

Cooperative Path Planning and Constraints Analysis for Master-Slave Industrial Robots

Regular Paper

Yahui Gan^{1,2,*}, Xianzhong Dai^{1,2} and Jun Li^{1,2}

1 School of Automation, Southeast University, No.2 Sipailou, Nanjing, China

2 Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, No.2 Sipailou, Nanjing, China

* Corresponding author E-mail: ganyahui@yeah.net

Received 04 May 2012; Accepted 06 July 2012

DOI: 10.5772/51374

© 2012 Gan et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract A strategy of cooperative path planning for a master-slave multiple robot system is presented in this paper. The path planning method is based on motion constraints between the end-effectors of cooperative robots. Cooperation motions have been classified into three types by relative motions between end-effectors of master and slave robots, which is concurrent cooperation, coupled synchronous cooperation and combined synchronous cooperation. Based on this classification, position /orientation constraints and joint velocity constraints are explored in-depth here. In order to validate the path planning method and the theoretical developments in motion constraints analysis, representative experiments based on two industrial robots, Motoman VA1400 and HP20, are provided at the end of the paper. The experimental results have proved both the effectiveness of the path planning method and the correctness of the constraints analysis.

Keywords Cooperative path planning, Master-slave multiple robots, Cooperation motions, Motion constraints

1. Introduction

Cooperative multiple robot systems is a hot topic in robotics and much literature has been produced in this field. A multiple robot system surpasses a single robot system greatly in terms of flexibility and versatility in task execution. In particular, multiple robot cooperation is required if we are to assemble and manipulate parts without the aid of fixtures or jigs or if we are to smoothly transfer heavy and voluminous objects from one place to another. An integral part of the development of a multiple robot cooperation system is how to automate motion planning and control in spite of the increased complexity from redundancy in system degrees-of-freedom. This paper focuses on how to automate the cooperative path planning problem for a multiple robot system based on motion constraints between the end-effectors of cooperative robots. The mathematical expressions developed in the motion constraints analysis here establish partly the theoretical foundation for construction of next generation robot controllers with cooperation function.

Until now, strategies for multiple robot control can be divided into three types: master-slave control [1–3], hybrid position-force control [4–6] and dynamic coordination control [7–9]. All these methods pay their attention to the control of the robots themselves. However, a multiple robot system is an intrinsic interactive system and there are certain kinematic and dynamic physical constraints that must be satisfied during their cooperation in task execution. So, in our opinion, in order to enable cooperation between two robots one must keep their kinematic and dynamic motion constraints satisfied. Therefore, cooperation control strategies for a multiple robot system should concentrate on the motion constraints analysis. References [10] and [11] support such opinions. In their work, a formulation of system model based on nonlinear differential algebraic system theory is presented that integrates the kinematic constraints as the algebraic part of system model. But their work excessively simplified the constraints expression. Reference [12] deeply explored motion constraints between two cooperative robots in three types of cooperation motion, but their classification of cooperation motion does not comply with some industrial applications, such as jigless arc welding or assembly. Motivated by these opinions, we investigated the kinematic and dynamic motion constraints between cooperative robots, especially for some industrial application orientated tasks. Based on such motion constraints, a cooperative path planning method is presented in this paper.

The remainder of this paper is organized as follows: Section 2 gives the problem statement and literature review of cooperative path planning for multiple robot systems. Position and orientation constraints between master-slave cooperative robots are introduced in Section 3. Section 4 analyses the joint velocity constraints between master-slave cooperative robots. In addition to the theoretical results, representative experiments for two cooperated industrial robots is presented in Section 5. Concluding remarks follow in Section 6.

2. Problem Statement and Literature Review

2.1 Problem Statement

In control of multiple robot cooperation systems, the master-slave method is still the state of art for industrial robots [13–15]. The master-slave control strategy has to identify one robot as master, the other as slaves. Cooperation control is based on Cartesian coordinate control of the master robot and the slaves are controlled through motion constraint relations. For a cooperative welding system, the positioning robot that holds the workpiece is usually identified as the master and the processing robots that hold the weld torch as slaves. For a cooperative transportation system working in load sharing mode, any robot can be identified as master.

To simplify the motion constraints analysis between such master-slave robots, cooperation motion has to be classified into three types according to relative motions between the end-effectors of the master and slave robot,

which is,

(1) *Concurrent cooperation* :

Irrespective of what motions they have been executing before, a simultaneous motion start time is forced for all robots at synchronization point t_0 . No position or orientation constraints exist in this type of cooperation.

(2) *Coupled synchronous cooperation* :

Irrespective of what motions they have been executing before, a simultaneous motion start time is forced for all robots at synchronization point t_0 . All robots are synchronized and perform identical line or circle arc motions and no relative motion exists between master and slave end-effectors.

The slave robot follows the motions of the master robot, without executing motion blocks of its own. This form of motion is used in load sharing. Figure 1 illustrates a typical two robot cooperation system performing the couple synchronous cooperation in load-sharing mode. In such a cooperation motion, the slave robot requires no separate motion commands of its own because it follows the motion of the master directly.



Figure 1. Two robots cooperating for transportation in coupled synchronous motion.

(3) *Combined synchronous cooperation* :

Irrespective of what motions they have been executing before, an identical motion time is forced for all robots in the synchronization period Δt . The master robot performs a basic line or circle arc motion, while the slaves perform different motion blocks relative to the end-effector frame of the master, producing a superposed motion on the basic motion of the master robot.

The slave robot follows the basic motions of the master robot, but also executes motion blocks of its own. This form of motion is used in process-dependent mode. Figure 2 shows a typical combined synchronous cooperation of two robots for welding. In such a case, the slave robot processes the workpiece while it is being transferred from one station to another by the master. The master acts as a transportation and positioning robot, while the slave is used as a process robot. During the transferring process,

the master keeps the workpiece constantly in an optimal processing position. The relative position and orientation between the workpiece and tool is retained during motions of the workpiece.

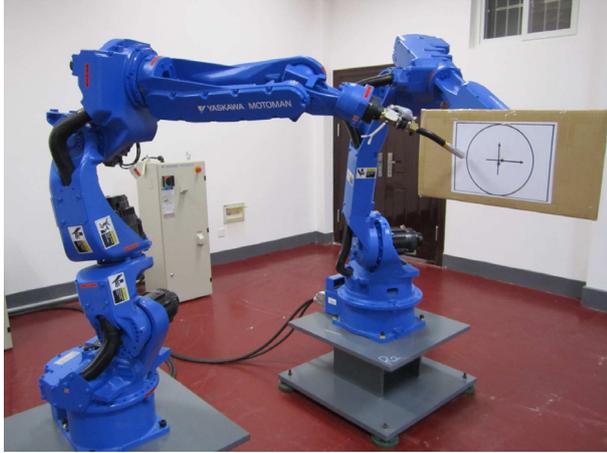


Figure 2. Two robots cooperating for curve tracking in combined synchronous motion.

In conventional robotic systems, path planning is usually carried out in two spaces, the Cartesian space and robot configuration space. The robot base frame is the frequently used Cartesian space where continuous-path trajectory, including line motion and circle arc motion, is planned. The robot tool frame and user defined frame also are Cartesian spaces, but they are seldom used for a single robot system. Robot configuration space is conventionally used in the point-to-point trajectory planning problem and is not suitable for arc welding application. Therefore, the path planning problem for a single robot is to identify two robot positions for line motion and three for circle arc motion in the robot base frame. However, when it comes to a multiple robot cooperation system, things become much more complex.

