \varepsilon_{n-1})) \end{cases} \quad (23)$$

where $K_v \in R^{m \times m}$, $K_p \in R^{m \times m}$, $K_\varepsilon \in R^{m \times m}$ are all diagonal positive definite matrices, $\hat{\theta}_i^d \in R^m$ is the

¹ For the simplicity of expression, (t) is omitted in the following context.

estimation of θ_i^d , the estimation error is defined as:

$$\Delta\theta_i^d = \theta_i^d - \hat{\theta}_i^d \quad (24)$$

Distributed kinematic adaptive law is designed as:

$$\begin{cases} \dot{\hat{c}}_1 & -K_{\varepsilon} \left(q_1, \dot{c}_1, \dots, \varepsilon_1 - \varepsilon_n \right) \\ \dot{\hat{c}}_2 & -K_{\varepsilon} \left(q_2, \dot{c}_2, \dots, \varepsilon_2 - \varepsilon_1 \right) \\ & \vdots \\ \dot{\hat{c}}_{n-1} & -K_{\varepsilon} \left(q_{n-1}, \dot{c}_{n-1}, \dots, \varepsilon_{n-1} - \varepsilon_{n-2} \right) \\ \dot{\hat{c}}_n & -K_{\varepsilon} \left(q_n, \dot{c}_n, \dots, \varepsilon_n - \varepsilon_{n-1} \right) \end{cases} \quad (25)$$

where $L_k \in R^{p \times p}$ is diagonal positive definite matrix.

Distributed dynamic adaptive law is designed as:

$$\dot{\hat{c}}_i = -L_d \left(Y_i^d \left(q_i, \dot{c}_i, \dots \right), i=1, \dots \right) \quad (26)$$

where $L_d \in R^{l \times l}$ is diagonal positive definite matrix.

Remark 3: In control law (23) $Y_i^d \hat{\theta}_i^d$ is feedforward compensation for robotic manipulator dynamics.

$\hat{J}_i^T \left(q_i, \hat{\theta}_i^k \right) \left(K_v \dot{c}_i \right)$ is feedback control to stabilize cross-coupling error to 0 asymptotically.

$\hat{J}_i^T \left(q_i, \hat{\theta}_i^k \right) K_{\varepsilon} \left(\varepsilon_i - \varepsilon_{i-1} \right)$ as $i=1, i-1=n$ is feedback control to stabilize synchronization error to 0 asymptotically. Note that estimated parameters of kinematics and dynamics are used in the control law, distributed kinematic (25) and dynamic adaptive law (26) are designed to update them online.

Theorem 1 Consider a multiple robotic manipulators system (1) with Assumption 1-3, if the proposed control law is designed with (23), (25) and (26), position tracking error Δx_i and synchronization error ε_i will be asymptotically stable, $i=1, \dots$.

Proof: Consider the following Lyapunov function candidate:

$$\begin{aligned}
V = & \sum_{i=1}^n \left[\frac{1}{2} r_i^T M_i(q_i) r_i + \frac{1}{2} (\Delta \theta_i^d)^T L_d^{-1} \Delta \theta_i^d + \frac{1}{2} (\Delta \theta_i^k)^T L_k^{-1} \Delta \theta_i^k \right. \\
& \left. + \frac{1}{2} e_i^T (K_p + \alpha K_v) e_i + \frac{1}{2} \varepsilon_i^T K_\varepsilon \varepsilon_i \right] \\
& + \frac{1}{2} \left(\int_0^t (\varepsilon_1(w) - \varepsilon_n(w))^T dw \right) K_\varepsilon \Lambda \beta \int_0^t (\varepsilon_1(w) - \varepsilon_n(w)) dw \\
& + \sum_{i=2}^n \frac{1}{2} \left(\int_0^t (\varepsilon_i(w) - \varepsilon_{i-1}(w))^T dw \right) K_\varepsilon \Lambda \beta \int_0^t (\varepsilon_i(w) - \varepsilon_{i-1}(w)) dw
\end{aligned} \tag{27}$$

Differentiating V with respect to time and considering equation (21) and Property 2, it yields:

$$\begin{aligned}
\dot{V} = & \sum_{i=1}^n \left[\tau_i - r_i^T Y_i^d(q_i, \dot{q}_i, \ddot{q}_i) - (\Delta \theta_i^d)^T L_d^{-1} \dot{\hat{\theta}}_i^d \right. \\
& \left. - (\Delta \theta_i^k)^T L_k^{-1} \dot{\hat{\theta}}_i^k - e_i^T (-\dot{K}_p + \alpha \dot{K}_v) e_i \right] \\
& + (\varepsilon_1 - \varepsilon_n)^T K_\varepsilon \Lambda \beta \int_0^t (\varepsilon_1(w) - \varepsilon_n(w)) dw \\
& + \sum_{i=2}^n (\varepsilon_i(w) - \varepsilon_{i-1}(w))^T K_\varepsilon \Lambda \beta \int_0^t (\varepsilon_i(w) - \varepsilon_{i-1}(w)) dw
\end{aligned} \tag{28}$$

Substituting control law (23) and adaptive laws (25)-(26) into (28), there must be:

$$\begin{aligned}
\dot{V} = & \sum_{i=1}^n \left[\dot{q}_i^T \left(-\tau_i + r_i^T Y_i^d(q_i, \dot{q}_i, \ddot{q}_i) \right) \right. \\
& \left. + e_i^T (K_p + \alpha K_v) \dot{e}_i \right] \\
& - \sum_{i=2}^n \left(\dot{q}_i^T \left(-\tau_i + r_i^T Y_i^d(q_i, \dot{q}_i, \ddot{q}_i) \right) \right) \\
& + (\varepsilon_1 - \varepsilon_n)^T k_\varepsilon \Lambda \beta \int_0^t (\varepsilon_1(w) - \varepsilon_n(w)) dw \\
& + \sum_{i=2}^n (\varepsilon_i(w) - \varepsilon_{i-1}(w))^T k_\varepsilon \Lambda \beta \int_0^t (\varepsilon_i(w) - \varepsilon_{i-1}(w)) dw
\end{aligned} \tag{29}$$

Consider the following two equations:

$$\begin{aligned}
& \sum_{i=1}^n \left[-(\dot{q}_i^T \left(-\tau_i + r_i^T Y_i^d(q_i, \dot{q}_i, \ddot{q}_i) \right)) \right] \\
= & \sum_{i=1}^n \left[-\dot{q}_i^T \left(-\tau_i + r_i^T Y_i^d(q_i, \dot{q}_i, \ddot{q}_i) \right) \right] \\
= & \sum_{i=1}^n \left[-\dot{q}_i^T \left(-\tau_i + r_i^T Y_i^d(q_i, \dot{q}_i, \ddot{q}_i) \right) \right]
\end{aligned} \tag{30}$$

$$\begin{aligned}
& \sum_{i=2}^n (r_i^x)^T K_\varepsilon (\varepsilon_i - \varepsilon_{i-1}) + (r_1^x)^T K_\varepsilon (\varepsilon_1 - \varepsilon_n) \\
&= (r_1^x)^T K_\varepsilon \varepsilon_1 - (r_1^x)^T K_\varepsilon \varepsilon_n \\
&+ (r_2^x)^T K_\varepsilon \varepsilon_2 - (r_2^x)^T K_\varepsilon \varepsilon_1 \\
&\quad \vdots \\
&+ (r_n^x)^T K_\varepsilon \varepsilon_n - (r_{n-1}^x)^T K_\varepsilon \varepsilon_{n-1} \\
&= \sum_{i=1}^{n-1} (r_i^x - r_{i+1}^x)^T K_\varepsilon \varepsilon_i + (r_n^x - r_1^x)^T K_\varepsilon \varepsilon_n
\end{aligned} \tag{31}$$

Substituting (30) and (31) into (29) and using (12), it yields:

$$\begin{aligned}
& \dot{V} - \sum_{i=1}^n \left[\varepsilon_i^T \Lambda K_\varepsilon \varepsilon_i \right] - \sum_{i=1}^{n-1} (\varepsilon_i - \varepsilon_{i+1})^T K_\varepsilon \beta (\varepsilon_i - \varepsilon_{i+1}) \\
& - (\varepsilon_n - \varepsilon_1)^T K_\varepsilon \beta (\varepsilon_n - \varepsilon_1) \leq 0
\end{aligned} \tag{32}$$

