CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Inverse kinematics strategies for physical human-robot interaction using low-impedance passive link shells

Jonathan Beaudoin, Thierry Laliberté and Clément Gosselin*

Laboratoire de Robotique, Département de génie mécanique, Université Laval, 1065 avenue de la médecine, Québec, QC, Canada, G1V 0A6
*Corresponding author. E-mail: gosselin@gmc.ulaval.ca

Abstract

This paper presents an investigation of the effectiveness of different inverse kinematics strategies in a context of physical human-robot interaction in which passive articulated shells are mounted on the links of a serial robot for manual guidance. The concept of passive link shells is first recalled. Then, inverse kinematics strategies that are designed to plan the trajectory of the robot according to the motion sensed at the link shells are presented and formulated. The different approaches presented all aim at interpreting the motion of the shells and provide an intuitive interaction to the human user. Damped Jacobian based methods are introduced in order to alleviate singularities. A serial 5-dof robot used in previous work is briefly introduced and is used as a test case for the proposed inverse kinematics schemes. The robot includes two link shells for interaction. Simulation results based on the different inverse kinematic strategies are then presented and compared. Finally, general observations and recommendations are discussed.

1. Introduction

Industrial applications in which human workers and robots share a common workspace are nowadays common in industry. In the past years, numerous studies have touched on the advancements of robots that are safe enough to assist humans, whether it be in industry or at home. Different safety metrics and safety-related issues are introduced in ref. [1]. A survey of the different forms of human-machine cooperation in assembly is presented in ref. [2], which explores different safety systems as well. In ref. [3], the development of a collaborative human-robot manufacturing cell compatible with the safety standards is described. A systematic evaluation of safety in human-robot interaction, covering the most significant injury mechanisms, is proposed in ref. [4]. If they are to fulfill their purpose, these robots should not only be safe but also simple and intuitive to use.

Different approaches are studied throughout the literature to safely control a manipulator for the application of physical human-robot interaction (pHRI). In most cases, a rigid manipulator is controlled through an admittance control scheme. Some control schemes also use function approximation techniques to improve the robustness of controllers in the presence of noise (see for instance [5] and [6]). In ref. [7], a velocity-based variable impedance control using the differentiation of the force to infer human intention is presented. Reference [8] presents a variable admittance control approach to improve system intuitiveness, using desired velocity and acceleration for the inference of human intentions. In ref. [9], a variable admittance control to deliver an optimal bilateral force amplification is introduced while a new variable admittance control law that guarantees the stability of the robot is proposed in ref. [10]. Admittance control requires the use of a force/torque sensor to read the user's inputs and infer their

intentions. The main drawback of this approach is that it introduces lag and delays, thereby affecting the intuitiveness of the interaction.

Alternatively, the macro-mini architecture overcomes this limitation by mounting a low-impedance mechanism (mini robot) at the rigid robot's end-effector. This method is first presented in refs. [11, 12]. Reference [13] builds upon this concept and introduces a novel low-impedance mini mechanism specifically designed for physical human-robot cooperation. The uMan is presented in ref. [14]. It consists of an underactuated manipulator designed with a novel passive mini mechanism, minimizing impedance. In reference [15], the mini mechanism is built upon the tripteron architecture [16] to eliminate parasitic motions. The macro-mini architecture can be used for different applications. For instance, refs. [17, 18] present a macro-mini robot designed for polishing and deburring using force control. In this approach, the user interacts only with the low-impedance mechanism, decoupling the robot's high impedance from the task. The mini mechanism's motion from its reference configuration relative to the end-effector dictates the robot's motion. However, in the case of a passive mini mechanism, since the payload is attached to the mini mechanism, the user feels its whole inertia, hindering the interaction for large payloads. Other approaches are also proposed in the literature in which the interaction is done mainly through force/torque sensors or load cells [19, 20, 21].

