ABSTRACT

A method, called the *Force Workspace (FW)* approach, is presented for the design and motion planning of mobile multi-limb robotic systems which must as part of their tasks apply large forces over large ranges of motion. These systems will be limited by actuator force / torque saturation limits, system-environment force and moment constraints such as friction, and kinematic joint limits. The FW can be used for the quantitative design of such systems and enables motions to be planned to perform tasks without violating the above constraints. Examples of FW based design and planning are given for a robotic climbing machine.

1. INTRODUCTION

Future mobile multi-limb robotic systems will often need to move through large ranges of motions while simultaneously applying large forces, (Meieran, 1991 and NASA, 1989), see Figure 1. Unlike conventional fixed-base systems, mobile systems must consider breaking handholds, losing footings, or overturning. Furthermore mobile systems will have limited actuator force / torque capacities due to weight and power consumption requirements. The need for umbilicals would be highly undesirable, particularly for space systems.

A mobile, multi-limb robotic system in contact with its environment will comprise redundantly actuated closed kinematic chains, see Figure 1. The actuator efforts and the contact forces and moments required to support the system and those required by its task are coupled and in general will be mathematically indeterminate. This complicates both design and planning for such systems. Related problems have been the subject of much research. For example, contact forces and the nature of force-distributions, independent of constraints, have been studied for systems in specified configurations, (Yoshikawa and Nagai, 1991 and Kumar and Waldron, 1981).

1 - Student Member, ASME
2 - Fellow, ASME

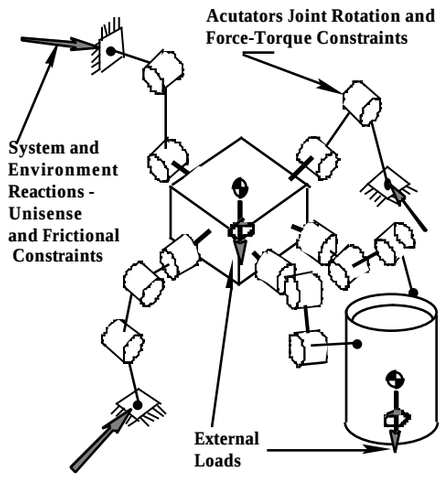

Figure 1 A mobile, multi–limb robotic system Concept

Contact forces and actuator efforts have been optimized based on frictional contact constraints independent of the mechanism which applies these forces, (Demmel and Lafferriere, 1989 and Ji and Roth, 1988) and contact forces and actuator torques have been solved for when subject to both frictional contact constraints and actuator effort limits, (Kerr, and Roth, 1986, Klein and Kittivatcharapong, 1990, Nahon, and Angeles, 1991, and Orin and Oh, 1981). These investigations were done within the contexts of grasping in multi-fingered robotic hands and robotic walking machines. While most of these studies have focussed on systems which are in a fixed configuration, methods have been developed to find smooth actuator efforts and contact forces and moments for *pre-specified* system motions, (Klein and Kittivatcharapong, 1990 and Nahon, and Angeles,

1991). These works provide important tools and insights to, for example, determine how redundant actuator efforts are to be distributed so as not to violate a system's constraints while applying forces and moments while in a given configuration. Such a result might be used prevent an object from slipping from the grasp of a robotic hand. Additional work has been done to study the effect of actuator force/velocity saturation limits on force application using cooperating manipulators in specified configurations, (Kokkinis, and Paden, 1989).

However, the question of how to *design* a system so that it is able to perform tasks requiring large forces over substantial ranges of motion while remaining within the constraints imposed by actuator capabilities, kinematic structure, as well as geometric obstacles in the envronment remains to be addressed; for example, how can a robotic device be best designed so that it is able to lift a heavy object in its "arms" while standing or climbing on a soft sandy hill? The problem of *planning* motions which will enable a system to perform such a task is tightly coupled to the design problem, and also remains unsolved.

In this paper, we address these problems using the *Force-Workspace Approach* (Madhani, 1991 and Madhani and Dubowsky, 1992). Fundamentally, the method 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 (Brooks and Lozano-Perez, 1983 and (Lozano-Perez, 1987). The resulting C-space map, termed the *Force-Workspace (FW)*, then uniformly represents system constraints. The FW approach is illustrated in Figure 2, which shows schematically a two-dimensional FW parameterized by $[q_1, q_2]$.

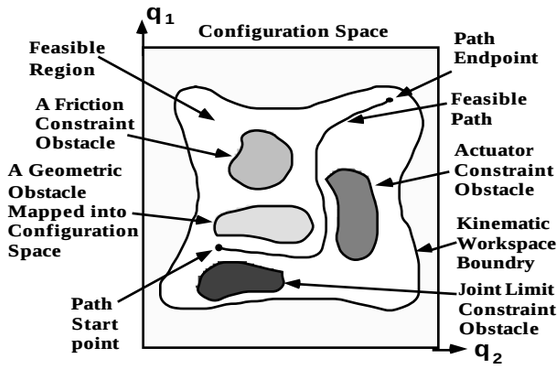

Figure 2. The Force-Workspace Concept.

