Simulated Framework for the Development and Evaluation of Redundant Robotic Systems

Ioannis Iossifidis
Computer Science Insitute, Ruhr West University of Applied Sciences, 45470 Mülheim an der Ruhr, Germany

Keywords: Simulated Environment, Redundant Manipulator, Closed Form Solution, Kinematics.

Abstract: In the current work we present a simulated environment for the development and evaluation of multi redundant open chain manipulators. The framework is implemented in Matlab and provides solutions for the kinematics and dynamics of an arbitrary open chain manipulator. For a anthropomorphic trunk-shoulder-arm configuration with in total nine degree of freedoms, a closed form solution of the inverse kinematics problem is derived. The attractor dynamics approach to motion generation was evaluated within this framework and the results are verified on the real anthropomorphic robotic assistant Cora.

1 INTRODUCTION

Focusing on the movement of arcticulated robots as part of autonomous acting robotic system, reaching and grasping are still demanding tasks with increasing complexity with respect to it's generalization properties. The performance depends on the underlying kinematical structure, the solution space of the invers kinematical mapping and the trajectory generating approach. Since both, the invers kinematics and the trajectory generation, depends on the underlying kinematical structure the manipulator has to be designed carefully. The design process again should linked and evaluated regarding the solution space of the invers kinematical mapping and the proposed motion planning scheme until the overall system convergences to a desired state. Although implementation on real hardware is of mandatory importance for the proof of concepts, the operational overhead is a limiting factor for the development, optimization and assessment of robotic systems and models. Simulation is a means to to overcome those limitations.

We developed a framework consisting of an mathematical description of the kinematics and dynamics of an arbitrary open chain manipulator and a simulator programmed in Matlab, implementing the inverse kinematics, simulating and visualizing it's motion. In the current implementation we propose a human like kinematical structure with a multi redundant arm and additional degree of freedoms in the shoulder and the body, (for further discussion see (Hollerbach, 1984)) (Iossifidis et al., 2003)). The source code of the simulator is freely available on *www.cstlab.net*.

In the following we first describe the structure of the trunk-arm-system represented by it's forward kinematics. Second we sketch the closed form solution for the whole system describing in detail how the null space motion can be utilized to avoid obstacles or joint limits. And in the last section the implementations of the simulator and some experimental results are described.

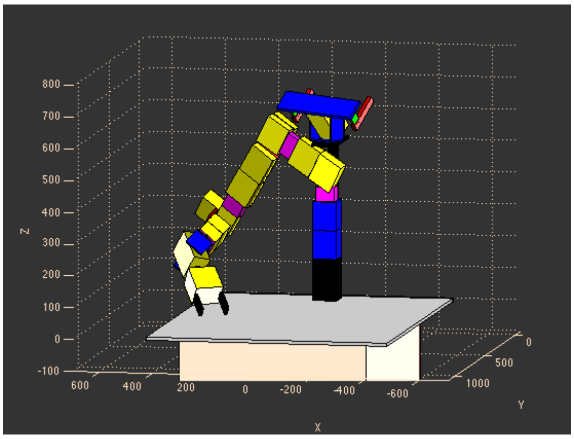

Figure 1: The MRobot-Simulator implements a robotic system consisting of an nine degree of freedom manipulator and a two degree of freedom sensor head.

Iossifidis I..
Simulated Framework for the Development and Evaluation of Redundant Robotic Systems.
DOI: 10.5220/0004802900550060
In *Proceedings of the 4th International Conference on Pervasive and Embedded Computing and Communication Systems* (PECCS-2014), pages 55-60
ISBN: 978-989-758-000-0

2 OPEN CHAIN MANIPULATOR

2.1 Initial Configuration

The sketched structure models (in correspondence to humans) a one degree of freedom trunk, a one degree of freedom shoulder elevation/depression with an attached seven degree of freedom arm. The arm consisting of a rotating trunk, spherical shoulder and wrist joints, and an elbow joint, for a total of nine degrees of freedom. The reference configuration is show in Figure 2.