Since $M_i(q_i)$ is uniformly positive definite, V is positive definite in r_i , $\Delta\theta_i^d$, $\Delta\theta_i^k$, e_i , ε_i and $\int_0^t (\varepsilon_i(w) - \varepsilon_{i-1}(w)) dw$ (as $i=1$, $i-1=n$). Since \dot{V} is bounded, r_i , $\Delta\theta_i^d$, $\Delta\theta_i^k$, e_i , ε_i and $\int_0^t (\varepsilon_i(w) - \varepsilon_{i-1}(w)) dw$ (as $i=1$, $i-1=n$) are all bounded vectors. This also means, that $\hat{\theta}_i^k$ and $\hat{\theta}_i^d$ are bounded. Due to x_i^d is bounded, x_i is bounded. Because the terms of Jacobian matrix are trigonometric function of q_i and $\hat{\theta}_i^k$, $\hat{J}_i(q_i, \hat{\theta}_i^k)$ is bounded. Then $\hat{r}_i^x = \hat{J}_i(q_i, \hat{\theta}_i^k) r_i$ is bounded with (18). Using (12), it can be concluded that \dot{r}_i is bounded. Since e_i and \dot{r}_i are bounded, u_i^x is also bounded with (11). Hence, u_i is bounded if inverse approximate Jacobian matrix is bounded ($\hat{J}_i(q_i, \hat{\theta}_i^k)$ is bounded and full rank). From (18), \dot{q}_i is bounded, and it also means that \dot{r}_i is bounded because that Jacobian matrix is bounded. Then, Δ^* is bounded, it also means that \dot{r}_i , \dot{e}_i , $\dot{\varepsilon}_i$, $\dot{\varepsilon}_{i-1}$ (as $i=1$, $i-1=n$) are bounded. The terms of $Y_i^k(q_i, \dot{r}_i)$ are trigonometric functions of q_i and \dot{r}_i , hence $Y_i^k(q_i, \dot{r}_i)$ is bounded. From kinematic adaptive law (25), $\dot{\hat{r}}_i$ is bounded. It is obvious that \dot{r}_i is bounded if $\dot{\hat{r}}_i$ is bounded. From closed-loop equation (21) with control law (23), (25) and (26), it can be concluded that \dot{r}_i is bounded. Because $\hat{J}_i^{-1}(q_i, \hat{\theta}_i^k)$ and $\dot{\hat{r}}_i$ are bounded, \dot{r}_i is bounded. Due to \dot{r}_i is bounded, \dot{e}_i is bounded. Note

that the terms of $\dot{1}_{i \setminus \dots}$ are trigonometric functions of q_i multiplied with \dot{c}_i and/or \ddot{c}_i , $\dot{1}_{i \setminus \dots}$ is bounded. Because of $\Delta\theta_i^k$, \hat{t}_i , \dot{c}_i and \hat{c}_i are bounded, \ddot{c}_i is bounded.

Differentiating (32) with respect to time, it yields:

$$\begin{aligned} \ddot{J} &= \sum_{i=1}^n [-2\dot{c}_i \ddot{c}_i \dots] \\ &- \sum_{i=1}^n [2\varepsilon_i^T \Lambda K_\varepsilon \dot{c}_i] - \sum_{i=1}^{n-1} \dots \\ &- 2(\varepsilon_n - \varepsilon_1)^T K_\varepsilon \beta(\dot{c}_1, \dots) \end{aligned} \quad (33)$$

It is obvious that \ddot{J} is bounded since e_i , \dot{c}_i , \ddot{c}_i , ε_i , \dot{c}_i , $\varepsilon_i - \varepsilon_{i+1}$ and \dot{c}_i ($i=1, \dots, n$, as $i=n$, $i+1=1$) are all bounded. Hence, \dot{J} is uniformly continuous. Using Barbalat's Lemma (Khalil 2002), one can obtain that as $t \rightarrow \infty$, $e_i \rightarrow 0$, $\dot{c}_i \rightarrow 0$, $\varepsilon_i \rightarrow 0$ and $\varepsilon_i - \varepsilon_{i+1} \rightarrow 0$ ($i=1, \dots, n$, as $i=n$, $i+1=1$).

When $\varepsilon_i = 0$ for all $i=1, \dots, n$, the synchronized objective defined by (8) can be achieved. All equations in expression (10) are combined in the following form:

$$\Delta x_1 + \Delta x_2 + \dots + \Delta x_n = 0 \quad (34)$$

With (8), it yields:

$$\Delta x_1 = \Delta x_2 = \dots = \Delta x_n = 0 \quad (35)$$

It also means that the invariant set of closed-loop equation (21) subjected to control law (23), (25), (26) includes zero position errors, i.e., $\Delta x_i = 0$, $i=1, \dots, n$. By using Barbalat's Lemma (Khalil 2002), $\Delta x_i \rightarrow 0$, as $t \rightarrow \infty$.

The motivation of proposing the ASTC has been justified as below:

- (1) By using the synchronization error and cross-coupling error (Sun and Mills 2002; Sun 2003), design synchronized controller to make each robotic manipulator track its own desired trajectory while synchronize its motion with the other manipulators according to the synchronized objective.
- (2) By using robot adaptive Jacobian tracking control theory (Cheah et al 2006; Liu et al 2008), design a kinematic adaptive law to estimate uncertain kinematic parameters online.
- (3) By using robot adaptive control theory (Slotine and Li 1991), design a dynamic adaptive law to estimate uncertain dynamic parameters online.

Remark 4: If let $\beta = 0$ and $K_\varepsilon = 0$, control law (23), (25) and (26) will be an independent AJTC without accommodating synchronization (Cheah 2006):

$$\tau_i = Y_i^d \hat{\theta}_i^d - \hat{J}_i^T(q_i, \hat{\theta}_i^k) (K_v \dot{e}_i + \dot{q}_i^d) \quad (36)$$

$$\dot{\hat{e}}_i = -K_p e_i - K_v \dot{e}_i \quad (37)$$

$$\dot{\hat{e}}_i = -L_d Y_i^d (q_i, \dot{q}_i, \ddot{q}_i^d) \quad (38)$$

where K_p , K_v , L_k and L_d are the same as the ones of ASTC, $i = 1, \dots, n$. Note that independent AJTC did not consider synchronization among the multiple robotic manipulators. Hence, it cannot be used directly for synchronized control of MRMS.

Remark 5: Compared with PASC (Sun 2003), the proposed approach can cope with kinematic uncertainties explicitly by using adaptive law (25). However, PASC assumed that kinematic parameters and Jacobian matrices of MRMS can be obtained exactly. Kinematic uncertainty is a very practical problem that should be dealt with in the controller design. Neglect of uncertain kinematics in controller design will decrease the performance of close-loop system or even affect system stability.

