Path Planning for Assembly Mode Changes for the 3-RRR using Global Workspace Roadmaps

Wesley Au[1], Hoam Chung[2] and Chao Chen[3]
Monash University, Australia
[1]wesley.au@monash.edu
[2]hoam.chung@monash.edu
[3]chao.chen@monash.edu

Abstract

Path planning for assembly-mode changes in the joint space for parallel manipulators is a challenging task due to parallel singularities. The process is made even more difficult by multiple direct kinematic solutions that these manipulators generally exhibit, especially for manipulators with a high number of degrees of freedom. This paper proposes a path planning scheme to tackle this problem; a hierarchical path planning method that uses a *global workspace roadmap*. It is a very powerful tool which allows us to efficiently determine path feasibility in joint space and to find the existence of possible assembly-mode changes. However in our previous works, it has only been shown to be viable on 2-DOF manipulators. With enhancements to the algorithm, the proposed path planning process is applied to the 3-RRR, a 3-degree of freedom parallel manipulator which exhibits up to 6 direct kinematics solutions in its joint space. Through numerical experiments, we show that the global workspace roadmap allows us to check path validity of a proposed assembly-mode change in 3-degrees of freedom that is viable in both time and memory constraints.

1 Introduction

Path planning for parallel manipulators is difficult due to their often complex workspace boundaries as a result of singularities. In particular, parallel singularities pose a significant challenge for these types of manipulators, as coming close to a parallel singularity configuration results in the immediate loss of output force, eventuating in the loss of control of the driven payload. Naturally, path planning is utilised to avoid these singularities, but naïvely planning to avoid all singularity cases can result a much-reduced and fragmented overall reachable workspace.

Much work has been done in an effort to understand parallel singularities and how they can be avoided without severely fragmenting the reachable workspace. A simple method for avoid singularities to selectively choose manipulator's parameters such that parallel singularities lie beyond the proposed operating workspace [Jiang and Gosselin, 2006; Huang and Thebert, 2010]. However this does not address the original problem of small reachable workspaces for parallel mechanism.

The concept of assembly modes arises when disconnected regions of the reachable workspace are formed by the parallel singularity locus. Initially it was thought that these regions of workspace bound by the parallel singularity locus cannot be physically connected, but this has since been proven wrong [Innocenti and Parenti-Castelli, 1998], and it has been shown that it is possible to change assembly modes [Macho *et al.*, 2008]. Cusp points are an analytical method for forcing assembly-mode changes without encountering singularities [Zein *et al.*, 2008; Moroz *et al.*, 2010], where cusp points were found analytically for the unconstrained 3-RPR [Urízar *et al.*, 2012]. While cusp points give us a mechanism for forcing assembly-mode changes, using them can be difficult as they do not guarantee the correct assembly-mode change. It is also unknown to the authors whether cusp points can be feasibly utilised on parallel manipulators that are 3-DOF and greater.

The concept of joint-space path planning for parallel manipulators using a hierarchical path planning scheme was introduced in previous works [Au *et al.*, 2013; 2014], where analysis of the entire reachable workspace was performed using the *global workspace roadmap*. This is a graphical technique, which gives us the ability to efficiently determine whether assembly-mode changes are possible, and if it is, tell us which *workspace patches* to visit. A *workspace patch* is defined as a connected set of valid points in joint space, in which any combination of trajectories between configurations within the set must be singularity-free. A patch also represents one single direct kinematic solution in the task space. This system-

atic approach for joint-space analysis via the parallel singularity locus was shown to be successful determining in path feasibility and generating singularity-free paths accurately, particularly for assembly-mode changes. However, this has only been applied to a 2-DOF case (3-R$\underline{P}$R with one leg constrained), which may result in more complicated workspaces and global workspace roadmap, possibly generating sub-optimal pathing results.

In this paper we have applied the hierarchical path planning technique and global workspace roadmap to the unconstrained 3-DOF 3-RRR planar parallel manipulator. Concerns about time performance and memory usage in implementing this systematic approach on one more DOF were alleviated by improving the *rotary disk search* algorithm [Au *et al.*, 2014]. When applied to the 3-DOF example, the path planning was completed in less time and memory than 2-DOF in previous works - and can be run on mid-performance desktop workstations. Hence the viability of this method in both speed and memory-usage for up to 3-DOF manipulators is confirmed.

2 Path Planning Scheme

Our proposed path planning scheme for the 3-RRR utilises a hierarchical structure where it is split into two process: *global* and *local* path planning (Figure 1). However, the majority of the analysis is done before typical local path planning, which is the generation of the *global workspace roadmap*.

2.1 Global Workspace Roadmap

The global workspace roadmap is a topological graph representing the overall reachable workspace that is used for singularity-free trajectory generation [Au *et al.*, 2014]. This graph of nodes and links is generated by means of deconstruction of the overall reachable workspace in the joint space variables into smaller *workspace patches*, bounded by the parallel singularity locus in the joint space. The primary purpose of this graph is to easily determine whether a path between two points in configuration space is feasible. If one were to determine whether a path between points in configuration space exists, we must perform the following steps:

1. Determine which workspace patches contain starting and ending configurations.
2. Determine the location of the nodes in the GWR associated with these patches.
3. Perform a search to see if a path exists between the two nodes.
4. If a path is found in the GWR, we know a path between these configurations exists.