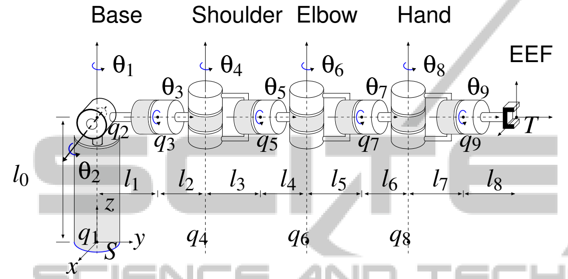

Figure 2: A total of eight revolute joints are ordered along the effector to simulate a human arm with trunk, shoulder, elbow and wrist.

The manipulator is composed of a series of *roll* and *pitch* joints. The combination of a *roll-pitch-roll*-joint is functionally equivalent to a spherical three DoF joint like the human shoulder or wrist. Similar to the example above, the twists are determined as

$$\omega_1 = \omega_4 = \omega_6 = \omega_8 = \begin{bmatrix}0\\0\\1\end{bmatrix}, \quad \omega_3 = \omega_5 = \omega_7 = \omega_9 = \begin{bmatrix}0\\1\\0\end{bmatrix}.$$

and

$$\omega_2 = \begin{bmatrix}0\\1\\0\end{bmatrix}$$

Support points can be chosen as

$$q_i = \begin{bmatrix}0\\ \sum_{k=1}^{i-1} l_k\\ 0\end{bmatrix} \qquad \text{for} \quad i = 1,4,6,8,$$

and

$$q_i = \begin{bmatrix}0\\ \sum_{k=1}^{i-1} l_k\\ l_0\end{bmatrix} \qquad \text{for} \quad i = 2,3,5,7,9.$$

This yields the twist coordinates

$$\xi_1 = \begin{bmatrix}0\\0\\0\\0\\0\\1\end{bmatrix}, \quad \xi_4 = \begin{bmatrix}l_1+l_2\\0\\0\\0\\0\\1\end{bmatrix}, \quad \xi_6 = \begin{bmatrix}l_1+l_2+l_3+l_4\\0\\0\\0\\0\\1\end{bmatrix},$$

$$\xi_8 = \begin{bmatrix}l_1+l_2+l_3+l_4+l_5+l_6\\0\\0\\0\\0\\1\end{bmatrix}, \qquad \xi_{2,3,5,7,9} = \begin{bmatrix}l_0\\0\\0\\0\\0\\1\end{bmatrix}.$$

The transformation between the base and end-effector coordinate frames in reference configuration is

$$g_{st}(0) = \begin{bmatrix} I & \begin{pmatrix}0\\ \sum_{i=1}^{9} l_i\\ l_0\end{pmatrix}\\ 0 & 1\end{bmatrix}.$$

2.2 Forward Kinematics

The forward kinematics is calculated as the product of exponential. On the basis of the simple ways of representing and calculating rigid transformations (for detailed description see (Murray et al., 1994)), we can move on to describe the transformation to the end effector of an open chain manipulator, that is, a chain of n joints. For each joint, we define a joint twist ξ_i that models the motion of the subsequent part of the robot.

Let $g_{st}(0)$ be the transformation between the base frame S of the manipulator and the end effector, or tool frame T in the reference configuration, that is, when all joint angles[1] $\theta_1,\ldots,\theta_n$ are set to zero.

Now we want to know what happens to the end effector after we apply a rigid transformation by moving a joint. Starting at the base, we get to the end effector via $g_{st}(0)$, which we then move by applying the rigid transformation $e^{\widehat{\xi}_i\theta_i}$. Combining these two transformation in the right order, we get

$$g_{st}(\theta) = e^{\widehat{\xi}_i\theta_i} g_{st}(0).$$

Here $g_{st}(\theta)$ maps the center of the base frame to the position of the end effector in the new configuration with $\theta_i \neq 0$. Thinking the other way round, it returns the S-coordinates of a point that we specify in T-coordinates. So if we specify "0", we refer to the position of the end effector in the frame T centered at that point. Applying $g_{st}(\theta)$ returns the coordinates of that point in frame S, centered at the base. Thus, we can think of $g_{st}(\theta)$ as enabling us to specify a point in a moving frame T, and get back its position in the motionless frame S.