Remark 6: If an approximate Jacobian matrix replaces adaptive law (25), control law (23) can be redesign as the following approximate Jacobian synchronized control (AJSC):

$$\begin{cases} \tau_1 = Y_1^d \hat{\theta}_1^d - \tilde{c}_1 \left(\frac{\partial Y_1^d}{\partial q_1} \dot{q}_1 + \frac{\partial Y_1^d}{\partial \dot{q}_1} \ddot{q}_1 \right) - \tilde{c}_1 \left(\frac{\partial Y_1^d}{\partial q_1} \dot{q}_1 + \frac{\partial Y_1^d}{\partial \dot{q}_1} \ddot{q}_1 \right) - \varepsilon \left(\frac{\partial Y_1^d}{\partial q_1} \dot{q}_1 + \frac{\partial Y_1^d}{\partial \dot{q}_1} \ddot{q}_1 \right) - \varepsilon \left(\frac{\partial Y_1^d}{\partial q_1} \dot{q}_1 + \frac{\partial Y_1^d}{\partial \dot{q}_1} \ddot{q}_1 \right) \\ \tau_2 = Y_2^d \hat{\theta}_2^d - \tilde{c}_2 \left(\frac{\partial Y_2^d}{\partial q_2} \dot{q}_2 + \frac{\partial Y_2^d}{\partial \dot{q}_2} \ddot{q}_2 \right) - \tilde{c}_2 \left(\frac{\partial Y_2^d}{\partial q_2} \dot{q}_2 + \frac{\partial Y_2^d}{\partial \dot{q}_2} \ddot{q}_2 \right) - \varepsilon \left(\frac{\partial Y_2^d}{\partial q_2} \dot{q}_2 + \frac{\partial Y_2^d}{\partial \dot{q}_2} \ddot{q}_2 \right) - \varepsilon \left(\frac{\partial Y_2^d}{\partial q_2} \dot{q}_2 + \frac{\partial Y_2^d}{\partial \dot{q}_2} \ddot{q}_2 \right) \\ \vdots \\ \tau_{n-1} = Y_{n-1}^d \hat{\theta}_{n-1}^d - \tilde{c}_{n-1} \left(\frac{\partial Y_{n-1}^d}{\partial q_{n-1}} \dot{q}_{n-1} + \frac{\partial Y_{n-1}^d}{\partial \dot{q}_{n-1}} \ddot{q}_{n-1} \right) - \tilde{c}_{n-1} \left(\frac{\partial Y_{n-1}^d}{\partial q_{n-1}} \dot{q}_{n-1} + \frac{\partial Y_{n-1}^d}{\partial \dot{q}_{n-1}} \ddot{q}_{n-1} \right) - \varepsilon \left(\frac{\partial Y_{n-1}^d}{\partial q_{n-1}} \dot{q}_{n-1} + \frac{\partial Y_{n-1}^d}{\partial \dot{q}_{n-1}} \ddot{q}_{n-1} \right) - \varepsilon \left(\frac{\partial Y_{n-1}^d}{\partial q_{n-1}} \dot{q}_{n-1} + \frac{\partial Y_{n-1}^d}{\partial \dot{q}_{n-1}} \ddot{q}_{n-1} \right) \\ \tau_n = Y_n^d \hat{\theta}_n^d - \tilde{c}_n \left(\frac{\partial Y_n^d}{\partial q_n} \dot{q}_n + \frac{\partial Y_n^d}{\partial \dot{q}_n} \ddot{q}_n \right) - \tilde{c}_n \left(\frac{\partial Y_n^d}{\partial q_n} \dot{q}_n + \frac{\partial Y_n^d}{\partial \dot{q}_n} \ddot{q}_n \right) - \varepsilon \left(\frac{\partial Y_n^d}{\partial q_n} \dot{q}_n + \frac{\partial Y_n^d}{\partial \dot{q}_n} \ddot{q}_n \right) - \varepsilon \left(\frac{\partial Y_n^d}{\partial q_n} \dot{q}_n + \frac{\partial Y_n^d}{\partial \dot{q}_n} \ddot{q}_n \right) \end{cases} \quad (39)$$

$$\dot{\hat{e}}_i = -L_d Y_i^d (q_i, \dot{q}_i, \ddot{q}_i^d) \quad (40)$$

where K_p , K_v and L_d are same to the ones of ASTC, $i = 1, \dots, n$. Note that approximate Jacobian matrices $\tilde{c}_i \left(\frac{\partial Y_i^d}{\partial q_i} \dot{q}_i + \frac{\partial Y_i^d}{\partial \dot{q}_i} \ddot{q}_i \right)$ and kinematic parameters \tilde{e}_i are estimated before controller design. If Jacobian matrices can be estimated exactly control law (39)-(40) will be PASC developed by (Sun and Mills; Sun 2003). Due to kinematic uncertainties, the estimation errors cannot be avoided and cannot be corrected online without adaptive law (23).

The following procedure are summarized to choose control gains and adaptive gains:

Step 1: Estimate the kinematic and dynamic parameters according to the designer's experience. Then substitute

them into control law (23).

Step 2: Use trial and error method to design feedback control gains: K_v , K_p and K_e until the closed-loop system is stable.

Step 3: Replace the estimated dynamic parameters by the ones updated by (25). Design dynamic adaptive gain L_d by using trial and error method. In this step, the control performance will be improved.

Step 4: Use the updated kinematic parameters according to (26) to replace the estimated ones. Then design kinematic adaptive gain L_k by using trial and error method. The control performance will be further improved.

Step 5: Tuning the control and adaptive gains slightly until the control performance is satisfied.

Remark 7: Lyapunov method is used to guarantee the system stability, which requires the control gains and adaptive gains are both positive definite. However, these parameters affect the system performance. Steps 1-5 are developed by using trial and error method to find the appropriate gains. Though the trial and error method depends on the designer's experience it is an effective solution with a few of trials in practice.
~~Remark 7: Lyapunov method is used to guarantee the system stability, which requires the control gains and adaptive gains are positive definite. However, these parameters affect the system performance. Steps 1-5 are developed by using trial and error method to find the appropriate gains. Though the trial and error method depends the designer's experience it is an effective solution in practice.~~

5 Illustrative Examples

To validate the performance of the proposed approach, numerical studies were presented in this section. The planar two-link robotic manipulator dynamic equation was described with (Zhao et al, 2009):

$$\begin{bmatrix} \alpha_{11}(q_2) & \alpha_{12}(q_2) \\ \alpha_{21}(q_2) & \alpha_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = \tau$$

$$\alpha_{11}(q_2) = (m_1 + m_2)r_1^2 + m_2r_2^2 + 2m_2r_1r_2 \cos(q_2) + j_1$$

$$\alpha_{12}(q_2) = m_2r_2^2 + m_2r_1r_2 \cos(q_2)$$

$$\alpha_{21} = \alpha_{12}$$

$$\alpha_{22} = m_2r_2^2 + j_2$$

$$\beta(q_2) = m_2 r_1 r_2 \sin(q_2)$$

$$\gamma_1(q_1, q_2) = (m_1 + m_2) r_1 \cos(q_2) + m_2 r_2 \cos(q_1 + q_2)$$

$$\gamma_2(q_1, q_2) = m_2 r_2 \cos(q_1 + q_2)$$

where q_1 , q_2 are angular position of each joint of robot. The parameters were assigned as: $r_1 = 0.3m$, $r_2 = 0.3m$, $m_1 = 1kg$, $m_2 = 1kg$, $g = 9.81m/s^2$, $j_1 = 5kg \cdot m$, $j_2 = 5kg \cdot m$. Assume these parameters unknown in controller design. Dynamic and kinematic adaptive laws were used to estimate them online.

Assumed that the base coordinates of the four robotic manipulators' were: $(-3.41, -2.41)m$, $(-3.15, 1.44)m$, $(3, 1.48)m$, $(3.26, -2.45)m$. The initial positions of robot end-effectors $x_i(0)$ were: $(-3, -2)m$, $(-3, 2)m$, $(3, 2)m$, $(3, -2)m$. The desired final positions $x_i(f)$ were: $(-2.8, -2)m$, $(-2.8, 2)m$, $(3.2, 2)m$, $(3.2, -2)m$. The desired trajectory of each robot end-effector was assigned as:

$$x_i^d = x_i(0) + (x_i(f) - x_i(0))(1 - \exp(-t)), \quad (i = 1, 2)$$

Dynamic uncertain parameters were given as:

$$\theta_i^d = [(m_1 + m_2)r_1^2 + m_2r_2^2 + j_1, m_2r_1r_2, m_2r_2^2, (m_1 + m_2)r_1, m_2r_2, m_2r_2^2 + j_2]$$

Kinematic uncertain parameters were given as:

$$\theta_i^k = [r_1, r_2]$$

The initial values of $\hat{\theta}_i^d$ and $\hat{\theta}_i^k$ were chosen as: $\hat{\theta}_1^d(0) = [1, 1, 1, 1, 1, 1]$, $\hat{\theta}_1^k(0) = [1, 1]$, $\hat{\theta}_2^d(0) = [0.5, 0.5, 0.5, 0.5, 0.5, 0.5]$, and $\hat{\theta}_2^k(0) = [0.5, 0.5]$ for ASTC and AJTC, respectively.

