WORKSPACE OPTIMIZATION OF ORIENTATIONAL 3-LEGGED UPS PARALLEL PLATFORMS

Mircea Badescu, Jeremy Morman and Constantinos Mavroidis
Robotics and Mechatronics Laboratory, Department of Mechanical and Aerospace Engineering
Rutgers University, The State University of New Jersey
98 Brett Rd, Piscataway, NJ, 08854-8058
E-mail: badescum@jove.rutgers.edu, jdmorman @eden.rutgers.edu, mavro@jove.rutgers.edu

ABSTRACT
In this paper the workspace optimization of an orientational 3-legged parallel platform with universal (U), prismatic (P) and spherical (S) joints at each leg is performed. The workspace is parameterized using four design parameters that span a large range of values. In this paper both the unconstrained and the constrained workspaces (i.e. workspace with joint limits) are used. For the workspace of each design configuration three performance indices are calculated using a Monte Carlo method: a) the workspace volume; b) the average of the inverse of the condition number and c) a Global Condition Index which is a combination of the other indices. A Genetic Algorithm procedure is used to determine the optimum design of the platform using the Global Condition Index as the objective function. Plots of the workspace are presented for the optimal design configuration with and without joint angle limitations for the whole workspace and for specific twist angles. Finally, for the optimal design, it is shown that by introducing limits to the universal and spherical joints, the platform's isotropy is improved.

Keywords: parallel platforms, optimal design, workspace, Genetic Algorithm, Monte Carlo method.

INTRODUCTION
In recent years, modular robots were increasingly proposed as means to develop reconfigurable and self-repairable robotic systems [1]. To perform impromptu custom tasks, increase the payload to weight ratio, and, in cases of emergency, self-repair, future inter-planetary robots and manipulation systems need to incorporate modularity and self-reconfiguration capabilities. Modular robots utilize many autonomous units, or modules, that can be reconfigured into a vast number of designs. Ideally, the modules will be homogeneous, small, and self-contained. The robot can change from one configuration to another by manual reassembly, or by itself. Self-reconfiguring robots adapt to a new environment or function by changing shape. Modules must interact with one another and cooperate in order to realize self-reconfiguration. Also, modular robots can repair themselves by removing and replacing failed modules. Since one self-reconfigurable modular robot can provide the functionality of many traditional mechanisms, they will be especially suited for space and planetary exploration, where payload mass must be kept minimum. Because they promise self-reparability and virtually limitless functionality, future self-reconfigurable modular robots are expected to be cheaper and more useful than current robot mechanisms in space missions.

In this project we investigate the use of 3-legged parallel platforms as joint modules of reconfigurable robots. Parallel platforms are currently being used in many applications as multi-degree of freedom systems with high rigidity, high payload to weight ratio, high precision and low inertia [2, 3]. These properties are also desired characteristics for the joint modules of reconfigurable robots. Six-legged, six degree-of-freedom (DOF) parallel platforms have been used as joint modules for reconfigurable robots in [4]. However, the high number of DOF, and thus the high number of actively controlled joints per module, increases complexity and cost. In addition, a purely 3 degree-of freedom translational or orientational motion would require activation of all six-module legs, which means increase in energy consumption. Our approach is to develop reconfigurable robots formed by a series of 3 DOF 3-legged parallel platform modules with purely translational or orientational motion so that they form hybrid kinematic chains with decoupled translation and orientation [5]. Figure 1 shows an example of such a hybrid system. In this example a 2-arm reconfigurable robotic system is formed from an alternating sequence of 3 DOF orientational and translational parallel platform modules. Recently, we studied the workspace optimization of a 3 DOF translational platform
In this paper we study the workspace of a 3 DOF orientational platform.