Within this space, the boundary of the kinematic workspace is found, defined as the set of q_i's where the system can reach as constrained by kinematic limits excluding joint range of motion limits. Within the kinematic workspace additional constraints have been mapped as the *constraint obstacles*. In this way, the interacting force and actuator constraints of the system can be treated in a unified manner with other system constraints. The resulting FW yields quantitative and qualitative information for design and planning. For example, it is possible to determine how changes in system design parameters, such as actuator torque limits and link lengths, affect a system's ability to effectively perform a task by observing the resulting changes in the shapes and sizes of constraint obstacles. Furthermore, to plan motions of the system which do not violate these system constraints, it is only necessary to choose a feasible path in C-space which does not intersect any of these obstacles. It is shown that the FW can be transformed into workspace coordinates to enhance their usefulness in some applications. In this paper, the FW is applied to the design of a climbing robot and to the planning of its motions. It is shown how overlapping FW maps can be used to plan multi-step gaits for such systems.

2 GENERATING THE FORCE WORKSPACE

2.1 System Description and Assumptions

The multi-limb systems studied here are assumed to consist of rigid links and ideal joints with no clearances or friction. External forces and moments supported or applied during a task are assumed to be known as functions of the system's configuration and may act at any location on the system's structure. The external forces and moments are assumed to be significantly larger than any dynamic forces

generated by the system's motion; hence these dynamic forces are neglected. 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 with ground a mechanism with n degrees of freedom (DOF). Its motion is represented by the vector $\underline{q}$ with n elements q_i. The system may contain both open and closed kinematic chains. In addition to the n q_i's an additional integer parameter, *P*, the system pose, is required to completely describe the system's configuration. *P* ranges from 1 to the total number of different system poses. The system C-space is defined as the space parameterized by $\{\underline{q}, P\}$.

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

To generate the FW of a system, the C-space is represented by a generalized quadtree or 2^n-tree. This and similar hierarchal representations have been used in the past to attack the problem of geometric obstacle avoidance for single manipulators with substantial success, (Lozano-Perez, 1987) and Faverjon, 1984). The 2^n-tree is a recursive data structure and the algorithms used to generate it are relatively simple, (Kerr and Roth, 1986). 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 every configuration within a node is entirely *feasible*, *infeasible*, or *mixed* with respect to system constraints, and the node is then labelled accordingly. At all configurations within a feasible node no constraints are violated, and at all configurations within an infeasible node at least one constraint is violated. If a node contains both feasible and infeasible configurations, then it is labelled mixed. The tree is generated by testing tree nodes, beginning with the rootnode, and by subdividing all mixed nodes into 2^n smaller nodes. The process is continued in recursive fashion, subdividing mixed nodes until a node is shown to be either completely feasible or completely infeasible, or if some pre-specified small node size is reached. A thorough introduction to 2^n-tree representations can be found in Samet, 1990. Using the 2^n-tree representation, the problem of mapping system constraints into the C-space to generate the force-workspace is reduced to generating appropriate and computationally practical tests to determine if a node is feasible, infeasible, or mixed with respect to the system's constraints.

2.3 Force–Workspace Node Tests

The first step in building the force-workspace, such as shown schematically in Figure 2, is to determine the kinematic workspace of the system, defined as the set of permissible system configurations which do not violate geometric constraints, excluding joint limits and physical obstacles in the environment. Once this has been done, joint limits, obstacles in the environment, actuator saturation limits, and frictional constraints between the system and the environment can be mapped as obstacles into the kinematically free regions of C-space. A great deal of work has been done to address kinematic workspaces, and obstacle avoidance for conventional serial chain manipulators (Kumar and Waldron, 1981, Yang, and Lee, 1983, Lozano-Perez, 1987, and Lozano-Perez and Taylor, 1989), although the work done to address kinematic workspaces for multi-limb systems is somewhat limited, and substantial additional work is required in this area (Kerr and Roth, 1986). In this paper, we address this problem for a particular case of a robotic climbing machine, see section 3.

The feasibility of a C-space cell, corresponding to a 2^n-tree node, with respect to actuator effort constraints and system-environment frictional constraints, is tested by extending a well-known linear programming technique originally developed to specify actuator torques in redundantly actuated robotic hands (Kerr and Roth, 1986) In this technique, 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 the three orthogonal components of force and the three orthogonal components of moment at each contact point between the system and its environment. These are the forces and moments which support the system, such as those 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 the set of 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. In this case, 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 determine how the components of $N(\mathbf{W})$ will be combined (Kerr and Roth, 1986). These null space components produce what are often referred to as "internal forces" in the overconstrained system (Kumar and Waldron, 1981 and Kerr and Roth, 1986). 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 Kerr, and Roth (1986), shown in Samet (1990).

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 condition on d^* 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 entirely *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 entirely *infeasible*. Figure 3 shows examples of feasible and mixed node cells. Since this test is computationally most expensive, it is applied to cells within the kinematic workspace that also have been found not to violate joint limits or geometric workspace obstacles. The resulting FW is of the form shown in Figure 2.

