

A Linkage Based Solution Approach for Determining 6 Axis Serial Robotic Travel Path Feasibility

R.J. Urbanic
University of Windsor

R. Hedrick
CAMufacturing Solutions Inc.

Ana M. Djuric
Wayne State University

ABSTRACT

When performing trajectory planning for robotic applications, there are many aspects to consider, such as the reach conditions, joint and end-effector velocities, accelerations and jerk conditions, etc. The reach conditions are dependent on the end-effector orientations and the robot kinematic structure. The reach condition feasibility is the first consideration to be addressed prior to optimizing a solution. The 'functional' work space or work window represents a region of feasible reach conditions, and is a sub-set of the work envelope. It is not intuitive to define. Consequently, 2D solution approaches are proposed. The 3D travel paths are decomposed to a 2D representation via radial projections. Forward kinematic representations are employed to define a 2D boundary curve for each desired end effector orientation. The feasible region for all orientations is determined by the use of Boolean intersections of the boundary curves or by overlapping valid regions in which points are placed. Combining the tool path and functional work space regions allow designers to readily visualize regions of concern. A KUKA and Comau robot are used to illustrate the methodology.

CITATION: Urbanic, R., Hedrick, R., and Djuric, A., "A Linkage Based Solution Approach for Determining 6 Axis Serial Robotic Travel Path Feasibility," *SAE Int. J. Mater. Manf.* 9(2):2016, doi:10.4271/2016-01-0336.

INTRODUCTION

The development of a wide array of automation strategies and industrial manipulators has facilitated manufacturing operations such as assembly, inspection, material handling, grinding, painting, applying sealant, and so forth. Many tasks are repetitive and are performed in hazardous environments (i.e., a paint booth). A comprehensive set of robot structures have since been designed and built to fulfill the various industry requirements. These multi-Degrees of Freedom (DOF) structures are highly complex in their form and control. Most manipulators used in the industry today are articulated with a combination of six rotational joints linked serially. These robotic arms are inspired from human arms and their ability to rotate, position, and orient hands as shown in [Figure 1](#) [1].

There are several distinct 'standard' kinematic structure families: for example, the ABB and the Fanuc robot family types vary based on the twist angle. This 6 axis serial link structural form provides the manipulators with a great deal of flexibility, and dexterity. In several environments, manufacturing cells are utilized, where machines and material handling systems (i.e. conveyors) are integrated with several 6 axis industrial robots, allowing designers to rearrange system elements for new product variants and/or address volume changes.

Significant research has been performed with respect to modelling industrial robots, and robotic systems. Much of this research focuses on the development and optimization of a manipulator design and its functionality within an industrial setting. Due to the kinematic structures of the serial six axis industrial robots, there are challenges related to identifying the kinematic singularities, the complex inverse kinematic solution(s), trajectory planning, and travel path validation for a design configuration.

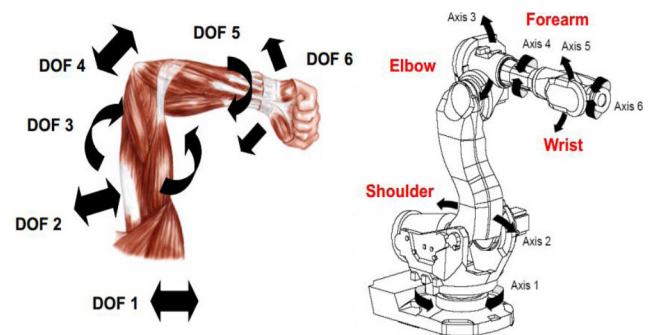


Figure 1. Articulated robot arm inspired from a human arm [1].

Various simulation tools allow designers to experiment with various solution approaches. Typically, the Denavit-Hartenberg notation (i.e., DH parameters) is presented to provide the background for the subsequent analyses, as the DH parameters are commonly used in the robotics domain. They provide a standard methodology to write the kinematic equations of a manipulator or end-effector. Each joint in a serial kinematic chain is assigned a coordinate frame. Using the DH notation [2], 4 parameters are needed to describe how a frame relates to a previous frame as shown in Figure 2. The kinematic structure for a KUKA KR robot family is shown in Figure 3, and the Comau family is shown in Figure 4.

