Motion Planning of Mobile Multi-Limb Robotic Systems Subject to Force and Friction Constraints

Akhil Madhani
Steven Dubowsky
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

Multi-limb, mobile robotic systems will need to apply large forces over large ranges of motion. This paper presents the Force-Workspace Approach *which can be used to plan the activities of such systems allowing them to execute such tasks without violating actuation limits, frictional constraints with their environments, and joint range of motion limits. The technique is applied to a robotic climbing machine.*

1 Introduction

Mobile, multi-limb systems are being developed for space construction and repair, planetary exploration, toxic-waste clean-up, and other hazardous missions [1,2]. Their tasks will involve assembling structural members in space, excavating material, and moving heavy containers, see Fig. 1.

Fig. 1 A System Concept

A fundamental system requirement will be the ability to move through large ranges of motions while simultaneously applying large forces. Unlike conventional fixed base industrial manipulators, mobile systems will need to consider breaking handholds, losing footings, and overturning. In addition, their actuator capacities will be limited by weight and power efficiency requirements, particularly in the case of space systems. Planning motions to enable application of large static forces while moving through large ranges of motion subject to actuator constraints, contact force and moment constraints, and of course, kinematic constraints is an important problem that past studies have not yet addressed.

A mobile, multi-limb robotic system in contact with its environment comprises redundantly actuated closed kinematic chains, hence the actuator efforts and contact forces and moments required to support it and those required by its task are indeterminate. Related problems have been studied. For example, [3,4] have solved for contact forces and studied the nature of force-distributions independent of constraints for systems in specified configurations, [5,6] have optimized contact forces and actuator efforts based on frictional contact constraints independent of the mechanism which applies these forces, and [7,8,9,10] have solved for contact forces and actuator torques subject to both frictional contact constraints and actuator effort limits. These investigations were done within the context of multi-fingered robotic hands, and robotic walking machines. While most of these studies have focussed on systems in fixed configurations, some have treated systems with *pre-specified* motions. For example, [8,9] use methods to find smooth actuator efforts, and contact forces and moments along pre-specified motions. These studies provide tools and insights to treat problems such as resolving redundant actuator efforts without violating a system's constraints while fixed in a given configuration. This might be used, for example, to prevent an object from slipping from the grasp of a robotic hand. However, planning a system's motions so that it is able to perform tasks requiring large forces over substantial ranges of motion remains to be addressed; for example, how can a robotic device transport a heavy object with its "arms" while standing or climbing on a soft sandy hill?

This paper presents a method to solve this problem. The approach, the *Force-Workspace (FW)* approach, maps constraints into the system's *configuration space (C-space)* to form *constraint obstacles* in a similar vein to geometric C-space obstacles such as those described in [11,12]. This enables large system motions to be planned which consider, in a uniform way, system force capabilities, kinematic constraints, joint motion limits, as well as geometric obstacles in the environment. Fig. 2 shows schematically a two-dimensional FW parameterized by $[q_1, q_2]$. The FW consists first of the boundary of the kinematic workspace, the set of q_i's where the system can reach as constrained by kinematic limits excluding joint range of motion limits. Additional constraints are then mapped as constraint obstacles within the kinematic workspace. Motions which do not violate system constraints are those which do not intersect any of these obstacles.

The method is applied here to plan the motions of a three-limb, planar, climbing machine that must climb upwards between two vertical walls, pushing outwards to maintain frictional support. The machine uses two of its limbs to lift itself upwards, without letting its feet slip or saturating its actuators. Acceptable paths are found using the FW approach. These paths are used to generate gaits allowing the system to climb continuously.

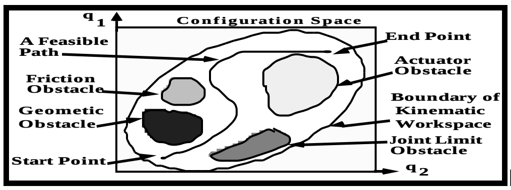

Fig. 2 The Force-Workspace Concept.

2 Generating the Force Workspace (FW)

2.1 System Description

This study assumes systems with rigid links and joints with no clearances or friction. The system's kinematic structure is assumed to consist of a single main body from which M serially actuated limbs extend to contact the environment or task; the contacts need not be fixed, but may be made with another kinematically constrained object such as a valve handle. The limbs may also extend freely from the system. Such a system forms a single mechanism with n degrees of motion freedom (DOF). This motion is represented by the n×1 vector $\underline{q}$. In addition to $\underline{q}$, a pose parameter, P, is required to completely describe the system's configuration. The system C-space is defined as the space parameterized by $\{\underline{q}, P\}$. External forces and moments supported or applied by the system are assumed functions of its configuration, and may act at any location on its structure.