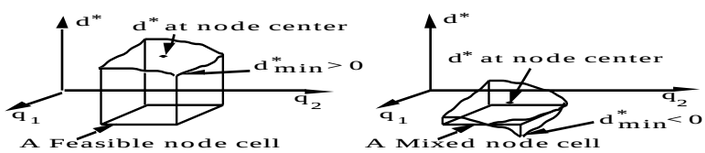

Figure 3. Feasible and Mixed cells in a 2–D F-W.

3 A CLIMBING MACHINE

To demonstrate the use of the Force-Workspace approach, consider the planar, three legged system, shown in Figure 4, which climbs between two vertical walls. The system must push outwards against the walls to maintain frictional support and prevent slipping, but is limited in how hard it can push by actuator torque saturation limits. It has been shown that this system can sucessfully climb by using two of its three limbs at a time to pull itself upwards. A gait is generated by placing

the third limb at an appropriate wall location, transferring weight to it, and repeating the procedure. The problems to be addressed are how to best design the system to permit it to lift itself using two limbs at a time, and how to plan these motions. Hence, a two limb system is studied comprising four links, two in each limb, and three actuated revolute joints where the center joint uses one actuator which applies a torque between the two limbs, and the body is free to swing about an extension of the axis of this actuator, Figure 4. The system body / payload load acts at joint 2, and is assumed much heavier than the system's limbs whose weight is neglected in this analysis. 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, F_T is a tangential contact force, and μ is the wall coefficient of friction. The system's FW is parameterized by $\{\underline{q} = [\theta_1\ \theta_2]^T, P = 1,2\}$. The node tests for this system are presented below.

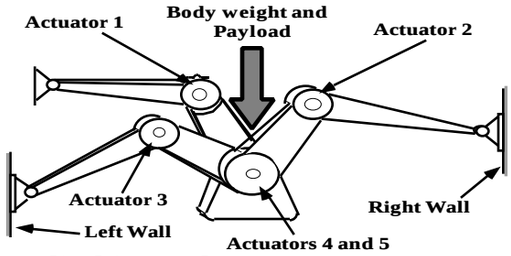

Figure 4. The Planar Climbing Robot Concept

3.1 The Climber FW Node Tests

Kinematic Workspace Test. The kinematic workspace (KW) test determines whether a node lies within the kinematic workspace. For this system, the kinematic workspace can be found by imagining the system to be separated at its body, joint 2, while each limb tip remains constrained by its respective contact location on the walls, as shown in Figure 5

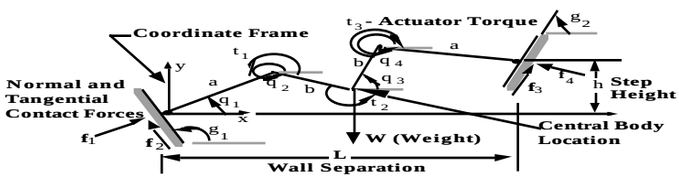

Figure 5. Kinematic Schematic of the Climbing Robot

The location of joint 2 on *limb 1* can easily be found as a function of θ_1 and θ_2. If these values of θ_1 and θ_2 represent a kinematically feasible position, they must lie within the region which can be reached by joint 2 on *limb 2* . The vector $\underline{x}$ in Figure 6 represents the location of the joint 2 as constrained by limb 1, and W represents the boundary of the region reachable by joint 2 as constrained by limb 2. Figure 6 shows $\underline{x}$ lying a distance $d(\underline{x}, W)$ from this region, which is bounded by two concentric circles. Note that this is true if limb 2 resides in either an elbow-up or elbow-down pose, and hence is independent of the pose parameter P. Clearly, a configuration, parameterized by $\{\underline{q} = [\theta_1\ \theta_2]^T, P = 1,2\}$ is within the KW if $\underline{x}(\underline{q})$ lies within W.

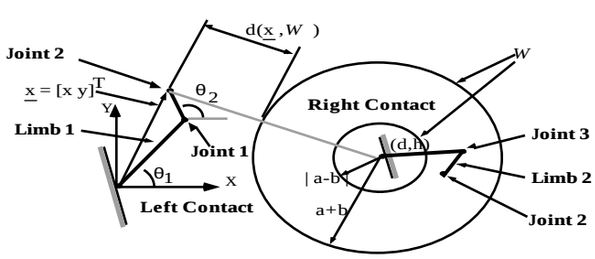

Figure 6. Finding the climbing robot's kinematic workspace.