After assigning coordinate frames the four D-H parameters can be defined as following:

- a_i - Link length is the distance along the common normal between the joint axes
- α_i - Twist angle is the angle between the joint axes
- θ_i - Joint angle is the angle between the links
- d_i - Link offset is the displacement, along the joint axes between the links

The numerical values for the general KUKA KR D-H parameters are $d_1 = -865$, $d_4 = 1200$, $d_6 = -210$, $a_1 = -410$, $a_2 = 1000$, and $a_3 = 45$. The D-H parameters for KUKA KR and Comau family of robots are summarized in Tables 1 and 2. The similarities and differences can be readily highlighted. The nomenclature used for the end-effector orientation angle for this work is presented in Figure 5.

The workspace for different configurations and degrees of freedom has been extensively analyzed [3,4,5,6,7,8,9,10,11,12,13,14] for several diverse applications and differing manipulator types. However, the workspace or work envelope region does not ensure the feasibility of reaching a point in space for an end-effector orientation. The reach conditions are dependent on both the end-effector orientations and the robot kinematic structure, and the functional space needs to be determined.

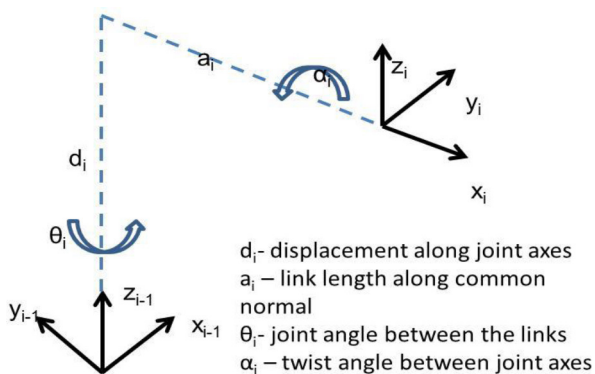


Figure 2. D-H parameters of a frame i with respect to frame $i-1$ illustrating the joint angle θ_i , and twist angle α_i , link length a_i , and the displacement along the joint axes d_i .

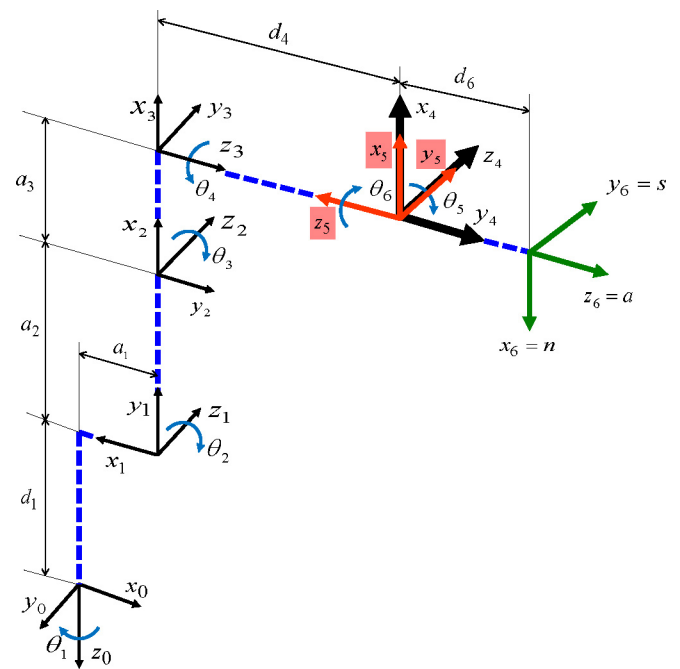


Figure 3. Kinematic structure for KUKA KR family

The D-H parameters can be employed in algorithms to determine the bounded reachable space for a given end-effector length and orientation, but this can also be determined by reducing the complex 3D problem into a series of 2D problems [15, 16]. The forward kinematics are solved by fixing θ_4 and θ_6 . This results in the $\theta_2, \theta_3, \theta_5$ rotation angles being analyzed via forward kinematics.