Rather than placing the mini mechanism at the robot's end-effector, ref. [22] proposes to mount a six-degree-of-freedom (6-dof) low-impedance passive shell on the robot's last link, thereby decoupling the task from both the structure of the robot and the payload. In reference [23], the 6-dof passive shell is replaced by two 3-dof shells mounted on two distinct links to make the interaction more intuitive.

The motion of the shells relative to the links is measured using encoders included in the mechanisms connecting the shells to the links. This motion is then used to infer the intentions of the human user. In refs. [22] and [23], the shell's motion is interpreted as desired link velocities. The inverse kinematics corresponding to the link on which a given shell is mounted is then solved to obtain the joint velocities. Nevertheless, different strategies can be employed to solve this problem. In order to study the impact of the different strategies and select the most appropriate approaches, this paper compares the behavior of different inverse kinematics strategies when used with low-impedance passive link shells. Based on the general principle and architecture presented in refs. [22] and [23], this paper aims at developing motion planning schemes that fully exploit the capabilities of the robots equipped with low-impedance passive shells. Simulations are used to study the effect of the motion control scheme on the intuitiveness of the interaction, which is characterized by the direction of the motion produced.

This paper is then structured as follows. First, the objective of the study is clearly stated. Second, a review of existing inverse kinematics strategies is briefly presented. Then, the serial manipulator and the low-impedance passive link shells used in this study are introduced. Next, the inverse kinematics strategies are adapted to the current objective, and simulation results are provided. Finally, the results are analyzed and discussed to provide insightful recommendations.

2. Objective

As mentioned above, in ref. [22], a low-impedance 6-dof passive link shell was designed to capture a human user's interaction commands and decouple the user's motions from the high-impedance structure of the robot in order to yield more natural physical human-robot interactions. It is pointed out that the shells provide the low-impedance interaction, while the robot itself is a high-impedance device. The shell is mounted on the last link of a custom-built 5-dof serial manipulator to evaluate its performances. Reference [22] uses the damped least-square pseudo-inverse method to solve the inverse kinematics.

In ref. [23], the low-impedance 6-dof passive link shell is replaced by two redesigned 3-dof low-impedance passive link shells. The two shells are mounted on the third and fifth link of the same 5-dof serial manipulator.

The work introduced in this paper builds upon and expands the results presented in refs. [22] and [23]. In ref. [23], a general formulation was presented that can be used to handle an arbitrary number of link shells placed along any of the links of a serial robot. Such an approach allows a user to control the

motion of a robot by manipulating any of the links that are equipped with shells. However, the strategies that should be used to exploit the motion of the shells was not explored in detail.

In this context, the objective of this paper is to study different strategies for the inverse kinematics based on the input provided by a human user, that is, based on a measured motion of the link shells. In other words, this paper addresses the issue of interpretation of the motion of the link shells to produce the motion of the robot that will most closely correspond to the intentions of the user. The strategies investigated are then applied to the arrangement presented in ref. [23], namely a 5-dof robot with two low-impedance passive link shells mounted on the third and fifth links, in order to compare their effectiveness.

3. Inverse kinematics and trajectory planning of serial robots

In physical human-robot interaction, a user moves a robot from its current position rather than following a prescribed Cartesian trajectory. The robot's instantaneous motion, and hence its velocity, is then more relevant than its absolute position and orientation. As such, the objective of the paper is not to revisit the kinematics of the robot itself but to develop means of interpreting the motion of the shells with respect to the robot in order to produce an intuitive motion of the robot. Because of that, velocity control is used (as opposed to position control) because the trajectory imparted to the shell by the user is unknown a priori. The velocities of the end-effector are obtained from the joint velocities through the Jacobian matrix, as

$$t = J(\theta)\dot{\theta} \tag{1}$$

where t is the six-dimensional end-effector velocity vector, $\dot{\theta}$ is the joint velocity vector, and J is the Jacobian matrix. The velocity vector t is defined by

$$t = \begin{bmatrix} \omega \\ \dot{p} \end{bmatrix} \tag{2}$$

where ω and $\dot{p}$ are respectively the angular velocity vector and the velocity of the reference point on the end-effector. Given Eq. (2), the Jacobian matrix can be written as

$$J(\theta) = \begin{bmatrix} A(\theta) \\ B(\theta) \end{bmatrix} \tag{3}$$

where A and B are the matrices related to the rotational and translational components, respectively.