From this kinematic constraint, a feasibility test for an entire node can be developed. The test follows from the fact that the displacement of the tip of a serial chain will be bounded if the displacements of each of its joints are bounded, (Paden, B., et al, 1989). Then if a given joint 2 location, $\underline{x}_0$, corresponding to a given $\underline{q}_0$, lies

sufficiently outside W, a test can be found which shows that all $\underline{x}$ corresponding to an entire node surrounding $\underline{q}_0$ will also lie outside W, hence the node will be infeasible. Similarly, if a given $\underline{x}_0$ lies sufficiently inside W, the test will show the node to be feasible. If neither case holds, the node will be found mixed. Following (Paden, B., et al, 1989), this test is found by bounding the displacements of $\underline{x}$ corresponding to finite displacements in the space of the variables $\underline{q}$, or

$$||\frac{d\underline{x}(\underline{q})}{d\underline{q}}(\Delta\underline{q})||_2 < B\,||\Delta\underline{q}||_\infty \qquad (3)$$

where $||\Delta\underline{q}||_\infty$ is the longest side length of the node, $d\underline{x}(\underline{q})/d\underline{q}$ is a jacobian matrix, and B is a finite bound on the displacement of $\underline{x}$, $||\Delta\mathbf{x}||_2$, over all values of $\underline{q}$. The function $d(\underline{x}, W)$ which gives the minimum distance between $\underline{x}$ and W, as shown in Figure 6, as:

$$d(\underline{x},W) = \min(|\sqrt{(x-d)^2 + (y-h)^2} - |a-b||, |\textbf{Error!}$$
$$(y-h)^2) - (a+b)|) \qquad (4)$$

By inspection of the kinematic structure of limb 1, it is easy to see $||\Delta\mathbf{x}||_2$ will be greatest when θ_1 is equal to θ_2 and:

$$B = a+b \qquad (5)$$

Then one can perform the following test for feasibility of a particular node. If the following inequality is satisfied:

$$\frac{||\Delta\underline{q}||_\infty}{2} < \frac{d(\underline{x},W)}{B} \qquad (6)$$

then the entire node will be feasible if $\underline{q}$ at its center is feasible, or infeasible if $\underline{q}$ at its center is infeasible. In other words, if $\underline{x}$ corresponding to a particular $\underline{q}$ is sufficiently inside or sufficiently outside W, then the node centered on that $\underline{q}$ will be entirely feasible or entirely infeasible, respectively. If equation (6) is not satisfied, then the node is found to be mixed.

Joint Limit Test. Joint limits are assumed in the form $\theta_{i,min} < \theta i < \theta_{i,max}$ i = 1,..., # joints. In mapping these constraints into the C-space, the joint limit test uses a non-linear programming technique to determine the feasibility of C-space cells. This is to avoid a brute-force, point by point approach. For each joint, the distance between the current joint position and a point directly between the upper and lower joint limits is used as an objective function. This objective function is then both minimized and maximized, subject to the linear constraints q node, in order to determine if any q within the current node causes the system to violate its joint limits. If any joint remains entirely within its infeasible joint region over all q within a node, then the node is labelled infeasible. If no joints violate their limits over all q within a node, then the node is labelled feasible. Otherwise, the node is labelled mixed.

Actuator Effort and Contact Wrench Constraint Test. This test follows from application of the general method described in section 2.3. The detailed equations and algorithms to implement it are given in Madhani and Dubowsky (1992).

3.2 The Climber Force-Workspace

The system FW corresponds to all configurations parameterized by $\{\mathbf{q} = [\theta_1,\theta_2]^T, P = 1, 2\}$. Figure 7 shows the FW in $\underline{\theta}$–space for $P = 1$, (right limb elbow-down) for a system with parameters given in Table 1. These parameters were chosen from a design study of a system currently being constructed, as further discussed below in Section 4.

Table 1. System parameters

Parameter	Value	Description
W	37.81 N	System weight
$\tau_{1,2,3,max/min}$	± 5.74 Nm	Joint torque limits
a, b	0.152 m	Link lengths
L	0.304 m	Wall separation
h	0.0 m	Vertical Contact Separation
μ	0.8	Wall coefficient of friction.

The dark grey cells in Figure 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 where the system may travel. Joint limit obstacles are not shown for clarity, see Madhani (1991). The white cells are formed from mixed cells. As described in Section 2.2, the white cells should be the same small size, however, a *merge* operation was performed on the final map after the subdivision process was completed. This process merges smaller cells of the same label into single cells, explaining the larger white cells in Figure 7.

For this system, it was found that a more intuitive and useful representation of the FW is obtained by mapping it into X–Y space, as shown in Figure 8. The variables X and Y are the coordinates of the center of the body in world space as shown in the figure. The figure shows a single X–Y space map for $P = 1$, and where the left limb is in an elbow-down configuration. The two-limb system is superimposed on the force-workspace and is shown in a feasible configuration, since the body lies within feasible regions of the force-workspace.

Figure 7. The force-workspace in [q1, q2]–space, P=1.

Figure 8 The force-workspace represented in X–Y space

4 USING THE FW IN DESIGNING THE CLIMBER

The Force-Workspace approach can be used for the design of multi-limb robotic systems. Here its use for the climber will be briefly described. Consider the system shown in Figure 4 whose task is to climb, carrying its body and payload upwards between two opposed walls. Such a system is currently being designed and built in our laboratory. The kinematic structure of this robot is shown in Figure 5. This system can climb by performing a series of "two-limb pullups" (Madhani, 1991 and Madhani and Dubowsky, 1992) where two limbs contact the walls and are used to lift the body of the system while the third waits to be brought into contact with a wall. The Force-Workspace approach is being applied to aid in the design of this system.

