ED6002 Project Report

Adarsh Somayaji ED15B001

Objective: To optimize the link lengths of a robotic manipulator so as to maximize force isotropy within a given usable workspace

Introduction: Designing a robot for modern applications such as haptic interfaces and surgical assistants has led to the performance demand increasing far beyond the traditional applications such as picking & placing objects and other repetitive tasks. Designing a robot to uphold a set of performance standards is complicated by the fact that the relationship between the robot's actuators & end effector varies with position & direction. Only after minimizing this variation, that is, maximizing the mechanical **isotropy**, can one choose suitable actuators & design a controller. The question of how to do the same is described in the following sections.

Background:

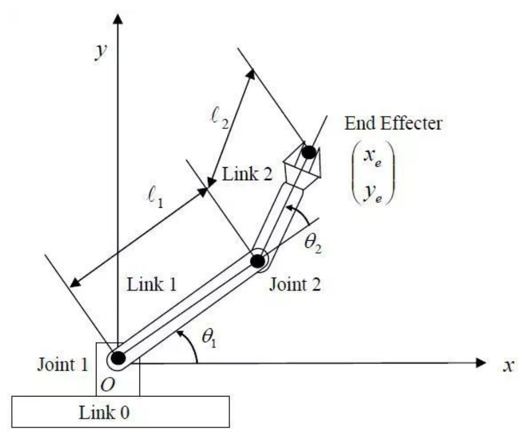

Fig 1: A 2R planar manipulator schematic

Consider a 2R planar manipulator as shown above. The 2 links are actuated by individual motors situated at their respective bases. The force exerted by the end effector of the manipulator is related to the motor torques by a linear map $\boldsymbol{\tau} = \mathbf{J}^T F$ where:

$\boldsymbol{\tau}$ → vector of motor torques $[\tau_1 \ \ \tau_2]$
J → The Jacobian matrix $[\delta x/\delta\Theta_1, \delta x/\delta\Theta_2; \delta y/\delta\Theta_1, \delta y/\delta\Theta_2]$
F → Vector of force exerted by the manipulator $[F_x \ F_y]$

Even though the map above is linear, the elements of the Jacobian matrix J are non-linear functions of the joint angles Θ. The Jacobian is fixed for a particular configuration. Therefore, an interesting question that arises is:
Given certain force requirements, which configurations are better than others at converting the input torques to the output forces?

For the sake of comparison, the magnitude of the vectors $|\boldsymbol{\tau}|$ & $|F|$ will be used to characterize these vectors. If the magnitude of force to be produced is restricted to unity in all directions, the actuator torques produce an ellipse in the torque domain[1].

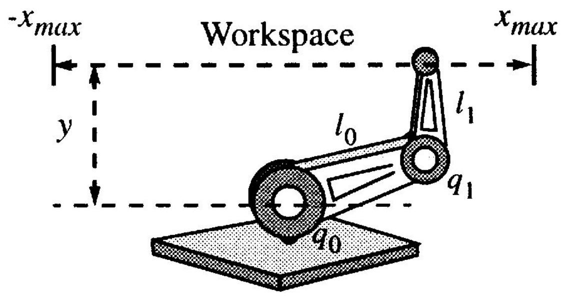

Fig 2: A 2R manipulator with a specified workspace

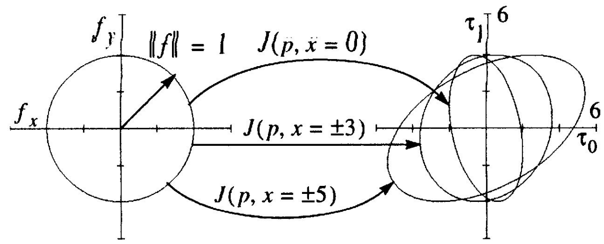

Fig 3: Torque ellipses for different positions in the workspace

As it can be observed, for the same output force to be produced in any direction at a particular point in the workspace, the motors have to exert different torques. Further, for different points in the workspace, the ellipse produced is different. Therefore, to define the optimality of the link lengths is a difficult proposition.

Here, the global isotropy index(GII) is used as a measure to define the "goodness" of a set of link lengths. The GII is defined as the ratio of the smallest singular value to the largest singular value in the entire workspace.