The initial value of $\hat{\theta}_i^d$ of AJSC were chosen as $\hat{\theta}_1^d(0) = [1, 1, 1, 1, 1, 1]$ and $\hat{\theta}_2^d(0) = [0.5, 0.5, 0.5, 0.5, 0.5, 0.5]$. The estimated kinematic parameter of AJSC were chosen as $r_{10} = 0.5$ and $r_{20} = 0.5$.

Figures 2-4 are the performances of the proposed ASTC. Figure 2 illustrates the synchronization errors, where solid lines and dashed lines denote the X and Y directions, respectively. Inspection of Figure 2, it shows that synchronization errors converge to 0 asymptotically. Figure 3 illustrates the task space position errors of the

robotic manipulators. Figure 3 shows that the configurations of position errors of robot 1 and robot 2 are similar to each other to avoid large synchronization error. Figure 4 illustrates the control input of ASTC which shows that the control inputs are bounded. Figures 5-7 are the performances of independent AJTC. This control algorithm does not include synchronized objective in the controller design. Figure 5 shows synchronization errors. Figure 6 shows task space position errors. Figure 7 shows the control input variations against time. Compare Figure 2 with Figure 5, it is clear that synchronization errors of independent AJTC are larger than those of the proposed ASTC especially in transient process. Though synchronization errors of independent AJTC converge to 0 eventually as position errors converge to 0, large synchronization errors in transient process are undesired in practice of MRMS. Figures 8-10 are the performances of AJSC. This control algorithm does not include kinematic adaptive law in the controller design. Figure 8 shows synchronization errors. Figure 9 shows task space position errors. Figure 10 shows the control inputs. Due to kinematic uncertainties are not included in controller design, control performances are not satisfactory. By comparing the control inputs shown in Figures 4, 7 and 10, it can be seen that all of them are bounded. The control inputs of ASTC and AJTC is similar in the amplitude. However the control inputs of AJSC are much larger than those of ASTC and AJTC. This is because approximate Jacobian parameters are used in the controller design, which cannot eliminate the kinematic modeling errors online during the system operation. From these comparisons, it is shown that the proposed ASTC is more effective to cope with system synchronization, kinematic uncertainty and dynamic uncertainty.

Table 1 Controller parameters

Control approach	Controller parameters
ASTC	$\beta = \text{diag}([50 \ 50]), \lambda = \text{diag}([2 \ 2]), K_p = \text{diag}([100 \ 100]),$ $K_v = \text{diag}([40 \ 40]), K_e = \text{diag}([100 \ 100]), L_k = \text{diag}([0.2 \ 0.2]),$ $L_d = \text{diag}([0.2 \ 0.2])$
AJTC	$\beta = \text{diag}([0 \ 0]), \lambda = \text{diag}([2 \ 2]), K_p = \text{diag}([100 \ 100]),$ $K_v = \text{diag}([40 \ 40]), K_e = \text{diag}([0 \ 0]), L_k = \text{diag}([0.2 \ 0.2]),$ $L_d = \text{diag}([0.2 \ 0.2])$
AJSC	$\beta = \text{diag}([60 \ 60]), \lambda = \text{diag}([2 \ 2]), K_p = \text{diag}([120 \ 120]),$

$$K_v = \text{diag}([50 \ 50]), \quad K_\varepsilon = \text{diag}([110 \ 110]), \quad L_d = \text{diag}([0.5 \ 0.5])$$