The first step in the application of the FW approach to the design of this system is to reduce the dimension of the design parameter space via a non-dimensionalization and several practical assumptions. The independent system parameters and variables for this system are taken to be θ_1,θ_2,a, b, h, τ_{1max}, τ_{2max}, τ_{3max}, τ_{1min}, $\tau_{2min,}$ τ_{3min}, μ_1, μ_2, W, see Table 1 and Figure 5. Assuming the design parameters which can be varied are the link lengths, a and b, the machine weight, W, and the actuator torque limits τ_{1max}, τ_{2max}, τ_{3max}, τ_{1min}, $\tau_{2min,}$ τ_{3min}. Recall the key parameter in generating the force-workspace is d^*, the maximum distance from the nearest constraints in the space of interaction forces. It can be written as a function of system parameters and variables as follows:

$$d^* = f(\theta_1,\theta_2,P,\mu_1,\mu_2,W,\tau_{1max},\tau_{2max},\tau_{3max},\tau_{1min},\tau_{2min},\tau_{3min},L,h,a,b) \quad (7)$$

These parameters are non-dimensionalized by scaling with respect to the distance between walls, L, and the weight of the machine W, as shown below:

$$d^*_{ND} = \frac{d^*}{W} = g(\theta_1,\theta_2,P,\mu_1,\mu_2,\frac{\tau_{1min}}{WL},\frac{\tau_{2min}}{WL},\frac{\tau_{3min}}{WL},\frac{\tau_{1max}}{WL},\frac{\tau_{2max}}{WL},\frac{\tau_{3max}}{WL},\frac{a}{L},\frac{b}{L}) \quad (8)$$

where $d^*_{ND} = d^*/W$ is a non-dimensional distance in the space of internal forces, $\tau_{iND} = \tau_{imin,max}/(WL)$ are non-dimensional actuator torque limits, and $a_{ND} = a/L$ and $b_{ND} = b/L$ are non-dimensional link lengths. Since generation of the force-workspace relies on sign(d*), (section 2.3), which is equal to sign(d^*_{ND}), the force-workspace can be generated using these non-dimensional design parameters, thus reducing the dimension of the design space by two. In the following example, the number of design parameters can be reduced further by assuming identical actuators at each joint, and by assuming that $|\tau_{i,min}|$ is equal to $|\tau_{i,max}|$ for each actuator. The effect of independently varying the relative strength of each actuator can be easily considered as shown in Madhani (1991). To maximize the area which can be reached by the central body for a given link length, we set a equal to b. As a result of the dimensional analysis and these simplifications, the dimension of the design space is reduced from ten dimensional parameters to two non-dimensional parameters: the non-dimensional link length, a_{ND}, and the non-dimensional torque limit, τ_{ND}.

The effect of changing system parameters on the feasible FW regions, where the system can effectively support itself, can be seen in Figure 9. This figure shows a sequence of four force-workspaces for the system in which both limbs reside in elbow down configurations, where h = 0, a_{ND} = 0.5, and τ_{ND} is varied from 0.411 to 0.675. Note the top right map of Figure 9, generated with non-dimensional parameters, corresponds to that of Figure 7 and 8, and the parameters given in Table 1. In Figure 9, the constraint obstacles in the force-workspace are labelled A, B, and C. As τ_{ND} is increased, we see a steady decrease in size of each of the constraint obstacles. It can be shown that within obstacle A, both the actuator 1 and the left wall friction constraint are violated, within obstacle B, both the actuator 2 and either the left or right wall friction constraints (depending on location within the obstacle) are violated, and within obstacle C, both the actuator 3 and the right wall friction constraint are violated. In these figures, it can be seen precisely how the feasible area of motion for the system increases as the torque limits increase, system weight decreases, and link lengths are changed. This shows how the FW's can represent graphically the effect of design parameter selection on the system's ability to climb (to lift itself statically), and how the designer might vary system parameters in order to modify the design. Of course, these choices will be limited by such factors as the coupling between increases in motor torque and increases in system weight. Nevertheless, the FW has been found in our experience to be a valuable tool in judging "what if" trade-offs in studying the effect of varying coefficients of friction at the walls, angles of the walls, relative location of system-wall contacts, link lengths, individual actuator torques, and system weight, to give the designer a more complete picture of how the system will perform under a variety of conditions, and what changes should be made to improve performance, (Madhani, 1991).

Figure 9. FW's as a function of actuator torque limits

5. MOTION PLANNING IN THE FORCE-WORKSPACE

The Force-Workspace approach can also be used to plan motions of a multi-limb mobile robotic system so that they can apply large forces and moments over

large ranges of motion without violating their constraints. If this motion involves fixed contact points with the environment, then only a single FW is required. To plan motions which involve relocating contacts, a series of FW's are required. These issues are discussed in the following sections.

