2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

On Robot Gymnastics Planning with Non-zero Angular Momentum

Evangelos Papadopoulos, *Senior Member IEEE*, Ioannis Fragkos, and Ioannis Tortopidis

***Abstract* – Conservation of angular momentum that introduces nonholonomic behavior, underactuation and time dependence, makes the trajectory planning of gymnastic robots difficult. By defining appropriate values for the initial angular momentum, a method is developed that can lead a mechanism to a desired final configuration from an initial given one, in prescribed time. This method is optimization-based and fully exploits the initial mechanism angular momentum. Obstacle avoidance during flight is achieved by setting additional constraints. The method results in smooth, small in magnitude, and therefore easily applicable joint torques.**

Index Terms* – *Gymnast robots, underactuated, nonholonomic planning, non-zero angular momentum.

I. Introduction

Research in gymnastic robots and related subjects has a number of purposes, such as the development of gymnast robots and the understanding of human torque requirements to execute a gymnastic movement. Designing gymnast robots improves our capabilities in developing humanoid robots and can provide us with useful knowledge about the importance of motion-related parameters. Also, the study of gymnastic motions is needed because it is difficult for an athlete to realize a priori the amount of torques required for a figure, and this, in turn, can result in injuring himself or herself.

Gymnastic motions include various tasks, such as jumping, dismounting from a bar, somersault, diving, back flips and so on. A number of researchers have dealt with these problems and have produced some interesting results. In the early 80's, Raibert achieved jumping, somersault control and dynamic stabilization of a 3D biped robot, [1]. Spong has succeeded in swinging up the acrobot and stabilizing it in its upright position, [2]. Mita *et al.* have introduced an analytical time optimal solution for an acrobot with non-zero initial momentum, [3]. They have shown that time optimal trajectories can be obtained using singular control and switching controllers. Grizzle *et al.* have constructed a scalar function, similar to the one in [3], by partially integrating the angular momentum equation and implemented it in a two-link structure undergoing ballistic motion, [4]. Trajectory planning has been studied by using analytical formulas subject to a switching condition in the neighborhood of a singular point. It is worth noticing that, till now, there are no analytical expressions for 3-link structures like these in [3,4]. Some researchers have given numerical solutions in gymnastic problems, like Godhavn *et al.* [5] or Kamon *et al.* [6] for somersault motions. Also related is the work of Sang-Ho Hyon *et al.*, where they simulated a 4-link mechanism in jumping, back handspring and somersault using target dynamics-based control, [7]. They focused on controlling physical quantities such as the center of mass (CM) motion or the angular momentum. Furthermore, they have constructed a 4-link model to implement the results of their work.

In this paper, a novel planning technique for planar rigid multibody systems undergoing ballistic motions is implemented. The method is extended so that the mechanism can also perform obstacle avoidance. This problem tackled is modeled after the gymnastic high jump problem, see Fig. 1.

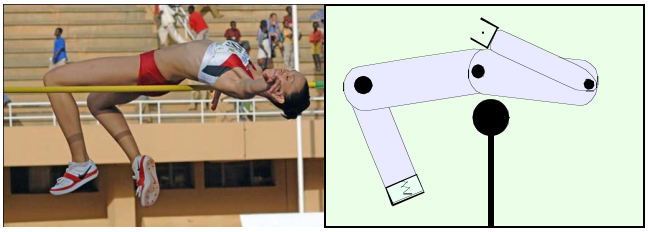

Fig. 1. A four link robot imitates the high jump movement of an athlete.

The main problem that is presented and solved in this paper is the construction of a trajectory that transfers the robot from a given initial configuration to a desired final one, in given time. This is a common phase for many gymnastic problems like the ones mentioned above, while it is also related to the famous cat problem, [8]. The main characteristic in this system's dynamical model is that the conservation of angular momentum poses a velocity-type constraint to it. At this point, the problem's nonholonomic nature and all the subsequent difficulties in trajectory planning are revealed: we are capable of controlling explicitly n actuated dof while we intent to affect the behavior of $n+1$ dof. Therefore, as we have fewer actuators than dof, the system is underactuated. Leading the system to the desired final configuration is all but trivial. Another important characteristic is that base orientation does not affect the robot kinetic energy and the variable representing the orientation is called cyclic. The planning

Manuscript received September 15, 2006. This work was supported by the General Secretariat for Research and Technology EPAN Cooperation Program 4.3.6.1.b (Greece-USA 035). Additional support provided by public funds (European Social Fund 75% and GSRT 25%) and private funds, (Zenon SA), within 8.3 of Op. Pr. Comp., 3rd CSP - PENED 2003.

E. Papadopoulos is with the Department of Mechanical Engineering, National Technical University of Athens, (NTUA) 15780 Athens (tel: +30-210-772-1440; fax: +30-210-772-1455; e-mail: egpapado@central.ntua.gr).