Different types of spherical or orientational parallel platforms have been proposed and analyzed. Merlet described several types of 2 and 3 DOF parallel platforms that can be used as wrists / orientation mechanisms [3]. Tsai classified the parallel platforms with rotational DOF in spherical and spatial orientation mechanisms [2, 7]. All the moving points of a spherical parallel platform move on concentric spheres. Analysis and fabrication of a spherical 3 DOF platform were presented in [8]. Optimization of a spherical five bar parallel linkage was done in [9]. Gosselin and Angeles used three design criteria, symmetry, workspace volume and isotropy, to optimize a 3 DOF spherical platform [10].

![Figure 1: Two-Arm Reconfigurable Robot with 3-DOF Parallel Platform Modules.](image)

In this paper the design optimization of orientational 3-UPS parallel platforms with prismatic, universal, and spherical joint constraints is performed. The workspace is parameterized using four design parameters, which are the prismatic joint stroke, the difference between the distances of the center to one of the corners of the triangular plate of the base and of the moving platform, the height of the platform in zero orientation, and the ratio of the heights of the two tetrahedrons. For a large range of values for these design parameters the workspace of the corresponding 3-UPS platforms is calculated. In this paper both the constrained and unconstrained workspaces are used. The difference between these two is that for the constrained workspace angular universal and spherical joint limits are imposed. For both workspaces of each design configuration three performance indices are calculated: a) the workspace volume; b) the average of the inverse of the condition number and c) a Global Condition Index which is a combination of the other two performance indices. For determining the optimal design a Genetic Algorithm was implemented. The Global Condition Index was chosen as the objective function for optimization. The optimization was performed with and without imposing joint constraints. Plots of the 3-D workspace of the optimal designs are generated. Finally, for the optimal design, it is shown that by introducing limits to the angles of the universal and spherical joints, the isotropy of the parallel platform is improved.

**MATHEMATICAL TOOLS**

In this section we present the mathematical and kinematical tools that were used to formulate the workspace optimization problem. These tools include the direct and inverse kinematics, the platform’s Jacobian matrix, its condition number and the Global Condition Index. For the
direct and inverse kinematics and for the calculation of the Jacobian matrix, we used the solution methodology proposed in [2]. For the Global Condition Index we used the definition introduced in [10].

**FIGURE 3:** 3-UPS Parallel Platform.

The 3-UPS parallel platform consists of a fixed tetrahedron, a moving tetrahedron and three identical limbs (see Figure 3). The tetrahedrons have equilateral triangular bases of different sizes (triangles $A_1A_2A_3$ and $B_1B_2B_3$). The tips of the tetrahedrons are connected using a spherical joint. The limbs are connected to the moving tetrahedron base with 2-DOF universal joints and to the fixed tetrahedron base with spherical joints. A linear actuator controls the leg length, and forms a prismatic joint. In Figure 3 the spherical joints are marked with S, the universal joints with U, and the prismatic actuated joint with P. Each universal joint is treated as two revolute joints with axes perpendicular to each other and intersecting at a point. In Figure 4, $u_{is}$ is the unit vector along the prismatic joint axis for leg $i$. $u_{ia}$ and $u_{ib}$ are the unit vectors along the axes of the universal joint of leg $i$. $u_{i7}$ is the CB$_i$ edge of the moving tetrahedron.

**FIGURE 4:** Joint Axes in the 3-UPS Platform.

A coordinate system is chosen for each tetrahedron as shown in Figure 5, a) and b). Frames A and B are defined for the base and moving tetrahedrons respectively. Their origin is placed at the common tip of the tetrahedrons and represents the point around which the moving tetrahedron rotates. Their $z$-axis is perpendicular to each base triangle's plane, and their $x$ and $y$-axes are in planes parallel to the base triangle's plane. The $x_{ab}$ axes are oriented parallel to the OA$_1$ and EB$_1$ lines, respectively. The frame A, C$_x$y$_x$z$_x$, is rigidly connected to the fixed tetrahedron and has center C at the tetrahedron’s tip. The point O is the center of the base triangle. The $x_a$ axis is oriented along OC. The $x_s$ axis is parallel to OA$_1$ and the $y_s$ axis is chosen to obtain a right hand rectangular coordinate system.