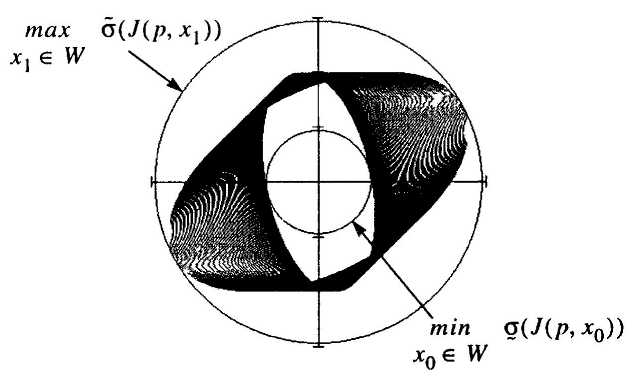

Fig 4: Torque ellipses and GII

Consider a particular set of link lengths. The workspace is discretized into several points and the force/torque ellipses is plotted for all points in the usable workspace. The GII is the ratio of the radius of the largest circle contained in all of these ellipses to the radius of the smallest circle containing all of these ellipses. A GII value of 1 implies that a mechanism is not only isotropic(direction insensitive) at each position in its workspace but also that it behaves consistently at all points in the workspace.

Problem Definition:

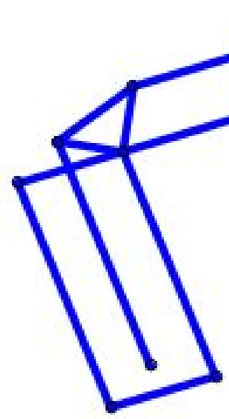

Fig 5: Kinematic diagram of a palletizing robot

A palletizing robot is to be used as a master arm in controlling a surgical robot through teleoperation. The configuration of a palletizing robot is shown in fig 5. As it is to be used as a master controller, the performance of the robot needs to be maximized within it's workspace. The parameter space is reduced to a single dimension by setting the length of the first link to be 1 unit and varying the length of the second link from a set lower limit to an upper limit. The workspace is defined to be a square of side 0.5 units and whose centre is located at a distance of 1 units from the robot's base at an angle of 45 degrees with the horizontal. The lower and upper limits of the length of the second link is calculated by ensuring that the boundaries of the usable and reachable workspaces are separated by a minimum safety margin of length K. This provides the constraints for the length of the second link.

Algorithm: The problem to be solved involves calculating the torque ellipses at all points in the usable workspace & then defining the GII. This needs to be done for all link lengths in the parameter space & then the length which has the maximum GII is chosen for best performance. Therefore, an optimization algorithm is to be defined which finds the globally optimum parameter from a discretized parameter space within a discretized workspace.

First, an exhaustive search algorithm will be used to find the optimum link lengths. This will be used as a baseline to evaluate the performance of a global minimax algorithm described in [2]. The minimax algorithm is described in the appendix and proceeds as follows:

- First, a link length is chosen from the parameter space
- Minimum and maximum singular values are computed for the chosen link length at each point in the discretized workspace
- The points at which the minimum & maximum singular values are computed are assigned as the worst and the best point in the workspace
- Singular values are calculated at these points for each link length in the parameter space
- Link lengths which produce a performance worse than the performance of the initially selected link length are culled from the parameter space
- A new point is then chosen and the process above is repeated until only one value of link length remains in the parameter space
- This link length will have the highest value of GII & is chosen as the optimal link length for best performance

Example: Consider the palletizing robot shown in fig 5. The desired workspace is a 0.2*0.2 units square with its centre located at a distance of 1.5 units from the origin at an angle 60 degrees with the +ve x axis. The link length of the first link is taken as 1 units. The link length of the second link is to be chosen from a parameter space {0.3, 0.4, 0.5, 0.6,..., 1.7, 1.8, 1.9}

The workspace is discretized into m*m points. The middle value from the parameter space is chosen as the initial link length for the second link(1.1 units). The GII is computed to be 0.483. The ratio of the singular values at the best and worst points are computed for all other link lengths. They are found to be {0.247, 0.363, 0.494, 0.641, 0.804, 0.743, 0.634, 0.549, 0.483, 0.431, 0.389, 0.356, 0.330, 0.311, 0.297, 0.289, 0.291}. Of these, all links corresponding to values less than 0.483684 are culled from the workspace. The remaining links are {0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1}.