I. Fragkos was with the Department of Mechanical Engineering, NTUA, 15780 Athens (e-mail: mc01027@mail.ntua.gr).

I. Tortopidis is with the Department of Mechanical Engineering, NTUA, 15780 Athens (e-mail: itor@central.ntua.gr).

scheme is strongly dependent on the initial angular momentum value, which the robot must acquire. This value is automatically determined since the flight time duration is chosen. Although the approach is numerical, it has the advantage of not allowing the robot to pass from a singular configuration and of always yielding a solution. The reason for this is the employment of smooth sinus functions with polynomials as arguments. Moreover, obstacle avoidance during the flight is presented. The developed method is extended while the use of optimization-based criteria, simplifies the problem leading to feasible solutions.

II. DYNAMICS AND PHYSICAL CONSTRAINTS OF MOTION

A. Dynamics

Free-flying robots consist of several joined rigid bodies with actuators mounted in every joint, as shown in Fig. 2.

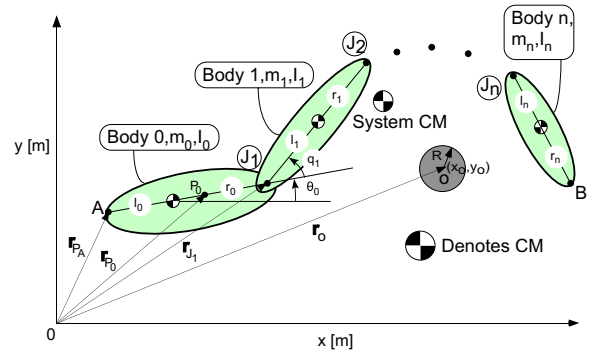

Fig. 2. An n-link robot undergoing ballistic motion in a vertical plane. Joints $q_1, q_2, \ldots, q_n$ are actuated. Obstacle parameters are also defined.

All joints are revolute, part of an open kinematic chain, so that, in a system with n joints, there are $n+3$ dof (under-actuated system). The system CM undergoes a planar ballistic motion due to the linear initial velocity given to the mechanism. The equations of motion have the form, [4],

$$\mathbf{M}(\bar{\mathbf{q}})\ddot{\mathbf{q}}+\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}+\mathbf{G}(\mathbf{q})=\begin{bmatrix}0\\ \boldsymbol{\tau}\end{bmatrix};\ \mathbf{q}(0)=\mathbf{q}^{in},\ \dot{\mathbf{q}}(0)=\dot{\mathbf{q}}^{in} \quad (1)$$

$$\ddot{x}_c=0;\ x_c(0)=x_c^{in},\ \dot{x}_c(0)=\dot{x}_c^{in} \quad (2)$$

$$\ddot{y}_c=-g_0;\ y_c(0)=y_c^{in},\dot{y}_c(0)=\dot{y}_c^{in} \quad (3)$$

where $\mathbf{M}(\bar{\mathbf{q}})$ is a positive definite symmetric matrix, called the system inertia matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ contains nonlinear velocity terms and $\mathbf{G}(\mathbf{q})$ contains gravity and angular momentum terms. The $n\times 1$ column vectors $\bar{\mathbf{q}}=[q_1,...,q_n]^T$ and $\boldsymbol{\tau}=[\tau_1,...,\tau_n]^T$ represent joint angles and torques respectively, while the $(n+1)\times 1$ vector $\mathbf{q}$ is defined as $\mathbf{q}=[\theta_0,\bar{\mathbf{q}}^T]^T$. The angle θ_0 stands for the orientation of the base (body 0), and x_c and y_c stand for CM coordinates whose dynamics are described by (2), and (3). Subscript in as indicates initial conditions. The first of the $n+1$ equations in (1) shows that the base orientation is not controllable explicitly, since no external torque is applied (i.e., the orientation dof is unactuated). After some algebraic manipulations in (1) (see [9] for details), the angular momentum constraint equation is obtained as,

$$D_0(\bar{\mathbf{q}})\dot{\theta}_0+D_1(\bar{\mathbf{q}})\dot{q}_1+...+D_n(\bar{\mathbf{q}})\dot{q}_n=h \quad (4)$$

where h is the total angular momentum wrt the system CM. Although in most works the initial angular momentum value is zero, here it is not. As a result, the whole mechanism rotates around its CM even if the joint angles are fixed. This characteristic makes the path planning problem seem more demanding. However, if we manage to exploit the existence of this momentum, the gymnastic task can be facilitated significantly. Due to (1), the configuration variables can be controlled explicitly, while the base orientation can be found from (4). On the other hand, the system CM motion cannot be controlled. Once the initial velocity is set, we cannot interfere with its motion. As it was previously mentioned, the initial angular momentum value is critical because it affects the base rotation boundaries. This is the subject of the next paragraph.