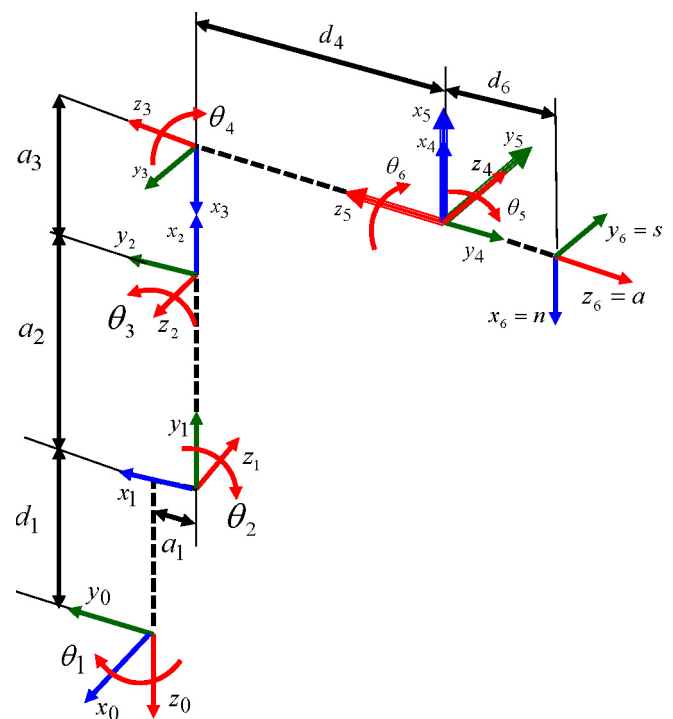


Figure 4. Kinematic structure for the Comau family

Table 1. The D-H parameters for the KUKA KR robot

i	D-H parameters				Joint Limits	
	d_i	θ_i	a_i	α_i	Minimum	Maximum
1	d_1	180°	a_1	-90°	-185°	185°
2	0	90°	a_2	0°	-40°	93°
3	0	0°	a_3	-90°	-210°	58°
4	d_4	0°	0	90°	-350°	350°
5	0	0°	0	90°	-120°	120°
6	d_6	180°	0	180°	-350°	350°

Table 2. The D-H parameters for the Comau robot

i	D-H parameters				Joint Limits	
	d_i	θ_i	a_i	α_i	Minimum	Maximum
1	1	d_1	90°	a_1	-90°	-171°
2	2	0	90°	a_2	180°	-55°
3	3	0	180°	a_3	90°	-162°
4	4	d_4	180°	0	-90°	-280°
5	5	0	0°	0	90°	-120°
6	6	d_6	180°	0	180°	-270°

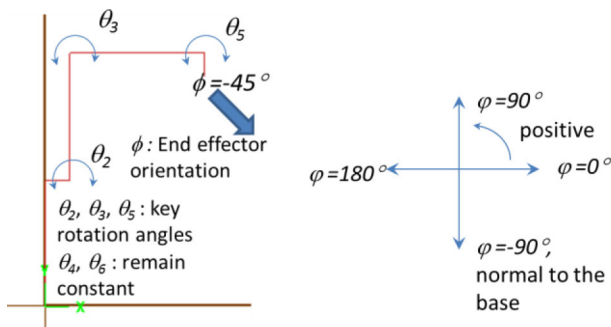


Figure 5. Reduced link set, and end-effector nomenclature for this research

A 2D feasible region can be determined, and then revolved around θ_1 (Figure 6). A complex 3D travel path can also be reduced to a 2D data set using radial projections from an origin point. This addresses the influence of θ_1 in 2D space. Combining these 2D solutions (Figure 7) the feasibility related to reach for a position-orientation set for a travel path (or travel path set) can be readily evaluated.

The derived solution approach developed using various CAD/CAM software tools (Mastercam®, Rhinoceros® and Grasshopper®), Workspace 5 (a robotics simulation tool), MATLAB, and Excel spreadsheet tools.