Each individual patch is represented as a unique node in the GWR. The boundaries of each patch are checked

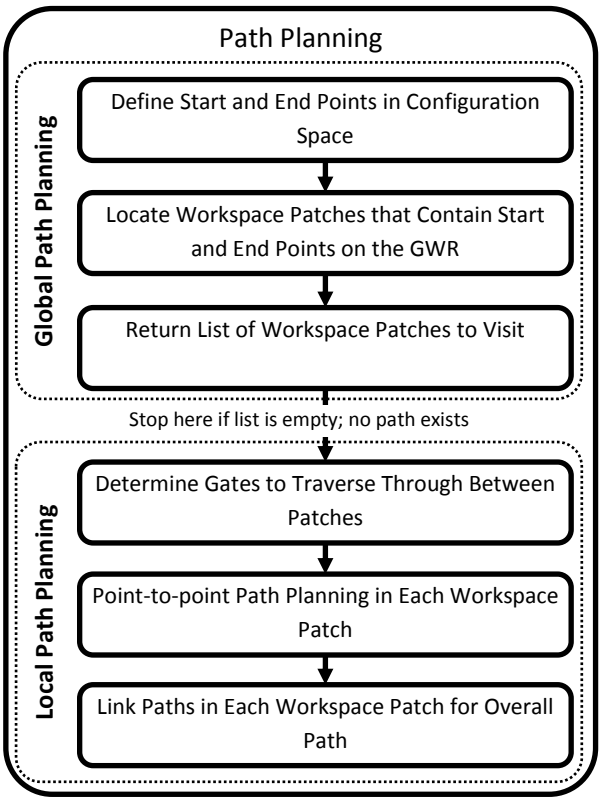

Figure 1: Work flow of the hierarchical path planning scheme.

for continuity to adjacent patches and are marked as links on the roadmap if adjacent patches are continuous and singularity-free. Once the roadmap is obtained, we complete the global path planning process of the hierarchical scheme by using the GWR to find path feasibility. Then local path planning algorithms complete the overall path in configuration space. Because each node represents a workspace patch, a set of connected and reachable configurations for a particular manipulator, we know in advance which nodes contain our starting and ending configurations during path planning.

2.2 Workspace Patch Boundaries

The configuration space is decomposed into workspace patches by projecting the parallel singularity locus into the 3-dimensional configuration space, the same method used in [Au *et al.*, 2014]. The logic behind this is that although parallel singularities do exist where the singularity locus is projected onto the joint space, not all direct kinematic solutions encounter singularities. A 2-dimensional example of this is shown in Figure 2 where the workspace surface represents the real roots of the univariate polynomial. The condition for the existence of a singularity for a certain layer of workspace is observed

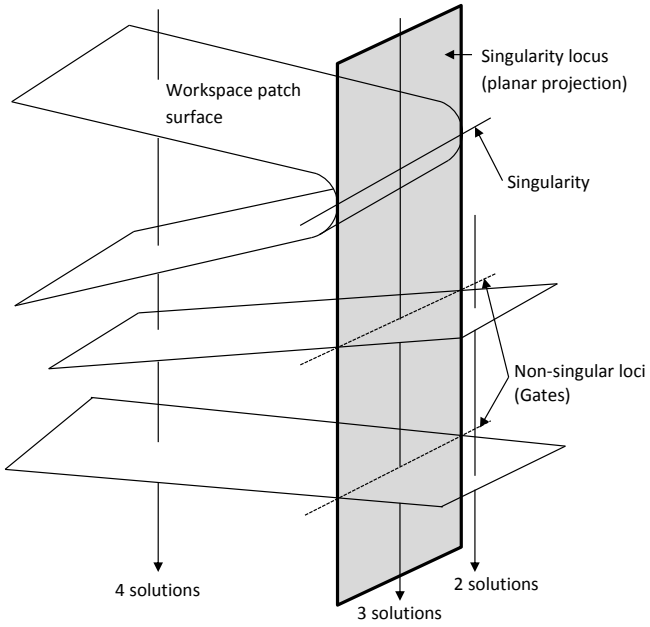

Figure 2: Singularity locus over multiple layers of workspace for a 2-dimensional case.

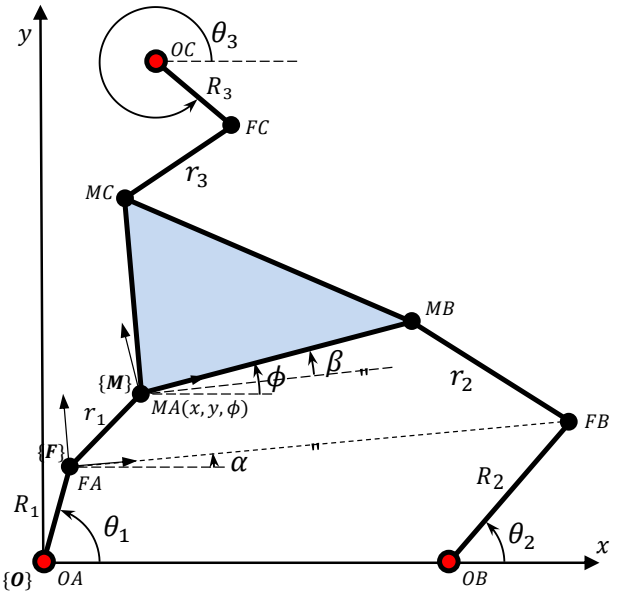

Figure 3: 3-RRR manipulator.

in polynomial's root multiplicity where it is greater than 1. This is observed as the workspace surface folding back over itself, where two real roots converge to the same value to cross into the complex plane. This feature gives a parallel manipulator the ability to change assembly modes without encountering parallel singularities, contradicting the parallel singularity locus assumptions. That means we must define for each workspace patch, what edge is a singularity and what edge is a *gate*. A *gate* is a non-singular boundary of a workspace patch, as observed in Figure 2. These boundaries segregate the overall reachable workspace and provide no hindrance in traversing between adjacent workspace patches. Gates play an important role in the global path planning as they will ensure the correct traversal between each workspace patch as dictated by the global path.

The parallel singularity locus is generated using the *n-solution map*, a discretised map that represents the number of direct kinematic solutions in the configuration space, solved by counting the number of real roots of the univariate polynomial derived from the kinematic equations. It was shown that the parallel singularity locus exists between differing regions of number of solutions in configuration workspace [Zaiter *et al.*, 2011], hence by simple inspection of the *n-solution map*, the parallel singularity locus was found without solving any equations related to the Jacobian.

3 3-RRR manipulator

Figure 3 shows the 3-DOF 3-RRR planar manipulator with active joint variables $\theta_{1,2,3}$ located at OA, OB and OC respectively. The output is located at MA, given as output variables $\mathbf{x} = (x, y, \phi)$.