A typical two robot cooperation system for welding is shown in Figure 3. For such a cooperation system, the trajectory teaching problem has many differences to a single robot system. First, many Cartesian frames exist, such as the absolute world frame, each robot base frame, robot tool frame and some user defined frames. Second, trajectory description for the welding robot only exists relative to workpiece frame, which is a moving dynamic frame. The trajectory description for the welding robot in its base frame or absolute world frame is not obvious, which makes the trajectory planning problem beyond the capabilities of a conventional robot controller. Third, simultaneous motion start/stop and motion synchronization have to be realized between different robot controllers. So far, trajectory teaching and the planning problem for multiple robot cooperation systems can be formed as follows:

Cooperative path planning for a multiple robot system is to identify one robot as master and the other as slaves, determine two Cartesian positions for line motion and three Cartesian positions for circle arc motion in the master base frame for the

master robot. For coupled synchronous cooperation, no further information needs to be collected for slaves. For concurrent cooperation or combined synchronous cooperation, determine two Cartesian positions for line motion and three Cartesian positions for circle arc motion in the slave base frame for the slave robots. After that, plan nominal trajectories in configuration space for each robot by this collected information to achieve cooperative motions between master and slave.

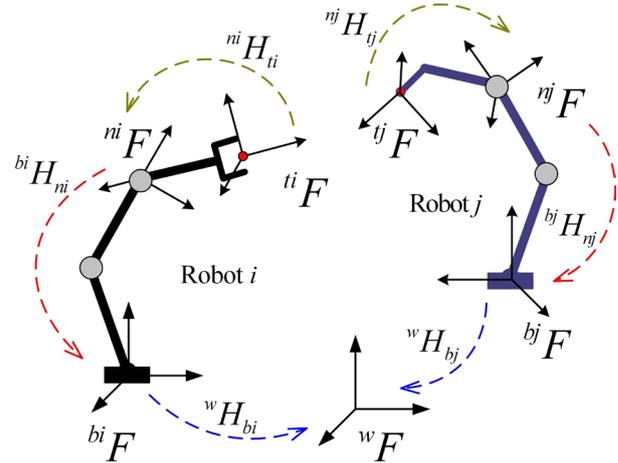


Figure 3. Schematic diagram of a two robot cooperation system for jigless welding.

2.2 Literature Review

By far, limited research has reported on motion constraints analysis between cooperative robots. Motion constraints for general cooperation process encounters great difficulty due to redundancy in system degrees-of-freedom and complexity in curve description of robot trajectory. However, when this problem is constrained to an industrial process, it becomes easier because only a few curves of motion trajectory exist, such as line, circle arc and helix.

Taking the flexible multiple robot cooperative welding system as an example, kinematic constraint relations between the welding robot and the positioning robot are of vital importance, which directly determines the weld quality. As for the arc welding process, reference [16] presents a simple and automated programming method of a redundant multiple robot system for concurrent motion. However, constraint relations between the end-effectors of cooperative robots are not further expounded. In reference [17], the kinematics of coordinated motion and the multi-robot programming schemes are reviewed, which expresses the motion of the welding robot as a Cartesian path in a moving reference frame attached to a 2 DOF part positioner. Reference [18] presents a strategy to simultaneously coordinate motions of a welding robot and a positioning table to track sharp corners along a Cartesian path. However, none of this research deeply expounds upon the kinematic constraints between cooperative robots. Constrained relations between two coordinated industrial robots for motion control have been explored in reference [12], where both the posture and joint velocity

constraint relations are discussed. There, three types of cooperation are classified according to object differences: two robots handling a rigid-body object, two robots handling a pair of pliers and two robots handling an object with a spherical joint. Their conclusions are very heuristic but not feasible for the jigless welding process which does not belong to any type of classification in reference [12].

3. Position and orientation constraints analysis between master-slave cooperative robots

As for the above three types of cooperation, concurrent cooperation only requires a simultaneous motion start time and no position or orientation constraints exist in this type, so no further investigations will be carried out for this type. As for the remaining two types, constraint relations in coupled synchronous cooperation is much simpler than the combined synchronous cooperation, because no relative motion exists between the master and slave end-effectors for couple cooperation.

A homogeneous transformation matrix is usually adopted to represent position and orientation of the robot end-effector, which forms like

$$H = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}_{4 \times 4}, \quad H^{-1} = \begin{bmatrix} R^T & -R^T T \\ 0 & 1 \end{bmatrix}_{4 \times 4} \quad (1)$$

where $R \in \mathbb{R}^{3 \times 3}$ is rotation matrix and $T \in \mathbb{R}^3$ is translation vector. Before this rotation matrix and translation vectors are applied, their coordinates must be established in the correct Cartesian frame. In a multiple robot system, many Cartesian frames exist, such as the world frame, the base frame and tool frame, for each robot. Relative position/orientation for these frames can also be represented by a homogeneous transformation matrix. Figure 3 shows such relative frames and the transformation matrix between two cooperated robots in a multiple robot system.

The desired motion trajectory for a robot can be specified as a time varying transformation matrix of its end-effector in the form of

$$M(t) = \begin{bmatrix} R(t) & T(t) \\ 0 & 1 \end{bmatrix} \quad (2)$$

Now, the position and orientation constraints between two cooperative master-slave robots for the last two types, coupled synchronous cooperation and combined synchronous cooperation, will be investigated.

3.1 Coupled synchronous cooperation

Consider two robots, each with n joints, handling a voluminous rigid body object (as shown in Figure 1) that is beyond the payload capacity of any individual robot. To move the object, two robot end-effectors must grasp it at two specified points. It is assumed that the end-effectors furnish tight grips so that there are no relative motions among the end-effectors and the object. Followed by the definition of coupled synchronous cooperation, no relative motion exists between master and slave end-effectors once their motion has been coupled, which means relative position/orientation between master end-effector and

slave end-effector will hold constant during the entire period of cooperation.

Let ${}^{mb}P_m(t)$ be the position/orientation of the master robot in the master base frame and ${}^{sb}P_s(t)$ be the position/orientation of the slave robot in the slave base frame at time t . Let coordinates for the slave end-effector in the master base frame be ${}^{mb}P_s(t)$, we have

$${}^{mb}P_s(t) = {}^{mb}H_{sb} \cdot {}^{sb}P_s(t) \quad (3)$$

where ${}^{mb}H_{sb}$ is the transformation matrix from slave base frame to master base frame. Once the robots have been mounted, their base will not be changed during the entire period of cooperation. So ${}^{mb}H_{sb}$ will be a constant homogeneous transformation matrix during the entire process, which can be determined by the method proposed in reference [19]. Let t_0 be the simultaneous start time for ${}^{mb}P_m(t)$ and ${}^{sb}P_s(t)$. Let mH_s be the transformation matrix from slave end-effector to master end-effector in the master base frame at time t_0 , which means

$${}^{mb}P_m(0) = {}^mH_s \cdot {}^{mb}P_s(0) \quad (4)$$

Substituting equation (3) into (4) at time $t = t_0$ yields

$${}^{mb}P_m(0) = {}^mH_s \cdot {}^{mb}H_{sb} \cdot {}^{sb}P_s(0) \quad (5)$$

By equation (5), we have

$${}^mH_s = {}^{mb}P_m(0) \cdot \left({}^{sb}P_s(0) \right)^{-1} \cdot {}^{sb}H_{mb} \quad (6)$$

As with the definition of coupled synchronous cooperation, mH_s will be hold constant during the entire period of cooperation, which means

$$\begin{aligned} {}^{mb}P_m(t) &= {}^mH_s \cdot {}^{mb}P_s(t) \\ &= {}^mH_s \cdot {}^{mb}H_{sb} \cdot {}^{sb}P_s(t) \\ &= {}^{mb}P_m(0) \cdot \left({}^{sb}P_s(0) \right)^{-1} \cdot {}^{sb}H_{mb} \cdot {}^{mb}H_{sb} \cdot {}^{sb}P_s(t) \quad (7) \\ &= {}^{mb}P_m(0) \cdot \left({}^{sb}P_s(0) \right)^{-1} \cdot {}^{sb}P_s(t) \end{aligned}$$

By equation (7), we have

$${}^{sb}P_s(t) = {}^{sb}P_s(0) \cdot \left({}^{mb}P_m(0) \right)^{-1} \cdot {}^{mb}P_m(t) \quad (8)$$