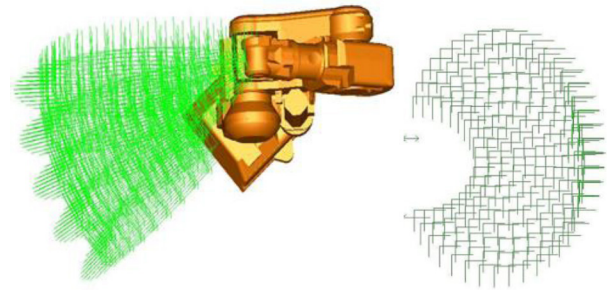
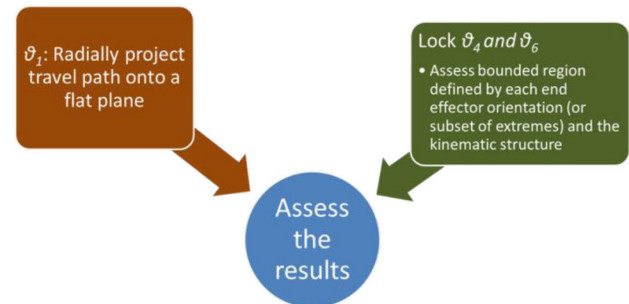
Figure 6. Reachable points for each θ_1 rotation angle, and end-effector criteria

Figure 7. Solution approach for travel path feasibility

SOLUTION APPROACH

Trajectory Projections

There are many software tools and optimization strategies are employed to create travel paths, and more are continuing to be developed [17,18,19]. Trajectory planning for coordinated motion of a robot for a particular task, within a cell containing multiple robots or a rail or positioning table, conveyors, and so forth are being developed and improved upon. However, the solution depends on the initial positioning and orientation, as well as criteria with respect to minimizing joint wrap, energy requirements, velocity changes, least travel time, etc. The inverse kinematics are challenging due to redundancies associated with joints 4 and 6 in addition to the 'elbow up or down' and other joint 'reflection' solutions. However, prior to optimizing a solution, it is essential to determine whether a solution is feasible. To quickly determine this, the travel paths are radially projected (or squashed) onto a plane. The general procedure is:

- Determine reach points or
 - Create travel path (curves)
 - Determine segment length Δl
 - Segment a curve path into points by Δl
 - Create points at the segment end points
- Establish a reference point vector $(\Delta x_r, \Delta y_r, \Delta z_r)$ from the joint 1 (0,0,0) to the part (0,0,0)
- Determine Cartesian x, y, z values of the points with respect to the reference point using the Pythagorean Theorem
- Calculate the 2D 'radial projection' equivalent Cartesian x, y, 0 values of the points with respect to the reference point
- Compare results to the functional work space boundaries

Consider the cube shown in Figure 8 (a). The reference position is 0,0,0, and the initial z depth is 8 units. From the reference point (0,0,0), the radial projection of point (0, 10, 8) is (8, 10, 0) (Figure 8 (b)). The resulting corner points for this reference point are shown in Figure 8 (c).

