Kinematic Inversion of Functionally-Redundant Serial Manipulators: Application to Arc-Welding

Liguo Huo and Luc Baron
Department of Mechanical Engineering
École Polytechnique de Montréal
P.O. 6079, station Centre-Ville
Montréal, Québec, Canada, H3C 3A7
liguo.huo@polymtl.ca, luc.baron@polymtl.ca

Abstract: This paper presents a new resolution scheme to solve redundant robotic tasks requiring less than six-degrees-of-freedom. Instead of projecting the secondary task onto the null space of the Jacobian matrix in order to take advantage of the redundancy, our approach directly decomposes the task into two orthogonal subspaces where the main and secondary tasks lie, respectively. This approach has shown to be efficient, *i.e.*, having a low numerical cost, and accurate, *i.e.*, having a low round-off error amplification. A numerical example is shown for an arc-welding robotic task.

1 Introduction

Since the late sixties, the control of serial robotic manipulators has received great attentions from the robotic research community. The earliest work is probably the one from Pieper [1], who proposed a scheme based on the Newton-Gauss method in order to iteratively converge toward the desired position and orientation, namely *pose*, of the end-effector (EE). At each iteration, a small displacement in joint space is computed from the inverse of a Jacobian matrix times the desired EE displacement. Whitney [2, 3] proposed to replace the differential form of the Jacobian matrix of Pieper by a more convenient form based on the translational and angular velocity vectors associated to the EE, which resulted into the well-known *resolved-motion rate method*. Since the inverse of the Jacobian matrix is required, many research works have been conducted on the conditions for obtaining non-singular Jacobian matrices, including the use of manipulators having more than six degrees-of-freedom in order to cope with singularities. Liégeois [4] was the first to propose a method to take advantage of the kinematic redundancy by using the generalized inverse together with the projection onto the null space of the Jacobian matrix of an arbitrary vector chosen as the gradient of an objective function. Nakamura [5] analyzed the kinematic redundancy of manipulators by the use of matrix theory. Klein and Huang [6] reviewed the algorithms for computing the generalized-inverse for redundant manipulators. Angeles *et al.* [7] introduced an approach-descent algorithm to solve the inverse kinematics of redundant manipulators. Many research works have considered the computational expense, the roundoff-error amplification and different ways to take advantage of the kinematic redundancy. More recently, Arenson *et al.* [8] proposed a redundancy-resolution scheme that avoid the squaring of roundoff errors while projecting the secondary task onto the null space of the Jacobian. In all these research works, the kinematic redundancy is studied without regard for the task to be performed, and hence, the redundancy comes from the kinematics of the manipulator itself.

In this paper, the sources of *kinematic redundancy* of a pair of manipulator-task are characterized into two groups, *i.e.*, the *intrinsic redundancy* and the *functional redundancy.* For the case of functional redundancy, an approach based on the orthogonal decomposition of the twist is proposed in order to take advantage of the redundancy without having to project the secondary task onto the null space of the Jacobian matrix, thus avoiding roundoff-error amplification and superfluous computations.

2 Background on Kinematic Inversion

Before introducing the redundancy-resolution algorithm, a brief review of the kinematic redundancy of a manipulator with respect to a given task and the kinematic inversion of serial manipulators in these contexts is provided.

2.1 Intrinsic and Functional Redundancy

Let $\mathcal{J}$ denote, the *joint space* of a robotic manipulator having $n+1$ rigid bodies serially connected by n joints, either revolute R or prismatic P. The position of the manipulator in $\mathcal{J}$ is given by the n-dimensional vector, namely $\boldsymbol{\theta}$, and hence, we have: $n = \dim(\mathcal{J}) = \dim(\boldsymbol{\theta})$. Moreover, let $\mathcal{O}$ denote, the *operational space* of the EE of the robotic manipulator resulting from the joint space $\mathcal{J}$. Since any free-moving rigid body in space can have at most six degrees-of-freedom (DOFs), the dimension of $\mathcal{O}$ is also at most six, and hence, we have: $o = \dim(\mathcal{O}) \leq 6$. Furthermore, let $\mathcal{T}$ denote, the *task space* such as required by the functional mobility of the EE, independent of the manipulator architecture and hence, we have: $t = \dim(\mathcal{T}) \leq 6$. Now, let us introduce the following three definitions:

Definition 2.1: Intrinsic redundancy

A serial manipulator is said to be intrinsically redundant when the dimension of the joint space $\mathcal{J}$*, denoted by* $n = \dim(\mathcal{J})$*, is greater than the dimension of the resulting operational space* $\mathcal{O}$ *of the EE, denoted by* $o = \dim(\mathcal{O}) \leq 6$, i.e., *when* $n > o$*. The degree of intrinsic redundancy of a serial manipulator, namely* r_I*, is computed as*

$$r_I = n - o. \tag{1}$$

Definition 2.2: Functional redundancy[1]