To simplify the direct kinematic solutions, a secondary frame $\{F\}$ is defined at the passive joint of the first chain. Frame $\{M\}$ is positioned on the moving platform (and output) such that $MB = (x_{MB}, 0)$ and $MC = (x_{MC}, y_{MC})$ in $\{M\}$. This simplifies the univariate polynomial to 6th-order, but all DK solutions for frame $\{M\}$ are now with respect to $\{F\}$. However, the transformation matrix ${}^{O}_{F}T$ is easy to calculate to return DK solutions back to $\{O\}$.

The parameters used in this paper are assumed to be $OA(x, y) = (0, 0)$, $OB = (6, 0)$, $OC = (3, 2)$, $X_{MB} = 4$, $X_{MC} = 1$ and $Y_{MC} = 5$, where leg lengths $R = (3, 3, 3)$ and $r = (3, 4, 5)$ arbitrary unit lengths.

3.1 Direct Kinematics

Using kinematic mapping as defined by [Chen *et al.*, 1982], we can avoid working with non-linear simultaneous equations, generated by forward kinematics for each leg. Instead, the coefficients of the univariate polynomial can be generated straight from the co-ordinates of MB and MC in $\{M\}$ (which are constant), FB and FC in $\{F\}$ and $\mathbf{q} = (\theta_1, \theta_2, \theta_3)$. The univariate polynomial in the mapping variable X_3 [Hayes *et al.*, 2004] is given as

$$A_6 X_3^6 + A_5 X_3^5 + A_4 X_3^4 + A_2 X_3^2 + A_1 X_3 + A_0 = 0, \quad (1)$$

where

$$
\begin{aligned}
A_6 &= e_1^2 + a_6 d_5^2 + d_1^2 \\
A_5 &= 2d_1 d_2 + 2a_6 d_5 d_6 + 2e_1 e_2 \\
A_4 &= a_6 d_5^2 + 2d_1 d_3 + e_2^2 + 2e_1 e_3 + d_2^2 + 2a_6 d_5 d_7 + a_6 d_6^2 \\
A_3 &= 2d_1 d_4 + 2e_2 e_3 + 2a_6 d_6 d_7 + 2a_6 d_5 d_6 + \\
&\quad + 2d_2 d_3 + 2e_1 e_4 \\
A_2 &= 2e_2 e_4 + e_3^2 + d_3^2 + a_6 d_7^2 + 2d_2 d_4 + 2a_6 d_5 d_7 + a_6 d_6^2 \\
A_1 &= 2e_3 e_4 + 2d_3 d_4 + 2a_6 d_6 d_7 \\
A_0 &= a_6 d_7^2 + e_4^2 + d_4^2
\end{aligned}
$$

$$
\begin{aligned}
K_a &= -r_a^2 \\
K_b &= {X_{FB}}^2 - r_b^2 \\
K_c &= {X_{FC}}^2 + {Y_{FC}}^2 - r_c^2
\end{aligned}
$$

$$
\begin{aligned}
a_6 &= \tfrac{1}{4} K_a \\
b_1 &= -X_{FB} - X_{MB} \\
b_4 &= -X_{FB} + X_{MB} \\
b_6 &= \tfrac{1}{4} 2X_{FB} X_{MB} + K_b + {X_{MB}}^2 \\
b_7 &= \tfrac{1}{4} {X_{MB}}^2 - 2X_{FB} X_{MB} + K_b \\
c_1 &= -X_{FC} - X_{MC} \\
c_2 &= -Y_{FC} - Y_{MC} \\
c_3 &= -Y_{MC} + Y_{FC} \\
c_4 &= -X_{FC} + X_{MC} \\
c_5 &= X_{FC} Y_{MC} - Y_{FC} X_{MC} \\
c_6 &= \tfrac{1}{4} \left(2Y_{FC} Y_{MC} + 2X_{FC} X_{MC} + K_c + {Y_{MC}}^2 + {X_{MC}}^2 \right) \\
c_7 &= \tfrac{1}{4} \left({X_{MC}}^2 + {Y_{MC}}^2 - 2X_{FC} X_{MC} + K_c - 2Y_{FC} Y_{MC} \right) \\
d_1 &= -c_2 b_6 + c_2 a_6 \\
d_2 &= c_4 a_6 - c_4 b_6 + c_6 b_4 - a_6 b_4 \\
d_3 &= c_5 b_4 - c_2 b_7 + c_2 a_6 \\
d_4 &= -a_6 b_4 - c_4 b_7 + c_7 b_4 + c_4 a_6 \\
d_5 &= c_2 b_1 \\
d_6 &= -c_1 b_4 + c_4 b_1 \\
d_7 &= -c_3 b_4 \\
e_1 &= b_6 c_1 - b_1 c_6 + b_1 a_6 - a_6 c_1 \\
e_2 &= b_6 c_3 - a_6 c_3 - b_1 c_5 \\
e_3 &= -b_1 c_7 + b_1 a_6 - a_6 c_1 + b_7 c_1 \\
e_4 &= b_7 c_3 - a_6 c_3
\end{aligned}
$$

The conversion of variable X_3 to output variables $({}^F x_{MA}, {}^F y_{MA}, \beta)$ in $\{F\}$ is performed by

$$\beta = 2 \ \mathrm{Atan2}(\mathrm{X}_3, 1) \tag{2}$$

$$ {}^F x_{MA} = 2 \left(\frac{X_1 X_3 + X_2}{X_2^3 + 1} \right) \tag{3}$$

$$ {}^F y_{MA} = 2 \left(\frac{X_2 X_3 - X_1}{X_2^3 + 1} \right) \tag{4}$$

where

$$X_1 = \frac{d_1 X_3^3 + d_2 X_3^2 + d_3 X_3 + d_4}{d_5 X_3^2 + d_6 X_3 + d_7} \tag{5}$$

$$X_2 = \frac{e_1 X_3^3 + e_2 X_3^2 + e_3 X_3 + e_4}{d_5 X_3^2 + d_6 X_3 + d_7}. \tag{6}$$

Hence

$$ {}^F_M T = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & {}^F x_{MA} \\ \sin(\beta) & \cos(\beta) & {}^F y_{MA} \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

while given $\alpha = \alpha(\theta_1, \theta_2)$ and ${}^O FA = {}^O FA(\theta_1)$,

$$ {}^O_F T = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & {}^O x_{FA} \\ \sin(\alpha) & \cos(\alpha) & {}^O y_{FA} \\ 0 & 0 & 1 \end{bmatrix}. \tag{8}$$