As seen in Figure 5, the position vectors of points A$_i$ and B$_i$ with respect to frames A and B respectively, can be written as:

$$\begin{align*}
\vec{a}_i &= [a_{ix}, a_{iy}, -h_a]^T, & \vec{b}_i &= [d_{ix}, b_{iy}, -h_b]^T
\end{align*}$$

(1)

**FIGURE 5:** Coordinate Systems.

The scalar $a_i$ denotes the distance between the center, O, of the base triangle of the fixed tetrahedron and any of the corners of the base triangle. The scalar $b_i$ denotes the distance between the center, E, of the base triangle of the moving tetrahedron and any of the corners of the base triangle. The parameter $c_i$ denotes the difference between $a_i$ and $b_i$.

Parameter $d_i$ denotes the length of leg $i$. Subscript $i$ denotes one of the legs and can take the values 1, 2 or 3. Letters in bold represent three-dimensional vectors. Superscript A or B on the left of a vector denotes the reference frame where its coordinates are calculated. Superscript T on the right of a vector denotes the transpose of a vector. Vector $d_i$ determines the position of the leg $i$ ($A_iB_i$) in the base coordinate system:

$$\begin{align*}
\vec{d}_i &= [d_{ix}, d_{iy}, d_{iz}]^T
\end{align*}$$

(2)

The scalar $h_a$ denotes the height of the fixed tetrahedron and is the distance from point O to point C. The scalar $h_b$ denotes the height of the moving tetrahedron and is the distance from point E to point C. The parameter $h$ represents the total height of the parallel platform in (0,0,0) orientation and is equal to the sum of $h_a$ and $h_b$. The parameter $h_{ab}$ is the ratio between...
h_a and h_b. The parameters c_i and h are taken as ratios to the maximum length of the leg.

The transformation from the moving frame B to the fixed frame A can be described by a 3x3 rotation matrix \( R_{AB} \) defined by a z-x-z (q-0-ψ) Euler rotation. Given that in the initial position (0-0-0 rotation) the x_a and x_b axes coincide, z_a and z_b, and y_a and y_b are in opposite directions respectively, the resulting rotation matrix is given by:

\[
R_{AB} = \begin{bmatrix}
  c_{ψ}c_{θ}c{φ} & -c_{ψ}s{φ} & s_{ψ} \\
  c_{ψ}c_{θ}s{φ} & c_{ψ}s{φ} + s_{ψ}c_{θ} & c_{θ} \\
  -s_{ψ}c_{θ} & c_{θ}s_{ψ} & c_{ψ}
\end{bmatrix}
\]

where \( c \) represents the cosine and \( s \) the sine of the following angle.

For the direct kinematics the leg lengths \( d_i \) and the triangular bases' geometry are known. The three orientation angles of the moving platform are determined from expanding the following three scalar equations (see [2] for more details):

\[
d_i^2 = a_i^2 + b_i^2 - 2a_i b_i \cos(θ_i) \quad \text{for} \ i=1, 2, 3
\]

For the inverse kinematics the orientation of the moving platform is known and the limb lengths, \( d_i \) are calculated from the following three equations:

\[
d_i = \pm \sqrt{a_i^2 + b_i^2 - 2a_i b_i \cos(θ_i)} \quad \text{for} \ i=1,2,3
\]

The manipulator Jacobian matrix \( J \) relates the end-effector velocities \( \omega \) to the actuated joint velocities \( d \):

\[
J \omega = J d
\]

For the 3-UPS platform \( J_x \) and \( J_a \) are given by:

\[
J_x = \begin{bmatrix} (u_{1,7} \times u_{1,4})^T \\ (u_{2,7} \times u_{2,4})^T \\ (u_{3,7} \times u_{3,4})^T \end{bmatrix}, \quad J_a = [1]
\]

The condition number of the Jacobian matrix is defined by Equation (8):

\[
k = ||J|| \cdot ||J^{-1}||
\]

where \( ||.|| \) denotes the norm 2 of a matrix [11].

