2-1-1990

Time Scaling of Cooperative Multi-Robot Trajectories

Seungbin B. Moon
Purdue University

Shaheen Ahmad
Purdue University

Time Scaling of Cooperative Multi-Robot Trajectories

Seungbin B. Moon
Shaheen Ahmad

TR-EE 90-15
February 1990

School of Electrical Engineering
Purdue University
West Lafayette, Indiana 47907

TIME SCALING OF COOPERATIVE MULTI-ROBOT TRAJECTORIES

Seungbin B. Moon, Shaheen Ahmad
Real-time Robot Control Laboratory
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907-0501
U.S.A.

ABSTRACT

In this paper we develop an algorithm to modify the trajectories of multiple robots in cooperative manipulation. If a given trajectory results in joint torques which exceed the admissible torque range for one or more joints, the algorithm slows down or speeds up the trajectory so as to maintain all the torques within the admissible boundary. Our trajectory modification algorithm uses the concept of time scaling developed by Hollerbach[10] for single robots. A multiple robot system in cooperative manipulation has significantly different dynamics compared to single robot dynamics. As a result, time scaling algorithm for single robots is not usable with multi-robot system. The trajectory scaling schemes described in this paper requires the use of linear programming techniques and is designed to accommodate the internal force constraints and payload distribution strategies. As the multi-robot system is usually redundantly actuated, the actuator torques may be found from the quadratic minimization which has the effect of lowering energy consumption for the trajectory. A scheme for generating a robust multi-robot trajectories when the carried load mass and inertia matrix are unknown but vary within a certain range is also described in this paper. Several examples are given to show the effectiveness of our multi-robot trajectory scaling scheme.

I. INTRODUCTION

There has been a considerable research interest in the area of multirobot systems in the recent years[1, 2, 9, 11, 13, 15, 21, 22, 23]. This paper addresses the problem of modifying the given trajectory for the multirobot system such that the joint motor torques are maintained within torque constraints. We assume that an initial trajectory which includes path, velocity, and acceleration of the end-effector is given. If the given trajectory does not satisfy the motor torque constraints, our multi-robot trajectory scaling algorithm allows us to find new trajectory velocity such that the torque constraints are not violated. Our algorithm also allows us to incorporate constraints on the internal forces and accommodates a prespecified load distribution.

The trajectory time scaling concept was initially developed by Hollerbach[10] in 1984 to modify a given trajectory for single robots. Sahar and Hollerbach[18] used this idea to find the minimum time trajectory using a joint space tessellation technique. Graettinger and Krogh[8] used time scaling concept to modify the trajectory which violates the torque constraints for a wheeled mobile robot. It is not possible to directly apply Hollerbach's[10] algorithm to the multi-robot case because of the closed kinematic chains, and force interaction between robots and the object held by the robots. Unlike single robot case, we need to consider the object dynamics because the object might be so heavy that it affects dynamics of the entire multi-robot system.

Additionally, if there are n robots each with six degree-of-freedom involved in cooperative manipulation, there are total of $k = 6n$ actuators available to actuate the load. Such a multi-robot system is redundantly actuated[15, 23]. This allows us to minimize the energy expenditure during the trajectory execution. Such issues were not addressed by other trajectory scaling algorithms.

Minimum time trajectory planning for single robots was initially studied by Kahn and Roth[14] using Pontryagin's minimum principle[4]. Work of Kahn and Roth was rather limited because they linearized the nonlinear robotic system in order to apply the minimum principle easily. Without using the minimum principle, Bobrow, Dubowsky, and Gibson [5,6] developed a more practical algorithm to generate the minimum time trajectory under the assumption that the path is fixed and parameterizable. Shin and McKay[19, 20] and Pfeiffer and Johanni[16] also developed similar approaches.

The issue of incorporating robot dynamics into motion planning for coordinated multi-robot system was identified in the report by Koivo and Bekey[13]. The research on this subject is scarce, and a few works are available for purely kinematic approaches[3]. Ahmad and Yan[2] carried out an initial work on the minimum time trajectory for multirobots for a given path employing dynamics of both the robots and the object. In this paper, a multirobot trajectory scaling scheme which incorporates dynamics is presented. Assume that we are given the desired path and velocity profile which are planned without considering the dynamics of the manipulators and the object. The velocity profile could be a trapezoidal curve or a sine curve, for example. First we check if the trajectory violates the joint torques limits. If the torque constraints are violated, we can modify the trajectory by time scaling. Although the approach presented in this paper does not allows us to find the minimum time trajectory, we can identify the increased velocity of the end-effector and the corresponding time scaling factor, if it exists for the given torque constraints. Methodology to compute the joint torques for the redundantly actuated multirobot system is also presented in this paper.

This paper is organized into six sections. In Section 2, multirobot dynamic model is presented. The time scaling of the multirobot trajectory and an algorithm to modify the given trajectory is given in Section 3. Time scaling of an *a priori* known trajectory with prespecified force distribution schemes are addressed in Section 4. The effect of payload variation on the range of time scaling constant is discussed in Section 5. Examples with planar three degree-of-freedom manipulators are given in Section 6, and the conclusion is given in Section 7.

II. The Dynamic Model of The Multi-Robot System

Figure 1 shows n manipulators grasping a common object which is to be moved along a given path from an initial point P_o to the final point P_f. The origin of the world reference frame is shown as O_w. Origin of the base coordinate of the i-th manipulator is O_i and O_c is the origin of a coordinate fixed in the carried object. The center of the mass of the object is assumed to be at O_c. The dynamic equation for each robot is known. For $i = 1,.., n$

$$\tau_i(t) = \mathbf{D}_i(\mathbf{q}_i(t))\ddot{\mathbf{q}}_i(t)+\dot{\mathbf{q}}_i(t)\mathbf{C}_i(\mathbf{q}_i(t))\dot{\mathbf{q}}_i(t)+\mathbf{G}_i(\mathbf{q}_i(t))+\mathbf{J}_i^T(\mathbf{q}_i(t))\mathbf{F}_i \tag{1}$$

where, $\mathbf{q}_i(t) \in \boldsymbol{R}^{n_i}$ is the vector of joint position, and n_i is the number of degree-of-freedom of the i-th robot, $\mathbf{D}_i \in \boldsymbol{R}^{n_i \times n_i}$ is the manipulator inertia matrix and $\mathbf{C}_i \in \boldsymbol{R}^{n_i \times n_i \times n_i}$ is the tensor of Coriolis and centrifugal terms. The vector of gravitational terms is $\mathbf{G}_i \in \boldsymbol{R}^{n_i}$ and $\mathbf{J}_i \in \boldsymbol{R}^{6 \times n_i}$ is the manipulator Jacobian matrix. The vector of joint torques is $\tau_i \in \boldsymbol{R}^{n_i}$. The vector $\mathbf{F}_i = [\ \mathbf{f}_i^T, \mathbf{v}_i^T\ ]^T \in \boldsymbol{R}^6$ is expressed in the base coordinate of the i-th robot, where linear force $\mathbf{f}_i \in \boldsymbol{R}^3$, the moment $\mathbf{v}_i \in \boldsymbol{R}^3$ are exerted by the i-th manipulator onto the carried object. Note that $\mathbf{D}_i$, $\mathbf{C}_i$, $\mathbf{G}_i$, and $\mathbf{J}_i$ are dependent only on the joint position, while $\mathbf{F}_i$ is a function of joint position, velocity, and acceleration. The position of the object with respect to the world reference frame O_w is given by $\mathbf{p} \in \boldsymbol{R}^3$. The orientation of the object reference frame with respect to the world reference frame is given by the rotation matrix $\mathbf{R}_w^c \in \boldsymbol{R}^{3 \times 3}$. The position of the contact point between the object and the i-th robot from the origin of the world coordinate frame is given by $\mathbf{p}_i \in \boldsymbol{R}^3$, it is measured with respect to the world coordinate frame. The contact point from the origin of the object coordinate frame is denoted as $\mathbf{r}_i \in \boldsymbol{R}^3$ with respect to object coordinate frame. Thus we have

$$\mathbf{p}_i = \mathbf{R}_c^w \mathbf{r}_i + \mathbf{p} \tag{2}$$

The manipulated object dynamics are given by the below equations.

$$m\ddot{\mathbf{p}}(t) + m\mathbf{g} = \sum_{i=1}^{n} \mathbf{R}_i^w \mathbf{f}_i \tag{3}$$

$$\mathbf{I}\dot{\omega}(t) + \omega(t) \times \mathbf{I}\omega(t) = \sum_{i=1}^{n} \mathbf{r}_i \times \mathbf{R}_i^w \mathbf{f}_i + \sum_{i=1}^{n} \mathbf{R}_i^w \mathbf{v}_i \tag{4}$$

where, $m \in \boldsymbol{R}^+$ is the mass of the object and $\mathbf{I} \in \boldsymbol{R}^{3 \times 3}$ is the object inertia matrix. Linear acceleration, $\ddot{\mathbf{p}} = [\ \ddot{p}_x, \ddot{p}_y, \ddot{p}_z\ ]^T$, and angular acceleration, $\dot{\omega} = [\ \dot{\omega}_x, \dot{\omega}_y, \dot{\omega}_z\ ]^T$, of the object are expressed in the world coordinate frame. Since $\mathbf{f}_i$ and $\mathbf{v}_i$ are expressed in the i-th base coordinate frame, we need to premultiply by the rotation matrix, $\mathbf{R}_i^w$, to convert these into the world reference frame. Gravitational vector $\mathbf{g} \in \boldsymbol{R}^3$ is expressed in the world coordinate reference frame, and acts along the *positive* z-axis, thus $\mathbf{g} = [\ 0, 0, 9.8\ ]^T$, if z-axis of the world coordinate frame is upward with respect to the earth. Notice that we can ignore the first term in the right hand side of eq.(4), if we assume rigid contact between the object and the robots.

Combining eq.(3) and eq.(4), we obtain

$$\begin{bmatrix} m\ddot{\mathbf{p}}(t) + m\mathbf{g} \\ \mathbf{I}\dot{\omega}(t) + \omega(t)\mathrm{x}\mathbf{I}\omega(t) \end{bmatrix} = \sum_{i=1}^{n} \mathbf{B}_i(\mathbf{q}_i(t))\, \mathbf{F}_i \tag{5}$$

where, $\mathbf{B}_i \in \boldsymbol{R}^{6\mathrm{X}6}$ is defined as follows.

$$\mathbf{B}_i(\mathbf{q}_i(t)) = \begin{bmatrix} \mathbf{R}_i^w & \mathbf{0} \\ \mathbf{S}_i\,\mathbf{R}_i^w & \mathbf{R}_i^w \end{bmatrix} \tag{6}$$

where, $\mathbf{S}_i \in \boldsymbol{R}^{3\mathrm{X}3}$ is the skew symmetric matrix which is used to represent the vector product in the first term on the right hand side of eq.(4).

$$\mathbf{S}_i(\mathbf{r}_i) = \begin{bmatrix} 0 & -r_{iz} & r_{iy} \\ r_{iz} & 0 & -r_{ix} \\ -r_{iy} & r_{ix} & 0 \end{bmatrix} \tag{7}$$

where, $\mathbf{r}_i = [\, r_{ix}, r_{iy}, r_{iz}\,]^T$ is a vector from the center of mass of the object to the contact point with the i-th manipulator. Clearly, $\mathbf{B}_i$ is dependent only on the joint position.