Given these two transformation matrices, we can find ${}^O MA$ and its given angle ϕ from the resulting transformation matrix (9)

$$ {}^O_M T = {}^O_F T \, {}^F_M T. \tag{9}$$

3.2 Parallel Singularities

Because we are only concerned with joint space path planning, serial singularities for this manipulator do not pose any threat to controllability and hence can be generally ignored. However the same cannot be said for parallel singularities since it can result in the locking or loss of control of the end effector. There are three options for determining the location of the parallel singularities: the determinant of the parallel Jacobian, counting the number of direct kinematic solutions or using geometric means. As discussed previously, we have used the singularity locus via counting the number of DK solutions to determine workspace patch boundaries. However we still require a numerical method to calculate the singularity condition to aide in continuity checks. In analysing the parallel singularity profile, we can draw comparisons to the 3-R$\underline{\mathrm{P}}$R manipulator and use geometry for determining the conditions for parallel singularity. This can be done because the 3-R$\underline{\mathrm{P}}$R and the 3-RRR manipulators are mechanically the same, with the former's base joints situated on the latter's active link ends. The geometric condition for parallel singularity is satisfied when the axes of the 3 passive links of the manipulator meet at a single point (Fig. 4) or when they are all parallel (Fig. 5).

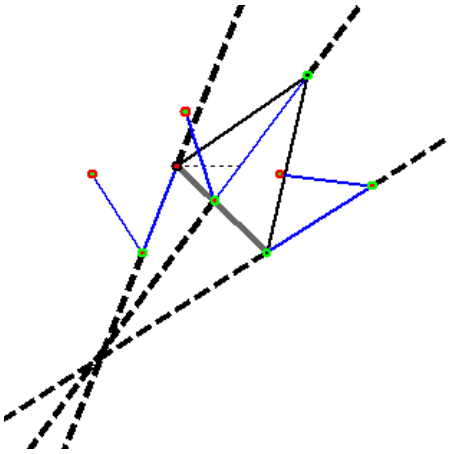

Figure 4: Parallel singularity where 3 axes coincide.

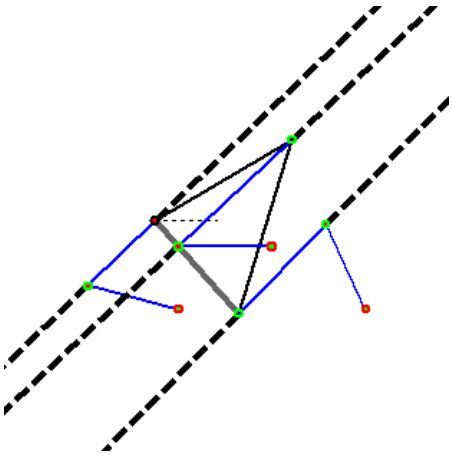

Figure 5: Parallel singularity where 3 axes are parallel.

The analytical solution to this method is much simpler than the traditional Jacobian and determinant method in that it does not involve very complicated algebraic equations [Zein *et al.*, 2008]. Derived from the kinematic and straight-line equations, a parallel singularity occurs when the following equation is satisfied:

$$Y_{FB}\cos(\theta_2)\sin(\theta_1-\theta_3)+\cos(\theta_1)\sin(\theta_2)(Y_{FC}\cos(\theta_3)+ (X_{FB}-X_{FC})\sin(\theta_3))-\sin(\theta_1)(X_{FB}\cos(\theta_3)\sin(\theta_2)+ \cos(\theta_2)(Y_{FC}\cos(\theta_3)-X_{FC}\sin(\theta_3)))=0. \quad (10)$$

As previously discussed, the projection of the parallel singularity manifold onto the joint space serves as the arbitrary boundaries for each proposed workspace patch. However, Equation (10) was not used in determining the parallel singularity locus, but rather was only used to determine the configuration's *aspect*.

An aspect is defined as a set of positions in the task space whose determinant of the parallel Jacobian (or in this case, Equation (10)) belongs to the same sign. Although there can only be positive or negative aspects, there may be multiple groups of aspects in each sign. It was shown that for the 3-RPR that there are only 2 aspects because there are no serial singularities if the active joint lengths were constrained [Coste, 2012]. Although this proof does not exist for the 3-RRR manipulator, it is understandable that there may be more than two aspects, given that the existence of serial singularities (ie. the workspace limit of each RR chain) can limit the overall reachable workspace.

4 Workspace Analysis

A summary of the generation of the global workspace roadmap is shown in Figure 6. In order to generate this map using numerical methods, we must discretise the joint space. However in doing so, we risk introducing noise into our results which can result in infeasibly planned paths. In this section we will discuss the methods used to handle the effects of discretisation of the joint space in order to accurately separate the overall workspace into patches for generation of the GWR.

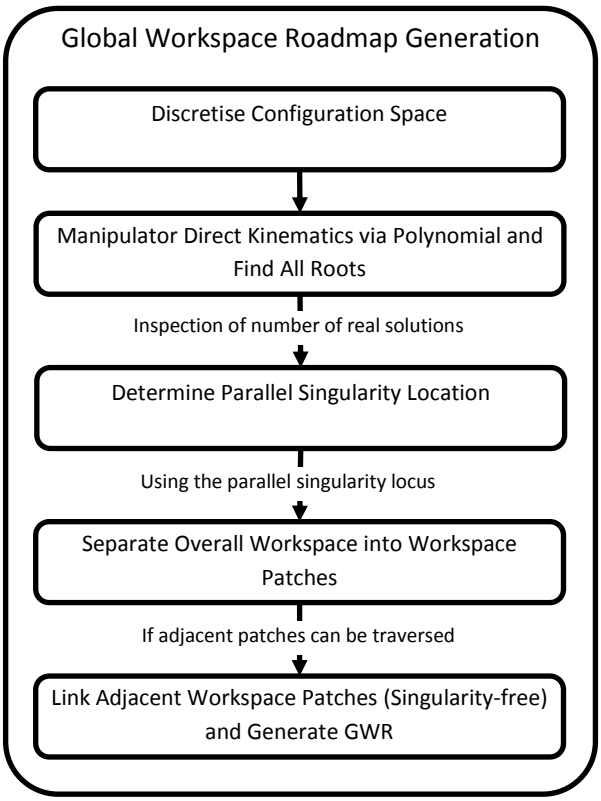

Figure 6: Work flow of the hierarchical path planning scheme.