5.1 Fixed Contacts—The Single FW

Once the FW has been generated, system motions between two configurations are planned by selecting a path within the C-space which avoids constraint obstacles and hence will not violate system constraints. To automatically generate such paths, the feasible nodes of the 2^n-tree representation of the force-workspace are transformed into a *search graph* whose edges represent the physical adjacency relationships between all feasible cells. Since cells adjacent in the force-workspace generally do not correspond to adjacent nodes in the 2^n-tree, a *neighbor-finding* algorithm is used to find the correct adjacency relationships in order to generate the search graph , and (Madhani, 1991 and Samet, 1982). A complete path between any two connected points which lie in the graph can be generated by using a minimum cost graph search, (Sedgewick, 1990). Finally, in order to tailor a path to suit a particular application, the edges of the search graph are weighted with a configuration based performance index, and then the path is selected with the minimum overall path weight using a weighted search.

Two examples of planning motions for the system shown in Figure 5 with parameters given in Table 2, are briefly discussed below. First the edges of the search graph were weighted by the distance between nodes, so the resulting path gives the minimum length travelled by the center body between points A and B, shown in Figure 10. This path avoids the constraint obstacle in the right of the FW. Figure 11 shows the actuator torques, and the ratios of tangential to normal contact forces along the path which must be less than the assumed coefficient of friction of 0.5.

Table 2. System parameters

Parameter	Value	Description
W	300 N	System weight
$\tau_{1,2,3,max/min}$	± 200 Nm	Joint torque limits
a, b	0.5 m	Link lengths
L	1.0 m	Wall separation
h	0.2 m	Vertical Contact Separation
μ	0.5	Wall coefficient of friction.

It can be shown for this example that the constraint obstacle circumscribed by the path in Figure 10 is due to the actuator 3 torque limit, and the right wall friction constraint, (Madhani, 1991). This can be seen in Figure 11, where both the actuator 3 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 0.5) both lie very near their limits as the path circumscribes the constraint obstacle.

Figure 10. A Minimum Distance Path

Figure 11. Actuator torques and contact forces ratios for minimum distance path.

Due to uncertainty in the system parameters, it would be desirable to avoid closely approaching system constraints. This behavior can be avoided by using a

different search graph edge weighting function. Recall that when a system is far from violating its actuator and friction constraints, d^*, the minimum distance from the nearest linearized actuator/friction constraints in the space of internal forces, will tend to be large. Hence we apply the following edge weight to the search graph:

$$\text{edgeweight} = \| \mathbf{x}_1 - \mathbf{x}_2 \|_2 \left[\left(1/d^*_1 + 1/d^*_2 \right)/2 \right]^{\alpha} \qquad (9)$$

where d^*_1 and d^*_2 are the values of d^* calculated at the centers of adjacent nodes which define an edge on the search graph, and x_1 and x_2 are the (x,y) coordinates of the body at the centers of these nodes. The resulting path minimizes an approximate integral of $(1/d^*)^{\alpha}$ along the path. Setting α equal to 5 strongly penalizes paths which enter areas of low d^*. Figure 12 shows the resulting path, and Figure 13 shows the corresponding actuator torques and the ratio of tangential to normal contact forces.

Figure 12. A d*-based path

Figure 13. Actuator torques and ratios of tangential to normal contact forces for a d*-based path.

The body traverses more than twice the distance, approximately 1.9 to 0.73 m, when the d^*-based criteria is used instead of the distance criteria. This is because, the second path seeks regions of large d^*, and steers well clear of infeasible regions. Figure 13 shows that the joint 3 torque and the ratio of tangential to normal contact forces remain away from their limits until near the end of the path, which is near an infeasible region, and hence cannot be avoided. Note that the joint 1 and 2 torques and the ratio of tangential to normal contact forces at the left contact are greater for the d^*-based path than for the shortest distance path, but overall none of these values are ever forced against their limits. It should be recalled that this simple example did not consider the effect of geometric obstacles in the system's environment.

5.2 Planning a "Gait"

The above procedure allows motions to be planned for a system given that a fixed set of contact conditions exist between the system and its environment and task. During many tasks, however, it may be desirable to change or relocate contacts, for example while making a step with a walking machine or while turning a valve "hand-over-hand". We use the term *stance* to define a system and a particular set of contacts with its environment and task; a system in a given stance forms a particular mechanism and has associated with it a single force workspace in which motions can be planned using the above computer search technique. In order to relocate contacts, a method is required to move between stances. In other words, each time a new set of contacts is chosen, the system forms a new mechanism, or *resides in a new stance*, and a new FW will be generated to plan motions for this mechanism; a method is needed to plan a motion across several force-workspaces. During the process of transferring between stances, or equivalently between FW's, we must ensure that system constraints are not violated. Figure 14 shows a technique for changing stances and transferring between FW's.