A pair of serial manipulator-task is said to be functionally redundant when the dimension of the operational space $\mathcal{O}$ *of the EE, denoted by* $o = \dim(\mathcal{O}) \leq 6$*, is greater than the dimension of the task space* $\mathcal{T}$ *of the EE, denoted by* $t = \dim(\mathcal{T}) \leq 6$*, while the task space being totally included into the operation space of the manipulator,* i.e., $\mathcal{T} \subseteq \mathcal{O}$*, and hence,* $o > t$*. The degree of functional redundancy of a pair of serial manipulator-task, namely* r_F*, is computed as*

$$r_F = o - t. \tag{2}$$

Definition 2.3: Kinematic redundancy

A pair of serial manipulator-task is said to be kinematically redundant when the dimension of the joint space $\mathcal{J}$*, denoted by* $n = \dim(\mathcal{J})$*, is greater than the dimension of the task space* $\mathcal{T}$ *of the EE, denoted by* $t = \dim(\mathcal{T}) \leq 6$*, while the task space being totally included into the resulting operation space of the manipulator,* i.e., $\mathcal{T} \subseteq \mathcal{O}$*, and hence,* $n > t$*. The degree of kinematic redundancy of a pair of serial manipulator-task, namely* r_K*, is computed as*

$$r_K = n - t. \tag{3}$$

Upon substitution of eqs.(1) and (2) into (3), it becomes apparent that the kinematic redundancy come from both the intrinsic and functional redundancies, *i.e.*,

$$r_K = r_I + r_F. \tag{4}$$

In the literature, most of the research works focusing on redundancy-resolution of serial manipulators suppose that $r_F = 0$, and thus, study $r_K = r_I$. In this paper, we will study the opposite case, *i.e.*, we suppose that $r_I = 0$, and thus, study $r_K = r_F$.

[1] This definition of functional redundancy of serial manipulator-task is directly expandable to other types of manipulators such as parallel and hybrid manipulators.

2.2 Non-Redundant Manipulators

For quick reference, we briefly summary of the resolved motion-rate method [2, 3] that is commonly used to iteratively solve the inverse kinematics of serial manipulators. The method is based on the relationship between the EE velocity, called *twist* and denoted $\mathbf{t}$, and the joint velocities, denoted $\dot{\boldsymbol{\theta}}$, given by

$$\mathbf{t} = \mathbf{J}\dot{\boldsymbol{\theta}}, \tag{5}$$

with $\mathbf{t}$ and $\dot{\boldsymbol{\theta}}$ defined as

$$\mathbf{t} \equiv [\boldsymbol{\omega}^T \dot{\mathbf{p}}^T]^T \in 2 \times \mathrm{IR}^3, \quad \dot{\boldsymbol{\theta}} \equiv [\dot{\theta}_1 \cdots \dot{\theta}_n]^T \in \mathrm{IR}^n, \quad \mathbf{J} \equiv \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix}, \quad \mathbf{A}, \mathbf{B} \in \mathrm{IR}^{3\times n}, \tag{6}$$

where $\boldsymbol{\omega} \in \mathrm{IR}^3$ is the angular velocity vector of the EE, $\dot{\mathbf{p}} \in \mathrm{IR}^3$ is the translation velocity vector of a point of the EE, while $\dot{\theta}_i$ is the velocity of joint i of the manipulator. It is noteworthy that $\mathbf{t}$ is not defined as a vector of IR^6, but rather as a set of two vectors of IR^3 casted into a column array, and hence, $\mathbf{t} \in 2 \times \mathrm{IR}^3 \neq \mathrm{IR}^6$. This distinction will be further used in section 3.0.

Upon substituting the finite displacement $\boldsymbol{\Delta}\mathbf{t}$ of a small time interval (see for example [9]) into eq.(5), the finite displacement $\boldsymbol{\Delta\theta}$ in $\mathcal{J}$ can be computed as

$$\boldsymbol{\Delta\theta} = \mathbf{J}^{-1}\boldsymbol{\Delta}\mathbf{t}, \tag{7}$$

where it is apparent that $\mathbf{J}$ must be square and non-singular.