To do the same for a configuration in which several joints have moved, i.e. $\theta_i \neq 0$ for more than one i, we just have to apply the appropriate transformation

[1] By abusive convention, we refer to the amount of translation θ_k of a prismatic joint k as *angle*, too, in order to prevent cluttering up our language.

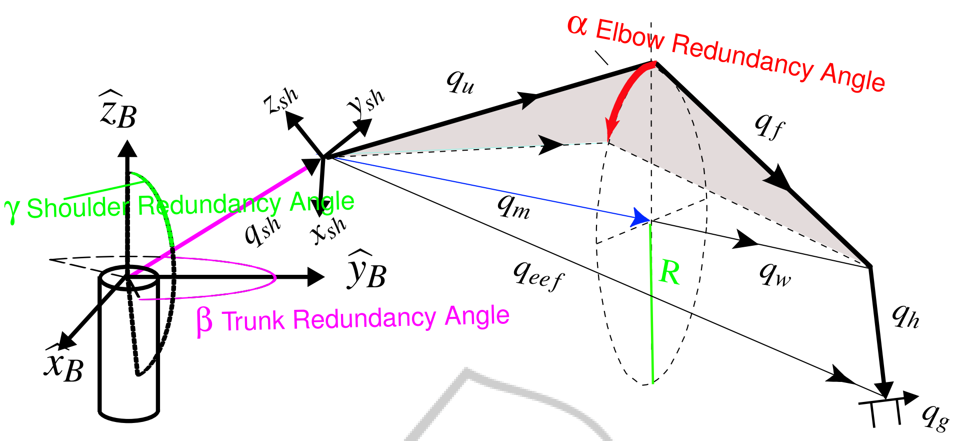

Figure 3: The figure depicts the kinematical structure and the associated coordinate systems of an multi redundant nine degree of freedom arm. α denotes the elbow angle, β denotes the trunk angle and γ the shoulder angle.

in the right order. Conceptually speaking, we start at the end effector and go up the joints along the manipulator till we reach the base, moving each joint by the appropriate angle θ_i on the way. This yields

$$g_{st}(\theta) = e^{\widehat{\xi}_1\theta_1} \cdots e^{\widehat{\xi}_n\theta_n} g_{st}(0) \qquad (1)$$

as the coordinate transformation from T to S in configuration $\theta = (\theta_1, \ldots, \theta_n)$.

Equation (1) is called the *product of exponentials* formula for the manipulator forward kinematics.

3 CLOSED FORM SOLUTION

The analytic solution for the inverse kinematics for an 9 DoF open chain manipulator will be descripted in current section.

For the solution of the overall system we assume that the shoulder is positioned due to the boundary condition determined by the object to be manipulated, like turning shoulder towards the object to extend it's workspace (distances to objects greater the the length of the arm) or to position the shoulder to an orientation that is orthogonal to the target direction (optimal grasping position). The shoulder elevation/depression is utilized to satisfy additional constraints, like the avoidance of undesired configurations.

Given the shoulder position, which is determined by the trunk angle and the shoulder angle, the inverse kinematic mapping of the seven degree of freedom has to be derived with respect to shoulder coordinate system.

In the following we derive first the inverse kinematics of the seven degree of freedom, exploiting it's geometric properties, and describe how the null space motion can be utilized to satisfy constraints like obstacle avoidance and joint limits.

3.1 Inverse Kinematics

Based on the work of (Kreutz-Delgado et al., 1990) and (Hollerbach, 1984)) the inverse kinematics problem for the nine degrees of freedom manipulator is solved in closed form. In the following derivation we denote the trunk angle with β and the shoulder angle angle with γ (correspond to θ_1 and θ_2 in the initial configuration in figure 2). We assume that the end effector position and hand orientation is determined by the object to be grasped and that the trunk angle β and shoulder angle γ will be choosen due to the boundary conditions and costraints of the given task. First we transform the given end effector position to the shoulder coordinate system by:

$$q'_{eef} = T^S_{sh} q_{eef}$$

with

$$T^S_{sh} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -(l_2+l_1) \\ 0 & 0 & 1 & -l_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(-\theta_1) & -\sin(-\theta_1) & 0 & 0 \\ \sin(-\theta_1) & \cos(-\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(-\theta_2) & 0 & \sin(-\theta_2) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(-\theta_2) & 0 & \cos(-\theta_2) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

beeing the homogeneous transformation.