4.1 Discretisation

The 3-dimensional joint space (θ_1, θ_2 and θ_3) was uniformly discretised over 51 steps between 0 and 2π inclusive. This results in a total of 132651 cells in the joint space, where each cell represents a unique manipulator posture. This allows easy implementation of searching algorithms. The minimal-connectivity scheme was used for this space, hence in our 3-dimensional case, we used 6-connectivity for connection to each adjacent cell. This restricts traversal between cells to occur at the cell faces only, which simplifies path planning and ensures traversal between cells is well-defined. Because higher connectivity schemes allow diagonal movement between cells via an edge or vertex, it is possible to pass through an infinitesimally small point in the grid space, hence we cannot guarantee singularity-free movement in anything greater than minimal-connectivity.

4.2 Search Algorithm

The search algorithm implemented is a modified version of the rotary disk search used in [Au *et al.*, 2014]. It was a search based on the depth-first search algorithm where a cell's edges represented vertices on a graph for the algorithm to process. The order of vertices to traverse was strictly ordered in an anti-clockwise fashion to ensure all cell edges were visited upon algorithm completion. In the new search algorithm, instead of arbitrarily ordering each vertex for each cell such that edges are visited anti-clockwise, the next direction (vertex) to visit is now ranked based on the adjacent cell's euclidean distance to the closest boundary cell in the proposed workspace patch. Hence all cells will now have their independent list of vertices to visit.

Assume the grey-marked boundary cells mark a singular configuration in joint space. Studying cell $(3, 4)$, whose cell is 1.41 cell units away from the closest boundary cell, the following ranking for the cell's edge is shown in Table 1.

(1,1) 1	(1,2) 1	0	0	0	0	0	1	2	3
(2,1) 1	0	0	0	0	0	1	1.41	(2,4) 2.24	2
1	0	0	0	0	0	1	(3,3) 2	(3,4) 1.41	(3,5) 1
0	0	0	0	1	1	1.41	2	(4,4) 1	0
0	0	0	0	0 (5,5)	2	2.24	2.24	1.41	1

Figure 7: (Left) Example of a proposed workspace patch with boundary cells marked as 1. (Right) Euclidean distance based on the boundary cells.

	Rotary Disk Search		Improved Search	
	Direction	Distance	Direction	Distance
1.	$+1$	1.00	-1	2.24
2.	$+2$	1.00	-2	2.00
3.	-1	2.24	$+1$	1.00
4.	-2	2.00	$+2$	1.00

Table 1: Comparison of edge rankings at cell $(3, 4)$. Direction indicates the dimensions to increment (1: vertical, 2: horizontal), where $\pm n$ increments dimension n by ± 1.

In the regular rotary disk search, if we began the search algorithm at cell $(3, 4)$, then the first adjacent cell we will visit is $(4, 4)$, which is scored 1.00, indicating that this cell is located directly next to a singular cell. Because of this cell's proximity to a singularity, its DK solutions may be ill-conditioned, possibly resulting in continuity-checking issues. If the entire patch contained 4 DK solutions, the worst-case scenario is that the wrong solution is linked when transitioning from cells $(3, 4)$ to $(4, 4)$. Not only will this generate continuity issues when generating a local path in the workspace patch, but will also generate an incorrect global workspace roadmap. For this reason, we have modified the search algorithm to its current version. In the improved search algorithm, we instead visit cell $(2, 4)$, which moves further away from any cells that are closer to singularities. This new algorithm significantly increases the accuracy of selecting the correct DK solutions to link for continuity.

The final modification to the search algorithm is that once a cell that is scored 1.00 is encountered, the search terminates at that cell and returns to cell it had previously visited. This ensures that the search does not unintentionally follow along any ill-conditioned singular surfaces. The resultant search pattern when the algorithm is applied to the workspace patch in Figure 7 is shown in Figure 8.

4.3 Continuity

Determining continuity of adjacent cells accurately in multiple-solution situations is very important in this path planning application. Not only does it help construct each workspace patch in the reachable workspace, but also determines which adjacent patches are continuous, which forms the basis of the global workspace roadmap generation. Because there are multiple direct kinematics solutions, we must define a continuity condition that accurately selects the correct solution, given any number of DK solutions between postures.

Because the moving platform of the 3-RRR is defined by 3 passive joints defined in frame $\{M\}$, ie MA, MB and MC (Figure 3), due to their dependence spatial co-

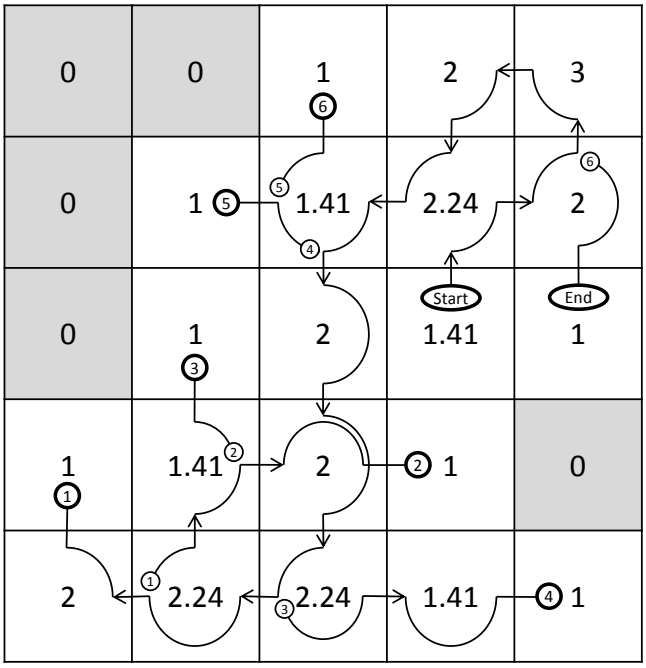

Figure 8: Search algorithm process. Dead ends and resume points are indicated.

ordinates only we can set our continuity conditions based on the delta of passive joints M. That is, we can condense the continuity condition into a comparison of a single number between the original configuration O, and the next configuration n, where n denotes the n-th solution in the adjacent cell by the following equation:

$$\Delta_n = |MA_n - MA_O| + |MB_n - MB_O| + |MC_n - MC_O|\,, \quad (11)$$

The continuity conditions between configurations O and n is therefore

$$\min(\Delta_n), \quad (12)$$

where n denotes the best direct kinematic solution in the next posture.