2.3 Intrinsically-Redundant Manipulators

For intrinsically-redundant serial manipulators, $\mathbf{J}$ always has more columns than rows, and hence, equation (5) becomes an under-determined linear algebraic system having infinitely many solutions. In this case, Liégeois [4] proposed to compute the finite displacement $\boldsymbol{\Delta\theta}$ as

$$\boldsymbol{\Delta\theta} = \underbrace{(\mathbf{J}^\dagger)\boldsymbol{\Delta}\mathbf{t}}_{\text{minimum}-\text{norm solution}} + \underbrace{(\mathbf{1} - \mathbf{J}^\dagger\mathbf{J})\mathbf{h}}_{\text{homogeneous solution}}, \tag{8}$$

where $\mathbf{J}^\dagger$ is defined as the right-generalized inverse of $\mathbf{J}$ such that

$$\mathbf{J}^\dagger \equiv \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}, \tag{9}$$

and $\mathbf{h}$ is an arbitrary vector of $\mathcal{J}$ allowing to satisfy a secondary task. The first term in the right-hand side (RHS) of eq.(8) is known as the minimum-norm solution of eq.(5), *i.e.*, the $\boldsymbol{\Delta\theta}_M$ that minimizes $\|\boldsymbol{\Delta\theta}\|$ among all the $\boldsymbol{\Delta\theta}$ that are solutions of eq.(5). The second part of the RHS of eq.(8) is known as the homogeneous solution of eq.(5), *i.e.*, the $\boldsymbol{\Delta\theta}_H$ that produce $\boldsymbol{\Delta}\mathbf{t} = \mathbf{0}$, *i.e.*, no displacement of the EE. This joint displacement $\boldsymbol{\Delta\theta}_H$ is also known as the self-motion of the manipulator. It is symbolically computed as the projection of an arbitrary vector $\mathbf{h}$ onto the nullspace of $\mathbf{J}$ with the orthogonal projector $(\mathbf{1} - \mathbf{J}^{\dagger}\mathbf{J})$. Equation (8) is used to solve the kinematic inversion of intrinsically-redundant manipulators by many researchers (*e.g.*, Siciliano [10], Arenson *et al.*[8]), including some who had put a special attention in avoiding the squaring of the condition number while solving eq.(8).

2.4 Functionally-Redundancy Manipulators

Typically, the motion of the EE required by a task is usually the full 6-DOF. However, many industrial tasks such as arc-welding, milling, deburing, laser-cutting, and many others, require less than 6-DOF, because of the presence of a symmetry axis or plane on the EE. For example, the general task of arc-welding requires 3-DOF for the displacement of the end-point of the electrode, but requires only 2-DOF for its orientation. The rotation of the welding-gun around the electrode axis is clearly irrelevant to the view of the task to be accomplished. In order to cope with this problem, Baron [11] proposed to add a virtual joint around the symmetry axis of the electrode, in order to transform the functional redundancy into an intrinsic redundancy thereby solving an augmented Jacobian matrix with eq.(8). However, this *augmented approach* to solve functionally-redundant robotic tasks suffers from the potential ill-conditioning of $\mathbf{J}$ and the additional computational cost required to solve an augmented $\mathbf{J}$. Below, we propose a *projected approach* to solve the same problem.

3 Kinematic Inversion of Functionally-Redundant Manipulators

After introducing the orthogonal decomposition of vectors and twists, we formulate the inverse kinematics of functionally-redundant manipulators by projecting the velocity relationship onto the instantaneous-task subspace, thereby producing the so-called *projected approach*.

3.1 Orthogonal-Decomposition of Vectors

Decomposing any vector $(\,\cdot\,)$ of $\mathbb{R}^3$ into two orthogonal parts, $[\,\cdot\,]_M$, the component lying on the subspace, $\mathcal{M}$, and $[\,\cdot\,]_{M^\perp}$, the component lying in the orthogonal subspace, $\mathcal{M}^\perp$, using the *projector* $\mathbf{M}$ and an *orthogonal complement* of $\mathbf{M}$, namely $\mathbf{M}^\perp$, as follows:

$$(\,\cdot\,) = [\,\cdot\,]_M + [\,\cdot\,]_{M^\perp} = \mathbf{M}(\,\cdot\,) + \mathbf{M}^\perp(\,\cdot\,) = (\mathbf{M} + \mathbf{M}^\perp)(\,\cdot\,) \tag{10}$$

It is apparent from eq.(10), that $\mathbf{M}$ and $\mathbf{M}^\perp$ are related by $\mathbf{M} + \mathbf{M}^\perp = \mathbf{1}$ and $\mathbf{M}\mathbf{M}^\perp = \mathbf{O}$, where $\mathbf{1}$ and $\mathbf{O}$ are the 3×3 identity and zero matrices, respectively. The orthogonal complement of