Given the hand orientation θ_h (elevation) and ϕ_h (azimuth) and the tool point q'_{eef}, the vector q_h from the wrist to the hand reference tool-point (Fig. 3) is determined as

$$q_h = R_z(\phi_h) R_y(\theta_h) \widehat{e}_y l_h \qquad (2)$$

where R_x, R_z denote rotation matrices around the z- and y-axes and l_h denotes the segment length.

The redundant degree of freedom is defined by the *redundancy circle*, the center

$$q_m = \frac{|q_u|^2 - |q_f|^2 + |q_w|^2}{2\,|q_w|^2} q_w \qquad (3)$$

of which lies on a ray pointing from the shoulder to the wrist joint. The spatial position of the elbow lies

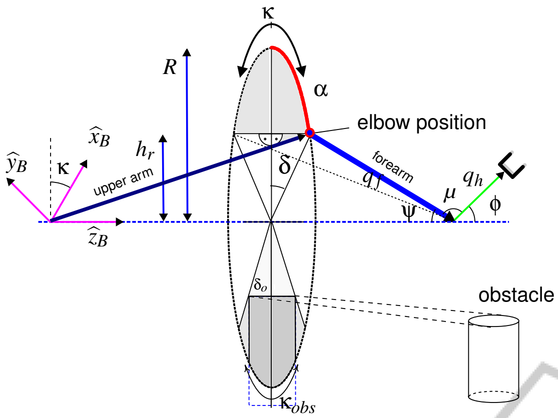

Figure 4: Geometrical construction to avoid joint configurations at the wrist and an obstacle reaching into the redundancy circle defines a to-be-avoided angular segment of that circle.

on this circle of radius r, with

$$r = \sqrt{|q_u|^2 - \left(\frac{|q_u|^2 - |q_f|^2 + |q_w|^2}{2\,|q_w|}\right)^2} \tag{4}$$

Expressing the wrist vector, q_w, through two angles, ϕ_w and θ_w, the elbow position can be written as

$$q_u = (R_x(\phi_w)R_z(\theta_w)R_x(\alpha)\hat{e}_y)\,r + q_m \tag{5}$$

where R_x and R_z are rotation matrices around the x- and the z-axis and the *redundancy angle* α characterizes the position of the elbow on the redundancy circle (Fig. 3). If α is specified, all limb vectors are known. A straightforward solution of the inverse kinematics determines the joint angles. $\theta_3, \theta_4, \theta_5, \theta_6, \theta_7, \theta_8, \theta_9$ (for more detail see (Iossifidis, 2013)).

3.1.1 Constraints for the Redundancy Circle

The redundancy angle α spans all redundant arm configurations consistent with the same tool position and orientation. The rate of change of the redundancy angle generates self-motion, which can be used to accommodate the two additional constraints of obstacle avoidance for the upper arm and joint limits at the wrist. To do that, we must compute which sectors on the redundancy circle are forbidden by these constraints.

Any obstacle o_i, that reaches up to the elbow, defines a to-be-avoided segment on the redundancy circle centered on κ_i and with an angular range of σ_i (see Fig. 4) Transforming limitations of joint angle range into constraints on the redundancy circle is more complicated. The most important joint angle limitation concerns the wrist, where the lower arm and the hand are spatially aligned. The angle μ between the hand q_h and the forearm q_f must be larger than $\pi/2$, which is true as long as

$$-(\pi - \mu) \leq \theta_8 \leq \pi - \mu. \tag{6}$$

Fig. 4 illustrates that this can be used to compute the corresponding allowed sector on the redundancy circle.