Equation (8) expresses the holonomic constraints for the position and orientation between master and slave robots during their coupled cooperation motion. This equation also implies that the slave trajectory can be uniquely determined in terms of those of the master and relative position/orientation at the initial time of master-slave couple cooperation. This equation establishes the theoretical foundation of our cooperative path planning method for coupled synchronous cooperation. Invoking (1), this constraint can be decomposed into two parts, one for position and the other for orientation. Equation (9) is the individual constraint relations for coupled synchronous cooperation, the first part for orientation and the second part for position. In equation (9) there are 12 scalar equations, but only six of them are independent because the orientation of the end-effector can be uniquely defined by three parameters, such as the Euler angles. Transformation between Euler angles and the rotation matrix has been provided in Appendix A.

$$\begin{cases} {}^{sb}R_s(t) = {}^{sb}R_s(0) \cdot {}^{mb}R_m^T(0) \cdot {}^{mb}R_m(t) \\ {}^{sb}T_s(t) = {}^{sb}R_s(0) \cdot {}^{mb}R_m^T(0) \cdot {}^{mb}T_m(t) - {}^{sb}R_s(0) \cdot {}^{mb}R_m^T(0) \cdot {}^{mb}T_m(0) + {}^{sb}T_s(0) \end{cases} \quad (9)$$

3.2 Combined synchronous cooperation

Consider two robots cooperate for a jigless welding process as shown in Figure 2. Here, the master robot holds the workpiece while the slave holds a processing tool. In such a case, motion trajectory for the master robot must be specified beforehand, whereas slave trajectory can not be specified directly and it must be determined by the processing requirement in order to achieve a cooperative motion which is a combined one. Usually, the processing requirement can only be given in the workpiece frame. It is supposed that no relative motion is allowed when the master robot holds the workpiece. In such a supposition, the processing requirement can also be specified or converted to the master end-effector frame because a constant transformation matrix exists between the workpiece frame and the master end-effector frame. Constraint relations for such cooperative motions of the master-slave robots will then be different from that of the coupled case. Unlike the constant relation in the first case, a definite time-varying constraint of the processing requirement must be retained during the combined cooperation.

Let ${}^{mb}P_m(t)$ be the motion trajectory for the master robot in its base frame. Let ${}^{me}P_s(t)$ be the processing requirement in the master end-effector frame, which is also the slave motion trajectory in the master end-effector frame. By robot inverse kinematics, trajectory representation ${}^{mb}P_m(t)$ can be converted to master joint space for executing the motion. While the slave trajectory ${}^{me}P_s(t)$ must be converted from the master end-effector frame to the slave base frame in order to execute the motion. By homogeneous transformation, we have

$${}^{sb}P_s(t) = {}^{sb}H_{mb} \cdot {}^{mb}H_{me}(t) \cdot {}^{me}P_s(t) \quad (10)$$

$$\begin{cases} {}^{mb}R_m(t) \cdot {}^{mb}R_{sb} \cdot {}^{sb}R_s(t) = {}^{me}R_s(t) \\ {}^{mb}R_m(t) \cdot {}^{mb}R_{sb} \cdot {}^{sb}T_s(t) + {}^{mb}R_m(t) \cdot {}^{mb}T_{sb} + {}^{mb}T_m(t) = {}^{me}T_s(t) \end{cases} \quad (13)$$

4. Constraints analysis for joint velocity between master-slave cooperative robots

Constraint relations for the position/orientation between master-slave robots have been shown in equations (9) and (13). Someone may suppose that constraints for joint velocity could be determined by taking the time derivative of such constraints for position and orientation. This could be effective for the translational velocity, however, the angular velocity for robot end-effector is not equal to the time derivative of orientation angular displacement. In reference [20], equations to calculate the angular velocity given Euler angles as in Appendix A are,

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} -\sin\theta & 0 & 1 \\ \cos\theta \sin\psi & \cos\psi & 0 \\ \cos\theta \cos\psi & -\sin\psi & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (14)$$

At the same time, derivative of rotation matrix with time has obscure meaning in physics and produces complex

where ${}^{sb}H_{mb}$ is the transformation matrix from the master base frame to the slave base frame, ${}^{mb}H_{me}(t)$ is the transformation matrix from the master end-effector frame to the master base frame. ${}^{sb}H_{mb}$ can be calibrated by the method proposed in reference [19]. As for the definition of robot forward kinematics, ${}^{mb}H_{me}(t)$ is the inverse of ${}^{me}H_{mb}(t)$, the position/orientation of the robot end-effector in its base frame,

$${}^{mb}H_{me}(t) = ({}^{me}H_{mb}(t))^{-1} = ({}^{mb}P_m(t))^{-1} \quad (11)$$

Substituting (11) to (10) yields,

$${}^{sb}P_s(t) = {}^{sb}H_{mb} \cdot ({}^{mb}P_m(t))^{-1} \cdot {}^{me}P_s(t) \quad (12)$$

Equation (12) expresses the holonomic constraints for the position and orientation between master and slave robots during their combined cooperation motion. This equation also implies that the slave trajectory can be uniquely determined in terms of the counterpart of the master and task processing requirement. This equation establishes the theoretical foundation of our cooperative path planning method for combined synchronous cooperation. Identically, this constraint can also be decomposed into two parts by invoking equation (1), one for position and the other for orientation. Equation (13) is the individual constraint relations for combined synchronous cooperation, the first part for orientation and the second part for position. In equation (13) there are also 12 scalar equations and only six of them are independent identically.

forms of orientation angles. Therefore, constraints for

angular velocity will be analysed directly from the definitions of different cooperation types.

Relations between the robot joint velocity and its end-effector position velocity/ angular velocity have been provided in reference [20] as,

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} J_\lambda \\ J_a \end{bmatrix} \dot{q} \quad (15)$$

where $q = (q_1, \dots, q_n)^T \in \mathbb{R}^n$ is the joint displacement vector for a robot with n degrees-of-freedom, $v = (v_x, v_y, v_z)^T \in \mathbb{R}^3$ is the translational velocity for robot end-effector in its base frame, $\omega = (\omega_x, \omega_y, \omega_z)^T \in \mathbb{R}^3$ is the angular velocity for the robot end-effector in its base frame, $J_\lambda = (J_{\lambda 1}, \dots, J_{\lambda n})^T \in \mathbb{R}^{3 \times n}$ is the positional part of Jacobian matrix and $J_a = (J_{a 1}, \dots, J_{a n})^T \in \mathbb{R}^{3 \times n}$ is

the orientational part of the Jacobian matrix. The Jacobian matrix $J = (J_{\lambda}^T, J_a^T)^T$ relates the position and angular velocities of the end-effector with reference to base frame respectively to \dot{q} .

4.1 Coupled synchronous cooperation

Let q_m and q_s be the joint displacement for master and slave robot respectively. Let ${}^{mb}v_m, {}^{mb}\omega_m$ be the translational and angular velocities of the master robot in the master base frame and ${}^{sb}v_s, {}^{sb}\omega_s$ be the translational and angular velocities of the slave robot in the slave base frame. By equation (15), we have

$$\begin{aligned} {}^{mb}v_m &= J_{\lambda m} \cdot \dot{q}_m, & {}^{mb}\omega_m &= J_{a m} \cdot \dot{q}_m \\ {}^{sb}v_s &= J_{\lambda s} \cdot \dot{q}_s, & {}^{sb}\omega_s &= J_{a s} \cdot \dot{q}_s \end{aligned} \quad (16)$$

Constraint relations for the translational velocity between master and slave end-effectors can be obtained by differentiating the second part of equation (9) with respect to time t , which is

$${}^{sb}\dot{T}_s(t) = {}^{sb}R_s(0) \cdot {}^{mb}R_m^T(0) \cdot {}^{mb}\dot{T}_m(t) \quad (17)$$

As the definition of translational velocity, we have

$$\begin{aligned} {}^{mb}v_m(t) &= {}^{mb}\dot{T}_m(t) \\ {}^{sb}v_s(t) &= {}^{sb}\dot{T}_s(t) \end{aligned} \quad (18)$$

Substituting (18) into (17) yields

$${}^{sb}v_s(t) = {}^{sb}R_s(0) \cdot {}^{mb}R_m^T(0) \cdot {}^{mb}v_m(t) \quad (19)$$

By invoking (16), we have

$$J_{\lambda s} \cdot \dot{q}_s(t) = {}^{sb}R_s(0) \cdot {}^{mb}R_m^T(0) \cdot J_{\lambda m} \cdot \dot{q}_m(t) \quad (20)$$

$$\dot{q}_s(t) = J_{\lambda s}^+ \cdot {}^{sb}R_s(0) \cdot {}^{mb}R_m^T(0) \cdot J_{\lambda m} \cdot \dot{q}_m(t) \quad (21)$$

where $J_{\lambda s}^+$ is the Moore-Penrose Pseudo Inverse of matrix $J_{\lambda s}$.