B. Physical Constraints in Motion Planning

Integrating (4) results in

$$\underbrace{\int_{\theta_0^{in}}^{\theta_0^{fin}} D_0(\bar{\mathbf{q}})\,d\theta_0}_{\Delta H_0}+\sum_{i=1}^{n}\underbrace{\int_{q_i^{in}}^{q_i^{fin}} D_i(\bar{\mathbf{q}})\,dq_i}_{\Delta H_i}=h\Delta t \quad (5)$$

where Δt is the flight time duration. Terms ΔH_i with $i=0,1,...,n$ show each link's dynamic reaction through time. Note that if the joint angle trajectories are known as a function of time and the flight time duration is specified, (5) can be integrated numerically to yield the final value of the base orientation. However, the orientation cannot be determined analytically, as it is depends on the particular joint space path that is taken.

The main problem we address here is to find a path, which in given time connects an initial configuration with a desired one, by actuating only the joints $J_1,J_2,...,J_n$. This is not trivial, as we must control $n+1$ variables by actuating n joints only. At the same time, due to the existence of initial momentum, the problem is time dependent. One can see that if the left side of (5) is bounded then the value of initial angular momentum must be bounded too. On the contrary, constraints imposed on the system by its CM linear motion are holonomic because (2), and (3) can be integrated analytically to yield the CM trajectory. This is related to $\mathbf{q}$ by definition as:

$$\begin{aligned} x_c&=A_0\cos(\theta_0)+\sum_{i=1}^{n}A_i\cos(q_i)+x_A\\ y_c&=A_0\sin(\theta_0)+\sum_{i=1}^{n}A_i\sin(q_i)+y_A \end{aligned} \quad (6)$$

The terms A_i are functions of system lengths and masses, while x_A, y_A, are the coordinates of the mechanism's initial contact point A, see Fig. 2.

Next, we estimate approximate bounds for the initial angular momentum, following a technique proposed in [10]. In the case of ΔH_0, its bounds are found as

$$\Delta H_0^{\min} = \begin{Bmatrix} \Delta\theta_0 \cdot \min_{\bar{\mathbf{q}}} D_0\left(\bar{\mathbf{q}}\right), \Delta\theta_0 > 0 \\ \Delta\theta_0 \cdot \max_{\bar{\mathbf{q}}} D_0\left(\bar{\mathbf{q}}\right), \Delta\theta_0 < 0 \end{Bmatrix}$$
$$\Delta H_0^{\max} = \begin{Bmatrix} \Delta\theta_0 \cdot \max_{\bar{\mathbf{q}}} D_0\left(\bar{\mathbf{q}}\right), \Delta\theta_0 > 0 \\ \Delta\theta_0 \cdot \min_{\bar{\mathbf{q}}} D_0\left(\bar{\mathbf{q}}\right), \Delta\theta_0 < 0 \end{Bmatrix} \quad (7)$$

while for terms ΔH_i, the bounds are

$$\Delta H_i^{\min} = \begin{Bmatrix} \Delta q_i \min_{\bar{\mathbf{q}}} D_i\left(\bar{\mathbf{q}}\right), \Delta q_i > 0 \\ \Delta q_i \max_{\bar{\mathbf{q}}} D_i\left(\bar{\mathbf{q}}\right), \Delta q_i < 0 \end{Bmatrix}$$
$$\Delta H_i^{\max} = \begin{Bmatrix} \Delta q_i \max_{\bar{\mathbf{q}}} D_i\left(\bar{\mathbf{q}}\right), \Delta q_i > 0 \\ \Delta q_i \min_{\bar{\mathbf{q}}} D_i\left(\bar{\mathbf{q}}\right), \Delta q_i < 0 \end{Bmatrix} \quad (8)$$

Here $\Delta\theta_0$ and Δq_i denote the difference between final and initial base orientation and joint angles respectively.

Note here that these bounds are approximate. Exact bounds cannot be found trivially, as they depend on the specific nature of each problem. However, the above ones are valid in the sense that, for given time duration, the motion may have a solution if the value of the initial angular momentum lies in the interval:

$$\frac{1}{\Delta t}\left[\sum_{i=0}^{n}\Delta H_i^{\min},\ \sum_{i=0}^{n}\Delta H_i^{\max}\right] \quad (9)$$

C. Preliminary Analysis

We must first choose an appropriate flight time duration. Here Δt is considered as an input and its value must be specified. Note that this value must be within some reasonable limits or else a large initial velocity will be required (case of $\Delta t >>$) or instantaneous, unrealistic configuration changes will have to occur (case of $\Delta t <<$). Once the time interval is specified, boundaries given by (9) are explicitly defined. By choosing the initial angular momentum equal to the average of the boundaries in (9), we ensure that its value is the most appropriate for a desired motion:

$$h = \frac{1}{2}\frac{1}{\Delta t}\sum_{i=0}^{n}\left(\Delta H_i^{\min} + \Delta H_i^{\max}\right) \quad (10)$$

Letting

$$\dot{\bar{\mathbf{q}}}(\mathbf{0}) = \dot{\bar{\mathbf{q}}}^{in} = [\dot{q}_1^{\ in}\ \ \dot{q}_2^{\ in} \ldots \dot{q}_n^{\ in}]^T$$

(4) results in

$$\dot{\theta}_0^{\ in} = \dot{\theta}_0\left(0\right) = \{h - \sum_{i=1}^{n} D_i(\bar{\mathbf{q}}^{in})\dot{q}_i^{\ in}\} / D_0\left(\bar{\mathbf{q}}^{in}\right) \quad (11)$$

Next, we can calculate the robot's initial y-velocity by integrating (3) twice, taking one boundary condition from (6) with $y_A^{\ in} = 0$ and another boundary condition if we write the CM equation as in (6), but starting for point B this time (it must be $y_B^{\ fin} = 0$). The robot's horizontal motion does not affect any parameter so it can be determined arbitrarily.