III. TIME SCALING OF THE MULTI-ROBOT TRAJECTORY

Assume that the preplanned joint trajectories, $\mathbf{q}_i(t)$, for $i = 1,.., n$, and object trajectories $\mathbf{p}(t)$, $\phi(t)$ for $t \in [\,0, t_f\,]$, are given. If it is necessary to modify the trajectories in order to satisfy the torque constraints, the velocity of the joint trajectory can be altered while keeping the end-effector path as previously planned. We assume that the joint torques are constrained by constants along the path, i.e., $\tau_i^- \leq \tau_i(t) \leq \tau_i^+$, $i = 1,..,n$ for all t , then

$$\tau^- \leq \tau(t) \leq \tau^+ \tag{8}$$

where, $\tau(t) = [\ \tau_1(t)^T\ ....\ \tau_n(t)^T\ ]^T$, $\tau^- = [\ (\tau_1^-)^T\ ....\ (\tau_n^-)^T\ ]^T$, and
$\tau^+ = [\ (\tau_1^+)^T\ ....\ (\tau_n^+)^T\ ]^T$

Let the new joint trajectory $\tilde{\mathbf{q}}_i(t)$ be defined as

$$\tilde{\mathbf{q}}_i(t) = \mathbf{q}_i(r(t)) \qquad i = 1,..,n \quad \text{for all } t \tag{9}$$

where, $r(t)$ is a monotonically increasing function of time t with

$$r(0) = 0 \text{ and } r(t_1) = t_f \qquad \text{for some } t_1 > 0. \tag{10}$$

Thus, the trajectory traversal time is now changed from t_f to t_1. If $t_1 > t_f$, trajectory traversal time is increased, otherwise traversal time is shortened. Differentiating $\tilde{\mathbf{q}}_i(t)$ in eq.(9) with respect to time, we obtain

$$\dot{\tilde{\mathbf{q}}}_i(t) = \frac{d\tilde{\mathbf{q}}_i(t)}{dt} = \frac{d\mathbf{q}_i(r)}{dr}\frac{dr}{dt} = \mathbf{q}_i'(r)\ \dot{r}(t) \tag{11}$$

$$\ddot{\tilde{\mathbf{q}}}_i(t) = \mathbf{q}_i''(r)\,\dot{r}^2(t) + \mathbf{q}_i'(r)\ \ddot{r}(t) \tag{12}$$

where $'$, $''$ represents d/dr, d^2/dr^2 and $\dot{}$, $\ddot{}$ represents d/dt, d^2/dt^2.

The equation of motion of the i-th manipulator along the new trajectory is

$$\begin{aligned}\tilde{\tau}_i(t) &= \mathbf{D}_i(\tilde{\mathbf{q}}_i)\ddot{\tilde{\mathbf{q}}}_i(t)+\dot{\tilde{\mathbf{q}}}_i(t)\mathbf{C}_i(\tilde{\mathbf{q}}_i)\dot{\tilde{\mathbf{q}}}_i(t) + \mathbf{G}_i(\tilde{\mathbf{q}}_i) + \mathbf{J}_i(\tilde{\mathbf{q}}_i)^T\,\mathbf{F}_i\,(\tilde{\mathbf{q}}_i,\dot{\tilde{\mathbf{q}}}_i,\ddot{\tilde{\mathbf{q}}}_i)\\ &=\{\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i''(r)+\mathbf{q}_i'(r)\mathbf{C}_i(\mathbf{q}_i)\mathbf{q}_i'(r)\}\dot{r}^2+\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i'(r)\ddot{r}+\mathbf{G}_i(\mathbf{q}_i)+\mathbf{J}_i(\mathbf{q}_i)^T\,\tilde{\mathbf{F}}_i\end{aligned} \tag{13}$$

where, $\tilde{\tau}_i$ is the new input joint torques, and $\tilde{\mathbf{F}}_i = \mathbf{F}_i\,(\tilde{\mathbf{q}}_i,\dot{\tilde{\mathbf{q}}}_i,\ddot{\tilde{\mathbf{q}}}_i)$ is the new force/moment applied onto the object. From now on, we will omit the dependence on a time t or on the function $r(t)$ in the arguments of terms, $\mathbf{D}_i$, $\mathbf{C}_i$, $\mathbf{G}_i$, $\mathbf{J}_i$, and $\mathbf{F}_i$ as long as it is clear from the context. From the above equation, we can derive $\tilde{\mathbf{F}}_i$. For simplicity, we will assume that $\mathbf{J}_i$ is a 6X6 square matrix and it is nonsingular for all $i = 1, .., n$ along the preplanned path. Then

$$\tilde{\mathbf{F}}_i = \mathbf{J}_i(\mathbf{q}_i)^{-T}[\tilde{\tau}_i(t) - \{\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i''(r)+\mathbf{q}_i'(r)\mathbf{C}_i(\mathbf{q}_i)\mathbf{q}_i'(r)\}\dot{r}^2-\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i'(r)\ddot{r} - \mathbf{G}_i(\mathbf{q}_i)] \tag{14}$$

The new trajectory of the object is given by $\tilde{\mathbf{p}}$ and $\tilde{\phi}$, where $\tilde{\mathbf{p}}(t) = \mathbf{p}(r)$, and $\tilde{\phi}(t) = \phi(r)$ for all t. We can express the time derivatives of the new trajectory of the object as follows.

$$\dot{\tilde{\mathbf{p}}}(t) = \frac{d\tilde{\mathbf{p}}(t)}{dt} = \frac{d\mathbf{p}(r)}{dr}\frac{dr}{dt} = \mathbf{p}'(r)\ \dot{r}(t) \tag{15}$$

$$\ddot{\tilde{\mathbf{p}}}(t) = \mathbf{p}''(r)\,\dot{r}^2(t) + \mathbf{p}'(r)\ \ddot{r}(t) \tag{16}$$

$$\tilde{\omega}(t) = \frac{d\tilde{\phi}(t)}{dt} = \frac{d\phi(r)}{dr}\frac{dr}{dt} = \phi'(r)\ \dot{r}(t) \tag{17}$$

$$\dot{\tilde{\omega}}(t) = \phi''(r)\,\dot{r}^2(t) + \phi'(r)\ \ddot{r}(t) \tag{18}$$

If we let,

$$r(t) = ct \qquad\qquad \text{for all } t \tag{19}$$

where, $c \in \boldsymbol{R}^+$ is a positive constant. Thus, $r(t_1) = ct_1 = t_f$. In such a case trajectory velocity is scaled by a constant throughout the entire trajectory. Here c is the trajectory time scaling constant. Notice if $c > 1$, we are speeding up and if $c < 1$, we are slowing down the trajectories.

The time scaled object dynamics is now given by

$$\begin{aligned}\sum_{i=1}^{n} \mathbf{B}_i(\mathbf{q}_i(r))\,\tilde{\mathbf{F}}_i &= \begin{bmatrix} m\ddot{\tilde{\mathbf{p}}}(t)+m\mathbf{g} \\ \mathbf{I}\dot{\tilde{\omega}}(t)+\tilde{\omega}(t)\mathrm{x}\mathbf{I}\tilde{\omega}(t) \end{bmatrix} \\ &= c^2\begin{bmatrix} m\mathbf{p}(r)'' \\ \mathbf{I}\phi(r)''+\phi(r)'\mathrm{x}\mathbf{I}\phi(r)' \end{bmatrix} + \begin{bmatrix} m\mathbf{g} \\ \mathbf{0} \end{bmatrix}\end{aligned} \tag{20}$$

We substitute the eq.(14) into eq.(20). Then we have following equation.

$$\begin{aligned}&\sum_{i=1}^{n} \mathbf{B}_i(\mathbf{q}_i(r))\,\tilde{\mathbf{F}}_i \\ &= \sum_{i=1}^{n} \mathbf{B}_i(\mathbf{q}_i(r))\mathbf{J}_i(\mathbf{q}_i)^{-T}[\tilde{\tau}_i(t)-\{\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i''(r)+\mathbf{q}_i'(r)\mathbf{C}_i(\mathbf{q}_i)\mathbf{q}_i'(r)\}c^2-\mathbf{G}_i(\mathbf{q}_i)]\end{aligned} \tag{21}$$

Let us define following quantities:

$$\mathbf{L}(r) = [\ \mathbf{B}_1(\mathbf{q}_1(r))\mathbf{J}_1(\mathbf{q}_1(r))^{-T} \ \ \ldots \ \ \mathbf{B}_n(\mathbf{q}_n(r))\mathbf{J}_n(\mathbf{q}_n(r))^{-T}\ ] \in \boldsymbol{R}^{6\mathrm{x}6n},$$

$$\tilde{\tau}(t) = [\ \tilde{\tau}_1(t)^T \ \cdots\ \tilde{\tau}_n(t)^T\ ]^T \in \boldsymbol{R}^{6n},$$

$$\mathbf{G}(r) = [\ \mathbf{G}_1(\mathbf{q}_1(r))^T \ \cdots\ \mathbf{G}_n(\mathbf{q}_n(r))^T\ ]^T \in \boldsymbol{R}^{6n},$$

$$\mathbf{P}(r) = \begin{bmatrix} m\mathbf{p}(r)'' \\ \mathbf{I}\phi(r)'' + \phi(r)' \mathrm{x} \mathbf{I}\phi(r)' \end{bmatrix} \in \boldsymbol{R}^6,$$

$$\mathbf{Q} = \begin{bmatrix} m\mathbf{g} \\ \mathbf{0} \end{bmatrix} \in \boldsymbol{R}^6,$$

$$\mathbf{E}(r) = \begin{bmatrix} \mathbf{D}_1(\mathbf{q}_1(r))\mathbf{q}_1''(r) + \mathbf{q}_1'(r)\mathbf{C}_1(\mathbf{q}_1(r))\mathbf{q}_1'(r) \\ \cdots \\ \mathbf{D}_n(\mathbf{q}_n(r))\mathbf{q}_n''(r) + \mathbf{q}_n'(r)\mathbf{C}_n(\mathbf{q}_n(r))\mathbf{q}_n'(r) \end{bmatrix} \in \boldsymbol{R}^{6n} \ .$$