Equation (21) establishes half of the constraint relations for master-slave joint velocity in the positional aspect. Constraints are also imposed on the angular velocities of

$${}^{mb}\dot{R}_m(t) \cdot {}^{mb}R_{sb} \cdot {}^{sb}T_s(t) + {}^{mb}\dot{R}_m(t) \cdot {}^{mb}R_{sb} \cdot {}^{sb}\dot{T}_s(t) + {}^{mb}\dot{R}_m(t) \cdot {}^{mb}T_{sb} + {}^{mb}\dot{T}_m(t) = {}^{me}\dot{T}_s(t) \quad (27)$$

However, the differential of the rotation matrix appears in equation (27) and this will lead to an unaffordably complex computation. Reference [21] provided a simple method to calculate the time differential of the rotation matrix R , which is

$$\frac{dR(t)}{dt} = W^A \cdot R(t) = R(t) \cdot W^B \quad (28)$$

where

$$W^A = \begin{bmatrix} 0 & -\dot{q}_2 q_1^T & \dot{q}_1 q_3^T \\ \dot{q}_2 q_1^T & 0 & -\dot{q}_3 q_2^T \\ -\dot{q}_1 q_3^T & \dot{q}_3 q_2^T & 0 \end{bmatrix}$$

$$W^B = \begin{bmatrix} 0 & -p_2^T p_1 & p_1^T p_3 \\ p_2^T p_1 & 0 & -p_3^T p_2 \\ -p_1^T p_3 & p_3^T p_2 & 0 \end{bmatrix}$$

the two end-effectors, but it cannot be obtained through differentiation of the corresponding constraint equations. As the definition of coupled synchronous cooperation, there are no relative motions between the master and slave end-effectors. So their angular velocities are always the same, that is

$${}^{mb}\omega_s = {}^{mb}\omega_m \quad (22)$$

Angular velocity can be converted from one frame to another by the equations provided in reference [21],

$${}^{sb}\omega_s = {}^{sb}R_{mb} \cdot {}^{mb}\omega_s \quad (23)$$

Substituting (22) and (16) into (23) yields

$$J_{a s} \cdot \dot{q}_s = {}^{sb}R_{mb} \cdot J_{a m} \cdot \dot{q}_m \quad (24)$$

$$\dot{q}_s = J_{a s}^+ \cdot {}^{sb}R_{mb} \cdot J_{a m} \cdot \dot{q}_m \quad (25)$$

where $J_{a s}^+$ is the Moore-Penrose pseudo inverse of the matrix $J_{a s}$. Equation (25) establishes another part of the constraint relations for master-slave joint velocity in the orientational aspect. This conclusion is quite different from the result in reference [12]. Combining the equation (21) and (25), we have

$$\dot{q}_s = \begin{bmatrix} J_{\lambda s}^+(q_s) \cdot {}^{sb}R_s(0) \cdot {}^{mb}R_m^T(0) \cdot J_{\lambda m}(q_m) \\ J_{a s}^+(q_s) \cdot {}^{sb}R_{mb} \cdot J_{a m}(q_m) \end{bmatrix} \dot{q}_m \quad (26)$$

Equation (26) gives the constraint relations between the joint velocities of the master-slave robots when they move in coupled synchronous cooperation. Actually, once the joint velocities of the follower are determined, its joint displacements can be obtained by integration. Consequently, the robot motions are uniquely specified by the joint velocities and the initial joint displacements.

4.2 Combined synchronous cooperation

Identically, constraint relations for the translational velocity of master-slave end-effectors can be obtained by differentiating the second part of equation (13) with respect to time t , as shown in equation (27).

in which $q_i = (r_{i1}, r_{i2}, r_{i3}), i = 1, 2, 3$ is the row vector of rotation matrix R , $p_j = (r_{1j}, r_{2j}, r_{3j})^T, j = 1, 2, 3$ is the column vector of rotation matrix R .

Let $v(t)$ be the relative translational velocity of the slave robot end-effector with reference to the master end-effector frame, we have

$$v(t) = {}^{me}\dot{T}_s(t) \quad (29)$$

By equation (18), (27), (28), and (29), we have the equation (30) and (31).

Substituting (16) into (31) yields the equation (32). Solving the equation (32), it leads to a equation for joint velocity as (33).

$$\mathbf{W}^A \left({}^{\text{mb}}R_{\text{m}}(t) \right) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot {}^{\text{sb}}T_{\text{s}}(t) + {}^{\text{mb}}R_{\text{m}}(t) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot v_{\text{s}}(t) + \mathbf{W}^A \left({}^{\text{mb}}R_{\text{m}}(t) \right) \cdot {}^{\text{mb}}T_{\text{sb}} + v_{\text{m}}(t) = v(t) \quad (30)$$

$${}^{\text{mb}}R_{\text{m}}(t) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot v_{\text{s}}(t) = -v_{\text{m}}(t) + v(t) - \mathbf{W}^A \left({}^{\text{mb}}R_{\text{m}}(t) \right) \left[{}^{\text{mb}}R_{\text{m}}(t) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot {}^{\text{sb}}T_{\text{s}}(t) + {}^{\text{mb}}T_{\text{sb}} \right] \quad (31)$$

$${}^{\text{mb}}R_{\text{m}}(t) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot J_{\lambda_{\text{s}}} \cdot \dot{q}_{\text{s}}(t) = -v_{\text{m}}(t) + v(t) - \mathbf{W}^A \left({}^{\text{mb}}R_{\text{m}}(t) \right) \left[{}^{\text{mb}}R_{\text{m}}(t) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot {}^{\text{sb}}T_{\text{s}}(t) + {}^{\text{mb}}T_{\text{sb}} \right] \quad (32)$$

$$\dot{q}_{\text{s}}(t) = J_{\lambda_{\text{s}}}^+ \cdot {}^{\text{sb}}R_{\text{s}}(0) \cdot {}^{\text{mb}}R_{\text{m}}^{\text{T}}(t) \cdot \left\{ -J_{\lambda_{\text{m}}} \cdot \dot{q}_{\text{m}}(t) - \mathbf{W}^A \left({}^{\text{mb}}R_{\text{m}}(t) \right) \left[{}^{\text{mb}}R_{\text{m}}(t) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot {}^{\text{sb}}T_{\text{s}}(t) + {}^{\text{mb}}T_{\text{sb}} \right] \right\} \quad (33)$$

Equation (33) establishes half of the constraint relations for master-slave joint velocity in the positional aspect. Constraints of the other part in the orientational aspect will also be established directly by the definition of combined synchronous cooperation. Let ${}^{\text{me}}\omega_{\text{s}}$ be the angular velocity of the slave end-effector with reference to the master end-effector frame. As the definition of combined synchronous cooperation, ${}^{\text{me}}\omega_{\text{s}}$ will be specified beforehand by the task processing requirement $\omega(t)$, that is

$${}^{\text{me}}\omega_{\text{s}} = \omega(t) \quad (34)$$

Transform this angular velocity from the master end-effector frame to the slave base frame by invoking the angular velocity conversion equation in [21],

$${}^{\text{me}}R_{\text{mb}}(t) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot {}^{\text{sb}}\omega_{\text{s}}(t) = \omega(t) \quad (35)$$

