

# Removing the Singularities of Serial Manipulators by Transforming the Workspace

John E. Lloyd  
Computer Science Dept., University of British Columbia  
Vancouver, B.C., Canada  
lloyd@cs.ubc.ca

## Abstract

A new method of handling the kinematic singularities of serial robotic manipulators is proposed. The idea is to transform the manipulator's workspace  $\mathcal{W}$  into a desingularized workspace  $\mathcal{W}^*$ . Robotic motions can then be planned anywhere in  $\mathcal{W}^*$ , subject to limits on spatial velocity and acceleration, and the resulting joint velocities and accelerations will be well-behaved and bounded.  $\mathcal{W}^*$  differs from  $\mathcal{W}$  only near a singularity surface, where a deformation is applied in the direction normal to the surface. While the technique does not handle self-motion singularities and may not be practical in some cases, it is very easy to implement for certain manipulators, such as the PUMA, which is studied in the paper. When applicable, the method offers various advantages when compared with other methods of singularity control.

## 1 Introduction

Kinematic singularities of robotic manipulators correspond to regions in the manipulator's workspace where execution of a prescribed spatial motion may lead to extremely large values of the joint velocities, accelerations, and higher time derivatives. In particular, for revolute-jointed serial robots, the outer workspace boundary usually corresponds to a singularity, making it quite difficult to perform tasks there. This boundary singularity reduces the usable workspace of the manipulator and is particularly evident to users of telerobotic systems.

In this paper, we demonstrate the possibility of desingularizing a robot by locally deforming its workspace  $\mathcal{W}$  in the vicinity of singularities. Motions can then be planned and directed anywhere within this deformed workspace  $\mathcal{W}^*$ , using only conventional bounds  $V_s$  and  $A_s$  on the spatial velocity and acceleration, and the resulting joint velocities and accelerations will remain well-behaved and bounded, in proportion to  $V_s$  and  $A_s$ .

Though not able to handle self-motion singularities (such as the well-known wrist singularity), and possibly not practical for robots with complex geometry, this method has a number of advantages that should make it of interest in cases where it can be applied:

1. It can be extremely easy to implement, as our experi-

ence with the PUMA (Section 5) suggests. Also, this implementation requires little or no modification to existing kinematic and control routines.

2. The desingularization is *complete*, in that joint velocities, accelerations, and higher time derivatives are always well-behaved and bounded.
3. Path errors can be understood and quantified exactly in terms of the applied workspace deformation.
4. The resulting motions are time efficient, and the robot will not get "stuck" at the singularity.
5. The method can be employed directly in online path generation.

The combination of these items represents a useful contribution to the state of the art. Although our method will produce path errors, these are easy to quantify (item 3) and may not be problematic in telerobotic or sensor-driven applications. Note that for the case where a robot is required to follow a fixed path without error, other methods can be used [1].

Only *regional* singularities (i.e., those associated with the translational part of the robot's kinematics) will be considered in this paper. When analyzing regional singularities, the workspace  $\mathcal{W}$  can be considered a subset of  $\mathfrak{R}^3$ , and a point  $\mathbf{X} \in \mathcal{W}$  can be described by a vector  $\mathbf{p}$ . The application of workspace reparameterization to more general singularities has not yet been studied.

Our analysis will be based mainly on a theoretical result [9] for non-redundant serial manipulators. However, experiments involving the boundary singularity of a planar 3R robot have shown that workspace reparameterization can be applied to redundant serial robots as well.

## 2 Related Work

Singularities are traditionally defined with respect to the manipulator Jacobian  $\mathbf{J}$ , which relates joint velocities  $\boldsymbol{\vartheta}$  to spatial velocities  $\mathbf{v}$  according to  $\mathbf{v} = \mathbf{J}\boldsymbol{\vartheta}$ . Near a singularity,  $\mathbf{J}$  becomes ill-conditioned and execution of a prescribed  $\mathbf{v}$  may require arbitrarily high values of  $\boldsymbol{\vartheta}$ . A common way to handle this is to use a damping factor to desingularize

the inverse computation which produces  $\boldsymbol{\vartheta}$  [2, 3, 4]. Problems exist with this, however: exact motion in the degenerate (singular) direction is usually not possible, the resulting path errors are hard to quantify, and accelerations and higher time derivatives are difficult to explicitly control.

If higher order derivative information is utilized, then it becomes possible to produce motion along a singular direction [5], and it has been shown that proper time-scaling of a trajectory can permit motion along any fixed path containing singularities [6, 7, 1]. This problem is closely connected with reparameterizing a prescribed path in order to remove the singularity [8, 9], the notion of which has been used to improve conventional Jacobian-based control [10]. Enhancing Jacobian-based computations with information from the manipulator's Hessian has also been studied [11, 12].