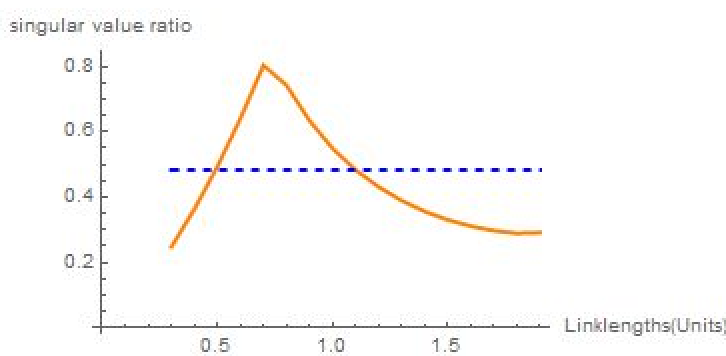

Fig 6: Plot of link lengths vs ratio of singular values in the first iteration

In the second iteration, the middle link is chosen as 0.7 and the GII is computed to be 0.804. The ratio of singular values at the best and worst points are computed for all the other links and they are found to be {0.494, 0.641, 0.804, 0.743, 0.634, 0.549, 0.483}. As 0.804 is greater than all other values, all links except 0.7 are culled from the parameter space. Therefore, the parameter space now consists of a single link 0.7. The search is terminated & the optimum value of the link length & GII is returned.

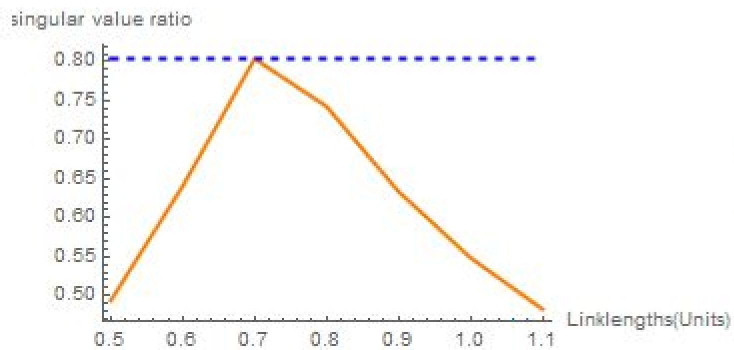

Fig 7: Plot of link lengths vs ratio of singular values in the second iteration

Since a parameter is only removed from the search space after it has produced singular values with a ratio worse than that of another parameter for which all singular values have been rigorously computed, the global optimum is guaranteed. Computational savings result from strategically exploring configurations which are likely to simultaneously identify many parameters as sub-optimal. Expected efficiency, however, relies on the presumption that within a continuous, bounded range of parameters, many of them, particularly those in close proximity to each other, will exhibit similarly favourable or poor behaviour at common configurations. This is a presumption that holds well in robot design problems.

Results:

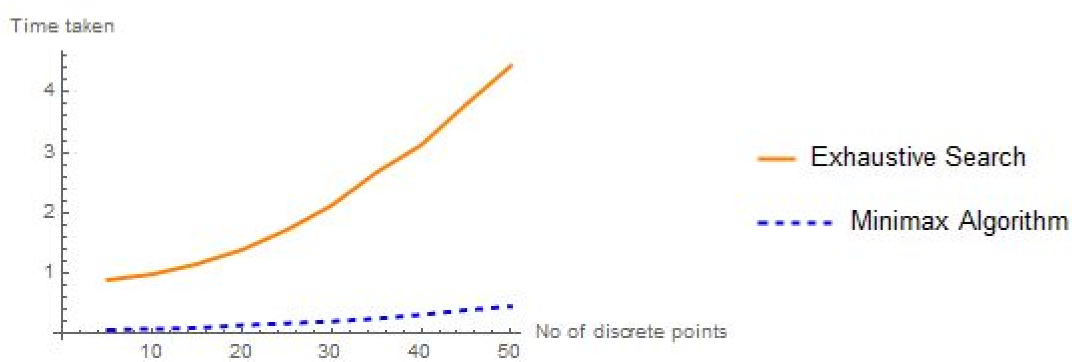

Fig 8: No of points in workspace vs time taken for computation

The number of points in the workspace versus the time taken for computation is plotted for by an exhaustive search and the minimax algorithm in Fig 9. For fair comparison, all other programmes were shut down and only Mathematica was running while computing the time taken for evaluations. Clearly, the minimax algorithm is orders of magnitude faster than an exhaustive search & serves as an efficient algorithm to compute the best link lengths for a given performance demand.