2.2 The 2^n-Tree Representation of C-space

To generate the FW, the C-space is represented by a generalized quadtree or 2^n-tree, [13,14]. The nodes of the tree represent cells in an n dimensional space, and the root node represents the entire space under investigation. To generate the 2^n-tree, one or more node tests are required to determine if a node is entirely *feasible*, *infeasible*, or *mixed*, with respect to system constraints. If at all configurations within a node no constraints are violated, it is labelled feasible; if at all configurations within a node at least one constraint is violated, it is labelled infeasible; and if a node contains both feasible and infeasible configurations, it is labelled mixed. The problem of mapping system constraints into the C-space to generate the FW is then reduced to generating appropriate and effective node feasibility tests.

2.3 Force-Workspace Node Tests

Three node tests are used. The first is used to determine the kinematic workspace, the set of permissible system configurations which do not violate geometric constraints excluding joint limits and physical obstacles in the environment. A great deal of work has been done to address kinematic workspaces [15,16], and obstacle avoidance for manipulators [12,17] which can be applied to this problem. Limited work, however, has been done to address kinematic workspaces for multi-limb systems, [7,18]. Addressing this problem in the general case is beyond the scope of this paper. The second test is used to map joint limits in the C-space, and is in general not difficult to develop, [18]. The test third, which is the focus here, and is described below, is used to map actuator saturation limits and contact constraints as obstacles into the C-space.

The first step in this last test is to determine if a single point in the C-space is feasible, meaning it can support itself and its task loading without violating actuator effort or system-environment frictional constraints. This is done by extending a well-known linear programming technique, originally developed to specify actuator torques in redundantly actuated robotic hands [7]. First the equations for static equilibrium of the system are written in the form:

$$\mathbf{W}\underline{c} = -\underline{F} \qquad (1)$$

Each column of the 6×m matrix $\mathbf{W}$, $\underline{w}_i$, represents the screw coordinates formed from each of the three orthogonal components of force and the three orthogonal components of moment generated through each contact point between the system and its environment. These are forces and moments which support the system, such as forces between its feet and the ground. The m×1 vector $\underline{c}$ has elements c_i which represent the scalar intensities of each contact wrench. The 6×1 vector $\underline{F}$ is a wrench representing the sum of any fixed, specified forces and moments acting on the system used to perform its task. In general the system will be overconstrained, where rank($\mathbf{W}$) = 6 and the null space of $\mathbf{W}$, $N(\mathbf{W})$ exists. The contact wrench intensities can be found by:

$$\underline{c} = -\mathbf{W}^{+}\underline{F} + \mathbf{N}\underline{\lambda} \qquad (2)$$

where $\mathbf{W}^{+}$ is the right generalized inverse of $\mathbf{W}$, $\mathbf{N}$ is a m×dim($N(\mathbf{W})$) matrix whose columns form a basis for $N(\mathbf{W})$, and $\underline{\lambda}$ is an dim($N(\mathbf{W})$)×1 array which may be chosen arbitrarily and which determines how the components of $N(\mathbf{W})$ will be combined, [7]. These null space components produce what are often referred to as "internal forces" in the overconstrained system, [19,7]. Unisense contact constraints (feet and fingertips for example can push when contacting objects, but cannot pull), linearized coulomb friction constraints, and actuator effort limits can be written as linear inequality constraints on the intensities of the contact wrenches, c_i. These linear inequality constraints on the c_i can be mapped into a space parameterized by the elements of $\underline{\lambda}$ in equation (2) to form a constraint polygon. If the largest possible circle, or in general hypersphere, is inscribed within the constraint polygon, its center, $\underline{\lambda}^*$, will be a maximum distance from the nearest constraint planes, and its radius, d^*, will be this distance. These can be solved for via a modification of the linear programming approach presented in [7], shown in [18].