Then, combining eq.(20) and eq.(21), and arranging terms according to c^2, we have

$$\begin{aligned} \mathbf{L}(r)\,\tilde{\tau}(t) - \mathbf{L}(r)\,\mathbf{G}(r) - \mathbf{Q} &= c^2\ \{\ \mathbf{P}(r) + \mathbf{L}(r)\,\mathbf{E}(r)\ \} \\ &= c^2\ \{\ \mathbf{P}(ct) + \mathbf{L}(ct)\,\mathbf{E}(ct)\ \} \end{aligned} \tag{22}$$

If we change the argument of equation (22) from t to t/c, then we have

$$\mathbf{L}(t)\,\tilde{\tau}(t/c) - \mathbf{L}(t)\,\mathbf{G}(t) - \mathbf{Q} = c^2\ \{\ \mathbf{P}(t) + \mathbf{L}(t)\,\mathbf{E}(t)\ \} \tag{23}$$

We can obtain the permissible range of c^2 from eq.(23) if the torques limits are given by eq.(8). In order to generate practically usable trajectories we need to limit the magnitude of the internal forces. This is because the internal forces do not contribute to the motion of the object. Internal forces are the ones used to hold the object. Obviously, if the internal forces are too large then the object could be squeezed or stretched. In order to prevent these undesirable effects, we introduce following constraints.

$$\mathbf{F}_i^{\min} \le \mathbf{F}_i \le \mathbf{F}_i^{\max} \qquad \text{for } i = 1,\ldots, n \tag{24}$$

where, $\mathbf{F}_i^{min}$ and $\mathbf{F}_i^{max}$ are given in advance. This inequalities may be transformed by substituting $\mathbf{F}_i$ from eq. (14).

$$\mathbf{F}_i^{min} + \mathbf{J}_i^{-T}\mathbf{G}_i \leq \mathbf{J}_i^{-T}[\,\tilde{\tau}_i(t/c) - \{\mathbf{D}_i\,\ddot{\mathbf{q}}_i + \dot{\mathbf{q}}_i\mathbf{C}_i\dot{\mathbf{q}}_i\}c^2\,] \leq \mathbf{F}_i^{max} + \mathbf{J}_i^{-T}\mathbf{G}_i \tag{25}$$

3.1. Linear programming method to find the range of the scaling constant

First, let $\mathbf{x} = [\,\tilde{\tau}^T(t/c)\,,\, c^2(t)\,]^T \in \boldsymbol{R}^{[6n+1]}$. Then we can formulate a linear programming problem[17] to find $c^2(t)$ as follows.

Find $\mathbf{x}$ which minimizes (or maximizes)

$$[\,0, 0, ..., 0, 1\,]^T\,\mathbf{x} = c^2(t) \tag{26}$$

subject to

$$\mathbf{A}_1\,\mathbf{x} = \mathbf{b}_1 \tag{27}$$

$$\mathbf{b}_2^- \leq \mathbf{A}_2\,\mathbf{x} \leq \mathbf{b}_2^+ \tag{28}$$

$$\mathbf{x}^- \leq \mathbf{x} \leq \mathbf{x}^+ \tag{29}$$

where,

$$\mathbf{A}_1 = [\;\mathbf{L}(t) \quad -\mathbf{P}(t)-\mathbf{L}(t)\mathbf{E}(t)\;] \in \boldsymbol{R}^{6\mathrm{x}[6n+1]}$$

$$\mathbf{b}_1 = \mathbf{L}(t)\,\mathbf{G}(t) + \mathbf{Q}$$

$$\mathbf{A}_2 = \left[\;\mathbf{J}_D \quad -\mathbf{J}_D\mathbf{E}(t)\;\right] \in \boldsymbol{R}^{6n\mathrm{x}[6n+1]}$$

$$\mathbf{J}_D = \begin{bmatrix} \mathbf{J}_1^{-T} & 0 & 0 & .. & 0 \\ 0 & \mathbf{J}_2^{-T} & 0 & .. & 0 \\ & & \cdots & & \\ 0 & 0 & 0 & .. & \mathbf{J}_n^{-T} \end{bmatrix} \in \boldsymbol{R}^{6n\mathrm{x}6n}$$

$$\mathbf{b}_2^+ = \begin{bmatrix} \mathbf{F}_1^{max} + \mathbf{J}_1^{-T}\mathbf{G}_1 \\ \cdots \\ \mathbf{F}_n^{max} + \mathbf{J}_n^{-T}\mathbf{G}_n \end{bmatrix} , \quad \mathbf{b}_2^- = \begin{bmatrix} \mathbf{F}_1^{min} + \mathbf{J}_1^{-T}\mathbf{G}_1 \\ \cdots \\ \mathbf{F}_n^{min} + \mathbf{J}_n^{-T}\mathbf{G}_n \end{bmatrix}$$

$$\mathbf{x}^- = [\;(\tau^-)^T\,,\,0\,]^T \quad , \quad \mathbf{x}^+ = [\,(\tau^+)^T\,,\,\infty\,]^T$$

If the solution to the above minimization(or maximization) problem exists, then the last element of the minimizing(or maximizing) vector of $\mathbf{x}$ is $c^{2-}(t)$ (or $c^{2+}(t)$). The dependence on the time argument t in $c^{2-}(t)$ (or $c^{2+}(t)$) needs to be emphasized since the coefficients matrices $\mathbf{A}_1$ and $\mathbf{b}_1$ are functions of time. In the Appendix A, we prove that once we have $c^{2-}(t)$ and $c^{2+}(t)$ it is guaranteed that any value of $c^2(t) \in [c^{2-}(t), c^{2+}(t)]$ satisfies the constraints given by eq. (27) to (29). This implies that $[c^{2-}(t), c^{2+}(t)]$ is the only interval which satisfies the torque and internal force constraints. If a feasible solution to the above problem does not exist, then we can conclude that the multi-robot trajectory is unrealizable. In such cases, we can change the path or try other velocity profiles. Finding the intersection of the intervals, $[c^{2-}(t), c^{2+}(t)]$, over the duration of the movement, $t \in [0, t_f]$, gives us the globally admissible range of time scaling constants, $[c^{2-}, c^{2+}]$.

$$[c^{2-}, c^{2+}] = \bigcap_{\Omega} [c^{2-}(t), c^{2+}(t)] \quad , \quad \Omega = \{ t \mid t \in [0, t_f] \} \tag{30}$$

After selecting an appropriate value of c^2 to modify the given trajectory, we can determine the joint torques for trajectory execution. The originally given trajectories can be employed by setting $c^2 = 1$, if $1 \in [c^{2-}, c^{2+}]$. In the Appendix B, we show that the shortest traversal time while satisfying joint torque and internal force constraints can be achieved by selecting $c = \sqrt{c^{2+}}$.

3.2. Dealing with the redundant actuation.

We indicated earlier the multirobot system is redundantly actuated, allowing us infinite possibilities in selecting the joint torques. This fact is reflected in that the system of equation (23) is underspecified. This allows us to determine the joint torques which may satisfy the minimum energy criterion. Consider the following minimization problem.

Find $\tilde{\tau}(t/c)$ which minimizes the quadratic cost function $\Phi(\tilde{\tau})$

$$\Phi(\tilde{\tau}) = \tilde{\tau}(t/c)^T W \tilde{\tau}(t/c) \tag{31}$$

subject to

$$\mathbf{L}(t)\, \tilde{\tau}(t/c) = \mathbf{L}(t)\, \mathbf{G}(t) + \mathbf{Q} + c^2 \{\mathbf{P}(t) + \mathbf{L}(t)\, \mathbf{E}(t)\} \tag{32}$$

$$\mathbf{b}_2^- + c^2\, \mathbf{J}_D \mathbf{E}(t) \le \mathbf{J}_D\, \tilde{\tau}(t/c) \le \mathbf{b}_2^+ + c^2\, \mathbf{J}_D \mathbf{E}(t) \tag{33}$$

$$\tau^- \le \tilde{\tau}(t/c) \le \tau^+ \tag{34}$$

where, the positive definite matrix $W \in R^{6n \times 6n}$ is a weight matrix. This matrix is selected by the trajectory planner according to the capacity of each manipulator. The matrix W can be selected to be a diagonal matrix for simplicity. The torque selection can be effected by the choice of the diagonal elements of the $\boldsymbol{W}$ matrix. Intuitively, we can prevent the torque saturation of the least powerful manipulator by setting the corresponding diagonal element of the $\boldsymbol{W}$ matrix larger than others. Similarly, the most powerful manipulator can be utilized by setting the corresponding diagonal element of $\boldsymbol{W}$ matrix smaller than others. As this is a quadratic programming problem with linear constraints, we can readily use available optimization packages to find torque $\tilde{\tau}(t/c)$[17].

Remark

Notice that the total energy expended by the multi-robot system over the duration of the movement, $t \in [\,0,\ t_1\,]$, is given by $\mathcal{E}(\,0, t_1\,)$:

$$\mathcal{E}(0, t_1) = \sum_{i=1}^{n} \int_{0}^{t_1} \dot{\tilde{q}}_i^{\,T}(t)\ \tilde{\tau}_i(t)\ dt$$

Therefore, the pointwise minimization $\Phi\,(\tilde{\tau})$ along the trajectory does not necessarily ensure global minimization of energy $\mathcal{E}(\,0, t_1\,)$ for the given trajectory. However, minimization of $\Phi\,(\tilde{\tau})$ does have the effect of lowering expenditure as this minimizes the magnitude of $\tilde{\tau}$.

In the Appendix C, we showed that a solution to the above quadratic programming problem satisfies the equations (8), (13), (20), and (24) which include the dynamics of the object and the robots, and constraints on the joint torques and internal forces. Therefore the algorithm in Section 3.2 can be viewed as the methodology to distribute forces $\mathbf{F}_i$ exerted by i-th robot onto the object. We now state an algorithm ***MMTSF****(Multiple Manipulators Time-Scaling for Fixed load)* to scale multirobot trajectories given the internal forces are constrained.

MMTSF Algorithm

Step 1. Given the initial robot trajectories, find the range of $c^2(t)$, $[\ c^{2-}(t), c^{2+}(t)\ ]$, using the linear programming approach described in Section 3.1.

Step 2. Find the global range of c^2, [c^{2-} , c^{2+}], by taking intersection of [$c^{2-}(t)$, $c^{2+}(t)$] over the duration of the movement.

Step 3.1. If the range of c^2 is not empty, then select a value of $c^2 \in$ [c^{2-} , c^{2+}] and go to Step 4. Notice that if you select $c^2 = 1$, the initially given trajectories are unchanged.

Step 3.2. If the range of c^2 is empty, the multi-robot trajectory is not realizable. Stop and generate an error message.

Step 4. Calculate the joint torques via the method described in Section 3.2.

Step 5. Stop.