Most likely, the desired trajectory is defined in the Cartesian space rather than in the joint space. Particularly, the approach described here interprets the inputs from link shells as Cartesian velocities of their associated links. The inverse problem must then be solved to yield the joint velocities from the desired end-effector velocity, namely

$$\dot{\theta} = J^I t \tag{4}$$

where J^I is a generalized inverse and where the Jacobian's dependence on the joint coordinates is dropped for convenience. Several methods can be used to solve the inverse kinematics, depending on the context. The most common methods are now briefly reviewed.

3.1. Inverse Jacobian

If the Jacobian matrix is square and non-singular, then one can write

$$J^I = J^{-1}. \tag{5}$$

However, in a context of physical human-robot interaction, this solution can be applied only to special cases. Indeed, the inverse kinematics must be solved for a link of the robot on which a shell is mounted.

In general, the degree of freedom of the link is different from the number of joints between the base of the robot and the link. In this case, the Jacobian matrix is therefore not square. Also, it is desired that the user be able to control the robot even when it is close to singular configurations.

The above considerations make this solution not generally applicable to pHRI robots in which the human user can apply motions to different links, which is the case in this work.

3.2. Pseudo-inverse Jacobian

If the Jacobian matrix is not square but of full rank, the inverse problem can be solved by taking the pseudo-inverse rather than simply the inverse of the matrix. The pseudo-inverse solution is also known as the *Moore-Penrose* [24] inverse, noted $\boldsymbol{J}^{\dagger}$. One then has

$$\boldsymbol{J}^{I} \equiv \boldsymbol{J}^{\dagger}. \tag{6}$$

For a full-rank matrix, the *Moore-Penrose* generalized inverse yields the least-square solution for an overdetermined system, namely

$$\boldsymbol{J}^{\dagger} = (\boldsymbol{J}^{T}\boldsymbol{J})^{-1}\boldsymbol{J}^{T}, \tag{7}$$

meaning here that there are more degrees of freedom at the point of application of the inverse kinematics than there are joints between the base of the robot and the point of application.

For a full-rank underdetermined system of equations, the *Moore-Penrose* generalized inverse corresponds to the minimum norm solution, namely

$$\boldsymbol{J}^{\dagger} = \boldsymbol{J}^{T}(\boldsymbol{J}\boldsymbol{J}^{T})^{-1}, \tag{8}$$

meaning here that there are more joints between the base and the point of application of the inverse kinematics than there are degrees of freedom at the point of application.

In the case of an overdetermined system, Eq. (7) yields the Cartesian velocity vector that is as close as possible (in the sense of the least squares) to the prescribed Cartesian velocity vector.

In the case of an underdetermined system, Eq. (8) yields the solution with the smallest norm for the joint velocity vector that produces the prescribed Cartesian velocity vector.

The pseudo-inverse method provides a solution for non-square Jacobian matrices, but requires nonetheless a matrix inversion. The problem of a rank-deficient matrix remains.

3.3. Damped pseudo-inverse Jacobian

As mentioned above, when the robot is in a singular configuration, the Jacobian matrix is not of full rank. This means that a matrix inversion is not possible. However, this problem is not unconquerable.

In order to avoid the inversion of a matrix which is not of full rank, one can modify the matrix to make it invertible while slightly altering the solution. This method was first used in refs. [25, 26]. For instance, considering Eq. (7), the damped pseudo-inverse can be written as

$$\boldsymbol{J}^{\dagger} = (\boldsymbol{J}^{T}\boldsymbol{J} + \lambda^{2}\boldsymbol{I})^{-1}\boldsymbol{J}^{T} \tag{9}$$

where λ is the damping coefficient and $\boldsymbol{I}$, the identity matrix.