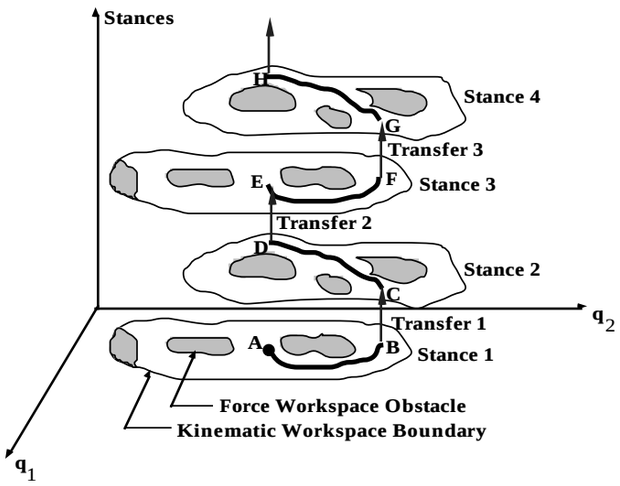

Figure 14. Transferring between force-workspaces when choosing new contacts.

The figure shows conceptually four FW's corresponding to a system in each of four different stances. The figure assumes that in each stance the system has 2 DOF, paramaterized by q_1 and q_2, though in general it may have many, and this number may differ between stances. The particular system variables used to parameterize the system DOF will in general differ between stances, hence q_1 and q_2 are referred to as the "active" motion variables, and may correspond to different system variables in each stance.

In Figure 14, the system begins at a configuration in one stance, such as configuration A in stance 1. In order to change stances, the topology of the mechanism formed by the system and the environment must change. For example, a limb may be brought into contact with the environment or the system's task, and subsequently, another may be lifted to complete the transfer between stances. Of course, any external loads or task forces must be supported throughout this process. The necessary and sufficient condition to allow a shift to the next stance is that the system configurations before and after the shift must lie in feasible regions in each of the corresponding FW's, (Madhani, 1991). As a result, as suggested in Figure 14, the system must move from configuration A in its current stance to a configuration which is compatible with a feasible configuration in the next stance, such as configuration B. The path for the motion from A to B must lie in the feasible FW for the current stance. At point B it possible to transfer to stance 2 at configuration C. During this transfer, the system is stationary, and after the transfer, a new set of q_i's will in general be required to paramaterize the motion of the new mechanism representing the system. In a similar manner, the system may proceed to point D and then transfer to point E in stance 3. The result can be referred to as a *gait* for the multi-limb system, in the sense of a walking machine, although it applies to any multi-limb system moving through a series of stances.

Figure 15 shows a gait generated using the system of Figure 5, with parameters given in Table 3. The medium grey region in Figure 11 represents feasible system configurations, and the dark grey region in Figure 15 (b) is the feasible intersection of the FW's corresponding to each stance. 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 the final gait, with a resulting body motion shown in Figure 15.

Table 3. System parameters

Parameter	Value	Description
W	300 N	System weight
$\tau_{1,3,max/min}$	± 200 Nm	Joint torque limits
$\tau_{3,max/min}$	± 300 Nm	Joint torque limits
a, b	0.5 m	Link lengths

L	1.0 m	Wall separation
h	0.4 m	Vertical Contact Separation
μ	0.8	Wall coefficient of friction.

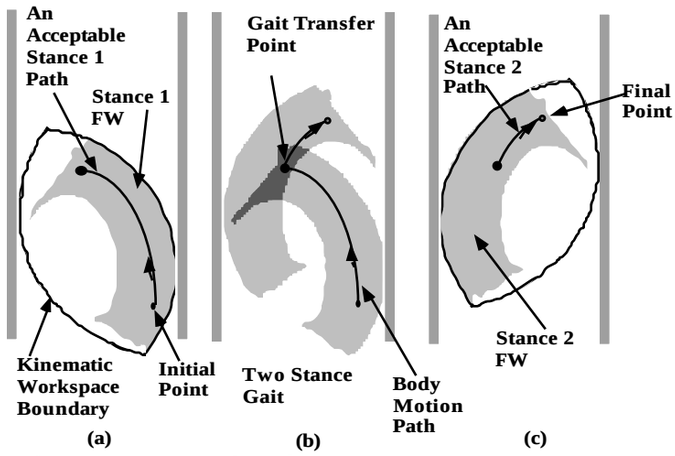

Figure 15. A gait generated using force-workspaces.

6. CONCLUSIONS

A method is presented for the design and motion planning of a class of mobile, multi-limb systems which must apply or support specified loads over large ranges of motion without violating actuator force/torque saturation limits, contact force/moment constraints between these systems and the environment (friction), and constraints on joint ranges of motion. The approach, termed the Force-Workspace Approach uses a recursive subdivision process to map the above constraints in a unified way into the system C-space to form *constraint obstacles*. The effects of changes in specific design parameters can be observed through changes in the sizes and shapes of the constraint obstacles. Motions can be generated which do not violate system constraints by selecting paths that avoid constraint obstacles. These paths can be optimized based on any configuration dependent performance index. "Gaits" for a multi-limb system can also be generated using the approach if a task requires that limbs change the type or location of contacts with the environment. The algorithms and analyses used to generate the FW and for planning motions within it can be applied directly to higher DOF systems. Visualization of high DOF FW's and their application to design is an area of current research. The method has been used as a design tool to specify actuator torque limits and link lengths for an experimental three-limb planar climbing robot, currently under construction, given the task of climbing upwards between two vertical walls. The method is applied to plan motions and gaits allowing this robot to climb.