IV. TIME SCALING OF TRAJECTORIES WITH A FIXED FORCE DISTRIBUTION

In many applications it may be desirable to predetermine the force distribution scheme, such as even distribution of the payload among the n -manipulators and given time history of the internal forces. We found that a predetermined force distribution reduces the computation time needed to find the trajectory scaling constant. In order to simplify the derivations, let us assume that the orientations of the base coordinates of each robots is identical to the one of the world coordinate and all the end-effectors grasps the object rigidly. Assuming an equal load distribution with zero internal forces, we observe from eq.(5) that

$$\tilde{\mathbf{F}}_i = \frac{c^2}{n}\begin{bmatrix} m\mathbf{p}(r)'' \\ \mathbf{I}\phi(r)'' + \phi(r)' \times \mathbf{I}\phi(r)' \end{bmatrix} + \frac{1}{n}\begin{bmatrix} m\mathbf{g} \\ \mathbf{0} \end{bmatrix} \qquad \text{for all } i \tag{35}$$

Therefore the new equations of motion are given as

$$\begin{aligned} \tilde{\tau}_i(t) &= \mathbf{D}_i(\tilde{\mathbf{q}}_i)\ddot{\tilde{\mathbf{q}}}_i(t) + \dot{\tilde{\mathbf{q}}}_i(t)\mathbf{C}_i(\tilde{\mathbf{q}}_i)\dot{\tilde{\mathbf{q}}}_i(t) + \mathbf{G}_i(\tilde{\mathbf{q}}_i) + \mathbf{J}_i(\tilde{\mathbf{q}}_i)^{\mathrm{T}}\,\mathbf{F}_i\,(\tilde{\mathbf{q}}_i, \dot{\tilde{\mathbf{q}}}_i, \ddot{\tilde{\mathbf{q}}}_i) \\ &= \mathbf{a}_i\,(r)\ c^2 + \ \mathbf{b}_i\,(r) \end{aligned} \tag{36}$$

where,

$$\mathbf{a}_i\,(r) = \{\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i''(r) + \mathbf{q}_i'(r)\mathbf{C}_i(\mathbf{q}_i)\mathbf{q}_i'(r)\} + \frac{1}{n}\,\mathbf{J}_i(\tilde{\mathbf{q}}_i)^{\mathrm{T}}\ \mathbf{P}(r)$$

$$\mathbf{b}_i\,(r) = \mathbf{G}_i(\tilde{\mathbf{q}}_i) + \frac{1}{n}\,\mathbf{J}_i(\tilde{\mathbf{q}}_i)^{\mathrm{T}}\ \mathbf{Q}$$

Similarly as in Section 3, by changing argument of eq.(36) from t to t/c, we obtain

$$\tau_i(t/c) = \mathbf{a}_i\,(t)\;c^2 + \;\mathbf{b}_i\,(t) \tag{37}$$

Since the torque is limited by constant values as in eq.(8), we have

$$\tau^- - \mathbf{b}_i\,(t)\;\leq \mathbf{a}_i\,(t)\;\;c^2\;\leq \tau^+ -\;\mathbf{b}_i\,(t) \tag{38}$$

Notice that $\mathbf{a}_i\,(t)$ is a 6-dimensional vector unlike the matrix $\mathbf{L}(t)$ which was 6x6n in Section 3. The global range of c^2 can be found by intersecting all the ranges $c^2_{ij}(t)$ over the duration of the trajectory.

$$[\;c^{2-}\;,c^{2+}\;] = \bigcap_K [c^{2-}_{ij}(t)\;,c^{2+}_{ij}(t)]\;, \tag{39}$$

where, K= $\{t\,,i,j \mid t \in [\,0,\,t_f\,],\, i,=1,..,n, \text{ and } j=1,..,6\,\}$, and $c^{2+}_{ij}(t)$ is the maximum value of c^2 obtained from the j-th component of the i -th robot in eq.(38), and similarly for $c^{2-}_{ij}(t)$.

The algorithm to find c^2 is similar to that of the single robot case, and the detailed procedure to find $c^{2-}_{ij}(t)$ and $c^{2+}_{ij}(t)$ is given in Hollerbach's work[10]. Because of the simplicity of this algorithm, the computation time to find c^2 is considerably lower. Notice also that the range of c^2 obtained from this algorithm is conservative because of the additional restriction on the load distribution.

V. Time Scaling of Trajectories With Load Variation

Quite often in manufacturing applications, robots are required to handle different loads in a manufacturing cycle. In this section, we investigate the problem of robustness with respect to the load variation. We design a method which will guarantee that the trajectory stays inside the torque boundary within the predefined range of load variation. In order to develop a robust trajectory scaling algorithm we will assume that the inertia matrix of the manipulated object is diagonal. This is not a restrictive assumption since the inertia matrix of an object is diagonal about its principal axes[7]. Therefore our assumption requires us to locate the object coordinate frame to coincide with the principal axes of the object. Let $\bar{m}$ and $\bar{\mathbf{I}}$ be the mass and the inertia matrix of the uncertain object, then:

$$\bar{m} = k_m m \tag{40}$$

$$\bar{\mathbf{I}} = \boldsymbol{K}\,\mathbf{I} = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix} \begin{bmatrix} \mathrm{I}_{xx} & 0 & 0 \\ 0 & \mathrm{I}_{yy} & 0 \\ 0 & 0 & \mathrm{I}_{zz} \end{bmatrix} \tag{41}$$

where, m and I_{ij} are the nominal mass and moments of inertia of the load. Assume the ranges of load parameter variation are known in advance as $0 \leq k_j^- \leq k_j \leq k_j^+$, for $j = m$, x, y, and z. Then the corresponding equation (23) for the uncertain object becomes

$$\mathbf{L}(t)\,\tilde{\tau}(t/c) - \mathbf{L}(t)\,\mathbf{G}(t) - \bar{\mathbf{Q}} = c^2\ \bar{\mathbf{P}}(t) + c^2\,\mathbf{L}(t)\,\mathbf{E}(t) \tag{42}$$

where

$$\bar{\mathbf{Q}} = k_m\,\mathbf{Q}\ \text{, and}\ \bar{\mathbf{P}}(t) = \begin{bmatrix} k_m m\ddot{\mathbf{p}}(t) \\ \boldsymbol{K}\mathbf{I}\dot{\omega}(t) + \omega(t)\mathrm{x}\boldsymbol{K}\mathbf{I}\omega(t) \end{bmatrix}$$

We can now find the permissible range of $c^2(t)$ from the solution of a nonlinear programming problem formulated in the below. If we let $\mathbf{y} = [\ \tilde{\tau}^T,\ k_m, k_x, k_y, k_z, c^2(t)\ ]^T$, we can find $\mathbf{y}$ which minimizes (or maximizes) $c^2(t) = [\,0, 0, ..., 0, 1\,]^T\,\mathbf{y}$

subject to

$$\mathbf{L}(t)\,\tilde{\tau}(t/c) = \mathbf{L}(t)\,\mathbf{G}(t) + k_m\ \mathbf{Q} + c^2(t)\,\bar{\mathbf{P}}(t) + c^2(t)\,\mathbf{L}(t)\,\mathbf{E}(t) \tag{43}$$

$$\tau^- \leq \tilde{\tau}(t/c) \leq \tau^+ \tag{44}$$

$$0 \leq k_j^- \leq k_j \leq k_j^+ \qquad \text{for } j = m, x, y, \text{ and, } z \tag{45}$$

$$0 \leq c^2(t) \tag{46}$$

Notice that constraints on the internal forces are not incorporated into this formulation. If it is desired to incorporate the internal force constraints, it can be done so using eq.(24) or (25). Since the above is a nonlinear programming problem, it may be difficult to find a solution. If we assume that the object is constrained to linear motion without any rotation, then we can ignore the variation on the inertia matrix because the angular velocity is zero. In this case, we can find the global range of c^2 by exhaustively searching along the range of k_m using linear programming techniques. The joint torques can be found similarly as described in Section 3.

Find $\tilde{\tau}(t/c)$ which minimizes the quadratic cost function $\Phi\,(\tilde{\tau})$

$$\Phi\,(\tilde{\tau}) = \tilde{\tau}(t/c)^{\mathrm{T}}\, W\ \tilde{\tau}(t/c) \tag{47}$$

subject to

$$\mathbf{L}(t)\,\tilde{\tau}(t/c) = \mathbf{L}(t)\,\mathbf{G}(t) + k_{\mathrm{m}}\mathbf{Q} + c^2\ \{\ \bar{\mathbf{P}}\,(t)\ +\ \mathbf{L}(t)\,\mathbf{E}(t)\} \tag{48}$$

$$\tau^{-}\ \leq\ \tilde{\tau}(t/c) \leq\ \tau^{+} \tag{49}$$

If it is desired to generate the reference torque profiles we need to know exact load parameters as in eq.(48). The significance of the above analysis is that, if the load changes within the predefined range, trajectory replanning is not necessary as the torques will remain within their limits. In addition, the trajectory execution time will also remain constant for the specified load variation, which allows for robust scheduling in a manufacturing workcell.

VI. Examples With Planar Robots

Consider two planar robots each with three degrees of freedom operating in the vertical plane manipulating an object as shown in Fig. 2. We assume that the end-effectors grasp the object rigidly, so there is no relative movement between the end-effectors and the object. The dynamic equations of motion can be found in [14]. If l_{ij}, m_{ij} are the length and mass of the j-th link of the i-th robot, then robot link lengths and masses are as follows; $l_{11} = l_{12} = l_{21} = l_{22} = 1\ m$, $l_{13} = l_{23} = 0.1\ m$, and $m_{11} = 5\ kg, m_{12} = 4\ kg, m_{13} = 0.5\ kg, m_{21} = 5\ kg,\ m_{22} = 4\ kg$, and $m_{23} = 0.5\ kg$. We will also assume each link is a cylinder with radius $r_{\mathrm{ij}} = 0.1\ m$, for all i, j. The inertia seen at joint j of the i-th robot is given by [7]

$$\mathrm{I}_{\mathrm{ij}} = \frac{1}{12}\, m_{\mathrm{ij}}\,(\,3\, r_{\mathrm{ij}}^{\,2} + l_{\mathrm{ij}}^{\,2}\,)$$

A trajectory for the object is selected such that it is moved along in the y_1-axis without any rotation. The world coordinate reference frame is attached to the base of manipulator 1 whose origin is at O_1. The position of O_2 is at (2.5, 0) with respect to O_1. The initial position of the object center of mass O_c is at (1.25, 1.0) . The imposed torque limits are $\tau_{11}^{\,+} = 100\ Nm,\ \tau_{12}^{\,+} = 80\ Nm,\ \tau_{13}^{\,+} = 50\ Nm,\ \tau_{21}^{\,+} = 100\ Nm,\ \tau_{22}^{\,+} = 80\ Nm, \tau_{23}^{\,+} =$

50 Nm, and the lower torque limits are given as $\tau_{ij}^{-} = -\tau_{ij}^{+}$ for all i and j. The desired end-effector velocity profile is a sinusoid.