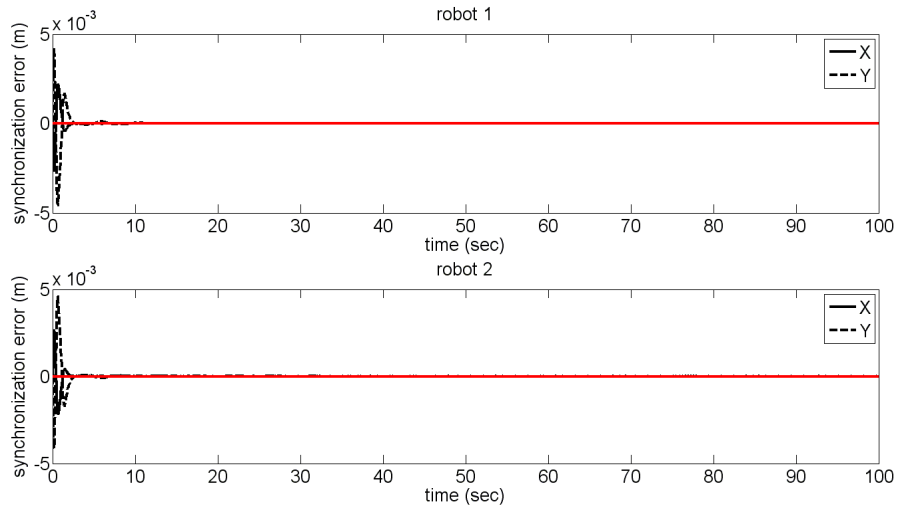


Figure 2 Synchronization error of ASTC

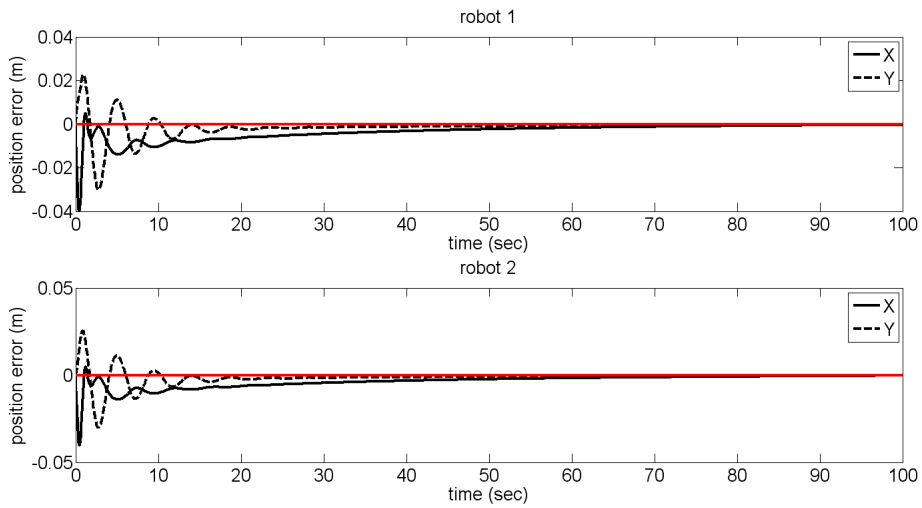


Figure 3 Position tracking error of ASTC

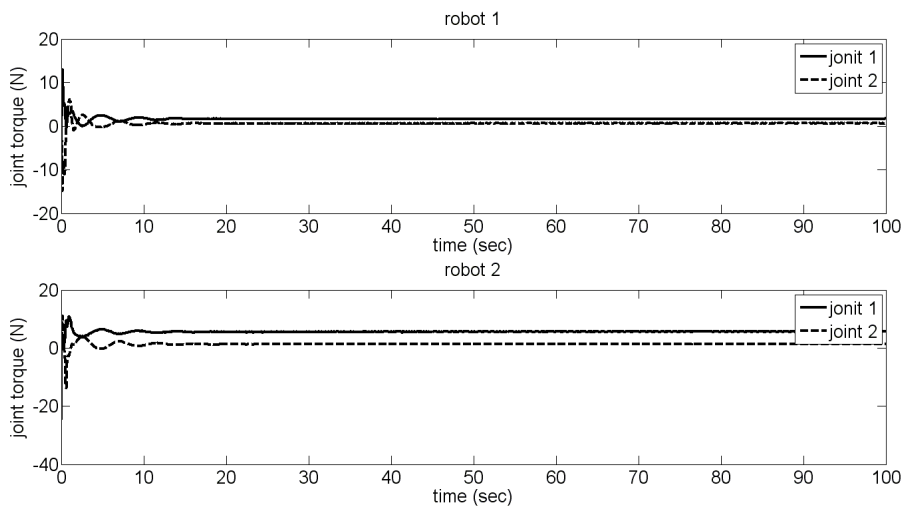


Figure 4 Control input of ASTC

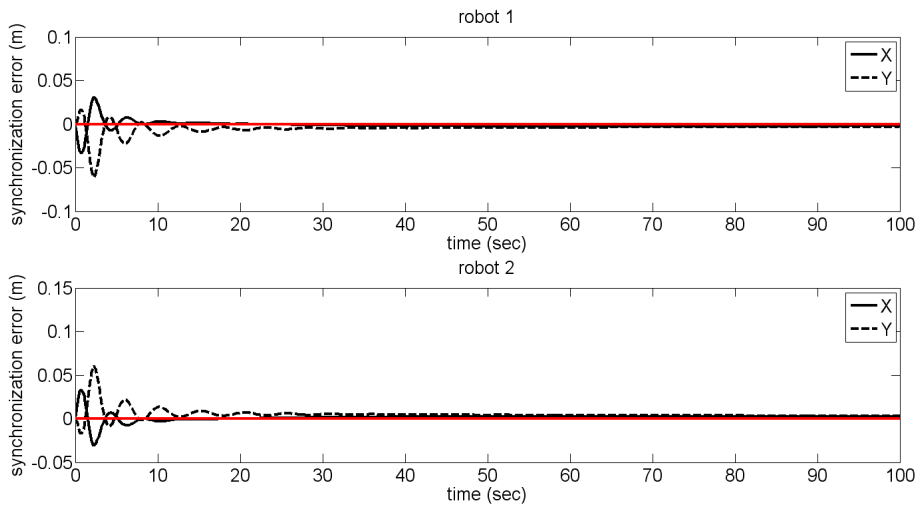


Figure 5 Synchronization error of AJTC

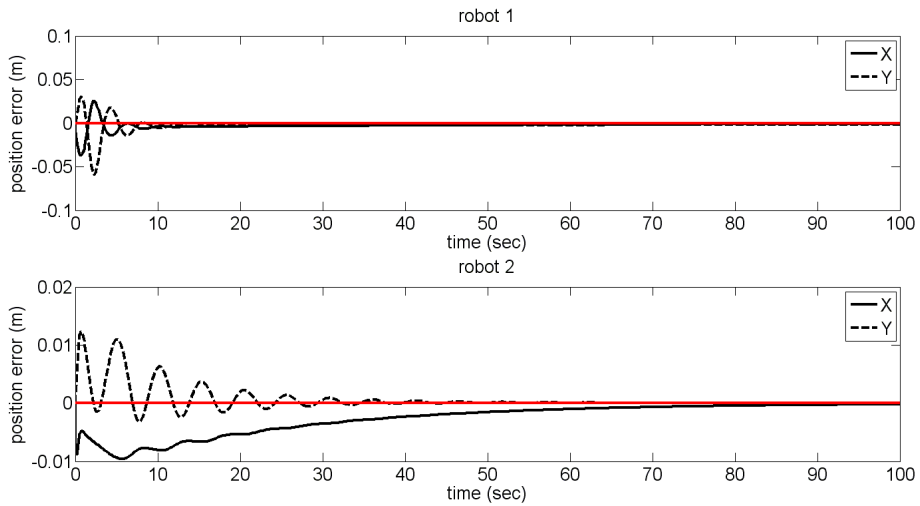


Figure 6 Position tracking error of AJTC

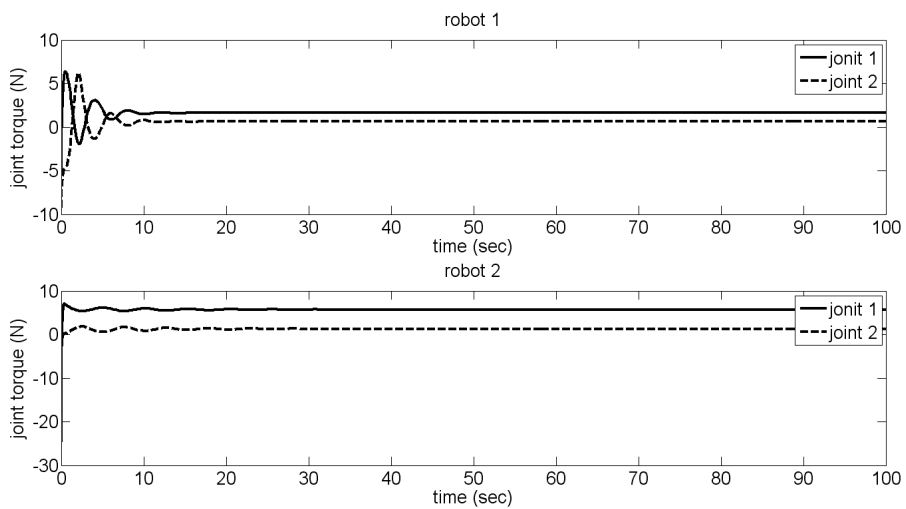


Figure 7 Control input of AJTC

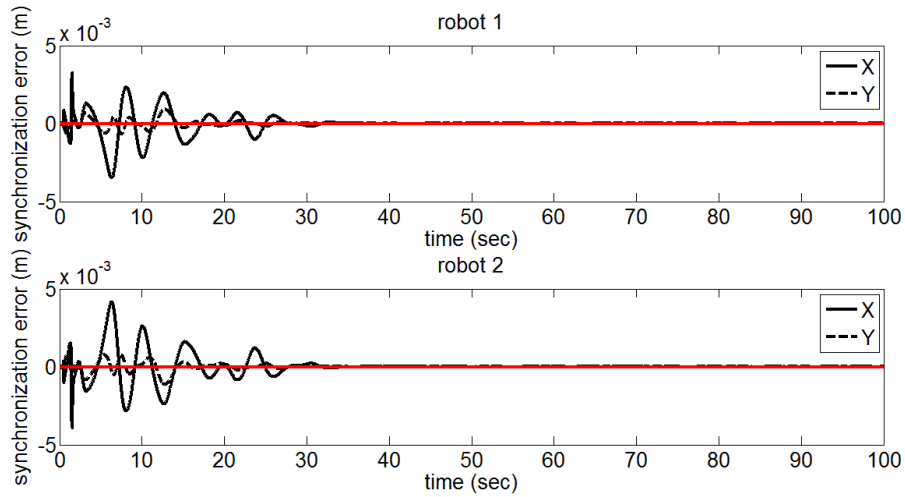


Figure 8 Synchronization error of AJSC

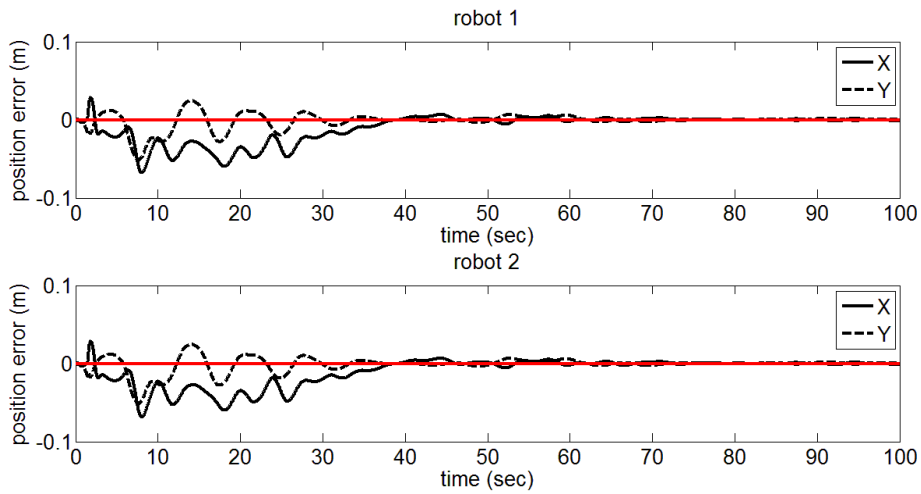


Figure 9 Position tracking error of AJSC

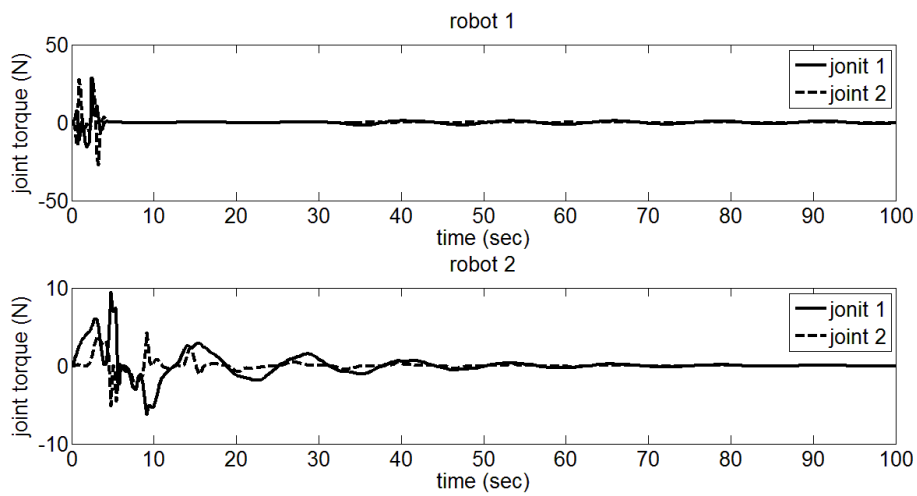


Figure 10 Control input of AJSC

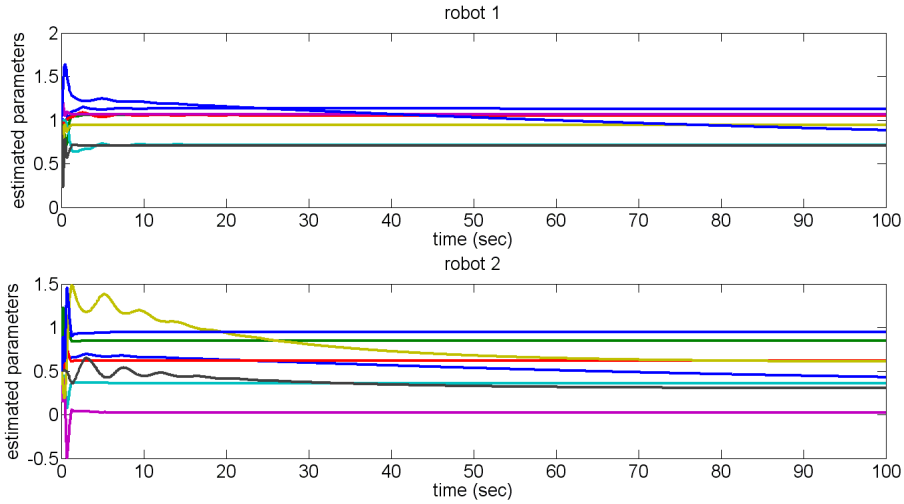


Figure 11 Adaptive law of ASTC

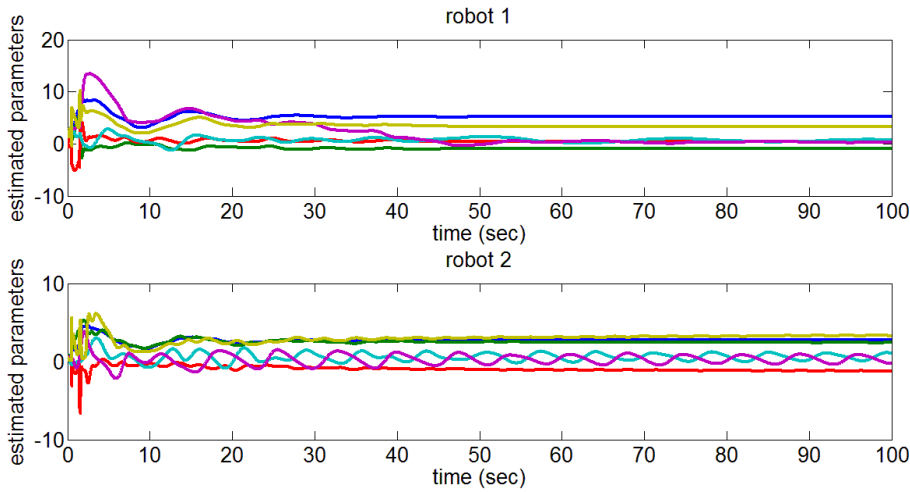


Figure 12 Adaptive law of AJSC

Figures 11 and 12 are adaptive laws of ASTC and AJSC, respectively. From these two figures, one can see that the proposed approach can estimate both kinematic and dynamic parameters. However conventional robot adaptive law (Slotine and Li 1991) only can estimate dynamic parameters. Note that the regressor matrix based adaptive law is very complex especially as the number of the estimated parameters is larger. In spite of the limitation, this adaptive law is still a good alternative to deal with parameters uncertainty.

Remark 8: [Three control approaches are compared in the Figures 2-10. ASTC has better performances than those of AJTC and AJSC. In comparison with AJTC, ASTC has better synchronization performance especially in the transient process. This is because ASTC doesn't consider the synchronized objective the controller design. The results illustrate that the proposed approach not only guarantees the position error convergence but also makes it converging in a synchronous manner. In comparison with AJSC, ASTC has better convergence performance with the position error and synchronization error. This is due to that ASTC uses the approximate Jacobian matrices in the controller design, which cannot eliminate kinematic uncertainty effect. In a word, it should consider the kinematic and dynamic uncertainty sensibly in the design of the synchronized controller of MRMS in task space.](#)