$\mathbf{M}$ thus defined, $\mathbf{M}^\perp$, is therefore unique, and hence, both $\mathbf{M}$ and $\mathbf{M}^\perp$ are projectors that verify the following properties:

- Symmetry: $[\mathbf{M}]^T = \mathbf{M}, \quad [\mathbf{M}^\perp]^T = \mathbf{M}^\perp$
- Idempotency: $[\mathbf{M}]^2 = \mathbf{M}, \quad [\mathbf{M}^\perp]^2 = \mathbf{M}^\perp$
- Rank-complementarity: $\text{rank}(\mathbf{M}) + \text{rank}(\mathbf{M}^\perp) = 3$
- Subspace-complementarity: $\mathcal{M} \oplus \mathcal{M}^\perp = \mathbb{R}^3$

The projector $\mathbf{M}$ projects vectors of $\mathbb{R}^3$ onto the subspace $\mathcal{M}$, while the orthogonal projector $\mathbf{M}^\perp$ projects those vectors onto the orthogonal subspace $\mathcal{M}^\perp$. These projectors are given for the four possible dimensions i of subspaces of $\mathbb{R}^3$ as:

$$\mathbf{M}_i = \begin{cases} \mathbf{1} \\ \mathbf{P} \\ \mathbf{L} \\ \mathbf{O} \end{cases}, \quad \mathbf{M}_i^\perp = \begin{cases} \mathbf{O} \\ \mathbf{L} \\ \mathbf{P} \\ \mathbf{1} \end{cases}, \quad \begin{array}{ll} i = 3 & \Rightarrow \text{3-D task} \\ i = 2 & \Rightarrow \text{2-D task} \\ i = 1 & \Rightarrow \text{1-D task} \\ i = 0 & \Rightarrow \text{0-D task} \end{array}, \tag{11}$$

where the plane and line projectors, $\mathbf{P}$ and $\mathbf{L}$, respectively, are defined as:

$$\mathbf{P} \equiv \mathbf{1} - \mathbf{L}, \quad \mathbf{L} \equiv \mathbf{e}\mathbf{e}^T, \tag{12}$$

in which $\mathbf{e}$ is a unit vector along the line $\mathcal{L}$ and normal to the plane $\mathcal{P}$. The null-projector $\mathbf{O}$ is the 3×3 zero matrix that projects any vector of $\mathbb{R}^3$ onto the null-subspace $\mathcal{O}$, while the identity-projector $\mathbf{1}$ is the 3×3 identity matrix that projects any vector of $\mathbb{R}^3$ onto itself.

3.2 Orthogonal-Decomposition of Twists

Any twist array $(\,\cdot\,)$ of $2 \times \mathbb{R}^3$ can also be decomposed into two orthogonal parts, $[\,\cdot\,]_\mathcal{T}$, the component lying on the task subspace, $\mathcal{T}$, and $[\,\cdot\,]_{\mathcal{T}^\perp}$, the component lying in the orthogonal task subspace (also designated as the redundant subspace), $\mathcal{T}^\perp$, using the *twist projector* $\mathbf{T}$ and an *orthogonal complement* of $\mathbf{T}$, namely $\mathbf{T}^\perp$, as follows:

$$(\,\cdot\,) = [\,\cdot\,]_\mathcal{T} + [\,\cdot\,]_{\mathcal{T}^\perp} = \mathbf{T}(\,\cdot\,) + \mathbf{T}^\perp(\,\cdot\,) = (\mathbf{T} + \mathbf{T}^\perp)(\,\cdot\,) \tag{13}$$

It is apparent from eq.(13), that $\mathbf{T}$ and $\mathbf{T}^\perp$ are projectors of twists that must verify all the properties of projectors of section 3.1. However, twists are not vectors of $\mathbb{R}^6$, and hence, projectors of twists cannot be defined as in eqs.(11) and (12), *e.g.*,

$$\mathbf{T} \neq \mathbf{t}\mathbf{t}^T, \quad \mathbf{T}^\perp \neq \mathbf{1} - \mathbf{t}\mathbf{t}^T, \tag{14}$$

but must rather be defined as block diagonal matrices of projectors of $\mathbb{R}^3$, *i.e.*,

$$\mathbf{T} \equiv \begin{bmatrix} \mathbf{M}_\omega & \mathbf{O} \\ \mathbf{O} & \mathbf{M}_v \end{bmatrix}, \quad \mathbf{T}^\perp \equiv \mathbf{1} - \mathbf{T} = \begin{bmatrix} \mathbf{1} - \mathbf{M}_\omega & \mathbf{O} \\ \mathbf{O} & \mathbf{1} - \mathbf{M}_v \end{bmatrix}, \tag{15}$$

where $\mathbf{M}_\omega$ and $\mathbf{M}_v$ are projectors of $\mathbb{R}^3$ defined in eqs.(11) and (12) which allow the projection of the angular and translational velocity vectors, respectively. It is noteworthy that the matrices of eq.(14) do not verify the properties of projectors, and hence, cannot be used for orthogonal decomposition. Finally, eq.(13) becomes

$$\mathbf{t} = \mathbf{t}_{\mathcal{T}} + \mathbf{t}_{\mathcal{T}}^{\perp} = \mathbf{T}\mathbf{t} + (\mathbf{1} - \mathbf{T})\mathbf{t}. \tag{16}$$