III. Nonholonomic Planning in Joint Space

A. Obstacle-free Environment

Here the basic methodology is presented. If we assume that each joint angle has a polynomial trajectory, the path planning problem is reduced in estimating polynomial coefficients, as in [10]. However, it is not sure that the corresponding joint trajectories will have reasonable bounds. Hence, we are interested in employing polynomial trajectories and at the same time guarantee that each link will not rotate without bounds. For these reasons, sinus functions with polynomial arguments are used. In this way, we can tune the length of each angle separately, while polynomial coefficients can be redundant with no effect in each angle's boundaries. Assuming that joint angles are $q_i \in [-\pi, \pi]$, we choose

$$q_i\left(t\right) = \pi \sin\left(b_0^{\ i} + b_1^{\ i} t + \ldots + b_{k_i}^{\ i} t^{k_i}\right), \quad i = 1,2,\ldots,n \quad (12)$$

were k_i implies the degree of polynomial i. The following boundary conditions are imposed:

$$\begin{Bmatrix} q_i\left(s\right) = q_i^{\ s} \\ \dot{q}_i\left(s\right) = \dot{q}_i^{\ s} \\ \ddot{q}_i\left(s\right) = \ddot{q}_i^{\ s} \end{Bmatrix}, \quad i = 1,2,\ldots,n \quad (13)$$

where $s = t^{in}, t^{fin}$. In other words, each joint angle must satisfy six conditions: two for its initial and final desired values and four for its initial and final velocities and accelerations. If (4) is solved for θ_0 and integrated, it yields,

$$\Delta\theta_0 = \int_{t^{in}}^{t^{fin}} \left\{ g_0\left(\bar{\mathbf{q}}\right)h + \sum_{i=1}^{n} g_i\left(\bar{\mathbf{q}}\right)\dot{q}_i \right\} dt \quad (14)$$

where terms $g_i\left(\bar{\mathbf{q}}\right)$ are given by,

$$g_0(\bar{\mathbf{q}}) = -h / D_0,\ \ g_i(\bar{\mathbf{q}}) = -[D_i / D_0]\dot{q}_i, \quad i = 1,2,\ldots,n \quad (15)$$

Equations (13) and (14) are $6n+1$ and the free parameters are $n + \sum_{i=1}^{n} k_i$. So the following condition must hold,

$$5n + 1 \le \sum_{i=1}^{n} k_i \quad (16)$$

If we choose the number of parameters equal to the number of constraints, a system of $6n+1$ equations has to be solved. On the other hand, if we use more coefficients than constraints, we can determine their value using the criterion,

$$\min \| [\mathbf{b}^1\, \mathbf{b}^2 \ldots \mathbf{b}^n]^T \| \quad (17)$$

where $\mathbf{b}^i = [b_0^{\ i}\, b_1^{\ i} \ldots b_k^{\ i}]$, and $i = 1,2,\ldots,n$. After joint space trajectories are specified, $\theta_0(t)$ is given by

$$\theta_0(t) = \int_{t^{in}}^{t} \left\{ g_0(s)h + \sum_{i=1}^{n} g_i(s)\dot{q}_i(s) \right\} ds \quad (18)$$

All other variables are directly estimated using (12). This approach leads always to a path, while the resulting joint torques are smooth and small in size.

As this optimization problem is easily manipulated with the above method, we may enter new constraints and satisfy more requirements. For example, we may set as constraint the maximum permissible value of joint torques, or the parameters for obstacle avoidance. The later case is developed next.

B. Obstacle Avoidance

Here the case where a circular obstacle, such as a pole, is interpolated in the robot's orbit is examined. The target is the same as before, i.e., reach a desired final configuration from an initial one in prescribed time, avoiding collision with the obstacle. Obviously, obstacle position is such that collision may happen but it may also be avoided by designing the trajectories in an appropriate way.