Remark 8: Three control approaches are compared in the Figures 2-10. ASTC has better performances than those of AJTC and AJSC. In comparison with AJTC, ASTC has better synchronization performance especially in the transient process. This is because ASTC doesn't consider the synchronized objective the controller design. The results illustrate that the proposed approach not only guarantees the position errors' convergence but also makes them converging in a synchronous manner. In comparison with AJSC, ASTC has better performances in the position errors and synchronization errors' convergence. This is due to that ASTC uses the approximate Jacobian matrices in the controller design, which cannot eliminate kinematic uncertainty effect. In a word, it should consider the kinematic and dynamic uncertainty in the design of the synchronized controller of MRMS in task space.

To further test the proposed approach, more complex desired trajectories, that is, two circles was tracked by two different manipulators, respectively. The manipulators' parameters were given in Table 2. This example can illustrate a more general situation. Figure 13 shows that the manipulators track the different circles in their task space. Figure 14 and Figure 15 are the synchronization errors and position tracking errors. From these figures, one can see that the proposed approach can make different robotic manipulators to track complex trajectories while keep synchronization among them. This example effectively demonstrates the proposed approach again.

Table 2 The robotic manipulators' parameters

Robot 1	Robot 2
$r_1 = 0.3m, r_2 = 0.3m, m_1 = 1kg, m_2 = 1kg$	$r_1 = 0.2m, r_2 = 0.2m, m_1 = 0.8kg, m_2 = 0.8kg$

The desired trajectory were given as:

$$\begin{cases} x_1^d = 2.3 + 0.1 * \cos(t) \\ y_1^d = 0.4 + 0.1 * \sin(t) \end{cases}$$

$$\begin{cases} x_2^d = 4.25 + 0.1 * \cos(t) \\ y_2^d = 0.4 + 0.1 * \sin(t) \end{cases}$$