5 Path Planning Results

With the current discretisation scheme of 51 steps between 0 to 2π and 3-RRR parameters defined in Section 3, we wish to perform an assembly-mode change in the *positive* aspect at

$$\mathbf{q} = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 \end{bmatrix} = \begin{bmatrix} 0.2513 & 1.5080 & 3.1416 \end{bmatrix}. \quad (13)$$

All results were generated on a system with MATLAB R2012b, on Windows 7 x64 with an Intel i7 2600K processor at 4.5 GHz with 16 GB of RAM.

5.1 Global Workspace Roadmap

The global workspace roadmap for this manipulator for the above parameters is shown in Figure 9. MATLAB's Bioinformatics toolbox was used to generate this graph. The naming convention for each workspace patch is as follows:

$$\boxed{\pm xyyz}$$

- $\pm$ indicates the positive or negative aspect.
- x indicates the number of DK solutions (assembly modes) in this set of connected points in configuration space.
- yy indicates the unique patch number, ordered descending of the number of cells the patch contains.
- z indicates the solution (assembly mode) ID for the particular patch.

For example, patch [+3022] indicates that this set of configurations contains 3 DK solutions (assembly modes), is the 2nd-largest patch of all 3-solution patches and represents the 2nd assembly mode out of 3. It is important to note that patches [+3021], [+3022] and [+3023] all contain the same points (cells) in configuration space and hence these workspace patches represent the 3 assembly modes.

All links between workspace patches are given a score, based on how close the gates are to singularities. Although this measurement is arbitrary, it aides in selecting a good *global path* for smoother trajectories between workspace patches. Generally, the path of shortest cost is chosen if more than one path exists.

As it turns out, the configuration defined in (13) has 3 direct kinematic solutions, and they are contained in workspaces +3011, +3012 and +3013. Therefore we want to check the connectivities of

AM change 1: $\boxed{+3011} \longleftrightarrow \boxed{+3012}$
AM change 2: $\boxed{+3011} \longleftrightarrow \boxed{+3013}$
AM change 3: $\boxed{+3012} \longleftrightarrow \boxed{+3013}$.

5.2 Global Path

The global path dictates the list of workspace patches to visit, if a path between two patches exist. For the joint values given in (13), the following global paths were generated:

Path 1: $\boxed{+3011} \leftrightarrow \boxed{+2012} \leftrightarrow \boxed{+3022} \leftrightarrow \boxed{+2031} \leftrightarrow \leftrightarrow \boxed{+3012}$ (Cost: 1.8)
Path 2: $\boxed{+3011} \leftrightarrow \boxed{+2012} \leftrightarrow \boxed{+3022} \leftrightarrow \boxed{+2011} \leftrightarrow \leftrightarrow \boxed{+3013}$ (Cost: 1.4)
Path 3: $\boxed{+3012} \leftrightarrow \boxed{+2011} \leftrightarrow \boxed{+3013}$ (Cost: 1.4).

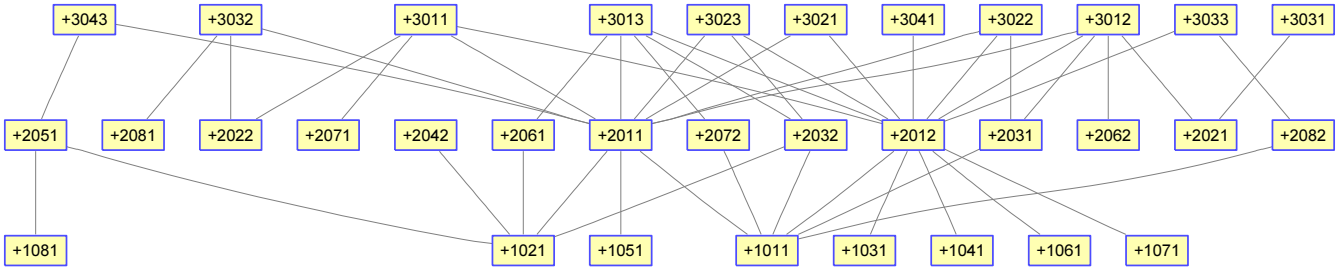

Figure 9: Global workspace roadmap for the *positive aspect only*. The costs between each workspace patch have been omitted for clarity.

From this result, we can conclude that all 3 assembly modes are connected, as shown by the existence of all 3 paths.

5.3 Local Path

For this paper, we will focus on performing local path planning on *Path 3* only, as results can be seen more clearly with a more simple global path. Figures 10 and 11 represent the two assembly modes (2 and 3) we wish to plan a path to transition between.

The blue circles on the figure indicate active revolute joints, with their respective active links are marked thick blue. Passive joints are marked with green circles and the output frame location (base of the platform) is indicated in red (the red circle is the output frame origin). Faint grey lines indicate axes of the passive links to determine parallel singularity conditions.

The path planning algorithm used within each workspace patch is the probabilistic roadmap, which was proven to be feasible performance-wise in multi-dimensional applications [Hsu *et al.*, 2006; LaValle, 2006]. The resultant path to achieve the proposed assembly-mode change in both joint and task space as returned by the PRM are given Figures 12 and 13. The parallel singularity profile is shown on figure 14. Note that these paths are artificially smoothed for illustration purposes.

We can observe in Figures 12 and 13 that we have successfully achieved an assembly-mode change in that we have created a looping path in the joint space that returns to its original configuration while the platform is in a different position in task space. The path remains singularity free as shown in Figure 14.

5.4 Notes on Performance

With this set-up, pre-processing of the workspace is completed in 2 minutes (separation of the joint space into patches), in which this data is stored in memory for local path planning algorithms to use. This was achieved using parallel processing with a 4-core, 8-threaded processor.