3.3 Twist Decomposition Algorithm in Solving Functional Redundancy

For functionally-redundant serial manipulators, it is possible to decompose the twist of the EE into two orthogonal parts, one lying into task subspace and another one lying into the redundant subspace. Substituting eq.(16) into eq.(7) yields, we have

$$\boldsymbol{\Delta\theta} = \underbrace{(\mathbf{J}^{\dagger}\mathbf{T})\boldsymbol{\Delta}\mathbf{t}}_{\text{task displacement}} + \underbrace{\mathbf{J}^{\dagger}(\mathbf{1} - \mathbf{T})\mathbf{J}\mathbf{h}}_{\text{redundant displacement}}, \tag{17}$$

where $\mathbf{h}$ is an arbitrary vector of $\mathcal{J}$ allowing to satisfy a secondary task. Vector $\mathbf{h}$ is often chosen as the gradient of an objective function to minimize (Baron [11]). For the avoidance of joint-limits, the objective function z can be written as to maintain the manipulator as close as possible to the mid-joint position $\bar{\boldsymbol{\theta}}$, *i.e.*,

$$z = \frac{1}{2}(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^T \mathbf{W}^T \mathbf{W} (\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \rightarrow \min_{\boldsymbol{\theta}}, \tag{18}$$

with $\bar{\boldsymbol{\theta}}$ and $\mathbf{W}$ being defined as

$$\bar{\boldsymbol{\theta}} \equiv \frac{1}{2}(\boldsymbol{\theta}_{max} + \boldsymbol{\theta}_{min}), \quad \mathbf{W} \equiv \mathbf{diag}(\boldsymbol{\theta}_{max} - \boldsymbol{\theta}_{min}). \tag{19}$$

Vector $\mathbf{h}$ is thus chosen as minus the gradient of z, *i.e.*,

$$\mathbf{h} = -\boldsymbol{\nabla} z. \tag{20}$$

The first part of the RHS of eq.(17) is the joint displacement required by the task, while the second part is the joint displacement in the redundant subspace (or irrelevant to the task). Clearly, equation(17) does not require the projection onto the null-space of $\mathbf{J}$ as most of the redundancy-resolution algorithms do, but rather requires an orthogonal projection based on the instantaneous geometry of the task to be accomplished.

Algorithm 3.1: Twist Decomposition Algorithm

1 $\{\mathbf{p}, \mathbf{Q}\} \Leftarrow \mathbf{DKP}(\boldsymbol{\theta})$

$\Delta\mathbf{Q} \Leftarrow \mathbf{Q}^T\mathbf{Q}_d$

$\Delta\mathbf{p} \Leftarrow \mathbf{p}_d - \mathbf{p}$

$\Delta\mathbf{t} \leftarrow \begin{bmatrix} \mathbf{Qvect}(\Delta\mathbf{Q}) \\ \Delta\mathbf{p} \end{bmatrix}$

$\mathbf{DKP}(\boldsymbol{\theta}) \Rightarrow \begin{cases} \mathbf{e} \Rightarrow \mathbf{M}_\omega \\ \mathbf{f} \Rightarrow \mathbf{M}_v \\ \mathbf{J} \end{cases},$

$\mathbf{T} \Leftarrow \begin{bmatrix} \mathbf{M}_\omega & \mathbf{O} \\ \mathbf{O} & \mathbf{M}_v \end{bmatrix},$

2 $\Delta\boldsymbol{\theta} \Leftarrow \mathbf{J}^\dagger\mathbf{T}\Delta\mathbf{t} + \mathbf{J}^\dagger(\mathbf{1} - \mathbf{T})\mathbf{J}\mathbf{h}$

if $\|\Delta\boldsymbol{\theta}\| < \epsilon$ then stop;

else

$\boldsymbol{\theta} \Leftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$

goto 1

The twist decomposition algorithm 3.1 is based on resolved motion rate method. Instead of using eq.(8) as many other RR schemes, equation (17) is used to compute $\Delta\boldsymbol{\theta}$. In step 1, $\mathbf{Q}_d$ and $\mathbf{p}_d$ are the desired orientation matrix and position vector of the EE, respectively; $\mathbf{vect}(\cdot)$ is the function transforming a 3×3 rotation matrix into an axial vector as defined in [9] (page 34); ϵ is the convergence criterion; the $\mathbf{DKP}(\boldsymbol{\theta})$ is used to compute firstly the position vector $\mathbf{p}$ and orientation matrix $\mathbf{Q}$, then the unit vectors of irrelevant rotation and translation axes, *i.e.*, $\mathbf{e}$ and $\mathbf{f}$, in order to build the twist projector $\mathbf{T}$.