By equation (35), we have

$${}^{\text{sb}}\omega_{\text{s}}(t) = {}^{\text{mb}}R_{\text{sb}}^{\text{T}} \cdot {}^{\text{me}}R_{\text{mb}}^{\text{T}}(t) \cdot {}^{\text{me}}\omega(t) \quad (36)$$

In equation (36), the rotation matrix ${}^{\text{me}}R_{\text{mb}}(t)$ is the orientational matrix of the master end-effector with reference to the master base frame. Actually, once

$$\dot{q}_{\text{s}} = \left[\begin{array}{c} J_{\lambda_{\text{s}}}^+ \cdot {}^{\text{sb}}R_{\text{s}}(0) \cdot {}^{\text{mb}}R_{\text{m}}^{\text{T}}(t) \cdot \left\{ -J_{\lambda_{\text{m}}} \cdot \dot{q}_{\text{m}}(t) - \mathbf{W}^A \left({}^{\text{mb}}R_{\text{m}}(t) \right) \left[{}^{\text{mb}}R_{\text{m}}(t) \cdot {}^{\text{mb}}R_{\text{sb}} \cdot {}^{\text{sb}}T_{\text{s}}(t) + {}^{\text{mb}}T_{\text{sb}} \right] \right\} \\ J_{\lambda_{\text{s}}}^+ \cdot {}^{\text{sb}}R_{\text{sb}} \cdot \Gamma^{\text{T}} \left(J_{\lambda_{\text{m}}} \cdot \dot{q}_{\text{m}} \right) \cdot \omega(t) \end{array} \right] \quad (41)$$

Equation (41) gives the constraint relations between the joint velocities of the master-slave robots when they move in coupled synchronous cooperation. By now, constraint relations of the position/orientation and joint velocity for the master-slave robots have been definitely proposed, both in couple synchronous cooperation and combined synchronous cooperation. These conclusions can be used to solve the cooperative path planning problem for cooperative motions in a multiple robot system.

5. Representative experiments

In order to validate the above theoretical developments and conclusions, representative experiments have been carried out in our lab. The testbed is composed of two industrial robots, Motoman VA1400 and HP20, which are produced by YASKAWA Electronic Corporation, Japan.

the angular velocity ${}^{\text{mb}}\omega_{\text{m}}(t)$ of the master end-effector with reference to the master base frame is determined, ${}^{\text{me}}R_{\text{mb}}(t)$ is determined. Appendix B illustrates how to calculate the rotation matrix $R(t)$ for the orientation of a rigid body given its angular velocity $\omega(t)$. Let the whole process in Appendix B be abbreviated as,

$${}^{\text{me}}R_{\text{mb}}(t) = \Gamma \left({}^{\text{mb}}\omega_{\text{m}}(t) \right) \quad (37)$$

Substituting the equation (16) and (37) into (36) yields,

$${}^{\text{sb}}\omega_{\text{s}}(t) = {}^{\text{sb}}R_{\text{mb}} \cdot \Gamma^{\text{T}} \left({}^{\text{mb}}\omega_{\text{m}}(t) \right) \cdot {}^{\text{me}}\omega(t) \quad (38)$$

$$J_{\lambda_{\text{s}}} \cdot \dot{q}_{\text{s}} = {}^{\text{sb}}R_{\text{mb}} \cdot \Gamma^{\text{T}} \left(J_{\lambda_{\text{m}}} \cdot \dot{q}_{\text{m}} \right) \cdot {}^{\text{me}}\omega(t) \quad (39)$$

By equation (39), we have

$$\dot{q}_{\text{s}} = J_{\lambda_{\text{s}}}^+ \cdot {}^{\text{sb}}R_{\text{mb}} \cdot \Gamma^{\text{T}} \left(J_{\lambda_{\text{m}}} \cdot \dot{q}_{\text{m}} \right) \cdot \omega(t) \quad (40)$$

where $J_{\lambda_{\text{s}}}^+$ is the Moore-Penrose pseudo inverse of matrix $J_{\lambda_{\text{s}}}$. Equation (40) establishes another part of the constraint relations for master-slave joint velocity in the orientational aspect. Combining the equation (33) and (40), we have equation (41).

An external PC, DELL Optiplex 780, is used as a top layer controller to implement all the control logic. This external PC is connected to the robot controller DX100 by Ethernet cable. The communication software MotoCom SDK, which is also provided by YASKAWA Electronic Corporation, is installed on the PC to help transfer robot data and job files between the PC and DX100 robot controller. Figure 4 shows the experimental testbed we used here in our lab.

Using the communication software MotoCom SDK, user applications can be developed and executed on a PC platform. Two different types of experiments have been carried out in our lab, one is the box transportation as shown in Figure 1 and the other is curve tracking as shown in Figure 2. MotoCom SDK provides users with the function to obtain the trajectory a robot moves, which forms the experiment data in the following analysis.

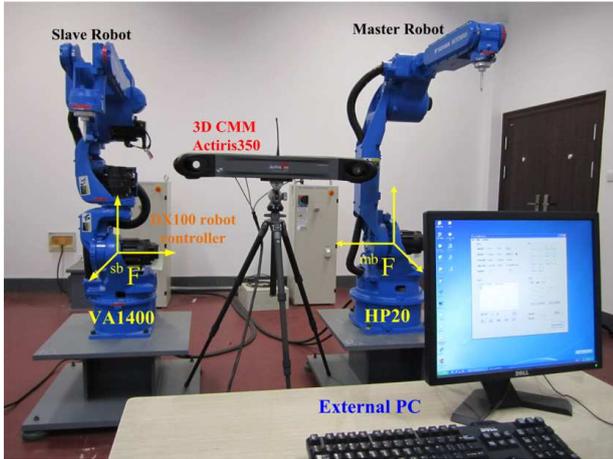


Figure 4. Experimental testbed in our lab.

5.1 Experiments of coupled synchronous cooperation

Experiments of Motoman VA1400 and HP20 robots cooperating in coupled synchronous motion are carried out here. As shown in Figure 1, the two robots grasp a cuboid box at two different points. To hold the box steady, there would be no relative motion between the end-effectors of the two robots.

In our experiments, relative position and orientation from the base frames of the VA1400 robot to the HP20 robot is

$${}^{mb}H_{sb} = (x, y, z, \phi, \theta, \psi) = [37.875, 1476.471, 54.463, 0.1266, -0.1009, -1.2169]$$

Here, the HP20 robot is identified as the master and VA1400 as the slave. A 6-dimensional vector $(x, y, z, \phi, \theta, \psi) \in \mathbb{R}^6$ is adopted here to represent the translation and rotation between based frames of the master robot and slave. The 3-dimension (x, y, z) indicates translation and 3-dimensional Euler angles (ϕ, θ, ψ) indicate rotation. The variables x, y, z has a unit of millimetre (mm) while variables ϕ, θ, ψ has a unit of degree ($^\circ$). This vector can be converted to a unique homogeneous transformation matrix. Values for ${}^{mb}H_{sb}$ are calibrated by an external sensor Actiris350. The Actiris350 system is a 3D coordinate measuring machine manufactured by ActCM Corporation, France. Actiris350 system has a precision of $\pm 15\mu m$ for single point according to ISO 10360-2, repeatability is $\pm 25\mu m$, acquisition speed is 15 measurements per second max. and measuring volume is $3.5m^3$. Figure 5 shows the 3D coordinate measuring machine Actiris350 we used.

For a line coupled synchronous cooperation motion, two points for the master robot trajectory must be taught by a teaching pendant. Here, the two teaching points for the HP20 robot in its base frame are

$${}^{mb}P_1 = [1041.655, -850.049, 470.555, 104.148, 18.007, -4.128]$$

$${}^{mb}P_2 = [1039.461, -851.654, -19.361, 104.149, 18.011, -4.127]$$



Figure 5. 3D coordinate measuring machine Actiris350.