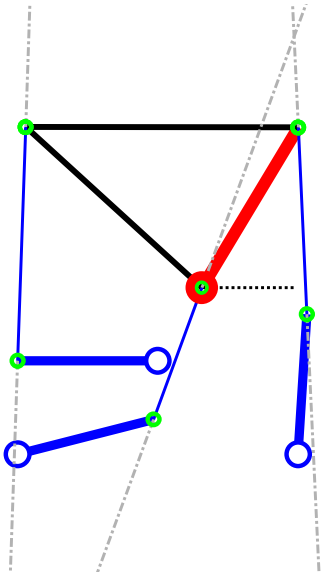

Figure 10: Assembly mode 2 for given **q**.

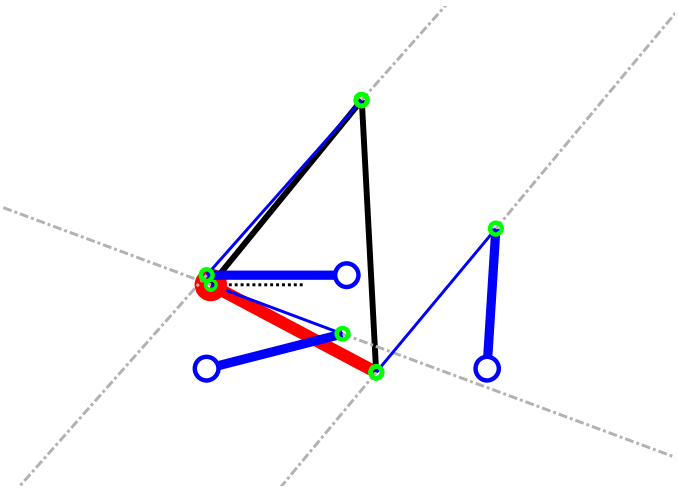

Figure 11: Assembly mode 3 for given **q**.

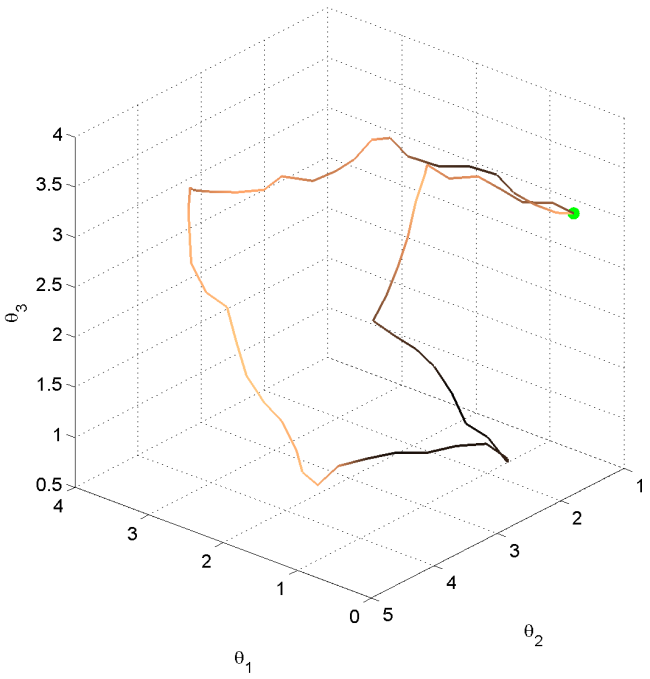

Figure 12: Path in joint space. Start and end point marked in green. The brightness of the path indicates its proximity to a parallel singularity (brighter is closer).

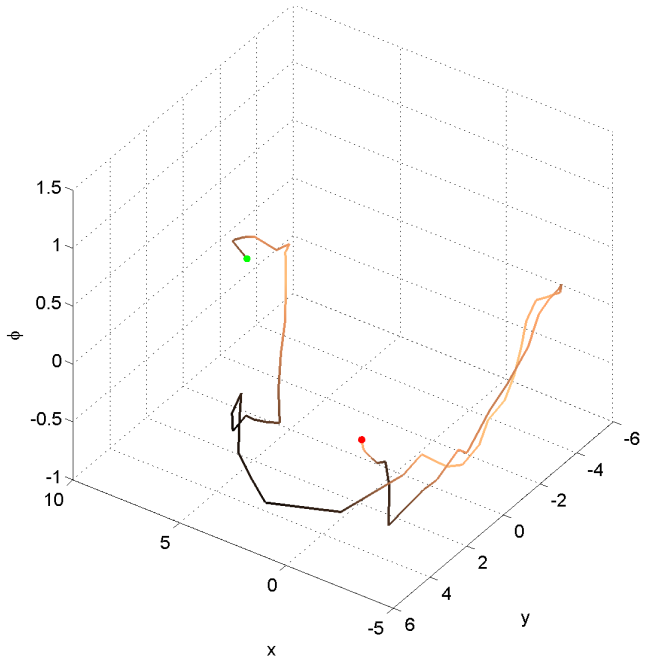

Figure 13: Path in task space. Start and end points of the path marked in green and red respectively.

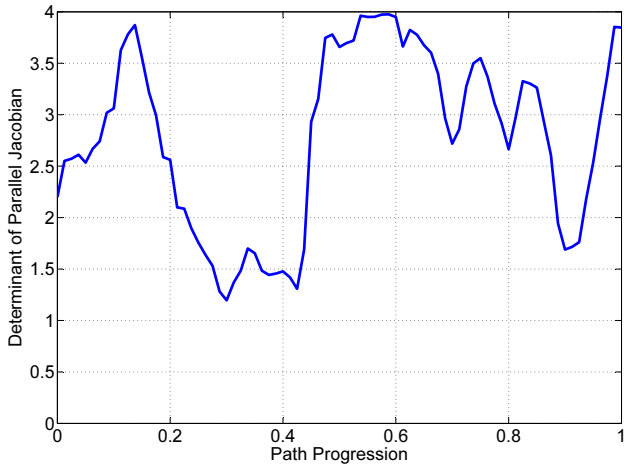

Figure 14: Parallel singularity profile of the path.