### 3 Workspace Desingularization

In this paper, the notion of desingularizing a manipulator's path is extended to the idea of desingularizing the entire workspace. We begin by reviewing the former.

Suppose that a manipulator is requested to follow some spatial path  $\mathbf{X}(s)$ , where  $s$  is a scalar path parameter. Let  $\boldsymbol{\vartheta}(s) \equiv (\vartheta_1(s), \dots, \vartheta_M(s))^T$  be the corresponding inverse kinematic solution. Motion along the path is specified in terms of a timing  $s(t)$  for the path parameter. Near a singularity,  $\vartheta'_j(s)$  and/or higher derivatives may become very large, implying, from the chain rules

$$\dot{\vartheta}_j = \vartheta'_j(s) \dot{s} \quad \text{and} \quad \ddot{\vartheta}_j = \vartheta'_j(s) \ddot{s} + \vartheta''_j(s) \dot{s}^2$$

that  $\dot{\vartheta}_j$  and/or  $\ddot{\vartheta}_j$  may also become very large for non-zero values of  $\dot{s}$  and  $\ddot{s}$ .

In *desingularizing* a path, the idea is to find a reparameterization  $s(\lambda)$  for which the corresponding  $\vartheta'_j(\lambda)$  (and higher derivatives) exist and are well-behaved at singularities. The work of Kieffer [8] showed that this can sometimes be done by equating  $\lambda$  with the arc-length of the spatial curve in  $\mathfrak{R}^7$  formed by  $\boldsymbol{\vartheta}(s)$  and  $s$ . In [9], it was shown that if  $\mathbf{X}(s)$  is piecewise analytic and the robot is non-redundant, then within either a right or left neighborhood of a singularity at  $s = s_0$ , an analytic reparameterization can always be formed using

$$\lambda = |s - s_0|^{1/\eta}, \quad (1)$$

where  $\eta$  is an integer guaranteed to be no greater than the root multiplicity associated with the singularity. In other words, even if  $\boldsymbol{\vartheta}(s)$  is not well behaved at  $s_0$ ,  $\boldsymbol{\vartheta}(\lambda)$  will be well behaved, and in fact analytic, at  $s = s_0, \lambda = 0$ . Because most singular points of non-redundant robots are associated with a root multiplicity of 2 (i.e., only two branches of the inverse kinematic solution meet there), this implies that in most situations a singularity can be removed

by the reparameterization

$$\lambda = \sqrt{|s - s_0|}. \quad (2)$$

Without loss, only such cases will be considered in the sequel.

We now turn our attention to the workspace  $\mathcal{W}$ . The regional singularities often form smooth 2D surfaces in  $\mathcal{W}$ , which frequently coincide with the boundary of  $\mathcal{W}$ . Let  $\mathcal{S} \in \mathcal{W}$  be such a singular surface, let  $\mathbf{p}_0$  be a point on  $\mathcal{S}$ , and assign a Euclidean  $x$ - $y$ - $z$  coordinate system at  $\mathbf{p}_0$  such that the  $x$  axis is aligned with the surface normal  $\mathbf{n}$  and points into  $\mathcal{W}$  (Figure 1). This means that  $x$  is parallel to the degenerate direction of the singularity, and so constant speed motion along  $x$  is not possible at  $\mathbf{p}_0$  (i.e., at  $x = 0$ ) without causing  $\boldsymbol{\vartheta}$  to blow up. Correspondingly,  $\partial\vartheta_j/\partial x$  is infinite at  $x = 0$  for at least one joint  $j$ . On the other hand, since the  $y$  and  $z$  axes are tangential to  $\mathcal{S}$ , motion in those directions is feasible and so  $\partial\vartheta_j/\partial y$  and  $\partial\vartheta_j/\partial z$  will exist.

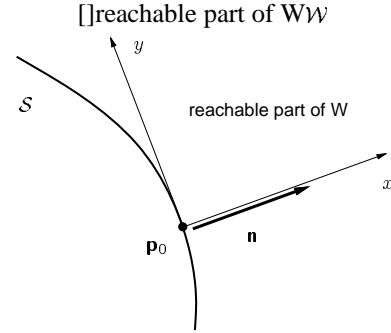


Figure 1: Cross section showing  $x$  and  $y$  axes of a local coordinate system centered at a point  $\mathbf{p}_0$  on  $\mathcal{S}$ .

Moreover, if we reparameterize the  $x$  axis by  $\lambda = \sqrt{x}$ , then  $\partial\vartheta_j/\partial\lambda$  will exist, for all  $j$ , at  $x = \lambda = 0$ . This follows from (2), by considering the  $x$  axis itself to be a path with  $s \equiv x, x \geq 0$  and  $s_0 = 0$ . This suggests that smooth robot motions can be created at  $\mathbf{p}_0$  by using the coordinate system  $\lambda$ - $y$ - $z$  in place of  $x$ - $y$ - $z$ .