7. ACKNOWLEDGEMENTS

The support of this work by the NASA Langley Research Center, Automation Branch under Grant NAG–1–801, and by the MIT Warren. M. Rohsenow Graduate Fellowship is acknowledged.

8. REFERENCES

Brooks, R. A. and Lozano-Perez, T., 1983, "A Subdivision Algorithm in Configuration Space for Findpath with Rotation," *Proc. of the VIII Intern. Joint Conf. on Artificial Intelligence*, Karlsruhe, pp 799-806.

Demmel, J. and Lafferriere, G., 1989, "Optimal Three Finger Grasps," *Proceedings of the IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, pp 936-942.

Faverjon, B., 1984, "Obstacle Avoidance Using an Octree in the Configuration Space of a Manipulator," *IEEE International Conference on Robotics and Automation*, Atlanta, GA.

Ji, K. and Roth, B., 1988, "Direct Computation of Grasping Force for Three-finger Tip-prehension Grasps,", *Trends and Developments in Robotics*, pp 125-138, ASME pub. DE-vol 15-3.

Kerr J., and Roth, B., 1986, “Analysis of Multifingered Hands,” *International Journal of Robotics Research*, vol. 4, no. 4, Winter, pp 3-17.

Klein, C. A. and Kittivatcharapong, S, 1990, “Optimal Force Distribution for the Legs of a Walking Machine with Friction Cone Constraints,” *IEEE Transactions on Robotics and Automation*, vol 6, no. 1,pp 73-85.

Kokkinis, T. and Paden, B, 1989, “Kinetostatic Performance Limits of Cooperating Robot Manipulators Using Force-Velocity Polytopes,” *Proceedings of the ASME Winter Annual Meeting*, San Fransisco, CA, pp 151-155.

Kumar, A., and Waldron, K. J., 1981, “The Workspaces of a Mechanical Manipulator,” *ASME Journal of Mechanical Design*, vol. 103, pp 665-672.

Kumar, V. R. and Waldron, K. J. 1988, “Force Distribution in Closed Kinematic Chains,” *IEEE Journal. of Robotics and Automation*, vol 4. no. 6, pp 657-664.

Lozano-Perez, T, 1987, “A Simple Motion-Planning Algorithm for General Robot Manipulators,” *IEEE Journal of Robotics and Automation*, vol RA-3, no 3.

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

Lozano-Perez, T, 1990, *Motion Planning*, Class Notes, 6.802/6.867 Robot Manipulation, MIT, Cambridge, MA..

Madhani, A., 1991, *Design and Motion Planning of Robotic Systems Subject to Force and Friction Constraints*, S.M. Thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology.

Madhani, A. and Dubowsky, S., 1992, “Motion Planning of Multi-Limb Robotic Systems Subject to Force and Friction Constraints,” *IEEE International Conference on Robotics and Automation*, Nice, France.

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

Nahon, M. A., and Angeles, J., 1991, “Optimization of Dynamic Forces in Mechanical Hands,” *ASME Journal. of Mechanical Design*, vol 113, pp 167-173.

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

Paden, B., Mees, A., and Fisher, M., 1989, “Path Planning Using a Jacobian-Based Freespace Generation Algorithm,” *Proc. IEEE International Conference on Robotics and Automation*, pp 1732-1737.

Orin, D. E. and Oh, S. Y., 1981, “Control of Force Distribution in Robotic Mechanisms Containing Closed Kinematic Chains,” *ASME Journal of Dynamic Systems, Measurement, and Control*, vol 102, pp 134-141.

Salisbury, J. K., 1982, *Kinematic and Force Analysis of Articulated Hands,*, Ph.D. Thesis, Depart. of Mech. Engr., Stanford University.

Samet, H., 1982, “Neighbor Finding Techniques for Images Represented by Quadtrees,” *Computer Graphics and Image Processing*, v. 18, pp 37-57.

Samet, H, 1990, *The Design and Analysis of Spatial Data Structures*, Addison-Wesley Publishing Co.

Sedgewick, R., 1990, *Algorithms in C*, Addison -Wesley Publishing Co.

Yang, D. C. H. and Lee, T. W. 1983, “On the Workspace of Mechanical Manipulators,” *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 105, pp 63-69.

Yoshikawa, T., and Nagai, K., 1991, “Manipulating and Grasping Forces in Manipulation by Multifingered Robot Hands,” *IEEE Transactions on Robotics and Automation*, vol. 7, no. 1.