As expected, we found the trajectory scaling to be sensitive to the configurations of the arms. In this example, there are four possible configurations according to two possible inverse kinematic solutions of the planar 3 degree-of-freedom manipulator with rigid contact, see Fig. 3. Notice in practice not all configurations may be physically usable as collision among robots and obstacles may occur during motion. Collision avoidance is not addressed in this paper, therefore we will assume all configurations are usable. A numerical example which demonstrates the effect of different configuration on the range of time scaling constants is given in Section 6.3.

6.1. Time scaling with 3 different sets of constraints

In this example, the given trajectory does not violate the torque constraints. However, we may want to speed up the velocity to shorten the traversal time. The mass of the manipulated object is $m = 0.5\ kg$. The trajectory velocity of the object is given as, $\dot{p}_x = 0$, $\dot{p}_y = 1.5 \sin 6t$, and $w_z = 0$ with $t_o = 0$ and $t_f = 0.523\ sec$. Thus the initial position of the object is (1.25, 1.0), and the final position is (1.25, 1.5) without any rotation.

(a) The range of c^2 was found to be , $[\, c^{2-}, c^{2+}] = [\, 0.0 , 3.988\,]$, if we do not impose any constraints on the internal forces. If we set $c^2 = 3.988$, the new traversal time is shortened to $t_1 = 0.523/\sqrt{3.988} = 0.262\ sec$, and torque profiles are as shown in Fig. 4(a). However, as we see in Fig. 4(b), the forces along the x-axis are very large and it could break the object.

(b) In order to prevent large internal forces, we imposed the following constraints on the internal forces, $-2\ N \leq f_{ix} \leq 2\ N$, and $-1\ Nm \leq v_{iz} \leq 1\ Nm$. Then the range of c^2 was now found to be reduced to $[\, c^{2-}, c^{2+}] = [\, 0.0 , 2.949\,]$. The corresponding torques and forces of the new trajectory in which we set $c^2 = 2.949$ are given in Fig. 5(a) and 5(b). The new traversal time was now shortened to $t_1 = 0.523/\sqrt{2.949} = 0.304\ sec$. Although the traversal time is now increased by 42 msec over the case with unconstrained internal forces, the internal forces exerted onto the object are now acceptable.

(c) Next, we found the range of c^2 with an even payload distribution and zero internal forces as suggested in Section 4. As we expected, the range [c^{2-} , c^{2+}] was smaller, [0.0 , 2.779]. The traversal time for this trajectory is $t_1 = 0.523/\sqrt{2.779} = 0.314$ *sec* and the joint torque profiles are as given in Fig. 6. Notice that this trajectory requires 10 *ms* more to execute over the constrained internal force example, despite the relative simplicity of the computation.

Notice the weight matrix $W = \mathbf{I}_{6X6}$ (a 6X6 identity matrix) was used to calculate the torques and a rigid grasp between the end-effectors and the object was assumed.

6.2. Trajectories which needs to be slowed down to satisfy the constraints

In this example the velocity along the y-axis is increased, such that $\dot{p}_y = 5\ sin\ 20t$. Then trajectory scaling was found to be [c^{2-} , c^{2+}] = [0.0 , 0.265], when the internal forces were constrained such that $-2\ N \le f_{ix} \le 2\ N$, and $-1\ Nm \le v_{iz} \le 1\ Nm$. The given trajectories are unrealizable and we need to modify the trajectories by choosing c^2 from the above range. If we select $c^2 = 0.265$, then the torque profiles are as shown in Fig. 5. The overall traversal time is now extended from $t_f = 157$ *msec* to $t_1 = 157/\sqrt{0.265}$ = 304 *msec* . This is consistent with the case in Section 6.1(b) as the motion is along the same trajectory except the initially given trajectory was fast.

6.3. Configuration dependence of trajectories

In this example, the manipulator configuration is changed from that in Section 6.1 to shoulder down-up configuration in Fig. 3 (b). Intuitively, this configuration is not suitable for lifting a heavy object. For the trajectory described in Section 6.1 we found the range of c^2 to be [c^{2-} , c^{2+}] = [0.0 , 2.516] when the internal forces were constrained such that $-2\ N \le f_{ix} \le 2\ N$, and $-1\ Nm \le v_{iz} \le 1\ Nm$. If we select c^2 = 2.516, the modified torque profiles are as shown in Fig. 7(a). The new traversal time is reduced from 304 *msec* in Section 6.1 to $t_1 = 523/\sqrt{2.516} = 330$ *msec* . Notice in Fig. 7(b) that the f_{1y} and f_{2y} have opposite signs after about 50 *msec* and the difference between f_{1y} and f_{2y} reaches 25 N near the end of the trajectory. Such large forces might lead to damage of the object. This can be avoided by employing even distribution of payload with zero internal forces. The even payload algorithm yields [c^{2-} , c^{2+}] = [0.0

, 2.406]. If we select $c^2 = 2.406$, then $t_1 = 523/\sqrt{2.406} = 337$ *msec* and the corresponding torque profiles are as given in Fig. 8.

6.4. Effect of Load Variation

In this example, we will assume that the mass of the object, m, may be varying between $0.1kg$ and $3kg$ and all other information pertaining to the system will remain as described in Section 6.1. We exhaustively searched along the range of the mass, and found that the global range of c^2 as $[\, c^{2-}, c^{2+}] = [\, 0.0, 1.358]$ which occurred at $m = 3$ kg when the internal forces were constrained such that $-2\,N \le f_{ix} \le 2\,N$, and $-1\,Nm \le v_{iz} \le 1\,Nm$. Now, the execution time is extended from 304 *msec* in Section 6.1 to $t_1 = 523/\sqrt{1.358} = 448$ *msec*. The reduction in the range of c^2 results from the fact that the trajectory is now designed to accommodate unspecified objects, $m \in [0.1, 3]\,kg$.

VII. Conclusion

In this paper we developed a methodology to alter the trajectory of cooperative multirobot operations, such that torque constraints were not violated. Our algorithm employed the concept of time scaling introduced by Hollerbach[10]. There is a significant difference between the scaling of multi-robot trajectories over that of scaling single robot trajectories. The difference results from the differences in the dynamic equations of the two systems. In the multi-robot system, there is the force interaction between the manipulators which is related to the payload distribution. Also it is usually redundantly actuated.

We have shown that we can determine the range of the time scaling constants by using linear programming methods. We can also accommodate constraints on the internal forces. Simplification to the time scaling can be found if a known payload distribution is desired. The joint torques can be found from a quadratic minimization which has the effect of lowering the energy consumption during cooperative operation. Extensions to our algorithm were developed which would allow robust trajectory planning when the mass and inertia of the payload is known to vary within certain bounds. Several simulations were given to show the effectiveness of our schemes to generate practically usable multi-robot trajectories.

APPENDIX A

Proposition 1 : Given the following equation

$$\mathbf{A}_1\mathbf{x} + \mathbf{b}_1 = k\,\mathbf{y}_1 \tag{A.1}$$

$$\mathbf{y}_2^- \leq \mathbf{A}_2\,\mathbf{x} - k\,\mathbf{b}_2 \leq \mathbf{y}_2^+ \tag{A.2}$$

$$\mathbf{x}^- \leq \mathbf{x} \leq \mathbf{x}^+ \tag{A.3}$$

where $\mathbf{A}_1 \in \boldsymbol{R}^{m\mathrm{x}n}$, $\mathbf{A}_2 \in \boldsymbol{R}^{m\mathrm{x}n}$, $\mathbf{b}_1 \in \boldsymbol{R}^m$, $\mathbf{b}_2 \in \boldsymbol{R}^m$, $\mathbf{y}_1 \in \boldsymbol{R}^m$, $\mathbf{y}_2^+ \in \boldsymbol{R}^m$, $\mathbf{y}_2^- \in \boldsymbol{R}^m$,

$\mathbf{x}^+ \in \boldsymbol{R}^n$, and $\mathbf{x}^+ \in \boldsymbol{R}^n$ are known. The vector $\mathbf{x} \in \boldsymbol{R}^n$ is bounded by equation (A.3) and $k \in \boldsymbol{R}^+$, $m \leq n$. Assume that k_0 and k_1 are greatest lower bound and least upper bound respectively, which satisfies the equations (A.1), (A.2), and (A.3). Then any $k \in [\,k_0 , k_1\,]$ satisfy the equations (A.1), (A.2), and (A.3).

proof : In order to show that any $k \in [\,k_0 , k_1\,]$ satisfies the equations (A.1), (A.2), and (A.3), it is enough to show that the convex combination of k_0 and k_1 satisfies the equations (A.1), (A.2), and (A.3). Consider the convex combination of k_0 and k_1, $k_\lambda = \lambda\, k_0 + (1-\lambda)\, k_1$, where $0 \leq \lambda \leq 1$. Let $\mathbf{x}_0$ and $\mathbf{x}_1$ be one of the corresponding values of $\mathbf{x}$ to k_0 and k_1, respectively, satisfying the equations (A.1), (A.2), and (A.3), i.e.,

$$\mathbf{A}_1\,\mathbf{x}_0 + \mathbf{b}_1 = k_0\,\mathbf{y}_1 \tag{A.4}$$

$$\mathbf{y}_2^- \leq \mathbf{A}_2\,\mathbf{x}_0 - k_0\,\mathbf{b}_2 \leq \mathbf{y}_2^+ \tag{A.5}$$

$$\mathbf{x}^- \leq \mathbf{x}_0 \leq \mathbf{x}^+ \tag{A.6}$$

$$\mathbf{A}_1\mathbf{x}_1 + \mathbf{b}_1 = k_1\,\mathbf{y}_1 \tag{A.7}$$

$$\mathbf{y}_2^- \leq \mathbf{A}_2\,\mathbf{x}_1 - k_1\,\mathbf{b}_2 \leq \mathbf{y}_2^+ \tag{A.8}$$

$$\mathbf{x}^- \leq \mathbf{x}_1 \leq \mathbf{x}^+ \tag{A.9}$$

Now it is enough to show that there exist $\mathbf{x}_\lambda$ such that which satisfies the equations (A.1), (A.2), and (A.3). Consider

$$k_\lambda\ \mathbf{y} = \{\lambda\, k_0 + (1-\lambda)\, k_1\}\mathbf{y}_1 = \lambda\, k_0\,\mathbf{y}_1 + (1-\lambda)\, k_1\mathbf{y}_1$$

$$= \lambda (\mathbf{A}_1 \mathbf{x}_0 + \mathbf{b}_1) + (1- \lambda) (\mathbf{A}_1 \mathbf{x}_1 + \mathbf{b}_1)$$
$$= \mathbf{A}_1 \{ \lambda \mathbf{x}_0 + (1- \lambda) \mathbf{x}_1 \} + \mathbf{b}_1 \tag{A.10}$$