Now, the idea is to apply the above transformation to every point on  $\mathcal{S}$ , creating a new workspace  $\mathcal{W}^*$  in which the effects of the singularity at  $\mathcal{S}$  have been removed. We do this as follows. For each  $\mathbf{p} \in \mathcal{W}$ , let  $d(\mathbf{p})$  be its distance to  $\mathcal{S}$ , let  $\mathbf{p}_0$  be the associated closest point on  $\mathcal{S}$ , and let  $\mathbf{n}$  be the normal directed from  $\mathcal{S}$  towards  $\mathbf{p}$ . Both  $\mathbf{p}_0$  and  $\mathbf{n}$  are well-defined if  $\mathcal{S}$  is smooth and  $d(\mathbf{p})$  is sufficiently small. Also, define the function  $\Gamma(d)$  (Figure 2) by

$$\Gamma(d) = \begin{cases} d + d_b & \text{if } d \geq d_b, \\ 2\sqrt{d_b}\sqrt{d} & \text{otherwise,} \end{cases} \quad (3)$$

where  $d_b$  controls the distance from  $\mathcal{S}$  at which the reparameterization starts to take effect. The point  $\mathbf{p}^* \in \mathcal{W}^*$

corresponding to  $\mathbf{p}$  is then given by

$$\mathbf{p}^* = \mathbf{p}_0 + [\Gamma(d(\mathbf{p})) - d_b] \mathbf{n}. \quad (4)$$

Qualitatively, this produces a local stretching of  $\mathcal{W}$  perpendicular to  $\mathcal{S}$ . Points in  $\mathcal{W}$  whose distance from  $\mathcal{S}$  exceeds  $d_b$  are unaffected, while the transformation of  $\mathcal{S}$  into  $\mathcal{W}^*$ , denoted by  $\mathcal{S}^*$ , is located a distance  $d_b$  away from  $\mathcal{S}$  (Figure 3).

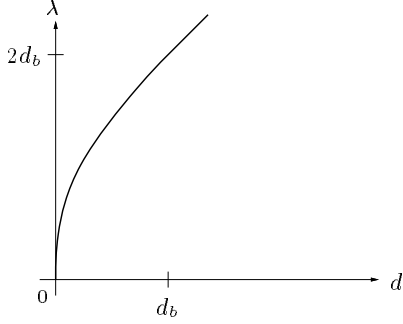


Figure 2: The function  $\Gamma(d)$ .

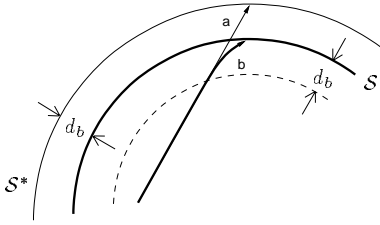


Figure 3: Transformation of  $\mathcal{W}$ , bounded by  $\mathcal{S}$  (dark circle), into  $\mathcal{W}^*$ , bounded by  $\mathcal{S}^*$  (light circle). Transformation affects only those points  $\mathbf{p} \in \mathcal{W}$  in the zone between  $\mathcal{S}$  and the dotted line indicating  $d(\mathbf{p}) = d_b$ . A straight-line motion (a) produced in  $\mathcal{W}^*$  may produce an actual motion (b) which is bent somewhat inside this zone.

By the arguments above, we can expect the inverse kinematic function to be differentiable everywhere in  $\mathcal{W}^*$ , implying that any motion generated in  $\mathcal{W}^*$  (subject to constraints on velocity and acceleration) should result in joint velocities and accelerations which are also well-behaved.

The actual motion in  $\mathcal{W}$  corresponding to a motion prepared in  $\mathcal{W}^*$  can be determined by the inverse transformation  $\mathcal{W}^* \rightarrow \mathcal{W}$ . If for each  $\mathbf{p}^* \in \mathcal{W}^*$  we let  $\lambda(\mathbf{p}^*)$  be its distance to  $\mathcal{S}^*$ , and  $\mathbf{n}^*$  and  $\mathbf{p}_0^*$  are the associated normal and closest point on  $\mathcal{S}^*$ , then the corresponding  $\mathbf{p} \in \mathcal{W}$  is given by

$$\mathbf{p} = \mathbf{p}_0^* + [\Gamma^{-1}(\lambda(\mathbf{p}^*)) + d_b] \mathbf{n}^*, \quad (5)$$

where

$$\Gamma^{-1}(\lambda) = \begin{cases} \lambda - d_b & \text{if } \lambda \geq 2d_b, \\ \lambda^2/(4d_b) & \text{otherwise.} \end{cases} \quad (6)$$

In particular, motions prepared as straight lines in  $\mathcal{W}^*$  will result in actual motions which are bent slightly within the threshold zone defined by  $d(\mathbf{p}) \leq d_b$  (Figure 3). There will not, however, be any bending of motions which are perpendicular to  $\mathcal{S}$ .