These values are directly read out from the robot controller DX100 by function $BscIsRobotPos()$, which is one of the member functions provided by the communication software MotoCom SDK. MotoCom SDK only provides a 6-dimensional vector $(x, y, z, \phi, \theta, \psi) \in \mathbb{R}^6$ to represent position and orientation of the robot end-effector, 3-dimension (x, y, z) for position and 3-dimensional Euler angles (ϕ, θ, ψ) for orientation. The variables x, y, z also have a unit of millimetre (mm) while variables ϕ, θ, ψ have a unit of degree ($^\circ$). These values can also be found on the teaching pendant of the robot controller. The following experiment data are in the same formation and acquired in the same way by the MotoCom SDK function $BscIsRobotPos()$.

For a coupled synchronous motion, the begin point for the slave robot also needs to be taught. Here, the begin point for the VA1400 robot in its base frame is

$${}^{sb}P_I = [1052.787, 520.290, 530.149, -98.055, 19.813, 0.454]$$

As the definition of coupled synchronous cooperation, slave robots require no further teaching operation. Slave trajectories should be calculated autonomously according to the master trajectory by the above conclusion of equation (8) or (9). By equation (8), we have

$${}^{sb}P_{II} = [1050.596, 518.684, 40.265, -98.053, 19.816, 0.455]$$

After ${}^{sb}P_{II}$ is calculated, motion trajectories in Cartesian space for both master and slave robot can be planned. Figure 6 shows robot trajectories for the VA1400 and HP20 robots moving in a couple line synchronous motion relative to the base frame of the HP20 robot. Here, the base frame of the master robot was identified as the absolute world frame. The corresponding joint positions for each robot, master robot HP20 and slave robot VA1400, have been shown in Figure 7 and 8 respectively. Data used in these figures are collected from the robot controller DX100 by position read functions in MotoCom SDK software.

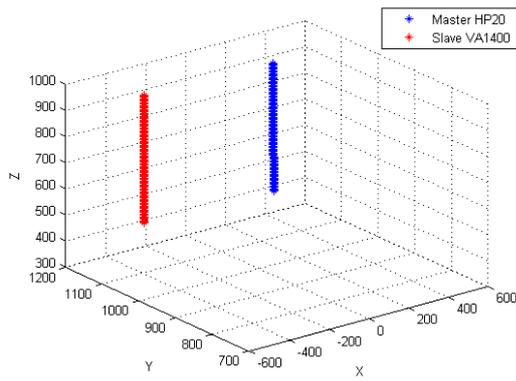


Figure 6. Trajectory of VA1400 and HP20 robots in couple line synchronous motion.

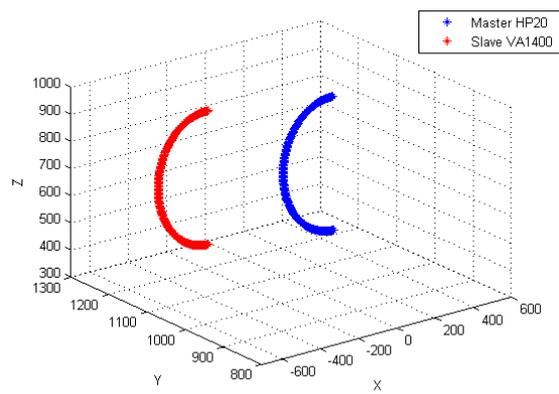


Figure 9. Trajectory of VA1400 and HP20 robots in couple circle synchronous motion.

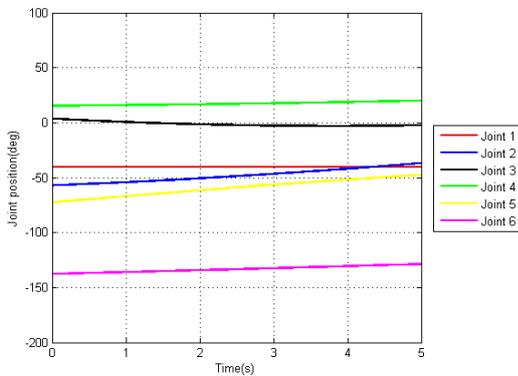


Figure 7. Joint position of master robot HP20 in couple line synchronous motion.

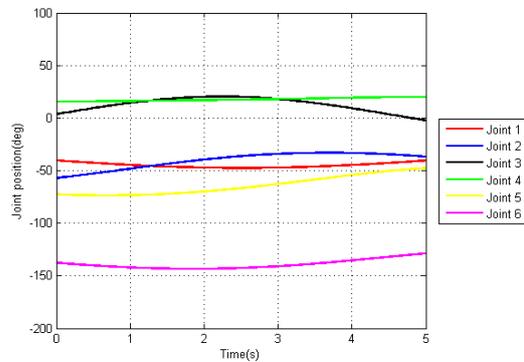


Figure 10. Joint position of master robot HP20 in couple circle synchronous motion.

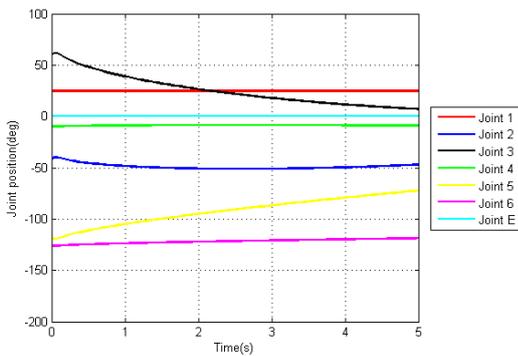


Figure 8. Joint position of slave robot VA1400 in couple line synchronous motion.

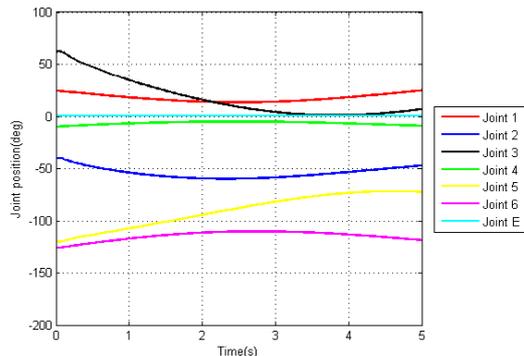


Figure 11. Joint position of slave robot VA1400 in couple circle synchronous motion.

Besides the couple line motion, couple circle synchronous motion has also been tested for the two robots. As for a circle arc trajectory, three points on the trajectory have to be identified. Here, only the third point ${}^{mb}P_3$ for master robot needs to be added, that is

$${}^{mb}P_3 = [1046.428, -1099.538, 239.018, 104.147, 18.011, -4.126]$$

Similarly, ${}^{sb}P_{III}$ can also be computed by equation (8),

$${}^{sb}P_{III} = [1057.506, 270.803, 298.586, -98.056, 19.816895, 0.456]$$

Figure 9 shows robot trajectories for the VA1400 and HP20 robots moving in a couple circle arc synchronous motion relative to the base frame of HP20 robot. The corresponding joint positions for each robot, master robot HP20 and slave robot VA1400, have been shown in Figure 10 and 11 respectively.

5.2 Experiments of combined synchronous cooperation

Experiments of Motoman VA1400 and HP20 robots cooperating in combined synchronous motion are carried out here. As shown in Figure 2, HP20 robot holds a panel while VA1400 carries an arc weld torch. In our experiments, the HP20 robot is identified as the master robot and VA1400 as the slave. The slave robot tracks the line or circle arc trajectories on the panel while the master transports the panel. The master robot adjusts the position and orientation of the panel constantly to keep it in an optimal posture for the slave processing.