Now define $\mathbf{x}_\lambda = \lambda \mathbf{x}_0 + (1- \lambda) \mathbf{x}_1$. Then eq. (A.10) becomes

$$k_\lambda \mathbf{y} = \mathbf{A}_1 \mathbf{x}_\lambda + \mathbf{b}_1 \tag{A.11}$$

Now consider

$$\mathbf{A}_2 \mathbf{x}_\lambda - k_\lambda \mathbf{b}_2 = \mathbf{A}_2 \{\lambda \mathbf{x}_0 + (1- \lambda) \mathbf{x}_1 \} - \{ \lambda k_0 + (1- \lambda) k_1 \} \mathbf{b}_2$$
$$= \lambda\{ \mathbf{A}_2 \mathbf{x}_0 - k_0 \mathbf{b}_2\} + (1- \lambda) \{ \mathbf{A}_2 \mathbf{x}_1 - k_1 \mathbf{b}_2\} \tag{A.12}$$

Thus we have

$$\mathbf{y}_2^- = \lambda \mathbf{y}_2^- + (1- \lambda) \mathbf{y}_2^- \leq \mathbf{A}_2 \mathbf{x}_\lambda - k_\lambda \mathbf{b}_2 \leq \lambda \mathbf{y}_2^+ + (1- \lambda) \mathbf{y}_2^+ = \mathbf{y}_2^+ \tag{A.13}$$

Since $\lambda \mathbf{x}^- \leq \lambda \mathbf{x}_0 \leq \lambda \mathbf{x}^+$ and $(1- \lambda)\mathbf{x}^- \leq (1- \lambda) \mathbf{x}_1 \leq (1- \lambda)\mathbf{x}^+$. Combining these two inequalities, we get

$$\mathbf{x}^- \leq \lambda \mathbf{x}_0 + (1- \lambda) \mathbf{x}_1 \leq \mathbf{x}^+ \tag{A.14}$$

We showed that for any k_λ , $0 \leq \lambda \leq 1$, there exists $\mathbf{x}_\lambda$ which satisfies the equations (A.1), (A.2), and (A.3).

Q. E. D.

Note that the vector notations in the Appendix A are different from the those in the paper. The following correspondence lists has been used.

$\mathbf{A}_1 = \mathbf{L}(t)$, $\mathbf{A}_2 = \mathbf{J}_D$, $\mathbf{b}_1 = -\mathbf{L}(t)\,\mathbf{G}(t) - \mathbf{Q}$, $\mathbf{b}_2 = \mathbf{J}_D\mathbf{E}(t)$,
$\mathbf{y}_1 = \mathbf{P}(t)+\mathbf{L}(t)\mathbf{E}(t)$, $\mathbf{y}_2^+ = \mathbf{b}_2^+$, $\mathbf{y}_2^- = \mathbf{b}_2^-$,

$\mathbf{x} = \tilde{\tau}(t/c)$, $\mathbf{x}^+ = \tau^+$, $\mathbf{x}^- = \tau^-$, $k = c^2(t)$

where, the left hand side terms are the ones used in the Appendix A and the ones on the right hand side are used in the paper.

APPENDIX B

Proposition 2 : Let $\mathbf{p}(t)$ be the initially given trajectory and $\tilde{\mathbf{p}}(t) = \mathbf{p}(c_1 t)$ be the modified trajectory with $c_1 = \sqrt{c^{2+}}$, where the range of c is given as $[\sqrt{c^{2-}}, \sqrt{c^{2+}}]$ from the algorithm ***MMTSF***. If t_f was the original traversal time, new traversal time is $t_1 = t_f / c_1$. Then t_1 is the shortest traversal time which can be obtained from time scaling of trajectories.

proof : Assume that there exists a time scaling constant c_2 such that $t_2 = t_f / c_2 < t_1$. Since $t_2 < t_1$, it implies $c_2 > c_1$. It means that $c_2 > \sqrt{c^{2+}}$, thus $c_2 \notin [\sqrt{c^{2-}}, \sqrt{c^{2+}}]$. Therefore, the trajectory corresponding to t_2 does violate the constraints at least one point during the motion. Thus t_1 is the shortest traversal time we can obtain from the trajectory scaling algorithm.

Q. E. D.

APPENDIX C

Proposition 3 : A solution to the quadratic programming problem, eq.(31) through eq.(34), satisfies the followings;

(i) torque constraints expressed by eq.(8),
(ii) internal force constraints expressed by eq. (24),
(iii) multi-robot dynamics expressed by eq. (13),
(iv) object dynamics expressed by eq. (20),

proof : Notice that a solution to the above quadratic programming problem satisfies eq. (32), (33), and (34). Since (i) and (ii) are restatements of eq.(34) and eq.(33), respectively, we only need to show that eq. (32) implies (iii) and (iv). Object dynamic equation and multi-robot dynamic equations are related by the forces, $\tilde{\mathbf{F}}_i$, for $i = 1, .., n$, which are unknown and found from a quadratic programming problem. Since eq.(22) results from the change of argument from t/c to t, eq.(32) implies eq.(22). Eq.(22) may be written as:

$$c^2\,\mathbf{P}+\mathbf{Q} = \sum_{i=1}^{n} \mathbf{B}_i(\mathbf{q}_i)\mathbf{J}_i(\mathbf{q}_i)^{-T}[\tilde{\tau}_i(t)-\{\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i''+\mathbf{q}_i'\mathbf{C}_i(\mathbf{q}_i)\mathbf{q}_i'\}c^2-\mathbf{G}_i(\mathbf{q}_i)] \tag{C.1}$$

If we let,

$$\mathbf{H}_i = \mathbf{J}_i(\mathbf{q}_i)^{-T}[\tilde{\tau}_i(t)-\{\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i''+\mathbf{q}_i'\mathbf{C}_i(\mathbf{q}_i)\mathbf{q}_i'\}c^2-\mathbf{G}_i(\mathbf{q}_i)] \tag{C.2}$$

Then from eq.(C.1) and (C.2), we obtain following two equations.

$$c^2\,\mathbf{P}+\mathbf{Q} = \sum_{i=1}^{n} \mathbf{B}_i(\mathbf{q}_i)\,\mathbf{H}_i \tag{C.3}$$

$$\tilde{\tau}_i(t) = \{\mathbf{D}_i(\mathbf{q}_i)\mathbf{q}_i''+\mathbf{q}_i'\mathbf{C}_i(\mathbf{q}_i)\mathbf{q}_i'\}c^2 + \mathbf{G}_i(\mathbf{q}_i) + \mathbf{J}_i(\mathbf{q}_i)^{-T}\,\mathbf{H}_i \tag{C.4}$$

Since $\tilde{\mathbf{F}}_i$ in equations (13) and (20) are arbitrary, (C.3) and (C.4) are equivalent expressions for eq.(20) and (13), respectively.

Q. E. D.

REFERENCES

[1] S. Ahmad and S. Luo, "Coordinated motion control of multiple robotic devices for welding and redundancy coordination through constrained optimization in Cartesian space," *IEEE Trans. Robotics and Automation*, vol. RA-5, no. 4, pp. 409-417, August 1989.

[2] S. Ahmad and H. C. Yan, "Minimum-time trajectory for multiple manipulators holding an common object," in *Proc. IFAC-AIPAC Conf.* (Nancy, France, July 1989).

[3] B. Benhabib, G. Zak, and E. Tabarah, "Position control of two-arm manipulators for coordinated point-to-point motion," *J. Robotic Sys.*, vol. 5, no. 2, pp. 103-124, 1988.

[4] L. D. Berkovitz, *Optimal Control Theory*, New York, NY: Springer-Verlag, 1974

[5] J. E. Bobrow, S. Dubowsky, and J. S. Gibson, "On the optimal control of robotic manipulators with actuator constraints," in *Proc. 1983 American Controls Conf.* (San Francisco, CA, June 1983), pp. 782-787.

[6] J. E. Bobrow, S. Dubowsky, and J. S. Gibson, "Time-optimal control of robotic manipulators along specified paths," *Int. J. Robotics Res.*, vol. 4, no. 3, pp. 3-17, 1985.

[7] J. H. Ginsberg, *Advanced Engineering Dynamics*, New York, NY: Harper & Row, 1988

[8] T. J. Graettinger and B. H. Krogh, "Evaluation and time-scaling of trajectories for wheeled mobile robots," *ASME J. Dyn. Syst., Meas., Contr.*, vol. 111, pp. 222-231, June 1989.

[9] S. Hayati, "Hybrid position/force control of multi-arm cooperating robots," in *IEEE Trans. Aerospace and Electronic Sys.*, vol. 24, no. 5, pp. 584-590, Sep. 1988.

[10] J. M. Hollerbach, "Dynamic scaling of manipulator trajectories," *ASME J. Dyn. Syst., Meas., Contr.*, vol. 105, pp. 102-106, Mar. 1984.

[11] T. Ishida, "Force Control in Coordination of Two Arms," in *Proc. 5th Int. Conf. A. I.*, pp. 717-722, Aug. 1977.

[12] M. E. Kahn and B. Roth, "The near-minimum-time control of open-loop articulated kinematic chains," *ASME J. Dyn. Syst., Meas., Contr.*, vol. 111, pp. 164-172, 1971.

[13] A. J. Koivo and G. A. Bekey, "Report of workshop on coordinated multiple robot manipulators: planning, control, and applications," *IEEE J. Robotics and Automat.*, vol. RA-4, no. 1, pp. 91-93, Feb. 1988.

[14] T. W. Nye, D. J. LeBlanc, and R. J. Cipra, "Design and Modelling of a computer controlled planar manipulator," *Int. J. Robotics Res.*, vol. 6, no. 1, pp. 85-95, 1987.

[15] D. E. Orin and S. Y. Oh, "Control of force distribution in robotic mechanisms containing closed kinematic chains," *ASME J. Dyn. Syst., Meas., Contr.*, vol. 102, pp. 134-141, 1981.

[16] F. Pfeiffer and R. Johanni, "A concept for manipulator trajectory planning," *IEEE J. Robotics and Automation*, vol. RA-3, no. 2, pp. 115-123, April 1987.

[17] S. S. Rao, *Optimization Theory and Applications*, India: Wiley Eastern Limited, 1984

[18] G. Sahar and J. M. Hollerbach, "Planning of minimum-time trajectories for robot arms," *Int. J. Robotics Res.*, vol. 5, no. 3, pp. 90-100, 1986.

[19] K. G. Shin and N. D. McKay, "An efficient robot arm control under geometric path constraints," in *Proc. 22nd IEEE Conf. Decision and Contr.*, pp. 1449-1457, 1983.