4 Application to Arc-Welding

When performing arc-welding operations, the electrode of the welding tool has an axis of symmetry around which the welding tool may be rotated without interfering with the task to be performed. This axis describes the geometry of the functional redundancy (or the redundant subspace of twists). The unit vector $\mathbf{e}$ denote the orientation of the axis of symmetry along the electrode. The projection of $\boldsymbol{\omega}$ along $\mathbf{e}$ is the irrelevant component of $\boldsymbol{\omega}$, while its projection onto the plane normal to $\mathbf{e}$ is the relevant component of $\boldsymbol{\omega}$. For a general arc-welding task around the electrode axis $\mathbf{e}$, the twist projector is defined as

$$\mathbf{T}_{weld} \equiv \begin{bmatrix} (\mathbf{1} - \mathbf{e}\mathbf{e}^T) & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}, \quad \mathbf{T}_{weld}^{\perp} \equiv \begin{bmatrix} \mathbf{e}\mathbf{e}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{21}$$

Now, substituting eq.(21) into eq.(17) yields

$$\boldsymbol{\Delta\theta} = \mathbf{J}^\dagger\mathbf{T}_{weld}\boldsymbol{\Delta}\mathbf{t} + \mathbf{J}^\dagger \begin{bmatrix} \mathbf{e}\mathbf{e}^T\mathbf{A}\mathbf{h} \\ \mathbf{0} \end{bmatrix}, \tag{22}$$

where $\mathbf{A}$ is the upper part of $\mathbf{J}$ as defined in eq.(6). Equation (22) can be used as line 2 of algorithm 3.1 in order to solve the inverse kinematics of serial manipulators while performing a general arc-welding task.

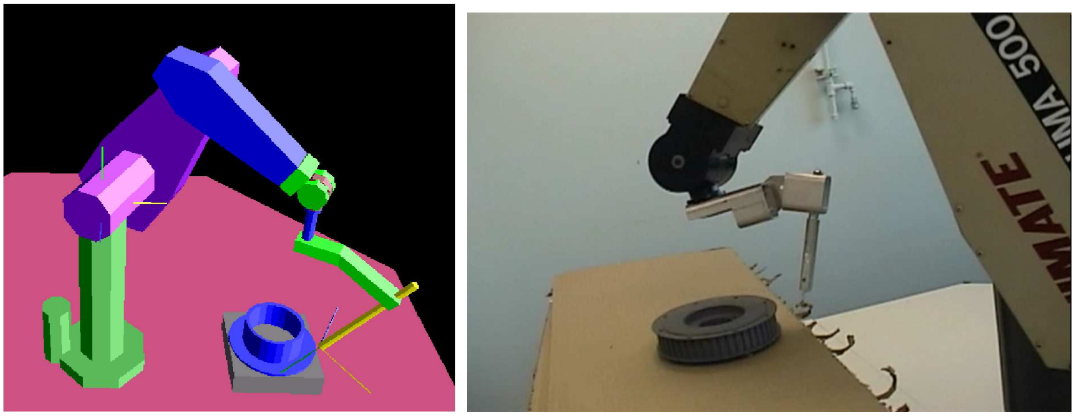

Figure 1: Arc-welding task with the PUMA 560 manipulator

As shown in Fig. 1, a PUMA 560 serial manipulator is used to perform a pipe-to-bride welding task. Its Denavit-Hartenberg parameters are described in Table 1. The welding tool has a transformation matrix $\mathbf{A}_{tool}$ as

$$\mathbf{A}_{tool} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\frac{\pi}{6}) & -\sin(\frac{\pi}{6}) & 0.1 \\ 0 & \sin(\frac{\pi}{6}) & \cos(\frac{\pi}{6}) & 0.501 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{23}$$