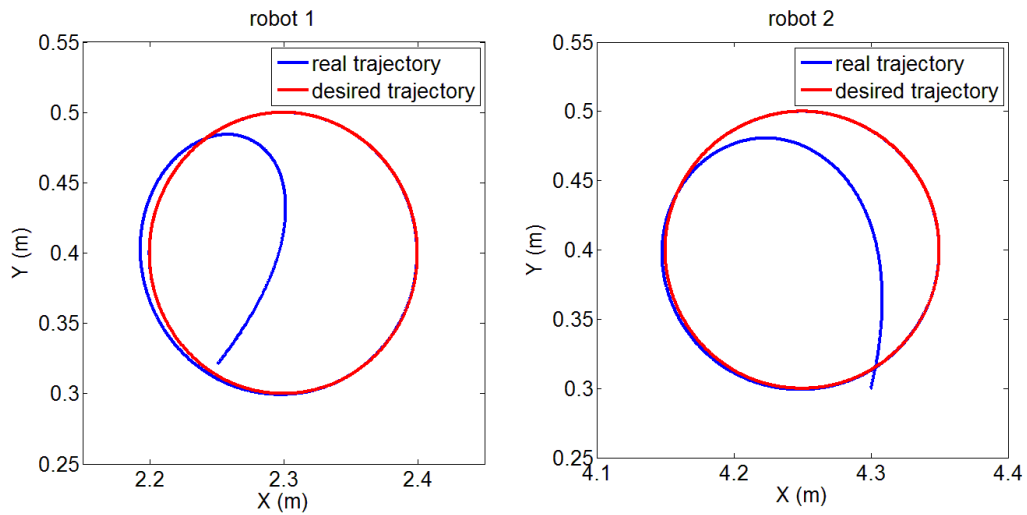


Figure 13 Two robotic manipulators track different circles in their task space

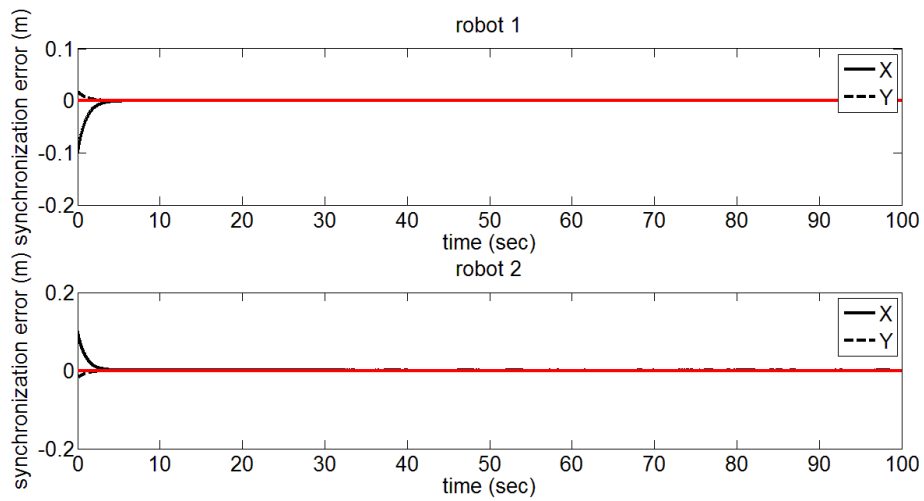


Figure 14 Synchronization error of ASTC (Circle tracking)

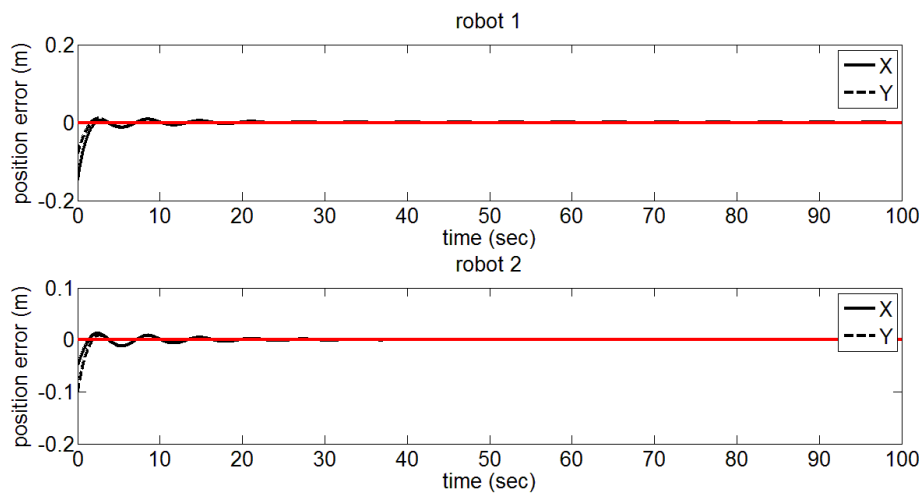


Figure 15 Position tracking error ASTC (Circle tracking)

Remark 9: The proposed approach has well satisfied performance in tracking of complex desired trajectories. It

illustrates that ASTC can track the complex desired trajectories while maintaining the synchronized objective. This is an expected capability in many industrial applications such as assembly, painting and machining by using multiple robotic manipulators.

~~**Remark 9:** The proposed approach has satisfied performances in the tracking of complex desired trajectories. It illustrates that ASTC can track the complex desired trajectories while maintains the synchronized objective. This is an expected capability in many industrial applications such as assembly, painting and machining by using multiple robotic manipulators.~~

The original purpose of this study is to present a MRMS synchronized control algorithm in task space in the presence of kinematic uncertainty and dynamic uncertainty. With reference to the numerical simulation results that are consistent with the analytical formulations, the proposed approach is effective to achieve the synchronized objective especially during the transient process. The proposed adaptive law can estimate kinematic and dynamic parameters online, which has strong robustness to these estimated parameters. The advantages of the proposed approach over the general robot adaptive control lies in the synchronized control and kinematic parameter estimation. Though synchronized control is proposed for multiple axes system in which dynamic adaptive law (Sun and Mills 2002; Sun 2003) is used, the kinematic uncertainty has not been considered by these literatures. It should be mentioned that (Cheng et al 2008a; Liu et al 2010) consider kinematic uncertainty in some aspect but not including robot dynamics in controller design. (Liu et al 2010) only considers two robot bilateral control teleoperation systems. These two methods cannot be used for MRMS directly.

6 Conclusions

By theoretical analysis and simulation demonstrations, a novel ASTC has been initially constructed to cope with kinematic uncertainties and synchronized control together in MRMS. In light of accommodating the cross-coupling errors the proposed approach can stabilize both position errors and synchronization errors that converge to zero asymptotically and simultaneously, which may achieve higher precision and more flexibility in manufacturing processes with multiple robotic manipulators. Note that the proposed approach expands the existing independent adaptive Jacobian tracking control algorithms (Cheah et al 2006; Liu et al 2008) to achieve synchronized objective of MRMS as well as expands existing adaptive synchronized control algorithms (Sun and Mills 2002; Sun 2003) to accommodate kinematic uncertainties in the controller design. 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 multiple robotic manipulators systems. In addition, this study provides an exemplary

showcase with effectively to integrate several cross boundary theoretical results in the fields of control and parameter estimation, which reflects the philosophy of interdisciplinary study having been the tendency in emerging research. The future work will be conducted in applying this new scheme to resolve some ad hoc problems (such as time delay and time varying information topology) encountered in MRMS.

Acknowledgement

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.

References

- Braganza, D., Dixon, W.E., Dawson, D.M., and Xian, B. (2008), 'Tracking control for robot manipulators with kinematic and dynamic uncertainty', *International Journal of Robotics and Automation*, 23, 117-126.
- Cheah, C.C., Liu, C., and Slotine, J.J.E. (2006), 'Adaptive Jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models', *IEEE Transactions on Automatic Control*, 51, 1024-1029.
- Cheah, C.C. (2008), 'Task-space PD control of robot manipulators: unified analysis and duality property', *International Journal of Robotics Research*, 27, 1152-1170.
- Chen, Z.H., Wang, Y., Ouyang, P.R. Huang, J., and Zhang, W. J. (2011), 'A novel iteration-based controller for hybrid machine systems for trajectory tracking at the end-effector level', *Robotica*, 29, 317-342.
- Cheng, L., Hou, Z.-G., Tan, M., and Zou An-Min. (2008a), 'Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties', *IEEE International Symposium on Intelligent Control*, 189-194.
- Cheng, L., Hou, Z.-G., and Tan, M. (2008b), 'Decentralized adaptive consensus control for multi-manipulator system with uncertain dynamics', *Proceedings of 2008 IEEE International Conference on Systems, Man, and Cybernetics*, Singapore, 2712-2717.
- Cheng, L., Hou, Z.-G., and Tan, M. (2008c), 'Decentralized adaptive leader-follower control of multi-manipulator

system with uncertain dynamics', *Proceedings of The 34th Annual Conference of the IEEE Industrial Electronics Society, Orlando, FL, USA*, 1608–1613.