It is clear that all previous constraints, plus some additional due to the obstacle, must hold. Denoting by P_j an arbitrarily selected point in link j, the distance between P_j and the obstacle's center, o, must be greater than R,

$$|| \mathbf{r}_{P_j} - \mathbf{r}_o \| > R, \quad j = 0,1,...,n \tag{19}$$

where all above variables are defined in Fig. 2. We must try to express the position vectors $\mathbf{r}_{P_j}$ as a function of $\mathbf{q}$ in order to model the constraints imposed on the problem. We express the joint position vectors $\mathbf{r}_{J_j}$ and the edge point position vector $\mathbf{r}_B$ as a function of $\mathbf{q}$ and $\mathbf{r}_A$, the position vector of the lowest point in the body, see Fig. 2:

$$\begin{aligned}
\mathbf{r}_{J_1} &= \mathbf{r}_A + \mathbf{R}_Z(\theta_0)\mathbf{L}_0 \\
\mathbf{r}_{J_i} &= \mathbf{r}_{J_{i-1}} + \mathbf{R}_Z(\theta_0)\left[\prod_{k=1}^{i-1}\mathbf{R}_Z(q_k)\right]\mathbf{L}_{i-1}, \; i = 2,3,...,n \\
\mathbf{r}_B &= \mathbf{r}_{J_n} + \mathbf{R}_Z(\theta_0)\left[\prod_{k=1}^{n}\mathbf{R}_Z(q_k)\right]\mathbf{L}_n
\end{aligned} \tag{20}$$

where $\mathbf{R}_Z$ is a Z axis principal rotation matrix, and,

$$\mathbf{L}_i = [r_i + l_i,\ 0,\ 0]^T, i = 0,...,n \tag{21}$$

Lengths r_i and l_i are defined in Fig. 2. Any body point can be described as a function of the joint position vectors and the position vectors of A and B, using $u \in [0,1]$:

$$\left\{\begin{aligned}
\mathbf{r}_{P_0} &= (1-u)\mathbf{r}_A + u\mathbf{r}_{J_1} \\
\mathbf{r}_{P_j} &= (1-u)\mathbf{r}_{J_{j-1}} + u\mathbf{r}_{J_j} \\
\mathbf{r}_{P_n} &= (1-u)\mathbf{r}_{J_n} + u\mathbf{r}_B
\end{aligned}\right\} \tag{22}$$

for each $j = 1,...,n-1$. By substituting the values of $\mathbf{r}_j$, $\mathbf{r}_B$ from (20) in (22), we can express each body's coordinates using only the variables $\mathbf{q}$, u and $\mathbf{r}_A$. Furthermore, we can use (6) to express $\mathbf{r}_A$ as a function of $\mathbf{q}$ and (x_c, y_c). This comes to the result that the position vector of every point can be expressed as $\mathbf{r}_{P_j} = \mathbf{r}_{P_j}(\mathbf{q}, u, x_c, y_c)$, or $\mathbf{r}_{P_j} = \mathbf{r}_{P_j}(\mathbf{q}, u, t)$. Thus, for obstacle avoidance (19) becomes,

$$|| \mathbf{r}_{P_j}(\mathbf{q}, u, t) - \mathbf{r}_0 \| > R, \forall\ j = 0,1,...,n \tag{23}$$

These are $n+1$ constraints that must be considered parametrically, for each $t \in [t^{in}, t^{fin}]$ and $u \in [0,1]$. The problem is posed on solving the system with constraints (13), (14) and (23) and, in comparison with the previous case, it is more intensive from a computational view point. However, some techniques have been proposed for the solution of this problem (see for example [11]). In the sequel, a variation of the SIP algorithm that constructs a grid in the parameter space and solves the problem in it is used. This is done using the Matlab function *fseminf*, imposing in it the equalities (13) as constraints, the semi-infinite inequalities (23) and optimizing the function (17). If the position of the obstacle is such that there is no feasible joint path, as expected, the above algorithm cannot find a solution. Else, the above methodology always yields a solution. This may not be global, but it is sufficient for our needs, as primarily we are interested in finding a feasible motion, and only secondly in minimizing the applied torques, as implied by the objective function in (17).

IV. Application Examples

To illustrate the developed methodology, we consider a three link mechanism (see Fig. 2, with $n = 2$). The system parameters used are displayed in Table 1.

TABLE I. System Parameters

Body	l_i $[m]$	r_i $[m]$	m_i $[kg]$	I_i $[kgm^2]$
0	0.2	0.2	3.0	0.16
1	0.2	0.2	2.5	0.107
2	0.15	0.15	1.0	0.053

As mentioned above, the duration of motion can be chosen arbitrarily. A normal value for time duration strongly depends on the mechanism parameters. If an athlete was asked to make a similar task, the time duration must be chosen, keeping in mind the maximum initial y-velocity that he or she can obtain. For the mechanism implemented above, a good interval for flight time duration is between $0.4s$ and $1.2s$.