The continuity of the transform  $\mathcal{W}^* \rightarrow \mathcal{W}$  is the same as the continuity of  $\Gamma^{-1}(\lambda)$ . As presently defined,  $\Gamma^{-1}(\lambda)$  has  $C(1)$  continuity, meaning that position and velocity continuity will be preserved when mapping from  $\mathcal{W}^*$  back to  $\mathcal{W}$ . If necessary, smoother functions  $\Gamma(d)$  and  $\Gamma^{-1}(\lambda)$  can be produced by smoothing the junction point at  $d = d_b$ .

## 4 Usage and Discussion

A standard way to employ workspace desingularization is as follows:

- Map the necessary task goals into  $\mathcal{W}^*$ ;*
- Compute the required motions;*
- Map back into  $\mathcal{W}$  and compute the inverse kinematics.*

For example, suppose a robot is required to track a reference point  $\mathbf{p}_d$  which is changing in real-time. Every control cycle, we can then: (a) determine  $\mathbf{p}_d^* \in \mathcal{W}^*$  corresponding to  $\mathbf{p}_d$ ; (b) adjust the robot's position  $\mathbf{p}^*$  in  $\mathcal{W}^*$  so as to try and rendezvous with  $\mathbf{p}_d^*$ ; and (c) find  $\mathbf{p} \in \mathcal{W}$  corresponding to  $\mathbf{p}^*$  and solve for the required joint values. The resulting  $\dot{\boldsymbol{\theta}}$  and  $\ddot{\boldsymbol{\theta}}$  will be well-behaved at singularities.

In step (c), joint solution determination is certainly easiest if the manipulator has a direct kinematic solution. If not, then a Jacobian-based solution can probably be used without too much difficulty, since forming the motion in  $\mathcal{W}^*$  implicitly provides much of the step-size adjustment needed near singularities.

To remove all the singularities of a manipulator, a reparameterization is required in the vicinity of each singular surface  $\mathcal{S}_i$ . It is possible that the above method may be difficult to apply to some surface types, or near the intersection of two or more surfaces (for the PUMA, however, surface intersection could be handled using composite mappings, as described in Section 5). Also, when two singular surfaces meet, the kinematic root multiplicity will be greater than 2, and so by equation (1), a higher order reparameterization may be required. For the PUMA, the inverse kinematics decouples sufficiently that this is not necessary.

Workspace desingularization is probably not directly applicable to self-motion singularities, which cannot be handled by path reparameterization [9].

An important quantity to consider is the threshold  $d_b$ , which controls how much of  $\mathcal{W}$  is deformed when constructing  $\mathcal{W}^*$ . There is no "required" value for  $d_b$ . Lowering  $d_b$  reduces the path error (Figure 3), but also necessitates lowering the spatial velocity and acceleration bounds,  $V_s$  and  $A_s$ , used to produce motions in  $\mathcal{W}^*$ . One way to

accommodate this would be to vary  $V_s$  and  $A_s$  over different regions of  $\mathcal{W}^*$ . While the exact relationship between  $d_b$ ,  $V_s$ ,  $A_s$ ,  $\vartheta_j$  and  $\dot{\vartheta}_j$  can in principle be derived analytically, it is probably easiest to determine a good value for  $d_b$  empirically with a few experiments.

## 5 Desingularizing a PUMA

In this section we will construct a  $\mathcal{W}^*$  which removes the elbow and shoulder singularities of a PUMA. The surfaces associated with both of these comprise an outer sphere  $\mathcal{S}_o$  with radius  $R_o$ , an inner sphere  $\mathcal{S}_i$  with radius  $R_i$ , and an internal cylinder  $\mathcal{C}$  with radius  $R_c$  (Figure 4). Together, these define the workspace boundary [13]. If  $a_2$ ,  $a_3$ ,  $d_3$ , and  $d_4$  denote the significant Denavit- Hartenberg parameters for the PUMA [14, 15], and we define  $l_4 \equiv \sqrt{d_4^2 + a_3^2}$ , then it can be shown that  $R_c = d_3$ , and

$$R_o = \sqrt{(a_2 + l_4)^2 + d_3^2} \quad \text{and} \quad R_i = \sqrt{(a_2 - l_4)^2 + d_3^2}.$$

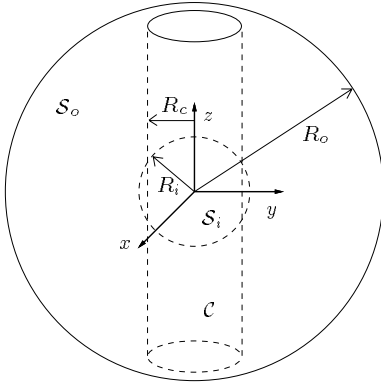


Figure 4: Surfaces associated with the elbow and shoulder singularities of the PUMA (not to scale).