To do that, define a coordinate system the z'-axis of which is aligned with the shoulder-wrist-axis of the robot arm. If ϕ_w and θ_w describe the azimuth and elevation of the wrist, then the transformed hand vector, q_h' is:

$$q_h' = \mathcal{R}_y^{-\theta_w} \mathcal{R}_z^{-\phi_w} q_h \tag{7}$$

After the transformation, the redundancy circle lies in a plane parallel to the $x'y'$-plane, so that we may project q_h' onto that plane to obtain:

$$\kappa = \arctan(q_h^{y'}, q_h^{x'}) \tag{8}$$

as the center of the prohibited sector of the redundancy circle. The angular extent of the prohibited region $\kappa \pm \delta$ follows from equation 6:

$$\delta = \arccos(h_r/r) \tag{9}$$

where h_r, ψ, c and φ are auxiliary quantities, which can be derived from figure 4:

$$h_r = c \cdot \tan(\psi) \tag{10}$$

$$c = |q_w| - |q_m| \tag{11}$$

$$\psi = \pi - (\mu + \phi) \tag{12}$$

$$\phi = \arccos\left(q_h^{z'} / |q_h{}'|\right). \tag{13}$$

The obstacle induced prohibited region on the redundancy cycle is calculated analogous, with κ_o as the center of the prohibited region and δ_o as it's extend (figure 4).

4 IMPLEMENTATION AND RESULTS

The goal was to develop a simulator for an arbitrary open chain manipulator. In sections 2 and 3 we introduced the necessary formularism to calculate the kinematic and dynamic terms. With the help of these formulas it was possible to develop a simulator evaluating the proposed solutions and visualize the motion of the robotic system. The software architecture of this simulator shall be described briefly in this section and in the following we evaluate the attractor dynamics approach to motion generation performing target acquisition and obstacle avoidance.

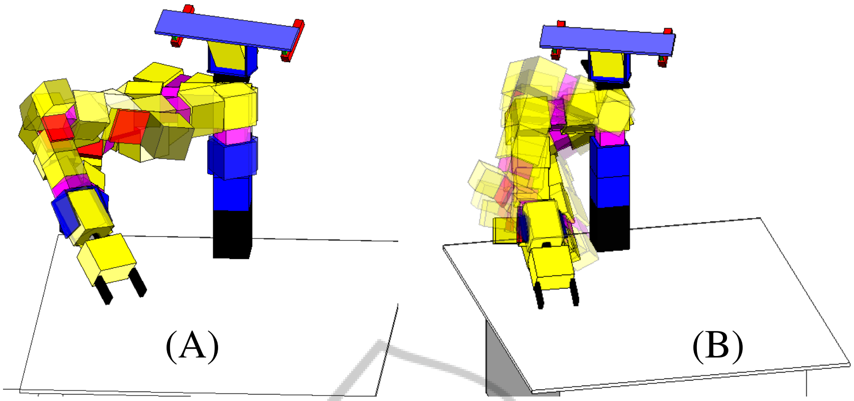

Figure 6: (A) shows how the shoulder angle varies between $-30°$ and $30°$. (B) The trunk angle varies from $-30°$ to $30°$. In both cases the position and orientation of the endeffector remains invariant.

4.1 MRobot

The MRobot-Simulator is an Matlab implementation of the presented nine degree of freedom manipulator. The simulator includes the constraints handling (see figure 5) as well as the shoulder and trunk motion (see figure 6). All parameters can be changed through the gui and the simulated arm motion will be updated immediately.

At this we would like to note that for a shoulder angle of 30 the presented manipulator correspond with the kinematical structure of one of the manipulators of the mobile robot Justin ((Borst et al., 2009)) and that the presented closed form solution solves it's inverse kinematics.

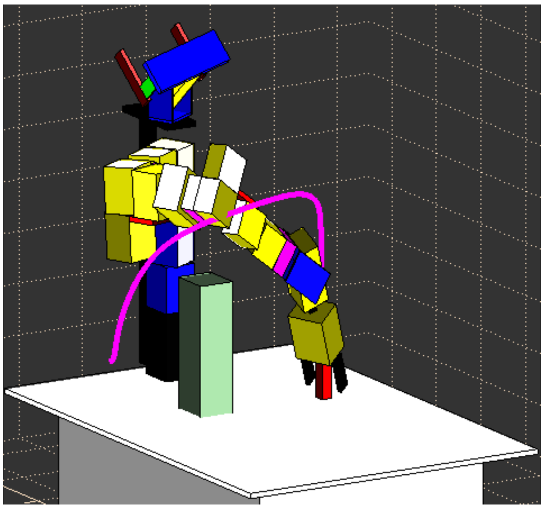

Figure 5: The robotic system utilizes redundancy to avoid obstacles.