In this study, this technique is used to determine the feasibility of a configuration by noting that if d^* is greater than zero, a configuration will be feasible, and if d^* is less than zero, it will be infeasible. This can be extended to determine the feasibility of all configurations within a C-space node cell. The node test begins by selecting the center point of a node and testing its feasibility using the above linear programming method. If the center point is feasible, ($d^* > 0$), a non-linear programming method is used to minimize d^* over $\underline{q}$, subject to the linear constraints: $\underline{q}$ node cell. If the resulting $d^*_{min} < 0$, the node is *mixed*, and if the resulting d^*_{min} is > 0, the node is *feasible*. If the center point of the node is infeasible, ($d^* < 0$), then instead of being minimized, d^* is maximized over $\underline{q}$, subject to the linear constraints: $\underline{q}$ node cell. If the resulting $d^*_{max} > 0$, the node is *mixed*, and $d^*_{min} < 0$, the node is *infeasible*. Fig. 5 shows examples of feasible and mixed node cells.

3. Motion Planning in Force-Workspace

Once the FW has been generated, motions can be planned which avoid constraint obstacles and hence do not violate system constraints. To automatically generate such paths, the feasible nodes of the 2^n-tree representation of the FW are transformed into a search graph whose edges represent the physical adjacency relationships between all feasible cells in the FW. By weighting the edges of the search graph with a configuration based performance criteria and using a minimum cost graph search, feasible paths can be planned and also tailored to suit particular applications [18].

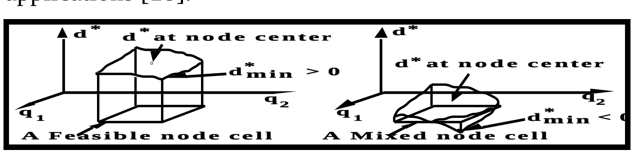

Fig. 3 Feasible and Mixed cells in a 2–D F-W.

When a new set of contact locations is chosen, for example while making a step with a walking machine or while turning a valve "hand-over-hand", the system forms a new mechanism and a new force-workspace will be generated to plan its motions. We define a *stance* as the system and a particular set of contact locations which define this mechanism. Hence relocating contacts requires transferring between stances, or equivalently between force-workspaces. During this process, we must ensure that system constraints are not saturated. Fig. 4 shows conceptually how this can be done. The system begins at configuration A in stance 1. To perform a step it must shift to the next stance in the gait. To do this the configuration of the system in the current stance must correspond to a feasible one in the next stance [18]. If it does not, the system must move in its current stance to a configuration which is feasible in both stances, such as configuration B in Fig. 4. The path for this motion must clearly be in the clear force-workspace for the current stance, avoiding any obstacles. At point B it possible to transfer to stance 2 at configuration C. In a similar manner, the system may proceed to point D and then transfer to point E in stance 3, and the gait can continue.

In section 4.3, we give the criteria which must be satisfied in order for transfers between stances and force-workspaces to occur successfully for the example of a planar climbing robot, including restrictions on choices of new contacts and system configurations while performing this transfer.

4.An Application—A Climbing Robot

The FW approach is applied to a planar, three-limb, climbing robot whose task is to climb upwards between two walls by pushing outwards against them in order to generate frictional support, as shown in Fig. 5. The system comprises six links, two in each limb, and five actuated revolute joints. The "body" carries two actuators which generate torques between the limb 1 and limb 2 shoulder joints relative to limb 3, which is rigidly attached to the body. The body carries a payload. The limbs are assumed weightless compared to the body and payload. The contacts made with the walls are point contacts with friction which do not support moments between the limb tips and the walls, but which support coulomb friction forces of the form $F_T \leq \mu F_N$ where F_N is a normal contact force and F_T is a tangential contact force.