A. New Configuration.

Here the initial system configuration is $(\theta_0, q_1, q_2)^{in} = (108.97^o, -108.97^o, 90^o)$ and the final one is $(\theta_0, q_1, q_2)^{fin} = (20^o, 70^o, 56.73^0)$, with initial joint velocities $\dot{\mathbf{q}}^{in} = (0.08, -0.08) rad/s$ and $\dot{\mathbf{q}}^{fin} = (0.01, -0.01) rad/s$. The approach presented in Section III.A., is employed here to specify the desired path. We assign a sinus function of the form (12) with a seventh order polynomial of t as argument, for both q_1 and q_2, i.e. we assume initially that we have $k_1 = 7$ and $k_2 = 7$, see (16). The parameters are calculated using initial and final positions of joints, with zero initial and final velocities and accelerations and solving by optimization the nonlinear equations, given by (13) and (14). The objective function is given from (17) for $n = 2$. The flight time duration is chosen 1.1 s. Increasing or decreasing this time has no effect on the reachability of the

desired position, but increases or decreases the torque requirements and the magnitude of initial y-velocity. The angular momentum is computed using (11), yielding $h = 0.13\ kgm^2s^{-1}$. Fig. 3 depicts snapshots of the mechanism motion, while Fig. 4 shows system trajectories.

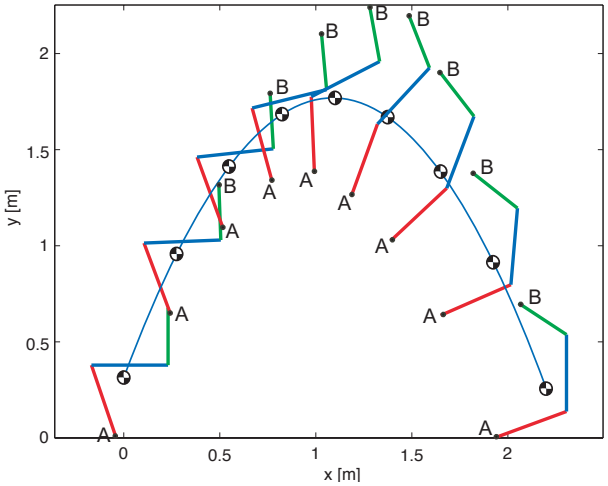

Fig. 3. Snapshots of a 3 link mechanism moving to a desired (θ_0, q_1, q_2).

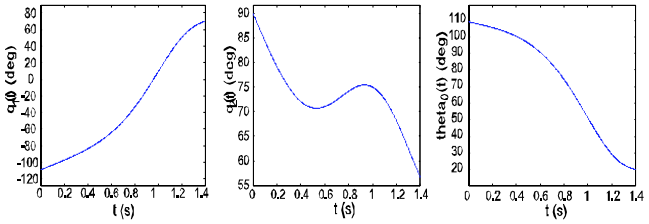

Fig. 4. Configuration variables that correspond to snapshots in Fig. 3.

As shown in Fig. 4, the desired configuration is reached in the specified time. Also, all trajectories are smooth throughout the motion, and the system starts and stops smoothly at zero velocities, as expected. This is an important characteristic of the method employed, and is due to the use of smooth functions, such as sinus with polynomial arguments. The corresponding joint torques are given in Fig. 5. These torques are computed using (3) and the elements of the reduced inertia matrix are given in the Appendix.

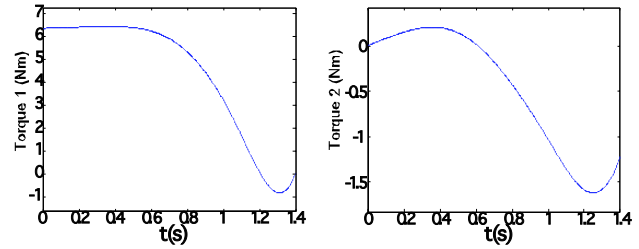

Fig. 5. Robot joint torques required for the motion shown in Fig. 2.

As shown in Fig. 5, the required torques are small and smooth and, as a consequence, they can be applied easily making the joint space path feasible.

B. Obstacle Avoidance.

In this case the mechanism employed in the previous example is required to make a similar flight and avoid an obstacle that is mounted close to its CM orbit. More specifically, the mechanism should reach the final configuration $(\theta_0, q_1, q_2)^{fin} = (-\pi/2^0, 0^0, 0^0)$ from the initial one $(\theta_0, q_1, q_2)^{in} = (72^0, 18^0, -45^0)$ with $\dot{\mathbf{q}}^{in} = \dot{\mathbf{q}}^{fin} = 0$, and avoid an obstacle with radius $R = 20\ cm$ mounted in such a position that the minimum distance from system CM is $20\ cm$. This is not a trivial task, because links have lengths $40\ cm$, $40\ cm$, and $30\ cm$ respectively. The flight time is chosen equal to $1.4s$ while the initial angular momentum is computed as $h = -1.17\,kgm^2s^{-1}$. In order to solve this parametrically constrained optimization problem, we have to define a grid in (u,t) space and find the minimum solution for every point of this grid. Here this grid was made with time step $0.1\ s$ and a length step $1\ cm$. One may support that, due to this discretization, obstacle avoidance is not guaranteed for every t and every u. To address this issue and increase the method's reliability, we use the expression $R+\varepsilon$ instead of R in (23), where ε is a small positive number. In this implementation, this number was chosen equal to $2\ cm$. Hence, we can conclude that the mechanism cannot reach in distance less that $2\ cm$ from the obstacle's surface at each point of the grid and, as we can realize, it is rather impossible to cover this distance in a point outside of the grid. This is because the grid is very dense and the coefficient ε relatively large. Fig. 6 shows snapshots of the mechanism motion.