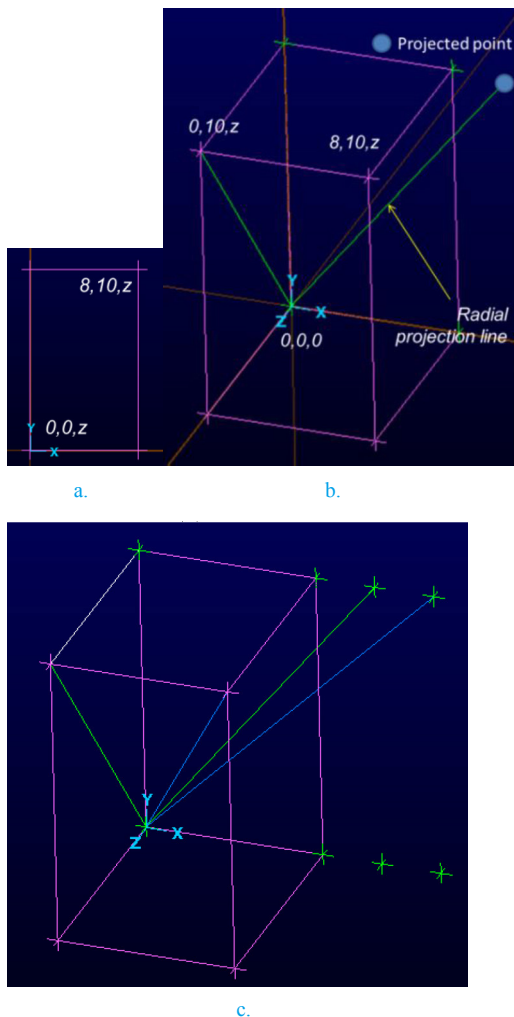


Figure 8. Radial projections for points at the corners of (a) a cube, (b) a projected point for a vector, and (c) the complete point set, with the two elevated corner vectors illustrated

Keeping the same block size, but offsetting the block two units (Figure 9 (a)) provides a different reach condition (Figure 9 (b)). The boundary box for the original point set is much wider due to the origin position, as shown in Figure 9 (c).

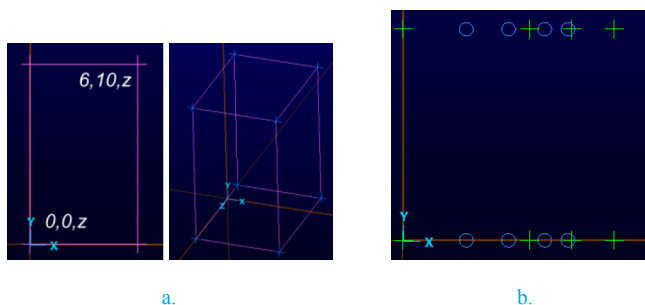


Figure 9.

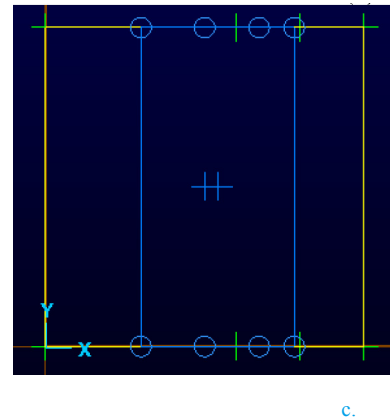


Figure 9. (cont.) (a) Offset cube, (b) projected points for the original block position (+ label), and for the offset position (o label), and the bounding regions for both point sets.

For a curve based travel path, positions on the curve need to be determined and then projected. A piece-wise linear model is created. The granularity is situation dependent. An example of a curve complete with points and their projected positions is shown in the Appendix (Figures 24, 25, 26, 27). This can be extended for all curves or curve sets, i.e., multiple operations such as those for a laser cladding travel path, which are presented in the case studies section.

The end effector orientation may vary for a travel path. A component may be picked in one end effector orientation and placed at another, or the orientation may be at a constant angle to a curve, but the curve wraps (Figure 10). Therefore, all the reachable regions need to be determined. Additional constraints can be added to the basic reach feasibility: no joint wrap, elbow up, etc. Additional constraints can be applied once the basic reach conditions are established. Different methodologies are described in the next section.

Empirical Point Generation Methodology

One basic empirical method to determine the reach space for any orientation angle ϕ is to increment the joints from the minimum joint limit to the maximum by an increment parameter Δ after fixing θ_i to 0 [15]. The end point is determined via the forward kinematics, and stored. A general process flow is developed for 2, 3, 5 or 6 axis articulated robots. Once this extensive point set is generated, sort strategies can be implemented to determine the boundary region points (Figure 10), and curve fitting methods employed to determine the bounded regions.

To initiate the sorting and boundary identification process, the points are sorted in y, and then x. The start point is at the minimum x coordinate for the minimum y value. Scanning from left to right, a boundary edge will occur when there is a positive change in y coordinate values for a large negative change in x coordinate values. After a 'left to right' scan is completed, the points are resorted, and a 'bottom to top' scan is performed. There are several practical tolerancing considerations with this method, as the points are not on an even grid, and there may not be matching x or y coordinate values.