The task is to perform the following trajectory Λ in $T = 285$ sec., *i.e.*,

$$\mathbf{p} = \begin{bmatrix} 0.1\cos(\omega t) \\ 0.6 + 0.1\sin(\omega t) \\ -0.59 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} \cos\alpha & -\sin\alpha\cos\beta & \sin\alpha\sin\beta \\ \sin\alpha & \cos\alpha\cos\beta & \cos\alpha\cos\beta \\ 0 & \sin\beta & \cos\beta \end{bmatrix}, \tag{24}$$

with $\alpha = \frac{\pi}{2} + \omega t, \quad \beta = \frac{-3\pi}{4}, \quad \omega = \frac{2\pi}{T}, \quad 0 \le t \le T$, where distances and angles are expressed in meter and radians, respectively. The electrode axis $\mathbf{e}$ can be computed as

$$\mathbf{e} = \mathbf{Q}_1\mathbf{Q}_2\cdots\mathbf{Q}_6\mathbf{k}, \quad \mathbf{k} \equiv [0 \;\; 0 \;\; 1]^T. \tag{25}$$

The secondary task $\mathbf{h}$ is to avoid the joint-limits such that:

$$\mathbf{h} = -\mathbf{W}(\boldsymbol{\theta} - \boldsymbol{\theta}_0), \tag{26}$$

where $\mathbf{W}$ is a positive-definite weighting matrix as in eq.(19) and $\boldsymbol{\theta}_0$ the mean-joint position defined as

$$\boldsymbol{\theta}_0 \equiv \begin{bmatrix} \pi/2 & -\pi/3 & \pi & \pi/4 & \pi/3 & \pi \end{bmatrix}^T. \tag{27}$$

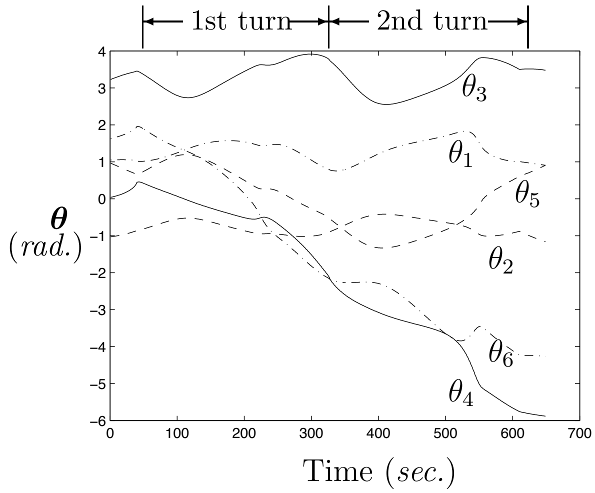

Figure 2: Joint position with respect to time without using RR scheme

As shown in Fig. 2, without taking advantage of the axis of symmetry of the electrode, the manipulator is able to perform the task while using the full amplitude of its joint motion. A second consecutive turn is, obviously, not possible without exceeding the joint limits. The left side of Fig. 3 shows the time history of the joint positions of the manipulator for two consecutive turns along the given welding trajectory with the augmented approach. Apparently, the manipulator is able to perform multiple consecutive turns without exceeding the joint limits. However, excessive joint velocities appear at every turn. The right side of Fig. 3 shows the time history of the joint positions for two consecutive turns along the curve welding trajectory with the projected approach. Apparently, the manipulator is still able to perform multiple consecutive turns without exceeding the joint limits. Excessive joint velocities appear only at the first turn because of the bad initial conditions, and not at all for the consecutive turns.

Table 2 shows the errors of the augmented approach and projected approach. The mean value of position error array is denoted as $\overline{e}_p$; the mean value of orientation error array in task space is denoted as $\overline{o}_e$. It is apparent from Table 2 that the projected approach has much lower position and the orientation errors in the task space than the augmented approach. In other words, the projected approach produces more accurate solutions than the augmented approach.

5 Conclusions

In this paper, the concept of functional redundancy is defined and discussed. The kinematic inversion of functionally-redundant serial manipulators is formulated using the orthogonal de-