Identically, relative position and orientation from the base frames of VA1400 robot to HP20 robot are

$${}^{mb}H_{sb} = [37.875, 1476.471, 54.463, 0.1266, -0.1009, -1.2169]$$

Firstly, the HP20 robot moves in a line motion from P_1 to P_2 in its base frame while the VA1400 robot moves in a line motion from P_1 to P_{II} relative to the end-effector frame of the HP20 robot. Teach points for master robot HP20 in its base frame are

$${}^{mb}P_1 = [1016.062, -757.138, 212.818, 96.433, -42.523, -7.834]$$

$${}^{mb}P_2 = [636.828, -765.975, 214.526, 96.433, -42.525, -7.833]$$

As the definition of combined synchronous cooperation, trajectory points also need to be taught for the slave robot. Here, two teach points in a line motion for the slave robot are

$${}^{sb}P_I = [1035.980, 639.812, 270.880, -92.319, 12.397, 0.795]$$

$${}^{sb}P_{II} = [1035.998, 639.818, 558.863, -92.319, 12.399, 0.792]$$

These points are relative to the base frame of the slave robot. By equation (12), coordinates of these points can be transformed relative to the end-effector frame of the master robot. After ${}^{me}P_I$, ${}^{me}P_{II}$ is computed, the trajectory of the slave robot could be achieved by equation (12) after the trajectory for the master robot is planned. Figure 12 shows the robot trajectories for VA1400 and HP20 robots moving in a combined line synchronous motion relative to the base frame of the HP20 robot. The corresponding joint positions for each robot, master robot HP20 and slave robot VA1400, have been shown in Figure 13 and 14 respectively.

Next, combined circle synchronous motion was tested for the two robots. During this motion, the HP20 robot acts as the master and still moves in a line motion from P_1 to P_2 as above. Meanwhile, the slave robot VA1400 moves in a circle arc motion from P_1 to P_{II} by a third point P_{III} relative to the end-effector frame of the HP20 robot. Coordinates of P_{III} point in VA1400 base frame are

$${}^{sb}P_{III} = [1035.978, 460.625, 270.878, -92.319, 12.398, 0.794]$$

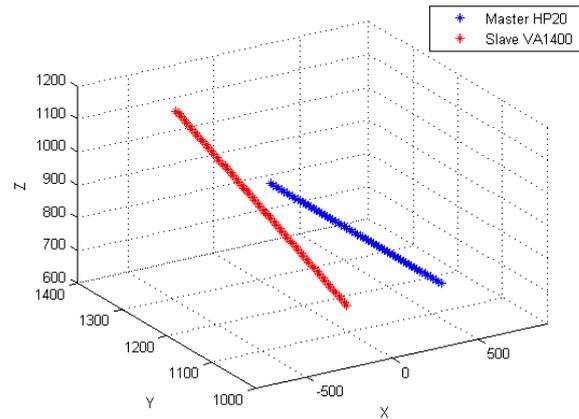


Figure 12. Trajectory of VA1400 and HP20 robots in combined line synchronous motion.

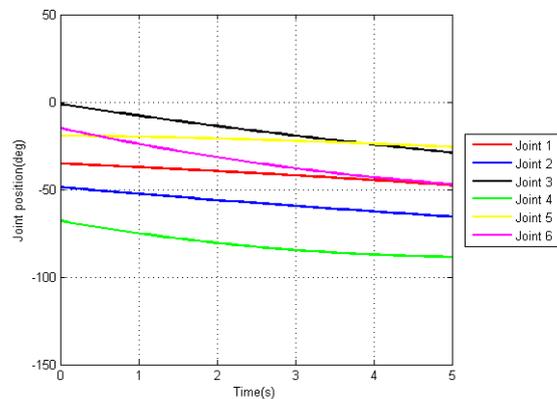


Figure 13. Joint position of master robot HP20 in combined line synchronous motion.

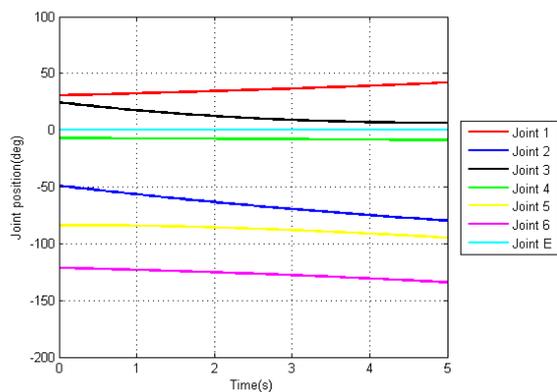


Figure 14. Joint position of slave robot VA1400 in combined line synchronous motion.

Similarly, transform the coordinates of P_{III} point from the slave base frame to the master end-effector frame and complete the trajectory planning for the slave robot. Figure 15 shows the robot trajectories for VA1400 and HP20 robots moving in a combined circle arc synchronous motion relative to the base frame of the HP20 robot. The

corresponding joint positions for each robot, master robot HP20 and slave robot VA1400, have been shown in Figure 16 and 17 respectively.

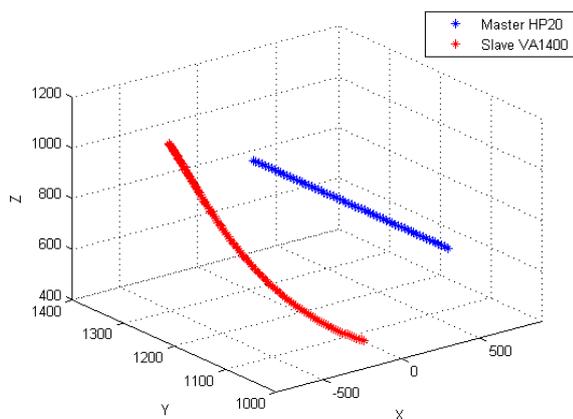


Figure 15. Trajectory of VA1400 and HP20 robots in combined circle synchronous motion.

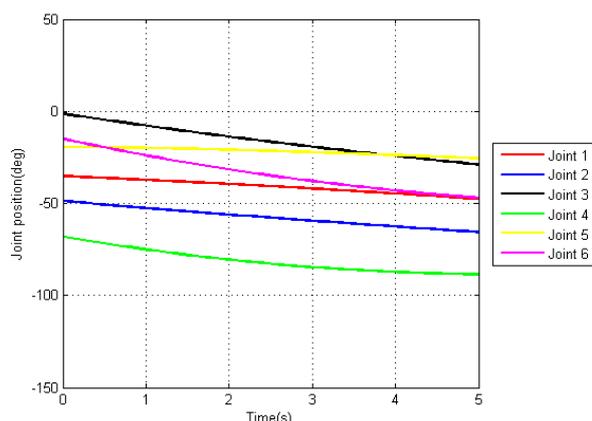


Figure 16. Joint position of master robot HP20 in combined circle synchronous motion.

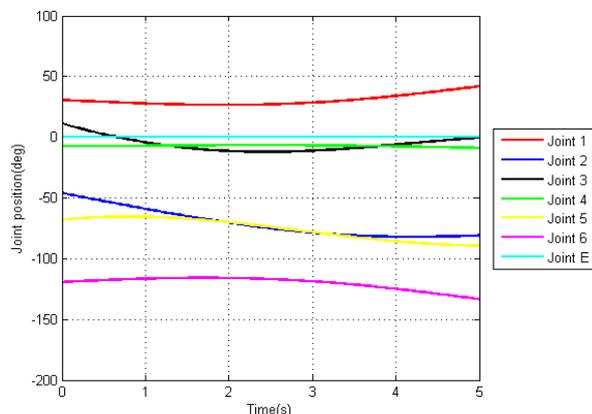


Figure 17. Joint position of slave robot VA1400 in combined circle synchronous motion.

So far we have finished all our test experiments. By the above figures, it is clear that the two robots can move in a cooperative motion whether for a line trajectory or circle trajectory, both for coupled synchronous cooperation and combined synchronous cooperation. Since the commercial industrial robots Motoman VA1400 and HP20 cannot be controlled in the velocity mode, a verification experiment for conclusions of joint velocity constraints has not been carried out here.

6. Conclusion

The aims of this paper are to establish the theoretical foundation for automated cooperative path planning and present a typical application use of this method. Focus of our cooperative path planning method is to autonomously calculate the trajectory of slave robots in the slave base frame given the master trajectory in the master base frame and task processing requirements in the workpiece frame. Motion constraints between cooperative master and slave robots are the basis for this method. Cooperative motions have been classified into three types: concurrent cooperation, coupled synchronous cooperation and combined synchronous cooperation. Constraints for each type of cooperation motion are introduced here. At the end of the paper, representative path planning experiments are presented on two industrial robots, Motoman VA1400 and HP20.