[20] K. G. Shin and N. D. McKay, "Minimum-time control of robotic manipulators with geometric path constraints," *IEEE Trans. Automat. Contr.*, vol. AC-30, no. 6, pp. 531-541, June 1985.

[21] T. J. Tarn, A. K. Bejczy, and X. Yun, "New nonlinear control algorithms for multiple robot arms" in *IEEE Trans. Aerospace and Electronic Sys.*, vol. 24, no. 5, pp. 571-583, Sep. 1988.

[22] Y. F. Zheng, "Kinematics and dynamics of two industrial robots in assembly," in *Proc. 1989 IEEE Conf. Robotics and Automation*, pp. 1360-1365.

[23] Y. F. Zheng and J. Y. S. Luh, "Optimal load distribution for two industrial robots handling a single object," *ASME J. Dyn. Syst., Meas., Contr.*, vol. 111, pp. 232-237, 1989.

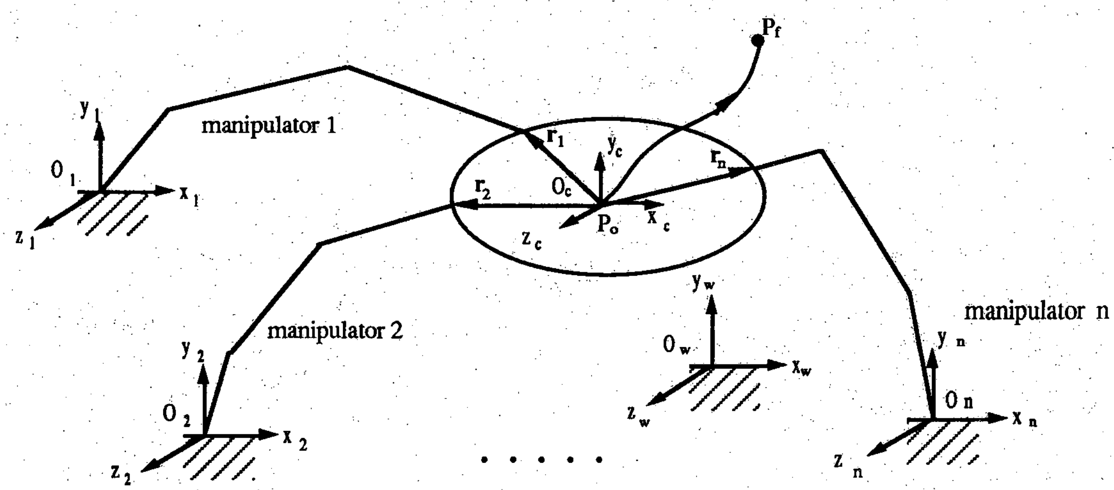

Figure 1. Multiple Manipulators Holding a Common Object.

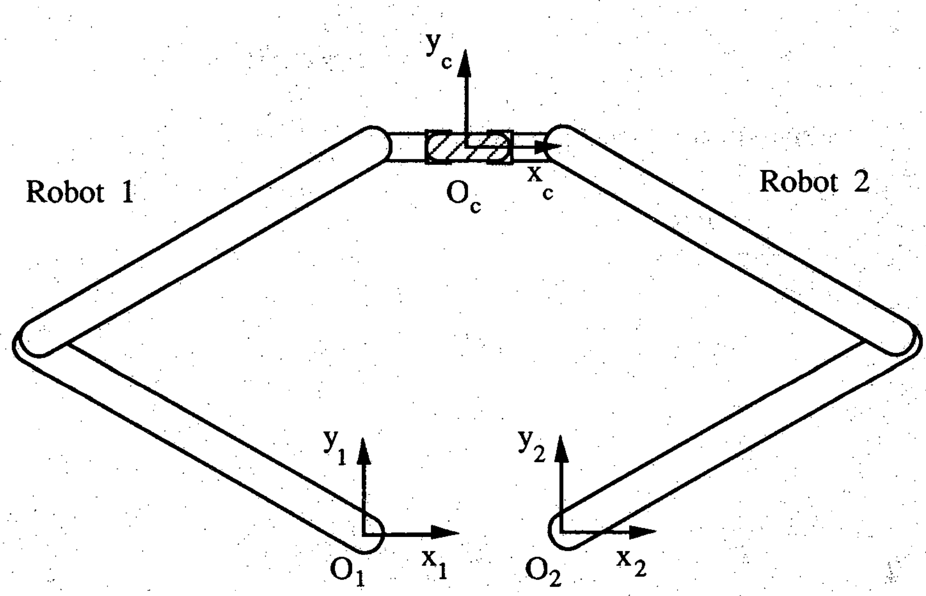

Fig. 2. Dual 3-DOF robots in cooperative manipulation.

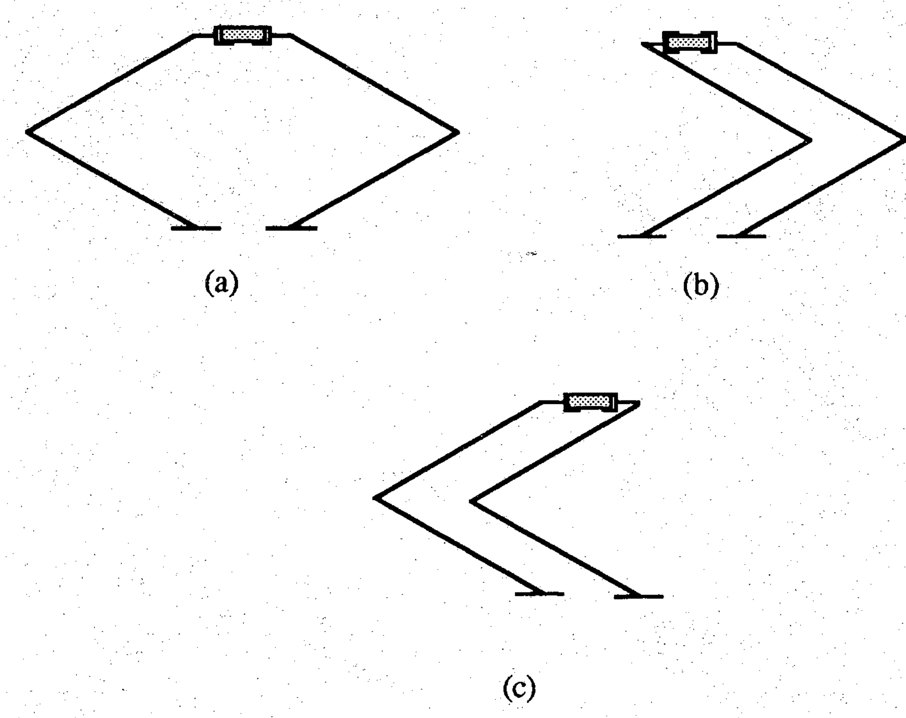

Fig. 3. Three configurations according to shoulder position
(a) shoulder up-up (b) shoulder down-up
(c) shoulder up-down

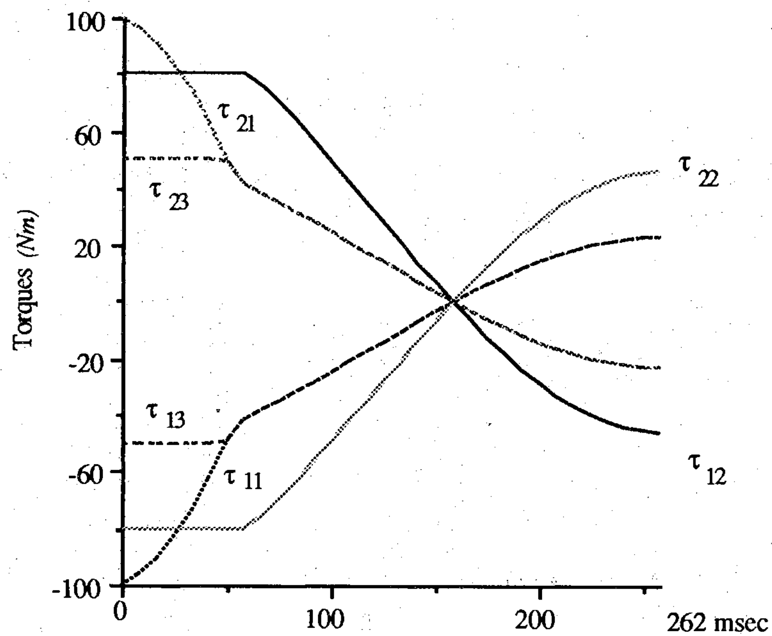

Fig. 4(a) Torque Profiles from the algorithm without constraints on the internal forces ($c^2 = 3.988$).

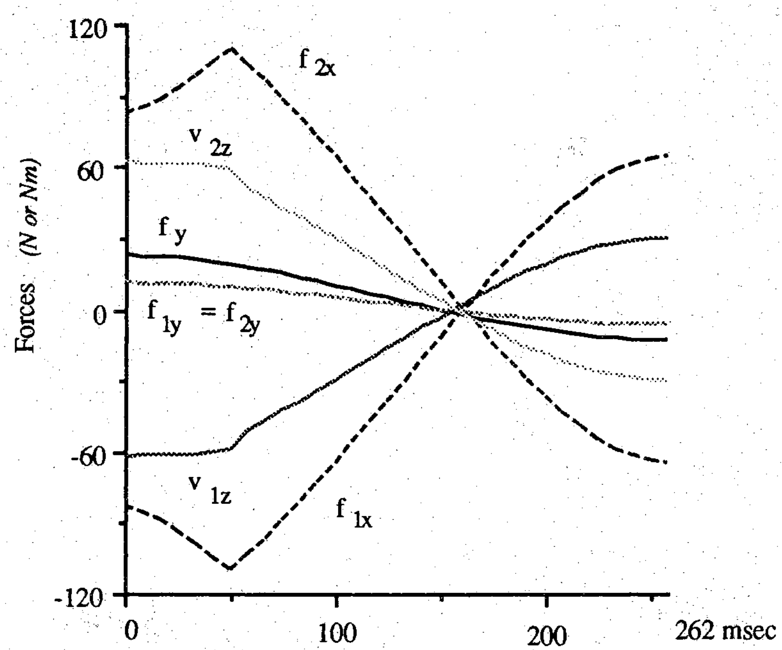

Fig. 4(b) Force profiles from the algorithm without constraints on the internal forces ($c^2 = 3.988$).