Cheng, L., Hou, Z.-G., and Tan, M. (2009), 'Adaptive neural network tracking control for manipulators with uncertain kinematics, dynamics and actuator model', *Automatica*, 45, 2312-2318.

Cheng, L., Hou, Z.-G., Tan, M., Lin Y., and Zhang, W. (2010), 'Neural-network-based adaptive leader-following control for multi-agent systems with uncertainties', *IEEE Transactions on Neural Networks*, 21(8): 1351–1358.

Cheong, J., Niculescu, S.-I., and Kim, C. (2009), 'Motion synchronization control of distributed multisubsystems with invariant local natural dynamics', *IEEE Transactions on Robotics*, 25, 382-389.

Chopra, N., Spong, M. W., and Lozano, R. (2009), 'Synchronization of bilateral teleoperators with time delay', *Automatica*, 44, 2142-2148.

Chung, S.-J., and Slotine, J.-J.E. (2009), 'Cooperative robot control and concurrent synchronization of Lagrangian systems', *IEEE Transactions on Robotics*, 25, 686-700.

Cui, R., Ren, B., and Ge, S.S., (2008), 'Synchronised tracking control of multi-agent system with high-order dynamics', *IET Control Theory and Applications*, 6, 603-614.

Dixon, W.E. (2007), 'Adaptive regulation of amplitude limited robot manipulators with uncertainty kinematics and dynamics', *IEEE Transactions Automatic Control*, 52, 488-493.

Gudino-Lau, J., Artagag, M.A. (2005), 'Dynamic model and simulation of cooperative robots: a case study', *Robotica*, 23, 615-624.

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

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

Hou, Z.-G., Cheng, L., and Tan, M. (2009), 'Decentralized robust adaptive control for the multiagent system consensus problem using neural networks', *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(3), 636–647.

Hou, Z.-G., Cheng, L., Tan, M., and Wang, X. (2009), 'Distributed adaptive coordinated control of multi-manipulator systems using neural networks', in *Robot Intelligence: An Advanced Knowledge Processing Approach*, Springer-Verlag, London, pp. 49–69.

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

- Khalil, H.K. (2002), *Nonlinear systems (3rd)*, New Jersey: Prentice-Hall.
- Koren, Y. (1980), 'Cross-coupled biaxial computer controls for manufacturing systems', *ASME Journal of Dynamic Systems, Measurement, and Control*, 102, 265-272.
- Lee, H.-K., and Chung, M.J. (1998), Adaptive controller of a master-slave system for transparent teleoperation. *Journal of Robotic Systems*, 15, 465-475.
- Liang, X., Huang, X., Wang, M., and Zeng, X. (2010), 'Adaptive task-space tracking control of robots without task-space-and joint-space-velocity measurements', *IEEE Transactions on Robotics*, 26, 733-742.
- Liu, C., Cheah, C.C., and Slotine, J.J.E. (2008), 'Adaptive task-space regulation of rigid-link flexible-joint robots with uncertain kinematics', *Automatica*, 44, 1806-1814.
- Liu, Y.-C., and Chopra, N. (2012), 'Controlled synchronization of heterogeneous robotic manipulators in the task space', *IEEE Transactions on Robotics*, 28, 268-275.
- Liu, X., Tavakoli, M., and Huang Q. (2010), 'Nonlinear adaptive bilateral control of teleoperation systems with uncertain dynamics and kinematics', *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4244-4249.
- Liu, Y.-H., Xu, Y., and Bergerman, M. (1999), 'Cooperation control of multiple manipulators with passive joints', *IEEE Transactions on Robotics and Automation*, 15, 258-267.
- Martinez-Rosas, J.C., Arteaga, M.A., and Castillo-Sanchez, A.M. (2006), 'Decentralized control of cooperative robots without velocity-force measurements', *Automatica*, 42, 329-336.
- Nijmeijer, H., and Rodriguez-Angeles, A. (2003), *Synchronization of mechanical systems*, Singapore: World Scientific.
- Ouyang, P.R., Dam, T., Huang, J., and Zhang, W.J. (2012), 'Contour tracking control in position domain', *Mechatronics*, 22, 934-944.
- Ren L., Mills, J.K., and Sun, D. (2006), 'Adaptive synchronized control for a planar parallel manipulator: theory and experiments', *ASME Journal of Dynamic Systems, Measurement, and Control*, 128, 976-979.
- Ren L., Mills, J.K., and Sun, D. (2008), 'Trajectory tracking control for a 3-DOF planar parallel manipulator using the convex synchronized control method', *IEEE Transactions on Control Systems Technology*, 16, 613-623.
- Rocha, R., Dias, J., and Carvalho, A. (2005), 'Cooperative multi-robot systems: A study of vision-based 3-D mapping using information theory', *Robotics and Autonomous Systems*, 53, 282-311.
- Rodriguez-Angeles, A., and Nijmeijer, H. (2004), 'Mutual synchronization of robots via estimated state feedback: a cooperative approach', *IEEE Transactions on Control Systems Technology*, 12, 542-554.

Shan, J. Liu, H.-T., and Nowotny, S. (2005), 'Synchronised trajectory-tracking control of multiple 3-DOF experimental helicopters', *IEE Proceedings Control Theory and Applications*, 152, 683-692.

Shan, J. (2008), 'Six-degree-of-freedom synchronised adaptive learning control for spacecraft formation flying', *IET Control Theory and Applications*, 2, 930-949.

Shang, W., Cong, S., Zhang, X., and Liang, Y. (2009), 'Active joint synchronization control for a 2-DOF redundantly actuated parallel manipulator', *IEEE Transactions on Control Systems Technology*, 17, 416-423.

Slotine, J.J. & Li, W. (1991), *Applied Nonlinear Control*, New Jersey: Prentice-Hall.

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

Sun, D. (2003), 'Position synchronization of multiple motion axes with adaptive coupling control', *Automatica*, 39, 997-1005.

Sun, D., Lu, R., Mills, J.K., and Wang, C. (2006), 'Synchronous tracking control of parallel manipulators using cross-coupling approach', *International Journal of Robotics Research*, 25, 1137-1147.

Sun, D., Shao, X., and Feng, G. (2007), 'A model-free cross-coupled control for position synchronization of multi-axis motions: theory and experiments', *IEEE Transactions on Control Systems Technology*, 15, 306-314.

Sun, D., and Tong, M. C. (2009), 'A synchronization approach for the minimization of contouring errors of CNC machine tools', *IEEE Transactions on Automation Science and Engineering*, 6, 720-729.

Sun, D. (2010), *Synchronization and Control of Multiagent Systems*, New York: CRC Press.

Su, Y., Sun, D., Ren, L., and Mills, J.K. (2006), 'Integration of saturated PI synchronous control and PD feedback for control of parallel manipulators', *IEEE Transactions on Robotics*, 22, 202-207.

Wang, H., and Xie, Y. (2009a), 'Passivity based adaptive Jacobian tracking for free-floating space manipulators without using spacecraft acceleration', *Automatica*, 45, 1510-1517.

Wang, H., and Xie, Y. (2009b), 'Prediction error based adaptive Jacobian tracking of robots with uncertain kinematics and dynamics', *IEEE Transactions on Automatic Control*, 54, 2889-2894.

Wang, H. (2013), 'Passivity based synchronization for networked robotic systems with uncertain kinematics and dynamics', *Automatica*, 49, 755-761.

Yu, S., Yu, X., Shirin zadeh, B., and Man, Z. (2005), 'Continuous finite-time control for robotic manipulators with terminal sliding mode', *Automatica*, 41, 1957-1964.

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

Zhao, D., Li, S., and Gao, F. (2008), 'Fully adaptive feedforward feedback synchronized tracking control for Stewart Platform systems', *International Journal of Control, Automation, and Systems*, 6, 689-701.

Zhao, D., Li, S., and Gao, F. (2009), 'Finite time position synchronised control for parallel manipulators using fast terminal sliding mode', *International Journal of Systems Science*, 40, 829-843.

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

Zhao, D. and Zou T. (2012), 'A finite-time approach to formation control of multiple mobile robots with terminal sliding mode', *International Journal of Systems Science*, 43, 1998-2014.

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