4.2 EXPERIMENTS

In order to prove the functionality of the simulator we have implemented the attractor dynamics approach to motion generation (Iossifidis and Schöner, 2006) in order to realize goal directed trajectories and to evaluate the obstacles avoidance properties of the approach. Target and obstacle related parameters were picked at random.

The experimental results generated by the simulator for multiple target-obstacle-configuration led to a systematical classification of the performance of the approach and pointed out it's limitations. In addition we could identify which part in the closed form solution of the inverse kinematics lead to undesired configuration changes. The detailed discussion of this experimental results is beyond this paper and will be part of future publications.

Figure 7 illustrate one experimental run. After the motion is initiated the evolving trajectory point directly in the direction of the target (A). Once the end effector approaches the obstacle, repelling forces pushes the end effector away from the initial direction (deforming trajectory – magenta line) and the end effector follows a trajectory over the obstacle (B-C). As soon as the distance between the end effector and obstacle becomes larger, and the end effector leaves the obstacles area of influence, the trajectory relaxes again to the target direction. Finally the end effector reaches the target (D).

5 CONCLUSIONS

A simulated framework consisting of an mathematical description of the kinematics and dynamics for arbitrary open chain manipulator and a simulator pro-

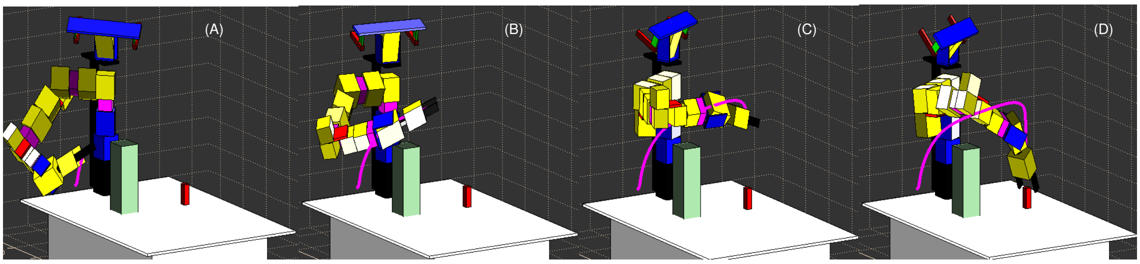

Figure 7: he figure shows the MRobot accomplishing a reaching task while avoiding obstacle. The magenta line depicts the trajectory of the end effector.

grammed in Matlab was introduced. Part of the framework is the implementation of a closed form solution for the inverse kinematical mapping for multi redundant manipulators (up to nine DoF). The framework provide ground truth data of all parameters for further analysis, an intuitive interface and a visualization front end for the whole robotic system. The software is freely available under `cstlab.net`.

REFERENCES

Borst, C., Wimbock, T., and Schmidt, F. (2009). Rollin'Justin-Mobile platform with variable base. In *Robotics and Automation*, 2009. ICRA'09. *IEEE International Conference on*, pages 1597—-1598. IEEE.

Hollerbach, J. (1984). Optimum kinematic design for a seven degree of freedom manipulator. In *2nd Int. Symp. Robotics Research*, pages 215 –222.

Iossifidis, I. (2013). Motion Constraint Satisfaction by Means of Closed Form Solution for Redundant Robot Arms. In *Proc. IEEE/RSJ International Conference on Robotics and Biomimetics (RoBio2013)*.

Iossifidis, I. and Schöner, G. (2006). Dynamical Systems Approach for the Autonomous Avoidance of Obstacles and Joint-limits for an Redundant Robot Arm. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 580–585. IEEE.

Iossifidis, I., Theis, C., Grote, C., Faubel, C., and Schöner, G. (2003). Anthropomorphism as a pervasive design concept for a robotic assistant. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, volume 4, pages 3465–3472. IEEE.

Kreutz-Delgado, K., Long, M., and Seraji, H. (1990). Kinematic analysis of 7 DOF anthropomorphic arms. In *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, pages 824 – 830.

Murray, R., Li, Z., and Sastry, S. (1994). A mathematical introduction to robotic manipulation. CRC Press.