Robot manipulator configurations where \( k \) is equal to 1 are called isotropic. In these configurations the system is able to develop the same amount of forces and velocities in all end-effector directions. For high values of \( k \), there are end-effector directions where the manipulator can develop much higher forces or velocities than in other directions. In many applications, this is not a desirable system property because the system loses its homogeneity in force and velocity development. Configurations in which \( k \) has an infinite value are singular configurations. In these configurations there are directions in space where the end-effector can either not move or not apply forces. The manipulator can also gain extra degrees of freedom in these configurations. Isotropy is a very local manipulator property because it changes from configuration to configuration. A manipulator can have good isotropy in one configuration but poor isotropy in another configuration. In many robot applications isotropy is an important property, and needs to be taken into account during the robot’s design phase. Defining a global isotropy index that will be able to characterize the system isotropy in the whole workspace is an important but difficult problem to solve. The problem is that a robotic system will always have some singular configurations or configurations with poor isotropy. By taking a type of an average of the condition number is only a rough indication of the quality of the system global isotropy and says nothing about the magnitude and number of bad isotropy configurations. Nevertheless, in this work we will use global isotropy indices based on the average of the condition number, as they are being used in the current literature.

In this work, three performance indices will be used to characterize a robotic system’s workspace:

a) The volume of the workspace. Obviously, using this performance index, optimal designs correspond to maximum workspace volume. It is obtained by multiplying the searching volume with the ratio of the number of points inside the workspace divided by the total number of points.

b) The average of the inverse of the condition number. The inverse of the condition number characterizes the system's isotropy. The average of the inverse of the condition number is calculated by summing the inverse of the condition number in every point in the workspace and then dividing it with the number of points considered in the workspace. Optimal designs correspond to values of the average condition number that are close to 1.

c) The Global Condition Index proposed in [10]. This index, defined in Equation (9), is the ratio of the integral of the inverse condition numbers calculated in the whole workspace, divided by the volume of the workspace. In reality it is calculated as the product of the sum of the inverse condition number with the searching volume divided by the number of points in the workspace. Therefore, it is like a combination of the first two performance indices:

\[
η = \frac{A}{B}, \quad \text{where} \ A = \int_{W} \left( \frac{1}{k} \right) dW \quad \text{and} \ B = \int_{W} dW
\]

in which \( k \) is the condition number at a particular point in the workspace \( W \). \( B \) is the total volume of the workspace. The Global Condition Index is bounded as:

\[
0 < η < 1
\]

Isotropic systems correspond to value of \( η \) equal to 1 and systems with bad isotropy correspond to values of \( η \) approaching zero.

**OPTIMIZATION ALGORITHM**

Based on the platform's direct kinematics, and due to the fact that all legs are equal and that the base and moving triangles are equilateral, the 3-UPS orientational platform workspace depends on four geometric parameters: a) the difference \( c_i \) between \( a_i \) and \( b_i \); \( c_i = a_i - b_i \); b) the total height of the platform considered as the sum of the heights of the two tetrahedra: \( h = h_a + h_b \) (see Figure 5); c) the ratio of the heights of the two tetrahedra: \( h_a / h_b \); and d) the stroke of the prismatic joint \( s \). We considered the following ranges of possible values for the design parameters.

The range of values for \( c_i \) is between 0.27 and 0.62. Platforms with very small values for \( c_i \) (i.e. platforms where the
triangular base and moving plates are almost equal) present extra DOF (self-motions) that could not be controlled from the actuation motion and such designs are not acceptable. Platforms with very large values for $c_i$ have very small workspace.