Using this approach, a solution that does not exactly meet the required Cartesian velocities is obtained. Nevertheless, it can be shown that by choosing an appropriate damping coefficient, a solution suitable for the application can be obtained. Using the damped pseudo-inverse of the Jacobian, it can be guaranteed that the maximum interaction force between the user and the shells will remain small, even near singular configurations. With other approaches, the interaction force may be higher, although limited.

3.4. Transpose Jacobian

Another possible approach to solve the inverse kinematic problem is to consider the robot as a quasi-static system and to assume that the desired speed at the point of application of the inverse kinematic is in fact a virtual force. Then, the resulting moment at the joints can be found from the transpose of the Jacobian matrix [27, 28]. In other words, the generalized inverse is taken as

$$J^I \equiv \alpha J^T \tag{10}$$

and Eq. (4) becomes

$$\Delta\boldsymbol{\theta} = \alpha J^T \boldsymbol{t} \tag{11}$$

where α is a scaling factor.

While this method does not require a matrix inversion, it is not exactly equivalent either (hence the $\Delta\boldsymbol{\theta}$ notation rather than $\dot{\boldsymbol{\theta}}$). In order to determine the value of α that minimizes the error introduced by the use of Eq. (10), it is first noted that $(JJ^T\boldsymbol{t})^T\boldsymbol{t} \geq 0$ for all J and $\boldsymbol{t}$. Indeed, one has

$$(JJ^T\boldsymbol{t})^T\boldsymbol{t} = (J^T\boldsymbol{t})^T(J^T\boldsymbol{t}) = \left\| J^T\boldsymbol{t} \right\|^2 \geq 0. \tag{12}$$

If the angles $\Delta\boldsymbol{\theta}$ are changed by a sufficiently small $\alpha \geq 0$, then the end-effector position should change by $\boldsymbol{t} \approx \alpha JJ^T\boldsymbol{t}$. The value of α that minimizes the error between $\boldsymbol{t}$ and $\alpha JJ^T\boldsymbol{t}$ is then obtained by

$$\alpha = \frac{\boldsymbol{t}^T(JJ^T\boldsymbol{t})}{(JJ^T\boldsymbol{t})^T(JJ^T\boldsymbol{t})}. \tag{13}$$

3.5. Singular value decomposition

An alternative method to compute the pseudo-inverse of a matrix is the singular value decomposition. If J is the Jacobian matrix, then its singular value decomposition can be written as [29]

$$J = UDV^T \tag{14}$$

where U and V are orthogonal matrices and D is a diagonal matrix containing the singular values, σ_i. The rank r of J is equal to the number of non-zero singular values.

The singular value decomposition of J can be rewritten as

$$J = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T \tag{15}$$

where $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ are the i-th columns of U and V, respectively.

Substituting Eq. (14) into Eqs. (7) or (8), one obtains

$$J^\dagger = VD^\dagger U^T \tag{16}$$

where $D^\dagger$ is the pseudo-inverse of D such that

$$d_{i,i}^\dagger = \begin{cases} 1/d_{i,i}, & d_{i,i} \neq 0 \\ 0, & d_{i,i} = 0 \end{cases}. \tag{17}$$

Then, Eq. (16) can be rewritten as

$$J^\dagger = \sum_{i=1}^{r} \sigma_i^{-1} \boldsymbol{v}_i \boldsymbol{u}_i^T \tag{18}$$

This method requires the computation of the reciprocal of non-zero scalars only, but it requires a singular value decomposition.