The workspace  $\mathcal{W}$  lies outside of  $\mathcal{C}$  and between  $\mathcal{S}_o$  and  $\mathcal{S}_i$ .  $\mathcal{S}_o$  is associated with the elbow singularity, reached when the arm is fully outstretched and  $\vartheta_3 = \pi/2 - \tan^{-1}(a_3/d_4)$ .  $\mathcal{C}$  is associated with the shoulder singularity, and  $\mathcal{S}_i$  is the counterpart to the elbow singularity that occurs when the arm completely folds up on itself. On a real PUMA, limits on  $\vartheta_3$  prevent  $\mathcal{S}_i$  from being reached, and so we need consider only  $\mathcal{S}_o$  and  $\mathcal{C}$ .

The deformation at  $\mathcal{S}_o$  will be considered first, with  $d_o$  denoting the associated deformation size  $d_b$ . Because  $\mathcal{S}_o$  is a sphere centered on the origin, the transformations given by equations (4) and (5) are particularly simple. For any point  $\mathbf{p} \in \mathcal{W}$ , we have  $\mathbf{n} = -\mathbf{p}/\|\mathbf{p}\|$ ,  $\mathbf{p}_0 = -R_o \mathbf{n}$ , and

$d(\mathbf{p}) = R_o - \|\mathbf{p}\|$ . Equations (4) and (5) then become:

$$\mathbf{p}^* = \frac{R_o + d_o - \Gamma(R_o - \|\mathbf{p}\|)}{\|\mathbf{p}\|} \mathbf{p}, \quad (7)$$

$$\mathbf{p} = \frac{R_o - \Gamma^{-1}(R_o + d_o - \|\mathbf{p}^*\|)}{\|\mathbf{p}^*\|} \mathbf{p}^*. \quad (8)$$

The transformed surface  $\mathcal{S}_o^*$  is a slightly larger sphere of radius  $R_o + d_o$  (Figure 5).

The deformation around  $\mathcal{C}$  is also easy to compute. Since the distance to  $\mathcal{C}$  is independent of  $z$ , it occurs entirely in the  $x$ - $y$  plane, so that  $p_z^* = p_z$ . If we then let  $\mathbf{p}_{xy} \equiv (p_x, p_y)^T$  and  $\mathbf{p}_{xy}^* \equiv (p_x^*, p_y^*)^T$ , and let  $d_c$  denote the desired deformation size  $d_b$ , the calculations corresponding to (4) and (5) are:

$$\mathbf{p}_{xy}^* = \frac{R_c - d_c + \Gamma(\|\mathbf{p}_{xy}\| - R_c)}{\|\mathbf{p}_{xy}\|} \mathbf{p}_{xy}, \quad (9)$$

$$\mathbf{p}_{xy} = \frac{R_c + \Gamma^{-1}(\|\mathbf{p}_{xy}^*\| + d_c - R_c)}{\|\mathbf{p}_{xy}^*\|} \mathbf{p}_{xy}^*. \quad (10)$$

The transformed surface  $\mathcal{C}^*$  is a slightly narrower cylinder of radius  $d_3 - d_c$ .

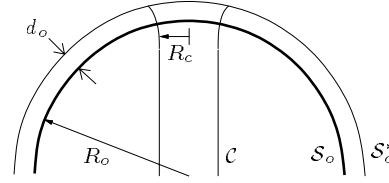


Figure 5: Cross section through the workspace, showing the transformation of  $\mathcal{S}_o$  (heavy circle) into  $\mathcal{S}_o^*$  (light circle), along with the induced bending of  $\mathcal{C}$  near  $\mathcal{S}_o^*$ .

The transformation of  $\mathcal{W}$  at points close to both  $\mathcal{S}_o$  and  $\mathcal{C}$  can be handled using composite mappings. First, the deformation at  $\mathcal{S}_o$  is performed, transforming  $\mathcal{W}^*$  into  $\mathcal{W}_o^*$  according to equation (8). Then the deformation at  $\mathcal{C}$  is performed, transforming  $\mathcal{W}_o^*$  into  $\mathcal{W}^*$  according to equation (10). The only complication is that the first deformation modifies the shape of  $\mathcal{C}$  near  $\mathcal{S}_o^*$ , so that its radius, rather than being constant, tapers outward slightly (Figure 5). To accommodate this, the value of  $R_c$  used in (10), which normally equals  $d_3$ , is made a function of the value of  $z$  within  $\mathcal{W}_o^*$ . This function  $R_c(z)$  is tedious to solve analytically, but can be quickly determined online using a spline approximation<sup>1</sup>. Moreover, by replacing  $d_c$  with  $d_c' = d_c + R_c(z) - d_3$ , it is possible to ensure that the transformed  $\mathcal{C}^*$  in  $\mathcal{W}^*$  remains a cylinder of radius  $d_3 - d_c$ .