The total height of the two tetrahedrons, $h$, has values between 0.2 and 1.0. Below 0.2 the workspace is very small and has no practical use. Total height equal to 1 corresponds to maximum length. The ratio of the heights of the two tetrahedrons, $h_{ab}$, was selected to vary between 0.33 and 3.0.

The stroke $s_i$ of the actuator is the maximum amount of travel for each leg’s prismatic actuator. It is expressed with a percentage value of the minimum length of the limb $d_{min}$ which is taken such that the maximum length is equal to 1. The fact that the leg's maximum length is considered to be equal to 1 means that all lengths are normalized with respect to the leg's maximum length as it is specified from each application’s requirements. In this project the range of values of the stroke of the actuators is considered to be between 20% and 87.5%. These values were selected based on the technical specifications of the majority of commercially available prismatic actuators. The minimum value of each leg’s length $d_{min}$ is equal to $1/(1+s_i/100)$. So in each platform's configuration the value of the leg length $d_i$ should be between $d_{min}$ and 1.

Due to the fact that the plots of all three performance indices for any value of the design parameters are highly non-monotone, a Genetic Algorithm was implemented to determine the optimal design configuration. It consists of the following steps:

a) An initial population of 100 members is randomly generated. Each member contains values for the four design parameters defined above. Each parameter has an eight digit resolution which means that 256 different values can be chosen for each parameter.

b) For each population member the platform’s geometric parameters are calculated using the design parameters.

c) To determine the values of the performance indices a Monte Carlo method was implemented. For each configuration with certain values of the design parameters, $s_i$, $c_i$, $h$, and $h_{ab}$, $1,728,000$ orientations of the moving tetrahedron are chosen randomly. The $\phi$, $\theta$ and $\psi$ Euler angles corresponding to a $z$-$x$-$z$ rotation are randomly chosen in the interval $[-180^\circ, +180^\circ]$.

d) For each orientation the lengths of the legs are determined using the inverse kinematics, Equation (5). If all legs have lengths in the allowed interval $d_{min}<d<d_{max}$ the interference of the legs with fixed and moving tetrahedrons is checked. Also, for the constrained workspace, the joint angles are determined and verified if they are between imposed limits. If the orientation verifies all constraints, then the Jacobian condition number, $k$, is determined using Equation (8).

e) For each configuration the performance indices are calculated using Equations (9) [Note: In Equation (9) $\eta$ is the global condition number, $A$ is the average of the inverse condition number and $B$ is the volume of the workspace]. The population member with the best objective function is determined and saved in a file. It will be called “the king” in further references. As the objective function, any one of the described performance indices can be chosen.

f) A mating pool is then created from the initial population selecting the members with better objective function values.

g) A crossover among the members of the mating pool is done with a 0.45 probability. At the end, the first member is replaced with the member with the best objective function (the king).

h) A mutation with 0.05 probability is then done over the new population. The first member of the population is not affected (the king).

i) For the new population the cycle starts over. It stops when the value of the objective function of the king is not improved with a certain relative quantity.

The unconstrained optimization is done without imposing the constraints mentioned in step d) above.

**RESULTS**

We used a program written in C++ to perform the optimization and MATLAB to plot the workspace. We run the optimization program for the constrained and unconstrained workspace.

**FIGURE 6: Workspace of the Optimal Unconstrained Configuration of the 3-UPS Platform.**

The optimal design for the unconstrained workspace has the following parameter values: stroke=83.9%, $c_i=0.602$, $h=0.285$, and $h_{ab}=0.344$. Figure 6 shows the workspace of the optimal unconstrained configuration. The points represent the positions that can be reached by the center, $E$, of the moving tetrahedron base. $E$ describes a sphere with the center at point $C$ and a radius $h$. To represent the twist angle, $\psi$, the radius of the sphere described by point $E$ was changed depending on the value of the twist angle. For $\psi = -180^\circ$ the radius was reduced by 10% of $h$, and for $\psi = +180^\circ$ it was increased by 10%. A linear dependence was used for the intermediary values. The color with which the points were represented reflects the value of the condition number of the platform’s Jacobian at the specific point. A value 1 of the condition number was represented with dark blue color. Because some of the condition numbers had extremely large values (more than 100), in our representation in MATLAB™ all condition numbers with a value over 10 were reduced to 10 (This was used just for representation but not for optimization). Dark red color is used to plot points with large condition numbers.