The cooperative path planning method introduced here has great application in the construction of next generation industrial robot controllers with the function of cooperative motion. It is believed that the application of multiple robot cooperative systems will herald a new era in the manufacturing industry. Whereas, traditional robot controllers cannot provide the function of cooperative path planning and cooperative motion control. It is believed that our method provides an effective solution to this problem and will greatly benefit industrial development.

7. Acknowledgements

This work was supported by the National Natural Science Foundation of China under grant no.61175113, no.61004035, National High Technology Research and Development Programme of China under grant no.2009AA043903-1-1, no.2007AA041703, Research Innovation Programme for College Graduates of Jiangsu Province under no. CX10B_076Z and Excellent Doctoral Dissertation Foundation of Southeast University.

8. References

- [1] K. Kosuge, J. Ishikawa, K. Furuta, and M. Sakai. Control of single-master multi-slave manipulator system using VIM, in: Proc. of IEEE International Conference on Robotics and Automation, New York, USA, pp. 1172–1177, 1990.
- [2] X. Yun, T. J. Tarn and A. K. Bejczy. Dynamic coordinated control of two robot manipulators, in: Proc. of IEEE International Conference on Decision and Control, New York, USA, pp. 2476–2481, 1989.

- [3] X. X. GU, C. B. FENG, and H. S. XIA. Dual-arm robotic coordination control based on Lyapunov globally exponential stability, *Robot*, vol. 13, no. 1, pp. 50–56, 1991.
- [4] J. J. Craig, M. H. Raibert. A systematic method of hybrid position/force control of a manipulator, in: Proc. of IEEE Computer Society's Third International Computer Software and Applications Conference, pp. 446–451, 1979.
- [5] M. Uchiyama, P. Dauchez. A symmetric hybrid position/force control scheme for the coordination of two robots, in: Proc. of IEEE International Conference on Robotics and Automation, New York, USA, pp. 350–356, 1988.
- [6] T. Yoshikawa and X. Zheng. Coordinated dynamic hybrid position/force control for multiple robot manipulators handling one constrained object, in: Proc. of IEEE International Conference on Robotics and Automation, New York, USA, pp. 1178–1183, 1990.
- [7] Z. W. Luo, Y. Uematsu, K. Ito, A. Kato and M. Ito. On cooperative manipulation of dynamic object, in: Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems, New York, USA, pp. 1064–1071, 1994.
- [8] M. Zribi, M. Karkoub and H. Loulin. Modelling and control of two robotic manipulators handling a constrained object, *Applied Mathematical Modelling*, vol. 24, no. 12, pp. 881–898, 2000.
- [9] A. Tavasoli, M. Eghtesad and H. Jafarian. Two-time scale control and observer design for trajectory tracking of two cooperating robot manipulators moving a flexible beam, *Robotics and Autonomous Systems*, vol. 57, no. 2, pp. 212–221, 2009.
- [10] N. H. Mcclamroch and D. Wang. Feedback stabilization and tracking of constrained robots. *Automatic Control*, IEEE Transactions on. 1988, 33 (5): 419–426.
- [11] H. Krishnan and N. H. Mcclamroch. Tracking in Nonlinear Differential-Algebraic Control Systems with Applications to Constrained Robot Systems. *Automatica*. 1994 (30): 1885–1897.
- [12] J. Y. S. Luh and Y. F. Zheng. Constrained relations between two coordinated industrial robots for motion control. *The International Journal of Robotics Research*, 1987. 3(6):60–70.
- [13] KUKA Roboter GmbH. KUKA.CR Motion Cooperation 2.1 For KUKA System Software (KSS) 5.3/5.4/5.5, Version: 00, Issued: 21 Aug 2007.
- [14] REIS GmbH & CO.KG MASCHINENFABRIK OBERNBURG. Cooperating Robots and Master/Slave Operation, Version: 15.0, Issued: 11 Jan 2005.
- [15] YASKAWA ELECTRIC CORPORATION. NX100 Options: Instructions For Independent / Coordinated Control Function. Manual NO. RE-CKI-A443, Printed in Japan, May 2005.
- [16] G. E. Cook, K. Andersen, Z. S. Saleh and K. R. Fernandez. Multiple-robot programming for coordinated motion, end-effector calibration, and part localization, *Industry Applications Society Annual Meeting 1989, Conference Record of the 1989 IEEE*, vol.2, pp. 1669–1674, 1989.
- [17] J. E. Agapakis, J. M. Katz and D. L. Pieper. Programming and control of multiple robotic devices in coordinated motion, *Proceedings of IEEE International Conference on Robotics and Automation*, New York, USA, pp. 362–367, 1990.
- [18] M. K. Jouaneh, Z. Wang and D. A. Dornfeld. Trajectory planning for coordinated motion of a robot and a positioning table. I. Path specification, *IEEE Transactions on Robotics and Automation*. 1990, 6(6):735–745.
- [19] Y. H. Gan, X. Z. Dai. Base frame calibration for coordinated industrial robots. *Robotics and Autonomous Systems*, 2011. 59(7-8): 563–570.
- [20] Z. X. CAI. *Robotics*. Beijing: Tsinghua University Press (2000).
- [21] S. S. HUANG, J. FAN. The proving and application of a new set of equations for kinematic analysis of robots. *ROBOT*, 1994. 16(5):257–263.
- [22] D. YUE, D. M. WU, Z. Z. WANG. Attitude Algorithm Based on Discrete Fourier Transform. *Computer Simulation*, 2010.27(7):21–25.

A. Transformation between Euler angle and the rotation matrix

In reference [20], successive rotations with ϕ, θ, ψ relative to axis z, y, z will lead to a rotation matrix R represented in equation (42).

$$\begin{aligned} \text{Euler}(\phi, \theta, \psi) &= \text{Rot}(z, \phi) \text{Rot}(y, \theta) \text{Rot}(z, \psi) \\ &= \begin{bmatrix} \cos \phi \cos \theta \cos \psi - \sin \phi \sin \psi & -\cos \phi \cos \theta \sin \psi - \sin \phi \cos \psi \cos \phi \sin \theta & \\ \sin \phi \cos \theta \cos \psi + \cos \phi \sin \psi & -\sin \phi \cos \theta \sin \psi + \cos \phi \cos \psi \sin \phi \sin \theta & \\ -\sin \theta \cos \psi & \sin \theta \sin \psi & \cos \theta \end{bmatrix} = \mathbf{R} \end{aligned} \quad (42)$$

Contrarily, a rotation matrix \mathbf{R} can be decomposed into three successive rotations with ϕ, θ, ψ relative to axis z, y, z ,

$$\text{Rot}(z, \phi) \text{Rot}(y, \theta) \text{Rot}(z, \psi) = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix} \quad (43)$$

$$\begin{aligned} \phi &= \text{atan2}(a_y, a_x) \\ \theta &= \text{atan2}(\sin \phi a_y + \cos \phi a_x, a_z) \\ \psi &= \text{atan2}(-\sin \phi n_x + \cos \phi n_y, -\sin \phi o_x + \cos \phi o_y) \end{aligned}$$

B. Conversion formulae from angular velocity to rotation matrix

In reference [22], conversion formulae from a given angular velocity $\omega = (\omega_x, \omega_y, \omega_z)^T \in \mathbb{R}^3$ to a rotation matrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ is provided.

First, quaternion $q = (q_0, q_1, q_2, q_3) \in \mathbb{R}^4$ has been adopted

to bridge $\omega(t)$ and $R(t)$. Conversion formula from the angular velocity $\omega(t)$ to its quaternion representation is

$$\dot{q}(t) = \frac{1}{2} \mathbf{W}(t) \cdot q(t) \quad (44)$$

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (45)$$

When quaternion representation $q(t)$ for a given angular velocity $\omega(t)$ have been achieved, the corresponding rotation matrix \mathbf{R} is,

$$\mathbf{R} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_2 - q_0q_3) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (46)$$