Benchmarking: The benchmarking results of the system used for computation were as follows:
"MachineName" -> "desktop-vlgh2fi"
"System" -> "Microsoft Windows (64-bit)"
"BenchmarkName" -> "WolframMark"
"FullVersionNumber" -> "11.3.0"
"Date" -> "November 21, 2018"
"BenchmarkResult" -> 1.193
"TotalTime" -> 11.6
"Results" -> {{"Data Fitting", 0.731}, {"Digits of Pi", 0.362}, {"Discrete Fourier Transform", 1.02}, {"Eigenvalues of a Matrix", 0.569}, {"Elementary Functions", 1.338}, {"Gamma Function", 0.451}, {"Large Integer Multiplication", 0.468}, {"Matrix Arithmetic", 0.766}, {"Matrix Multiplication", 0.626}, {"Matrix Transpose", 1.596}, {"Numerical Integration", 0.665}, {"Polynomial Expansion", 0.119}, {"Random Number Sort", 1.229}, {"Singular Value Decomposition", 0.726}, {"Solving a Linear System", 0.934}}}

References:
[1] Ashitava Ghosal, “Robotics: Fundamental Concepts and Analysis”
[2] L. Stocco, S.E. Salcudean, F. Sassani, “Fast Constrained Global Minimax Optimization of Robot Parameters”, Robotica, Int. J. Info., Edu. & Res. in Robotics & Artificial Intelligence, v. 16, pp. 595-605, 1998

Appendix:

List of Symbols

i = looping index

P_i = set of all parameters in parameter space

p_i = design parameter

$\hat{p}_i$ = best known design parameter

W = set of all positions in workspace

x = end-effector position

$\underset{\sim}{x}$ = position with the smallest singular value

$\tilde{x}$ = position with the largest singular value

$\underset{\sim}{\sigma}$ = minimum singular value at a position

$\tilde{\sigma}$ = maximum singular value at a position

$\underset{\sim}{\Sigma}_i : P_i \rightarrow \Re$ = minimum singular value upper bounding function

$\tilde{\Sigma}_i : P_i \rightarrow \Re$ = maximum singular value lower bounding function

s = performance measure; either GII or κ^{-1} as defined in (3)

$\hat{s}$ = performance measure of best known design parameter

ill Culling Algorithm

$$\text{Set} \quad i = 0, \quad \hat{s}_0 = 0 \tag{6}$$

$$\text{Set} \quad \begin{Bmatrix} \underset{\sim}{\Sigma}_0(p) = \infty \\ \tilde{\Sigma}_0(p) = 0 \end{Bmatrix}; \ \forall p \in P_0 \tag{7}$$

$$\text{Choose} \quad (p_0 = \hat{p}_0) \in P_0 \tag{8}$$

REPEAT

$$\text{Find} \quad \underset{\sim}{x}_i = \underset{x \in W}{\arg min} \ \underset{\sim}{\sigma}(p_i, x), \quad \tilde{x}_i = \underset{x \in W}{\arg max} \ \tilde{\sigma}(p_i, x) \tag{9}$$

$$\begin{array}{ll} \text{if} \ \left(\hat{s}_{i+1} = \dfrac{\underset{\sim}{\sigma}(p_i, \underset{\sim}{x}_i)}{\tilde{\sigma}(p_i, \tilde{x}_i)}\right) > \hat{s}_i; & \hat{p}_{i+1} = p_i \\ \text{otherwise} & ; \ \hat{p}_{i+1} = \hat{p}_i, \ \hat{s}_{i+1} = \hat{s}_i \end{array} \tag{10}$$

$$\text{Set} \quad \begin{Bmatrix} \underset{\sim}{\Sigma}_{i+1}(p) = min \ \{\underset{\sim}{\Sigma}_i(p), \ \underset{\sim}{\sigma}(p, \underset{\sim}{x}_i)\} \\ \tilde{\Sigma}_{i+1}(p) = max \ \{\tilde{\Sigma}_i(p), \ \tilde{\sigma}(p, \tilde{x}_i)\} \end{Bmatrix}; \ \forall p \in P_i \tag{11}$$

$$\text{Set} \quad P_{i+1} = \left\{ p \in P_i \ \middle| \ \frac{\underset{\sim}{\Sigma}_{i+1}(p)}{\tilde{\Sigma}_{i+1}(p)} > \hat{s}_{i+1} \right\} \tag{12}$$

$$\text{Choose} \quad p_{i+1} \in \underset{p \in P_{i+1}}{\arg max} \ \frac{\underset{\sim}{\Sigma}_{i+1}(p)}{\tilde{\Sigma}_{i+1}(p)} \tag{13}$$

$$i = i + 1 \tag{14}$$

$$\text{UNTIL} \quad \hat{p}_i = p_i \tag{15}$$