In summary, the desingularized workspace  $\mathcal{W}^*$  looks exactly like  $\mathcal{W}$  (Figure 4), except that  $\mathcal{S}_o^*$  is slightly larger and  $\mathcal{C}^*$  is slightly narrower.

<sup>1</sup>The surface normal of  $\mathcal{C}$  near  $\mathcal{S}_o^*$  also contains a small  $z$  component, but this effect can be ignored.

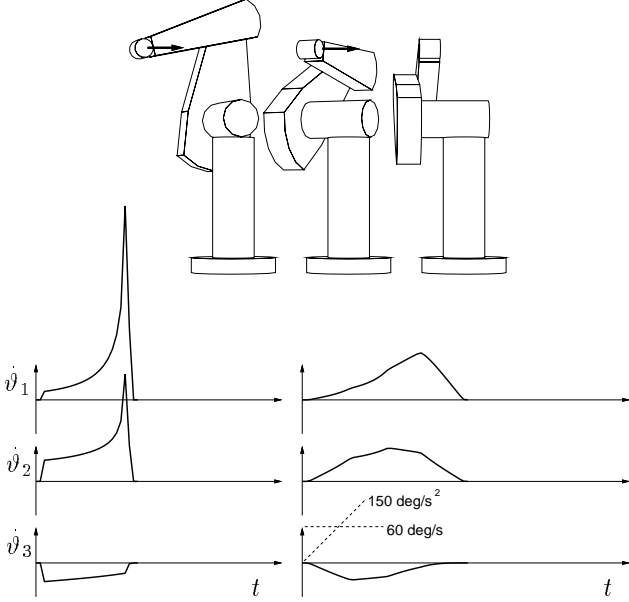


Figure 6: Straight line motion from  $\mathbf{p} = (0, 400, 300)^T$  into a shoulder singularity at  $\mathbf{p} = (0, d_3, 300)^T$ . Nominal velocities are shown on the left; desingularized velocities are shown on the right. Plots are scaled as indicated by the dotted lines on the rightmost graph for  $\dot{\vartheta}_3$ .

## 6 Demonstrations

The effectiveness of the reparameterization is now shown. Each example involves a straight-line motion toward a singular point, and is illustrated by a three-frame animation. Time profiles are shown of the resulting joint velocities, first for a nominal motion prepared in  $\mathcal{W}$  (with no singularity compensation applied), and second for a motion prepared in  $\mathcal{W}^*$ . The nominal motions were executed with constant path speed  $\dot{s} = V_s$ . The motions prepared in  $\mathcal{W}^*$  were executed using a trapezoidal profile for  $\dot{s}$ , with  $|\dot{s}|$  and  $|\ddot{s}|$  bounded by  $V_s$  and  $A_s$ . Values for  $V_s$  and  $A_s$  were 250 mm/s and 500 mm/s<sup>2</sup>.

The Denavit-Hartenburg parameters used for the PUMA were  $a_2 = 431.8$ ,  $d_3 = 149.09$ ,  $a_3 = -20.32$ , and  $d_4 = 433.07$ , with the kinematics described in [14]. The motions were undertaken using the “right-handed, elbow down” solution branch<sup>2</sup>.  $\mathcal{W}^*$  was constructed with  $d_o = d_c = 80$  mm.

In the first example (Figure 6), the robot is driven into a shoulder singularity (i.e., the path terminates on  $\mathcal{C}$ ), nominally causing large spikes in  $\dot{\vartheta}_1$  and  $\dot{\vartheta}_2$ . However, if the same motion is prepared in  $\mathcal{W}^*$ , with bounds on  $\dot{s}$  and  $\ddot{s}$ , the resulting  $\dot{\vartheta}_j$  and  $\ddot{\vartheta}_j$  are well-behaved.

Similar results hold for the other examples. In Figure 7, the robot is driven along a straight line that makes tan-

<sup>2</sup>meaning that that  $-\pi \leq \vartheta_1 - \text{atan2}(p_y, p_x) < 0$  and  $-\pi/2 < \vartheta_3 + \text{atan}(a_3, d_4) \leq \pi/2$ .