When the robot is near a singularity, at least one of the singular values σ_i is close to zero, making the inversion near singularities unstable. Similarly to Eq. (9), one can damp this behavior by introducing a
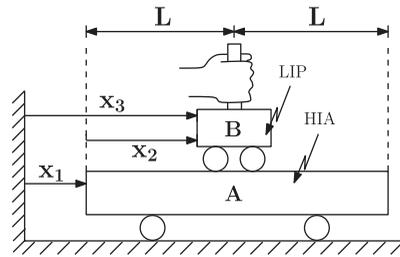
Figure 1. *Representation of the one-degree-of-freedom macro-mini manipulator, figure taken from* [13].

damping coefficient. Equation (18) then becomes

$$J^{\dagger} = \sum_{i=1}^{r} \frac{\sigma_i}{\sigma_i^2 + \lambda^2} v_i u_i^T.$$ (19)

In the *selectively damped least squares* method [30], a different damping value is chosen for each singular value σ_i. The damping coefficients depend not only on the current configuration of the manipulator but also on the distance between the end-effector and the target position.

4. Serial manipulator with low-impedance passive link shells

As mentioned above, this paper aims at developing strategies to exploit the concept of low-impedance link shells proposed in refs. [22] and [23]. The concept behind the low-impedance shell introduced in refs. [22] and [15] is now briefly recalled.

The low-impedance shell draws from the macro-mini architecture [11, 12] in which a small (mini) robot is attached to a larger (macro) active robot to control the motion of the end-effector and allow a low-impedance interaction with a human user. The mini manipulator can be either active or passive. The goal of this architecture is to decouple the macro's impedance from the task and from the human user when used in a context of pHRI. In some cases, the mini manipulator also carries the payload [13, 14, 15]. Also, the mini robot can be passive, in which case the user feels the payload's impedance.

The concept of the macro-mini architecture is explained in detail in refs. [13, 14, 15, 22] and briefly recalled here for convenience. Figure 1 illustrates a one-degree-of-freedom macro-mini manipulator for simplicity. The mini manipulator B is mounted on the macro manipulator A. The range of motion of the mini robot B, relative to the macro robot A, is given by $2L$. Its neutral position $x_2 = 0$ is determined, usually the center of its reachable range, relative to the macro robot A. When the user manipulates the mini robot B away from its neutral position, the macro robot A moves in order to follow it. Therefore, the user indirectly guides the macro robot A throughout the workspace. The maximum speed and acceleration that the user can impart to the mini robot B depends on the ability of the macro robot A to catch up with the mini robot. The complete kinematic analysis is presented in refs. [13, 14, 15, 22].

The principle of the macro-mini manipulators can be applied to a serial robot by mounting low-impedance shells around the links of the robot, as shown in Fig. 2 and described in refs. [22] and [23]. The shell is a low-impedance passive mechanism mounted on the link of a serial manipulator, rather than its end-effector. It is then decoupled from the structure of the robot and the payload. Although the shells are mounted on passive mechanisms, their joints are equipped with encoders so that the relative motion between the shells and the links can be measured. Given the robot's speed and desire to maintain the shell in its neutral configuration by following it, the user interaction is decoupled from the robot's impedance. The mechanism connecting the shell to the robot link includes preloaded springs and mechanical limits that tend to return the shell to a neutral configuration when no external forces are applied on it.
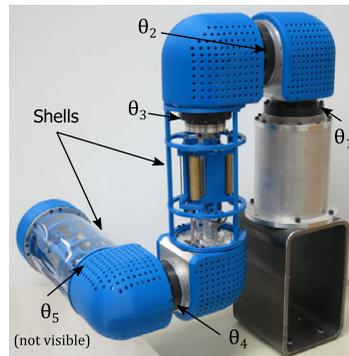
Figure 2. *Photograph of the experimental 5-DOF robot with the low-impedance displacement sensors mounted on links 3 and 5, figure taken from [23].*
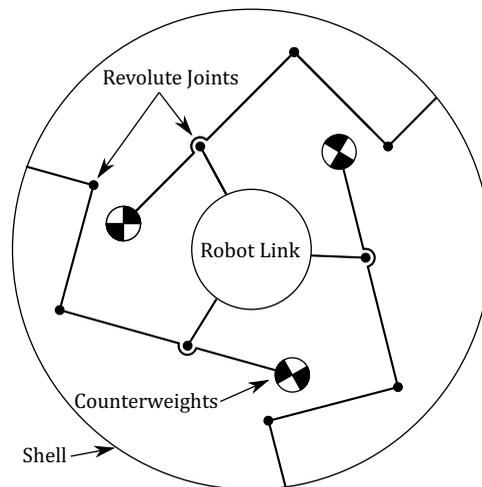