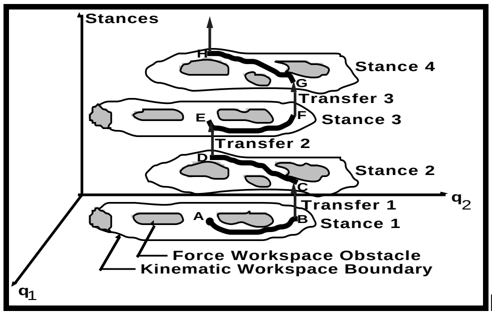

Fig. 4. Transferring Between Force-Workspaces

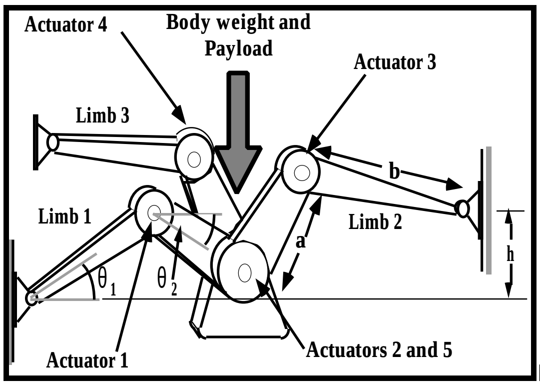

Fig. 5. A Planar Climbing Robot

The FW is used to find motions that will allow the system to climb upwards, and to generate a gait which allows continuous climbing. Not all motions will succeed since, in some configurations, the system will violate one or more of its friction or actuator constraints. As described in Fig. 6, the gait found uses two stances which breaks the problem into two subtasks. First, the system uses limb 1 and limb 2 to lift its body upwards; this is referred to as a two-limb pull-up. Then, the third limb, limb 3, is placed at a new contact and the supporting loads are transferred to limb 3, allowing limb 1 to be lifted from the wall. The payload can then continue being lifted using a new pair of limbs, limb 3 and 2, and the process repeated to generate a gait. The FW results for this motion are developed below.

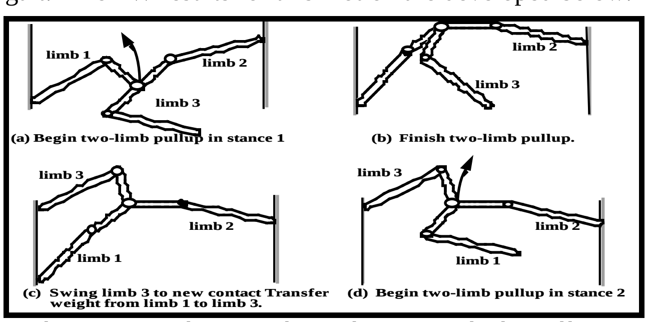

Fig. 6 A Gait Based on the Two Limb Pullup.

4.1 The Force–Workspace

The configuration space of the system is parameterized by $\{\mathbf{q} = [\theta_1, \theta_2]^T, P = 1,2\}$. Fig. 7 shows the FW in $\underline{\theta}$–space for $P = 1$, (right limb elbow-down configuration) for the system parameters given in Table 1. The dark grey cells in Fig. 7 represent configurations outside the kinematic workspace (KW), the medium grey cells are constraint obstacles within the KW where either actuator torques or wall frictional constraints are violated, and the light grey cells represent feasible configurations. Joint limit obstacles are not shown for clarity, see [18]. The white cells are mixed. Note that a *merge* operation was performed on the final map after the subdivision process was completed. This process merges smaller cells of the same feasibility into a single cell, explaining why there are white cells which are larger than the minimum cell size in Fig. 7. Also note that after the subdivision process, all feasible cells were further subdivided into small, equal sized nodes in order for the path planner to give smoother paths. This subdivision process requires no node tests and has a trivial computational expense.

Table 1 System Parameters

Parameter	Value	Description
$\lVert\mathbf{F}\rVert$	300 N	External wrench on "body"
$\tau_{1,3,max/min}$	±200 Nm	Joint torque limits
$\tau_{2\ max/min}$	±300 Nm	Joint torque limits
a, b	0.5 m	Link lengths
d	1.0 m	Wall Separation
h	0.4 m	Right contact "step height"
μ	0.8	Coef. of friction at walls

An intuitive representation of the FW is obtained by mapping all locations which can be reached by the body into world space, or X–Y space, as shown in Fig. 8. The variables X and Y are the coordinates of the center of the body as shown in the figure. A single X–Y space map for $P = 1$, and where the left limb is in an elbow-down configuration is shown. The two-limb system is superimposed on the force-workspace and is shown in a feasible configuration, since the body lies within the feasible force-workspace region.

Fig. 7. The FW in [_1, _2]–Space

4.2 Planning a Two–Limb Pull-up.

A path giving the minimum distance travelled by the center body between points A and B is shown in Fig. 8. This path avoids the large constraint obstacle in the left of the FW. Fig. 9 shows the resulting actuator torques, calculated using the method of [7], and the ratios of tangential to normal contact forces along the path which must be less than the coefficient of friction of 0.8.