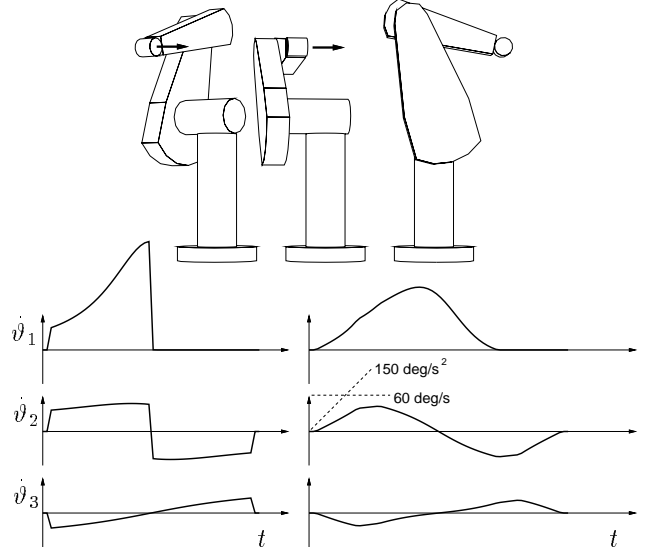


Figure 7: Straight line motion from  $\mathbf{p} = (-d_3, 300, 300)^T$  to  $\mathbf{p} = (-d_3, -300, 300)^T$ , which goes through a shoulder singularity by making tangential contact with  $\mathcal{C}$  at its midpoint. Nominal velocities are shown on the left; desingularized velocities are shown on the right.

gential contact with  $\mathcal{C}$ , so that it is driven *through* a shoulder singularity. Our choice of kinematic solutions on either side of the singularity caused nominal discontinuities in  $\dot{\vartheta}_1$  and  $\dot{\vartheta}_2$ . These are removed, however, when the motion is prepared in  $\mathcal{W}^*$ . Lastly, Figure 8 shows the results when the robot is driven into a double singularity near the ready position, such that the final path point makes contact with both  $\mathcal{S}_o$  and  $\mathcal{C}$ . The nominal motion therefore causes velocity spikes in all three joint angles, but again these are properly removed by preparing the motion in  $\mathcal{W}^*$ .

## 7 Conclusion

A new method for controlling manipulators at kinematic singularities has been presented, involving the creation of an alternate workspace  $\mathcal{W}^*$  in which the singularities are removed. Motions can then be prepared in  $\mathcal{W}^*$ , with limits on the spatial velocity and acceleration, and the resulting joint velocities and accelerations will be well behaved and bounded.  $\mathcal{W}^*$  equals  $\mathcal{W}$  except within a certain distance  $d_b$  of a singular surface. The actual motions resulting in  $\mathcal{W}$  will generally have a path error near the singularity. However, this error can be understood and quantified directly in terms of the mapping from  $\mathcal{W}$  to  $\mathcal{W}^*$ . Moreover, the size of the deformation (and hence the error) associated with  $\mathcal{W}^*$  can be controlled by adjusting the threshold  $d_b$ , with the understanding that as  $d_b$  is decreased, lower limits on the spatial velocity and acceleration will be required within  $\mathcal{W}^*$  to keep joint velocities and accelerations within bounds.

Loosely speaking, our approach turns the singularity

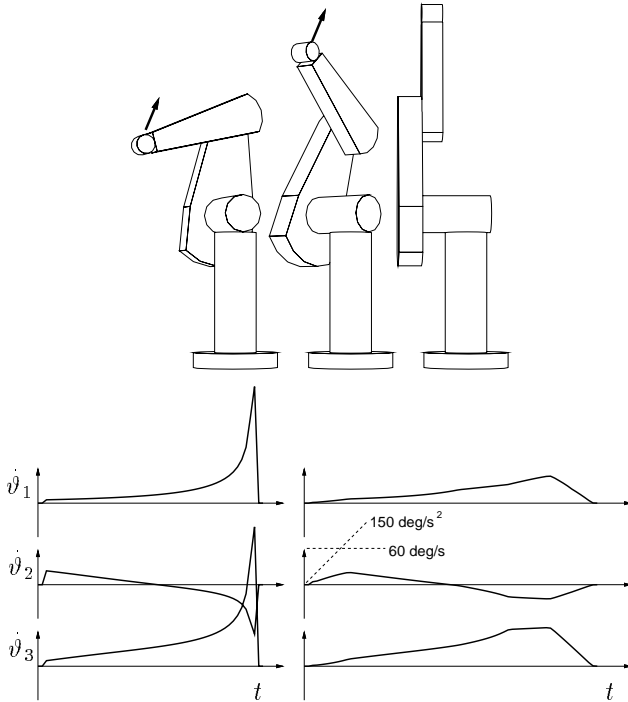


Figure 8: Straight line motion from  $\mathbf{p} = (0, 400, 300)^T$  to a combination elbow-shoulder singularity at  $\mathbf{p} = (0, d_3, a_2 + l_4)^T$ . Nominal velocities are shown on the left; desingularized velocities are shown on the right.

control problem into an ordinary collision control problem in  $\mathcal{W}^*$ .