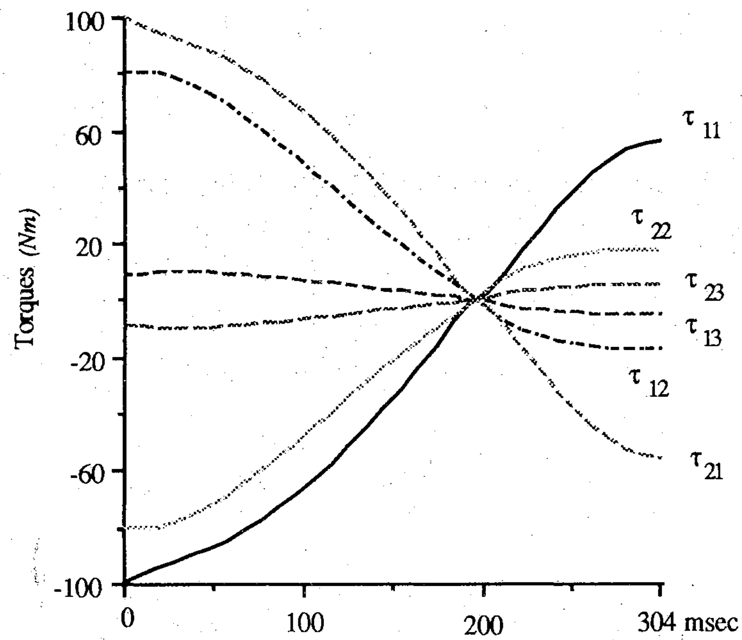

Fig. 5(a) Torque Profiles from the algorithm with constraints on the internal forces, $-2 \leq f_{ix} \leq 2$, and $-1 \leq v_{iz} \leq 1$ ($c^2 = 2.949$).

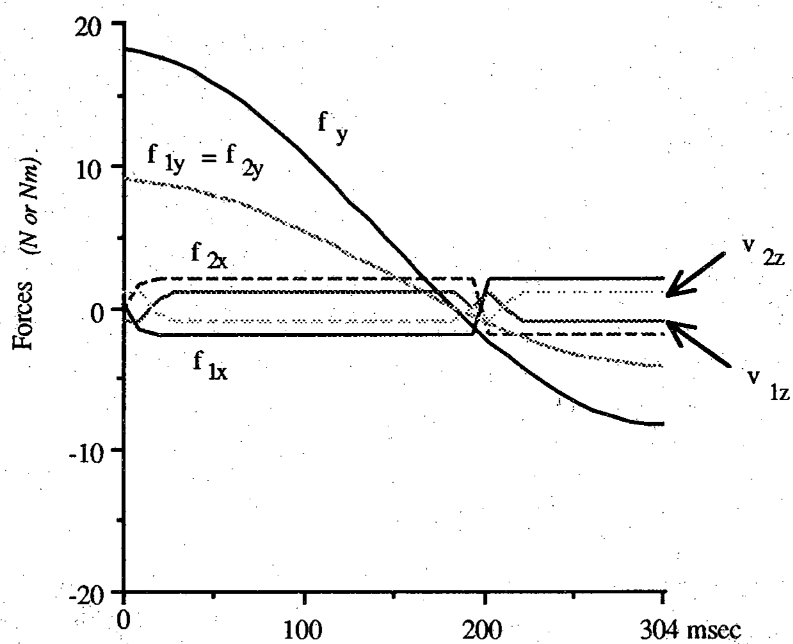

Fig. 5(a) Force Profiles from the algorithm with constraints on the internal forces, $-2 \le f_{ix} \le 2$, and $-1 \le v_{iz} \le 1$ ($c^2 = 2.949$).

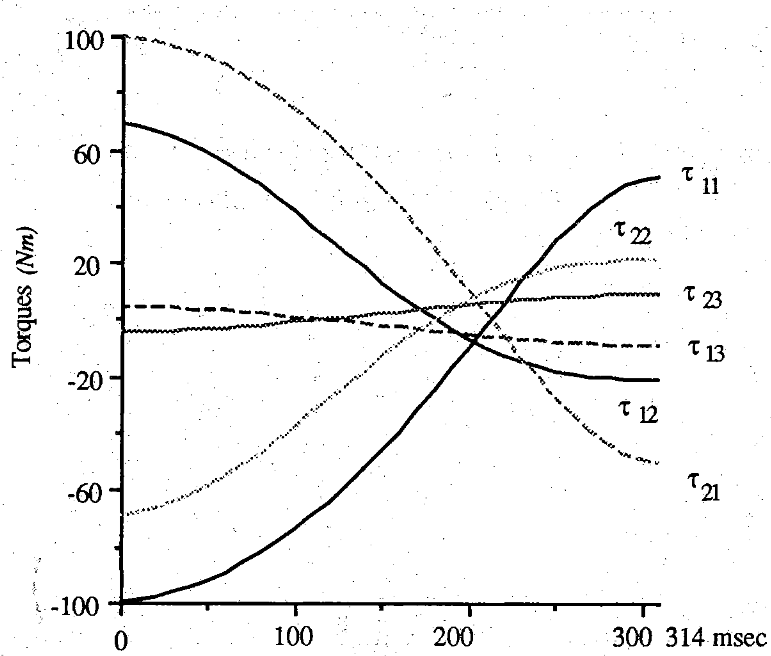

Fig. 6 Torque Profiles from the algorithm with the even distribution with zero internal forces ($c^2 = 2.779$).

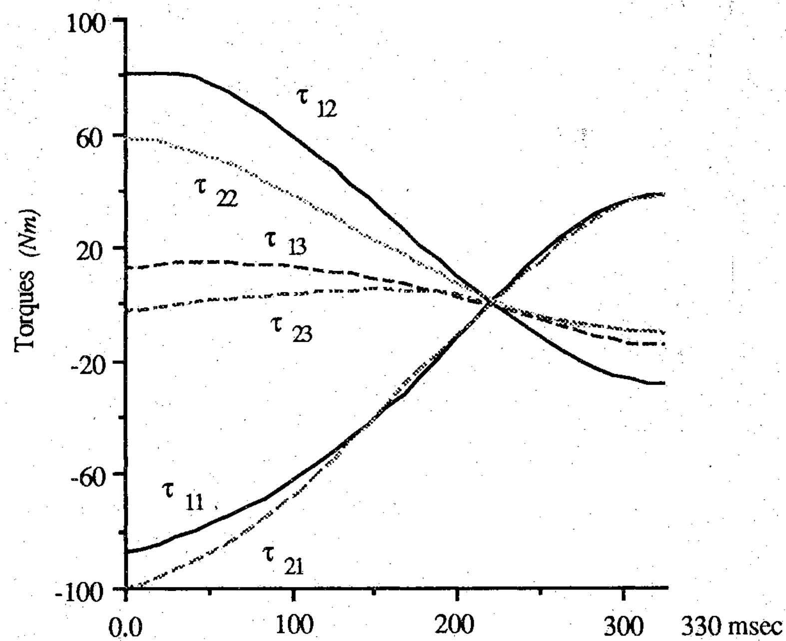

Fig. 7(a) Torque profiles from the algorithm with constraints on the internal forces, $-2 \leq f_{ix} \leq 2$, and $-1 \leq v_{iz} \leq 1$ in down-down configuration ($c^2 = 2.516$).

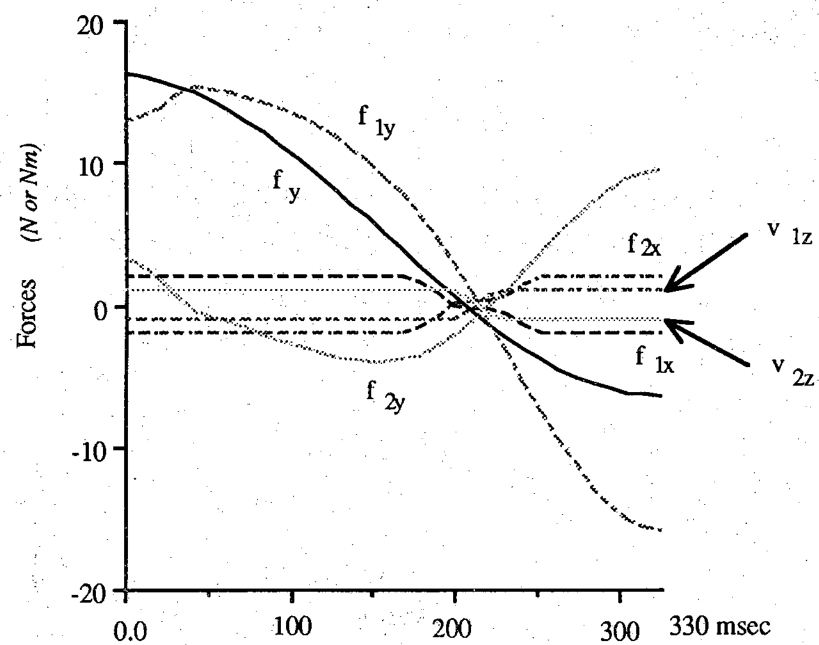

Fig. 7(b) Force profiles from the algorithm with constraints on the internal forces, $-2 \leq f_{ix} \leq 2$, and $-1 \leq v_{iz} \leq 1$ in down-down configuration ($c^2 = 2.516$).

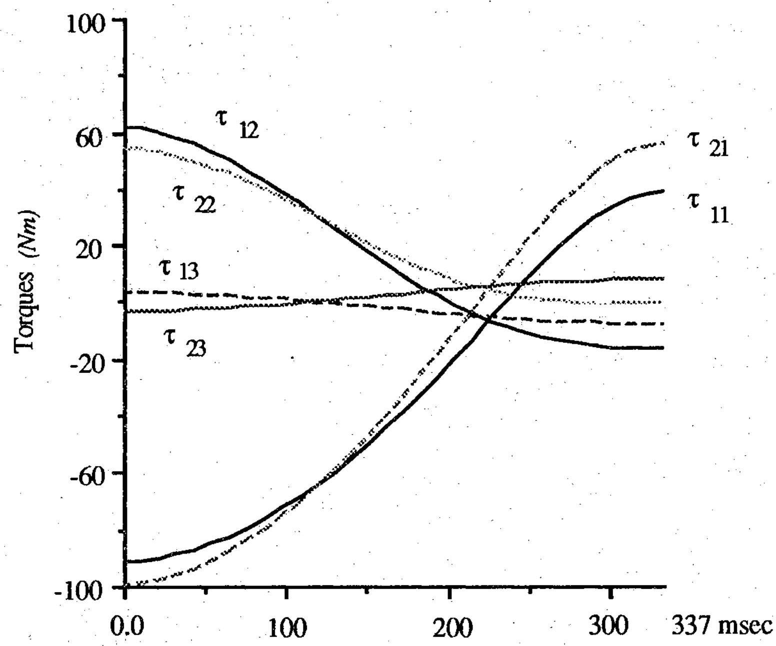

Fig. 8 Torque profiles from the algorithm with even load distribution with zero internal forces in down-down configuration ($c^2 = 2.406$).