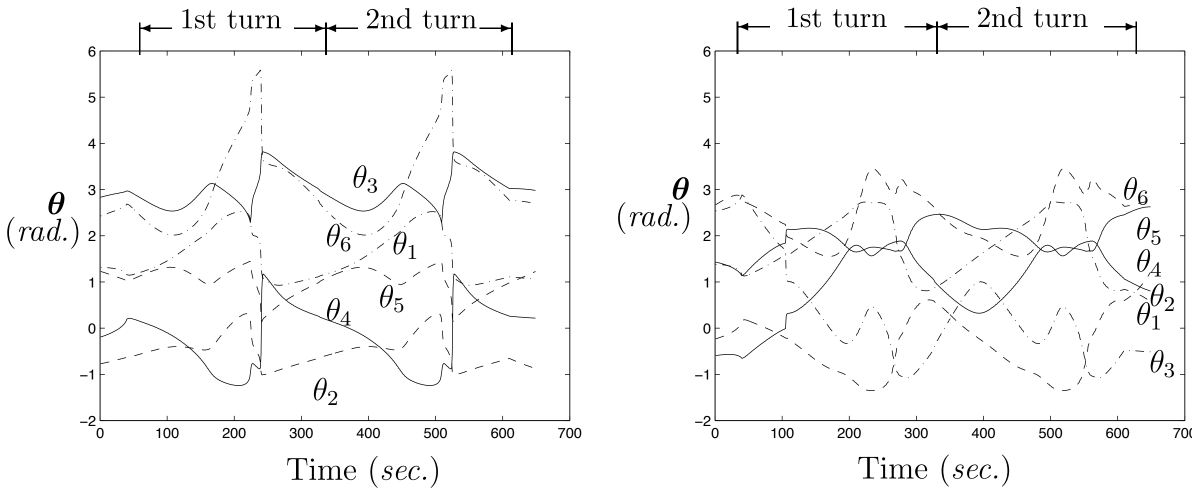

Figure 3: Joint position with respect to time for the augmented approach (left) and projected approach (right)

composition of the twist of the EE into a task subspace and a redundant subspace. The numerical simulation of the arc-welding of a pipe-to-bride with the PUMA 560 serial manipulator has shown to be effective relative to the augmented and also to the non-redundant approaches.

6 Acknowledgments

The financial support from the Natural Sciences and Engineering Research Council of Canada under grant OGPIN-203618 is gracefully acknowledged.

joint	θ_i	a_i	b_i	α_i
1	θ_1	0.0	0.0	$-\pi/2$
2	θ_2	0.4318	0.0	0.0
3	θ_3	-0.0203	0.1491	$\pi/2$
4	θ_4	0.0	0.4330	$-\pi/2$
5	θ_5	0.0	0.0	$\pi/2$
6	θ_6	0	0.055	0
unit	*rad.*	*m*	*m*	*rad.*

Table 1: DH parameters of PUMA 560

method	$\overline{e}_p$	$\overline{o}_e$
augmented	0.0789	2.7403×10^{-5}
projected	1.0533×10^{-7}	1.0123×10^{-5}
unit	meter	rad.

Table 2: Errors of the augmented approach and projected approach

References

[1] Pieper, D.L., "The kinematics of manipulators under computer control", *Ph.D. thesis*, Stanford University, 1968.

[2] Whitney, D.E., "Resolved motion rate control of manipulators and human prostheses", *IEEE Trans. Man-Machine Syst.*, vol. 10, no. 2, pp. 47–53. 1969.

[3] Whitney, D.E., "The mathematics of coordinated control of prosthetic arms and manipulator", *ASME J. Dynamics Syst., Measur. and Cont.*, vol. 94, no. 4, pp. 303–309, 1972.

[4] Liegeois, A., "Automatic supervisory control of the configuration and behavior of multibody mechanisms", *IEEE Trans. on System, Man, Cybern.*, vol. SMC-7, no. 12, pp. 245–250, 1977.

[5] Nakamura, Y., "Control of articulated robot arms with redundancy", *M.S. thesis*, Kyoto University, 1979.

[6] Klein, C.A. and Huang, C.-H. , "Review of pseudoinverse control for use with kinematically redundant manipulators", *IEEE Trans. on Systems, Man, and Cybernetics*, vol. SMC-13, no. 3, pp. 245–250, 1983.

[7] Angeles, J., Habib, M. and Lopez-Cajun, C.S., "Efficient Algorithms for the Kinematic Inversion of redundant Robot Manipulators", *Int. J. of Robotics and Automation*, vol. 2, no. 3, pp. 106–115, 1987.

[8] Arenson, N., Angles, J. and Slutski, L., "Redundancy-resolution algorithms for isotropic robots", *Advances in Robot Kinematics: Analysis and Control*, pp. 425–434, Salzburg, 1998.

[9] Angeles, J., *Fundamental of Robotic Mechanical Systems: Theory, Methods, and Algorithms*, second edition, Springer-Verlag, New York, 521 pages, 2003.

[10] Siciliano, B., "Solving manipulator redundancy with the augmented task space method using the constraint Jacobian transpose", *IEEE Int. Conf. on Robotics and Automation*, vol. 5, pp. 1–8, 1992.

[11] Baron, L., "A joint-limits avoidance strategy for arc-welding robots", *Int. Conf. on Integrated Design and Manufacturing in Mech. Eng.*, Montreal, Canada, 16-19 may 2000.