The method is not directly applicable to self-motion type singularities (including the wrist singularity), and the computation of  $\mathcal{W}^*$  may not always be easy or possible. However, our experience with the PUMA shows that the method can be very easy to apply to certain simple manipulators. Advantages of the method include: (a) computational and implementational ease, (b) control over accelerations and higher derivatives as well as joint velocities, (c) precise knowledge of the resulting path errors, (d) time efficient motions which don't get stuck at singularities, and (e) the ability to handle motions generated online.

Our experience with this method is very new, and so many questions remain. A more rigorous mathematical analysis is also required. Experiments involving the boundary singularity of a planar 3R arm indicate that the method can also be applied to redundant robots. It will be interesting to see how broadly applicable the method is, and how it relates to (or can be used to enhance) more traditional Jacobian-based singularity control methods.

## References

[1] J. E. Lloyd and V. Hayward, "A discrete algorithm for fixed-path trajectory generation at kinematic singularities," in *Proceedings of the IEEE International Conference*

on Robotics and Automation, (Minneapolis, Minnesota), pp. 2743–2748, Apr. 1996.

[2] C. W. Wampler II and L. J. Leifer, "Applications of damped least-squares methods to resolved-rate and resolved-acceleration control of manipulators," *Transactions of the ASME: Journal of Dynamic Systems, Measurement, and Control*, vol. 110, pp. 31–38, Mar. 1988.

[3] A. A. Maciejewski and C. A. Klein, "The singular value decomposition: Computation and applications to robotics," *International Journal of Robotics Research*, vol. 8, pp. 63–79, Dec. 1989.

[4] S. Chiaverini, B. Siciliano, and O. Egeland, "Review of the damped least-squares inverse kinematics with experiments on an industrial robot manipulator," *IEEE Transactions on Control Systems Technology*, vol. 2, pp. 123–134, June 1994.

[5] Z. S. Tumei and C. O. Alford, "Solving for manipulator joint rates in singular positions," in *Proceedings of the IEEE International Conference on Robotics and Automation???*, (Philadelphia, Pennsylvania), pp. 987–992, Apr.24-29 1988.

[6] L. Nielsen, C. C. de Wit, and P. Hagander, "Controllability issues of robots near singular configurations," in *Advances in Robot Kinematics, 2nd International Workshop*, (Linz, Austria), pp. 283–290, Sept.10-12 1990.

[7] C. Chevallereau, "Feasible trajectories for a non redundant robot at a singularity," in *Proceedings of the IEEE International Conference on Robotics and Automation*, (Minneapolis, Minnesota), pp. 1871–1876, Apr. 1996.

[8] J. Kieffer, "Differential analysis of bifurcations and isolated singularities for robots and mechanisms," *IEEE Transactions on Robotics and Automation*, vol. RA-10, pp. 1–10, Feb. 1994.

[9] J. E. Lloyd, "Using puiseux series to control non-redundant robots at singularities," in *Proceedings of the IEEE International Conference on Robotics and Automation*, (Minneapolis, Minnesota), pp. 1877–1882, Apr. 1996.

[10] D. N. Nenchev, Y. Tsumaki, M. Uchiyama, V. Senft, and G. Hirzinger, "Two approaches to singularity-consistent motion of nonredundant robotic mechanisms," in *Proceedings of the IEEE International Conference on Robotics and Automation*, (Minneapolis, Minnesota), pp. 1883–1890, Apr. 1996.

[11] E. D. Pohl and H. Lipkin, "A new method of robotic motion control near singularities," in *Fifth International Conference on Advanced Robotics (91 ICAR)*, (Pisa, Italy), pp. 405–410, June 19-22 1991.

[12] K. A. O'Neil, Y. C. Cheng, and J. Seng, "Removing singularities of resolved motion rate control of mechanisms, including self-motion," *IEEE Transactions on Robotics and Automation*, vol. 13, pp. 741–751, Oct. 1997.

[13] H. Zhang, "Feasibility analysis of displacement trajectories for robot manipulators with a spherical wrist," in *Proceedings of the IEEE International Conference on Robotics and Automation*, (Sacramento, California), pp. 1252–1257, Apr.9-11 1991.

[14] R. P. Paul, B. E. Shimano, and E. Mayer, "Kinematic control equations for simple manipulators," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-11, pp. 449–455, June 1981.

[15] S. Elgazzar, "Efficient kinematic transformations for puma 560 robot," *IEEE Transactions on Robotics and Automation*, vol. RA-1, pp. 142–151, Sept. 1985.