Processing time

In general, most of the computational time is used to determine the direct kinematic solutions and their continuity with adjacent cells. Although the 2 minutes is a somewhat arbitrary time to achieve to complete this pre-processing task, the goal is to still minimise this as much as possible. If we assume it takes 5 seconds to complete 1 dimension of joint space path planning ($5^3 = 125$ seconds for a 3-DOF manipulator), then exponential growth means that at for 6-DOF manipulators such as the Stewart platform [Stewart, 1965], we can expect processing to take $5^6 = 15625$ seconds, or approximately 4 hours. However, pre-processing is performed only once before any path planning is conducted, hence time performance for pre-processing is not critical for real-time applications, only local path planning is.

The time to analyse the global workspace roadmap for path existence is negligible as observed in our experiments, hence the taken to complete local path planning process depends on the path planning algorithm used thereafter. The probabilistic roadmap algorithm implemented in this paper is an iterative algorithm that improves the quality of solution the more iterations it is allowed to run. In general, we have allowed up to 2 minutes for the PRM to produce a shortest path that avoids parallel singularity areas in the workspace. This is not ideal for real-time applications, but the point is to prove feasibility and to test the attainable quality of the path generated.

Memory

Memory usage for the entire process was 1.7 GB for storing workspace patch and GWR data, and an extra 450 MB per core for parallel processing. The memory usage peaked at approximately 6 GB. These values were recorded from the operating system's performance mon-

itor. Although this is a highly feasible fesult, given that the intervals chosen for each joint variable is not considered coarse, we can expect this value to exponentially grow as this algorithm is applied to higher dimensions. In the event that memory is constrained, the only solution is to use a coarser interval for the joint space and apply interpolation methods to obtain a smooth trajectory.

6 Conclusion

The hierarchical path planning scheme was successfully demonstrated on the 3-RRR, utilising the entire reachable joint space as needed. The joint space was discretised and some processing of the workspace was applied in order to minimise noise introduced with this method. Using an efficient algorithm, the entire joint space was separated into smaller workspace patches and the global workspace roadmap was created successfully. Using this map, we were able to determine quickly whether a proposed assembly-mode change was possible. Finally, the local path planning scheme, utilising the probabilistic road map method, we have generated a singularity-free trajectory to achieve this assembly-mode change.

We have seen how quickly the global workspace roadmap can tell us what assembly-mode changes are possible. This is very valuable for path planning in higher DOF systems if a quick solution is needed to determine path feasibility.

References

[Au *et al.*, 2013] Wesley Au, Hoam Chung, and Chao Chen. Path planning of the 3-RPR using global workspace road maps. In *3rd IFToMM International Symposium on Robotics and Mechatronics*, Singapore, 2013.

[Au *et al.*, 2014] Wesley Au, Hoam Chung, and Chao Chen. Generation of the global workspace roadmap of the 3-RPR using rotary disk search. *Mechanism and Machine Theory*, 78:248–262, 2014.

[Chen *et al.*, 1982] Yong Chen, Oene Bottema, and Bernard Roth. Rational rotation functions and the special points of rational algebraic motions in the plane. *Mechanism and Machine Theory*, 17:335–348, 1982.

[Coste, 2012] Michel Coste. A simple proof that generic 3-RPR manipulators have two aspects. *Journal of Mechanisms and Robotics*, 4:1–6, 2012.

[Hayes *et al.*, 2004] M.D.J Hayes, P.J. Zsombor-Murray, and Chao Chen. Unified kinematic analysis of general parallel manipulators. *ASME Journal of Mechanical Design*, 126(5):866–874, 2004.

[Hsu *et al.*, 2006] David Hsu, Jean-Claude Latombe, and Hanna Kurniawati. On the probabilistic foundations of probabilistic roadmap planning. *The International Journal of Robotics Research*, 25(7):627–643, July 2006.

[Huang and Thebert, 2010] Ming Z. Huang and Jean-Luc Thebert. A study of workspace and singularity characteristics for design of 3-dof planar parallel robots. *International Journal of Advanced Manufacturing Technology*, 51:789–797, 2010.

[Innocenti and Parenti-Castelli, 1998] C. Innocenti and V. Parenti-Castelli. Singularity-free evolution from one configuration to another in serial and fully-parallel manipulators. *Journal of Mechanical Design*, 120:73–79, 1998.

[Jiang and Gosselin, 2006] Qimi Jiang and Clément M. Gosselin. The maximal singularity-free workspace of planar 3-RPR parallel mechanisms. In *International Conference on Mechatronics and Automation*, Luoyang, China, 2006.

[LaValle, 2006] Steven M. LaValle. *Planning Algorithms*. Cambridge, 2006.

[Macho *et al.*, 2008] E. Macho, O. Alturazza, C. Pinto, and A. Hernandez. Transitions between multiple solutions of the direct kinematic problem. In Jadran Lenarčič and Philippe Wenger, editors, *Advances in Robot Kinematics: Analysis and Design*, pages 301–310. Springer Science+Business Media B.V. 2008, 2008.

[Moroz *et al.*, 2010] G. Moroz, F. Rouiller, D. Chablat, and P. Wenger. On the determination of cusp points of the 3-RPR parallel manipulators. *Mechanism and Machine Theory*, 45(11):1555–1567, 2010.

[Stewart, 1965] D. Stewart. A platform with six degrees of freedom. *Proc Instn Mech Engrs*, 180(15):371–386, 1965.

[Urízar *et al.*, 2012] Mónica Urízar, Víctor Petuya, Oscar Alturazza, Eric Macho, and Alfonso Hernández. Assembly mode changing in the cuspidal analytic 3-RPR. *IEEE Transactions on Robotics*, 28(2):506–513, April 2012.

[Zaiter *et al.*, 2011] Abdel Kader Zaiter, Philippe Wenger, and Damien Chablat. A study of the singularity locus in the joint space of planar parallel manipulators: special focus on cusps and nodes. In *4th International Congress Design and Modeling of Mechanical Systems*, Sousse, Tunisia, 2011.

[Zein *et al.*, 2008] Mazen Zein, Philippe Wenger, and Damien Chablat. Non-singular assembly-mode changing motions for 3-RPR parallel manipulators. *Mechanism and Machine Theory*, 43:480–490, 2008.