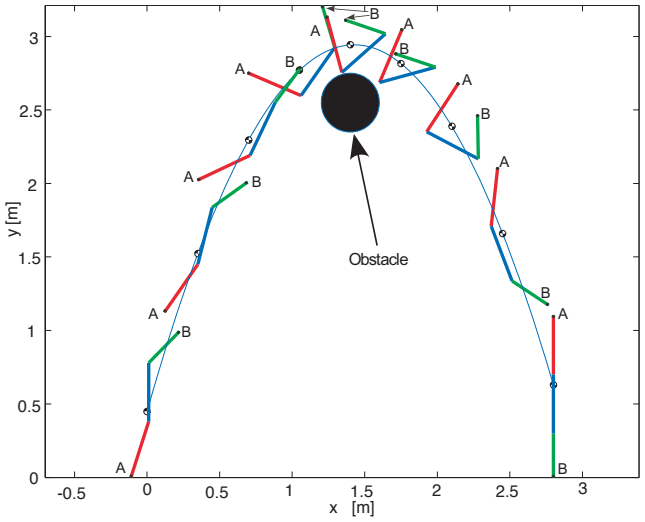

Fig. 6. Snapshots of a 3 link mechanism moving to a desired (θ_0, q_1, q_2) while avoiding an obstacle.

The resulting path leads to the desired final configuration, in the specified time. It is worth noting that without the use of (23), the obstacle would not have been avoided, see Fig. 7. In this figure, the black mechanism configuration (A^*B^*), which is the outcome of planning without using (22), results in a collision with the obstacle, while the non-stared configurations avoid the obstacle. Smooth trajectories that correspond to the motion in Fig. 6 are shown in Fig. 8. These were required to have zero initial and final joint velocities and accelerations. The required

torques are also smooth and applicable. Fig. 9 shows their time evolution and compares them with the ones needed in the absence of the obstacle.

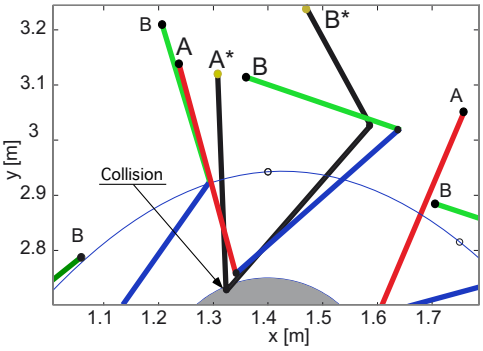

Fig. 7. The black mechanism cannot avoid collision with the obstacle.

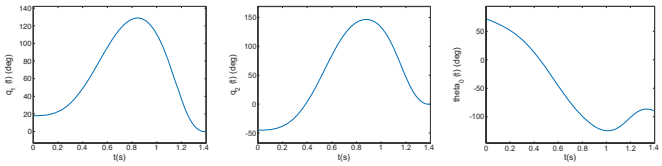

Fig. 8. Configuration variable trajectories that correspond to Fig. 6.

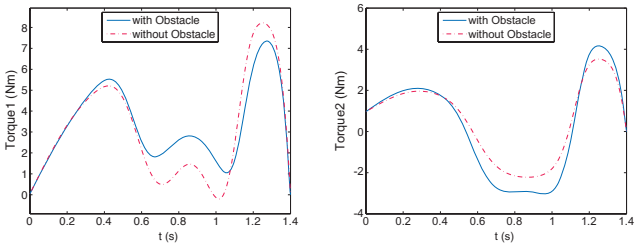

Fig. 9. Robot joint torques for the cases with and without obstacle.

V. Conclusions

A technique that allows simultaneous control of the joint angles and the orientation of a n link mechanism has been developed. The mechanism undergoes a ballistic motion and has initial angular momentum. The main characteristics of this method are the appropriate choice of initial angular momentum and the use of optimization techniques to determine the parameters of the problem. Furthermore, an obstacle avoidance method was developed. Both of the methods were implemented on a three link mechanism and the results have shown that required torques are applicable even in the case of an obstacle.

References

[1] J. K. Hodgins and M. H. Raibert, "Biped gymnastics", *International Journal of Robotics Research*, 9(2), pp. 115—132, 1990.

[2] M. W. Spong, "The swing up control problem for the Acrobot," *IEEE Control Syst. Mag.*, vol. 15, no. 1, pp. 49–55, Feb. 1995.

[3] T. Mita, S. Hyon, and T. Nam, "Analytical time optimal control solution for a two link planar acrobot with initial angular momentum," *IEEE Trans. on Robotics & Automation,* v. 17, no. 3, pp. 361–366, 2001.