Recall the key parameter used to determine if a configuration is feasible is the distance to the nearest constraint in $\underline{\lambda}$-space, the variable d^*. It can be shown for this example that two constraints will be active in determining d^*, and that the constraint obstacle circumscribed by the path in Fig. 12 is due to the actuator 1 torque limit, and the left wall friction constraint [18]. Fig. 9 shows that both the actuator 1 torque, and the ratio of the tangential force to the normal force at the left contact (which is limited by a coefficient of friction of 0.8) both lie very near their limits as the path circumscribes the constraint obstacle. An alternative weighting criteria which changes this behavior, keeping these values away from their limits is presented in [20].

4.3 Planning a Gait

The concept shown in Fig. 4 is used to plan the system's gait. After using limb 1 and limb 2, which form stance 1, to lift the body upwards, the next step in planning the gait is to select a contact location for limb 3 and to find a system configuration in which the load can be successfully transferred from limb 1 to limb 3, as shown in Fig. 6. The process of contacting a wall with limb 3 and transferring weight to it requires a foot force transfer from one stance to another. For both stances (limbs 1-2, and 3-2), FW's will be generated to plan two-limb pull-ups before and after this transfer. Clearly, the system must reside in a feasible configuration within each FW before and after the transfer. It can be shown that in addition to being a necessary condition to execute the foot force transfer, this is also a sufficient condition [18]. It follows that a test to determine if a new contact location for limb 3 exists is that an intersection of feasible regions exists when the stance 1 and stance 2 FW's are overlapped.

Fig. 8. The FW in X–Y Space

Fig. 10 shows a gait generated using the FW of Fig. 8. The dark grey region in Fig. 10 (b) is the feasible intersection of the FW's corresponding to each pair of limbs. Since such a feasible intersection exists, the chosen limb 3 contact point is feasible. The condition on the system configuration for the foot force transfer to occur is that the body lie within this intersection. Once contact forces are shifted to stance 2, the unloaded limb, limb 1, can be lifted and planning can continue within the new FW. Continuing cyclically in this manner produces a gait, with two steps of the resulting body motion shown in Fig. 10.

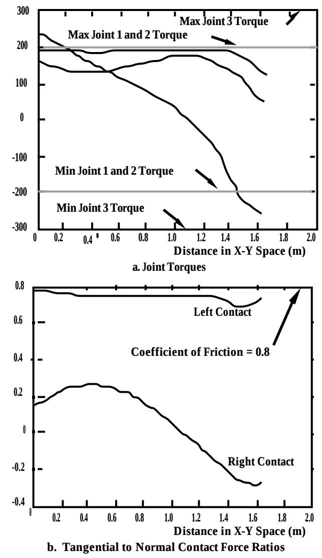

Fig. 9 Torques and Contact Force Ratios

5. Conclusions

A method has been presented to generate motions for a class of multi-limb robotic systems enabling them to apply large static forces over large ranges of motion without saturating actuator effort limits, system-environment friction constraints, kinematic joint limits, or geometric workspace obstacles. The approach, termed the *Force-Workspace (FW)* approach, maps these constraints into the system C-space to form *constraint obstacles* using a recursive subdivision process. To generate motions along which actuator efforts can be specified without violating system constraints, paths are planned that avoid these constraint obstacles. The method permits the shape of the paths to be controlled using any configuration dependent performance criteria. The FW approach was applied to a proposed three-limb planar climbing robot whose task is to climb upwards between two vertical walls by pushing outwards to generate frictional support. Motions were planned automatically within the system force-workspace enabling it to lift itself upwards using two limbs at a time, and a gait was planned to enable it to switch limbs and climb continuously. As a final note, the FW approach is currently being used as a tool to design an experimental version of the wall climbing robot [18].

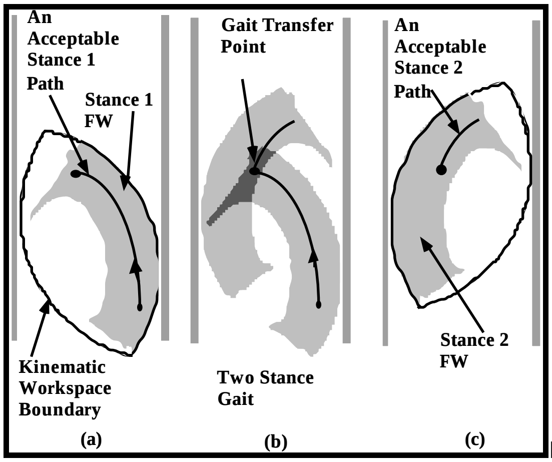

Fig 10. A FW Generated Gait.