For the optimization of the constrained workspace, we imposed limits at the various joints of the platform. In this paper we present results from two different schemes of joints constraints (I and II). For the constrained workspace I, the limits of the universal joint angles were set at $\pm60^\circ$ for the revolute joint that allows the leg rotation on CEA planes and $\pm60^\circ$ for the other revolute joint. We have to remind that the
universal joint can be treated as two revolute joints. The limits for the spherical joint angles were set at $\pm 65^\circ$. The optimal design for the first constrained workspace was found to have the following values: stroke=50%, $c_i=0.616$, $h=0.526$, and $h_{ab}=2.85$. Figure 7.a) shows the workspace for the optimal platform design using joint constraint scheme I. It can be noted, that due to the influence of joint constraints a stroke larger than 50% will no further increase the workspace volume.

We run the optimization algorithm for the second scheme of joint constraints. The limits of the universal joint angles were set at $\pm 75^\circ$ for the revolute joint that allows the leg rotation on CEA, planes and $\pm 85^\circ$ for the other revolute joint. The limits for the spherical joint angles were set the same, at $\pm 65^\circ$. The optimal design parameters that were obtained are: stroke=71%, $c_i=0.62$, $h=0.44$ and $h_{ab}=1.75$. In this case the average value of the inverse of the condition number, $k$, is increased and the workspace volume is increased relative to the previous optimal configuration.

For the constrained workspaces I and II we used the same representation method as for the unconstrained workspace. From the data provided by the optimization programs, it can be seen that the unconstrained workspace volume is less than half the constrained workspace volume. By imposing constraints on the joint angles, one can exclude not only many bad isotropy points, but many good isotropy points as well. Overall, the average value of the inverse of the condition number is increased, which means that the isotropy of the platform is improved, and the workspace volume is reduced relative to the previous optimal configuration.

Even though the workspaces shown in Figures 6-9 are continuous, not all of the points can be reached with any orientation. This is shown in Figures 8-b,c) and 9-b,c). Also it can be seen that for small values of the pitch angle, $\varphi$, the manipulator has a bad isotropy, no matter how small the yaw angle, $\theta$, or the twist angle, $\psi$, are.

From Figures 7, 8, and 9 we learn that points on the workspace can be reached only for certain twist angles. Also we learn that even though some points can be reached for a
large range of twist angles for some values of these we get a better isotropy. These results can be used to determine a path between two orientations maintaining a good isotropy if the twist angle in the intermediate positions is not critical.

CONCLUSIONS

In this paper the design optimization of orientational 3-UPS parallel platforms was presented. These platforms can be used as joint modules in reconfigurable robots and the quality of their unconstrained and constrained workspace is an important design feature. The workspace is parameterized using four design parameters. Three workspace performance indices are calculated as a function of the design parameters. These performance indices quantify the workspace volume, the system isotropy and a combination of the previous two properties. The optimal design is determined for both situations with and without joint limits imposed. Different design parameters are obtained in the two situations. The paper presents the total and fixed twist unconstrained and constrained workspaces.

ACKNOWLEDGMENTS

This work was supported by a Rutgers University Research Council Grant and by NASA's Jet Propulsion Laboratory. Mr. Mircea Badescu was supported by a Teaching Assistantship from the Department of Mechanical and Aerospace Engineering. Mr. Jeremy Morman was supported by an NSF REU supplement.

REFERENCES