[4] Grizzle, J.W., Moog, C., et Chevallereau, C., "Nonlinear Control of Mechanical Systems with an Unactuated Cyclic Variable," *IEEE Tr. on Automatic Control,* vol. 50, no. 5, pp. 559 – 576, May 2005.

[5] Godhavn, J., Balluchi, A., Crawford, L., and Sastry, S., "Steering of a class of nonholonomic systems with drift terms," *Automatica*, vol. 35, pp. 837–847, 1999.

[6] Kamon, M., and Yoshida, K., "3D attitude control methods for free-flying dynamic system with initial angular momentum," *J. Robot. Soc. Japan,* vol. 16, no. 2, pp. 223–231, 1998.

[7] Yokoyama, N., Hyon, S., Emura, T., and Suzuki, M., "Control of planar four-link gymnastic robot based on target dynamics," *SICE Tohoku Chapter 213rd Workshop,* 2003.

[8] Kane, T., and Scher, M., "A dynamical explanation of the falling cat phenonenon," *Int. J. Solid Structures,* Vol. 5, pp.663-670, 1969.

[9] Papadopoulos, E. and Dubowsky, S., "On the Nature of Control Algorithms for Free-floating Space Manipulators," IEEE Transactions on Robotics and Automation, Vol. 7, No. 6, Dec. 1991, pp. 750-758.

[10] Tortopidis, I. and Papadopoulos, E., "Point-to-Point Planning: Methodologies for Underactuated Space Robots," *Proc. IEEE International Conference on Robotics and Automation (ICRA '06),* May 2006, Orlando, FL, USA, pp. 3861-3866.

[11] Brosowski B., *Parametric semi-infinite optimization*, Verlag Peter Lang, Frankfurt, 1982.

Appendix

To compute the joint torques from (1), we substitute the first equation in the rest and we get a system with n equations:

$$\mathbf{H}(\bar{\mathbf{q}})\ddot{\bar{\mathbf{q}}} + \mathbf{C}^*(\bar{\mathbf{q}},\dot{\bar{\mathbf{q}}})\dot{\bar{\mathbf{q}}} + \mathbf{G}^*\left(\mathbf{q},h\right) = \boldsymbol{\tau} \tag{A.1}$$

In the examples, $n=2$, so $\bar{\mathbf{q}} = [q_1, q_2]^T$, $\boldsymbol{\tau} = [\tau_1, \tau_2]^T$, and,

$$\begin{aligned} H_{11} &= d_4 + 2d_6 c_2 + d_7 - (D_1 + D_2)^2 / D_0 \\ H_{12} &= H_{21} = d_7 + d_6 c_2 - D_2(D_1 + D_2)/D_0 \\ H_{22} &= d_7 - D_2^{\,2} / D_0 \end{aligned} \tag{A.2}$$

$$\begin{aligned} \mathbf{G}^* = &-g_0[(l_1 m_1 + l_2 m_2)\mathrm{c}(\theta_0 + q_1),\ l_2 m_2\,\mathrm{c}(\theta_0 + q_1 + q_2)]^T \\ &+ \frac{h^2}{2}\frac{\partial D_0^{-1}}{\partial \mathbf{q}} \end{aligned} \tag{A.3}$$

where $c_i = \cos(q_i)$, $c_{12} = \cos(q_1 + q_2)$, and the terms D_i, d_i are given by,

$$\begin{aligned} D_0(q_1,q_2) &= (d_1 + d_4 + d_7) + 2(d_2\,\mathrm{c}_1 + d_6\,\mathrm{c}_2 + d_3 c_{12}) \\ D_1(q_1,q_2) &= (d_4 + d_7) + d_5 c_1 + (d_6 + d_8)c_2 + d_9 c_{12} \\ D_2(q_1,q_2) &= d_7 + d_8 c_2 + d_9 c_{12} \end{aligned} \tag{A.4}$$

$$\begin{aligned} d_1 &= \mathrm{I}_0 + m_0(m_1 + m_2)r_0^2 / M \\ d_2 &= d_5 = m_0 r_0((m_1 + m_2)l_1 + m_2 r_1)/M \\ d_3 &= d_9 = m_0 r_0 m_2 l_2 / M \\ d_4 &= \mathrm{I}_1 + (m_0 m_1 l_1^2 + m_1 m_2 r_1^2 + m_0 m_2 (l_1 + r_1)^2)/M \\ d_6 &= d_8 = m_2 l_2 (m_0(l_1 + r_1) + m_1 r_1)/M \\ d_7 &= \mathrm{I}_2 + m_2(m_0 + m_1)l_2^2 / M \end{aligned} \tag{A.5}$$

where,

$$M = m_0 + m_1 + m_2 \tag{A.6}$$

Coefficients A_i in (6) for $n=2$ are given by,

$$\begin{aligned} A_0 &= l_0 m_0 + L_0(m_1 + m_2)/M \\ A_1 &= l_1 m_1 + L_1 m_2 / M \\ A_2 &= l_2 m_2 / M \end{aligned} \tag{A.7}$$

and all variables in (A.5) - (A.7) are defined in Fig. 2.