6. Acknowledgements

Support of the NASA Langley Research Center, Automation Branch under Grant NAG–1–801, and through the MIT Department of Mechanical Engineering Warren M. Rohsenow Graduate Fellowship is acknowledged.

7. References

[1] Meieran, H. B., “Mobile Robots Responding to Toxic Chemical Accidents”, *Proc. of the Fourth ANS Topical Meeting on Rob. and Remote Systems*, Albequerque, NM, Feb 1991, pp 383-391

[2] NASA, *Proceedings of the NASA Conference on Space Telerobotics*, JPL Publication 89-7, January, 1989.

[3] Yoshikawa, T., and Nagai, K., “Manipulating and Grasping Forces in Manipulation by Multifingered Robot Hands,” *IEEE Trans. on Rob. and Autom.*, vol. 7, no. 1, Feb. 1991.

[4] Kumar, V. R. and Waldron, K. J. “Force Distribution in Closed Kinematic Chains,” *IEEE J. of Rob. and Autom.*, vol 4. No. 6, Dec 1988, pp 657-664.

[5] Demmel, J. and Lafferriere, G., “Optimal Three Finger Grasps,” *Proceedings of the IEEE Int. Conf.* on Rob. and Autom., Scottsdale, AZ, May 1989, pp 936-942.

[6] Ji, K. and Roth, B., “Direct Computation of Grasping Force for Three-finger Tip-prehension Grasps,” *In Trends and Develop. in Rob*., pp 125-138, ASME pub. DE-vol 15-3, 1988.

[7] Kerr, J., and Roth, B., “Analysis of Multifingered Hands,” *Int. J. of Rob. Res*., vol. 4, no. 4, 1986, pp 3-17.

[8] Klein, C. A. and Kittivatcharapong, S, “Optimal Force Distribution for the Legs of a Walking Machine with Friction Cone Constraints,” *IEEE Trans. on Rob. and Autom*., vol 6, no. 1, Feb. 1990, pp 73-85.

[9] Nahon, M. A., and Angeles, J., “Optimization of Dynamic Forces in Mechanical Hands,” *ASME J. of Mech. Des*., vol 113, June 1991, pp 167-173.

[10] Orin, D. E. and Oh, S. Y., “Control of Force Distribution in Robotic Mechanisms Containing Closed Kinematic Chains,” *ASME J. of Dyn. Syst., Meas., and Cont.*, vol 102, June 1981, pp 134-141.

[11] Brooks, R. A. and Lozano-Perez, T., “A Subdivision Algorithm in Configuration Space for Findpath with Rotation,” *Proc. of the Eighth Int. Joint Conf. on Art. Intel*., Karlsruhe, August, 1983, pp 799-806.

[12] Lozano-Perez, T, “A Simple Motion-Planning Algorithm for General Robot Manipulators,” *IEEE J. of Rob. and Autom.*, vol RA-3, no 3, June 1987.

[13] Samet, H., Design and Analysis of Spatial Data Structures, Addison Wesley, 1990.

[14] Paden, B., Mees, A., and Fisher, M., “Path Planning Using a Jacobian-Based Freespace Generation Algorithm,” *Proceedings of the IEEE Int. Conf. on Rob. and Autom*. Scottsdale, AZ, May 1989, pp 1732-1737.

[15] Kumar, A., and Waldron, K. J., “The Workspaces of a Mechanical Manipulator,” *ASME J. of Mech. Des.*, vol. 103, July 1981, pp 665-672.

[16] Yang, D. C. H. and Lee, T. W. “On the Workspace of Mechanical Manipulators,” *ASME J. of Mech., Trans., and Autom*. in Des. vol. 105, Mar 1983, pp 63-69.

[17] Lozano-Perez, T. and Taylor, R. H., “Geometric Issues in Planning Robot Tasks,” *In Robotics Science*, Brady, J. M, editor, MIT Press, Cambridge, MA, 1989.

[18] Madhani, A., Design and Motion Planning of Robotic Systems Subject to Force and Friction Constraints, S.M. Thesis, Dept. of Mech. Eng., MIT, August, 1991.

[19] Salisbury, J. K., *Kinematic and Force Analysis of Articulated Hands*, Ph.D. Thesis, Dept. of Mech. Eng., Stanford, July 1982.

[20] Madhani, A. and Dubowsky, S., "Design and Motion Planning of Multi-Limb Robotic Systems: The Force Workspace Approach," Submitted, *ASME J. Mech. Des.*