Figure 3. *Cross-section link and a 3 DOFs low-impedance link shell architecture, figure taken from [23].*

For the purpose of this paper, the 5-DOF serial manipulator introduced in ref. [22] is used and its architecture is recalled here for convenience. The general architecture is presented in Fig. 2. The architecture is based on two clusters of joints: one with three motors near the base and the other with two motors near the end-effector.

When a single shell is used near the end-effector, the 6-DOF shell developed in ref. [22] works well. However, in such a case, only the last link of the robot can be manipulated by the human user. In order to enhance the interaction between a user and the robot, it is desirable to implement additional shells on the links of the robot closer to the base. Since these links have fewer degrees of freedom, using 6-DOF shells becomes inefficient. Furthermore, the high number of input signals becomes more complicated to interpret.

For this reason, the 6-DOF shell used in ref. [22] is replaced by two 3-DOF sensitive shells in ref. [23]. The 3-DOF shells are easier to balance and require lower pre-loads which means smaller and more intuitive interaction forces for the user. They also use revolute joints rather than prismatic joints. Instead of using a Gough-Stewart architecture, a planar 3-RRR parallel mechanism is used, thereby providing two translational degrees of freedom in the directions orthogonal to the link axis and one rotational degree of freedom around the link axis, as illustrated in Fig. 3. All sensors are positioned at the base of the shell, which simplifies the wiring.

The shells are mounted respectively on the third and fifth links of the 5-DOF serial manipulator. Each of the shells has two translational DOFs in the plane perpendicular to the link axis and one rotational DOF around the link axis.

5. Inverse kinematics and trajectory planning strategies

The different strategies presented in Section 3 are general approaches to solving the inverse kinematics problem at the velocity level. The objective of this research is to use these strategies in conjunction with the input motions measured between the shells and the links on which they are mounted. To this end, these general strategies must be adapted.

The mapping of the desired motion – measured by the encoders mounted on the shell mechanisms – onto the joint space of the robot is described in ref. [23] and is briefly repeated here for convenience. Considering link i of the serial n-DOF robot, one can write

$$t_i = J_i \dot{\theta}_i \tag{20}$$

where J_i is the Jacobian matrix, with dimension $(6 \times i)$, of link i and $\dot{\theta}_i$ is the vector containing the first i joint velocities.

Consider a shell mounted on link i, which has ℓ_i DOFs for the user inputs. The user input vector associated with shell i is noted c_i which contains ℓ_i components.

One can easily eliminate the rows of Eq. (20) that do not have any component corresponding to the DOFs of the shell, so as not to constrain them unnecessarily, by using a selection matrix S_i. Matrix S_i is of dimension $(\ell_i \times 6)$. By expressing the Jacobian matrices in their respective link's space, S_i is defined as a matrix whose components are all zero except for ℓ_i unit components spread between the rows. The j-th component of each row is one if the j-th row of J_i is to be kept.

Using the selection matrix S_i, Eq. (20) is reduced to the relevant DOFs as

$$S_i t_i = S_i J_i \dot{\theta}_i \tag{21}$$

and, therefore, $S_i J_i$ is of dimension $(\ell_i \times i)$. The expression $S_i t_i$ is a function of c_i.

In order to select which joints react to which shell, a reduction matrix R_i is introduced, of dimension $(i \times k_i)$ where $k_i \leq i$ is the number of joints that should be reacting to the shell of link i. Reduction matrix R_i is a matrix whose components are all zero except that each column has its j-th component equal to one if the j-th joint is active. Then, $\dot{\theta}_{ir} = R_i^T \dot{\theta}_i$ contains only the active components of $\dot{\theta}_i$. Eq. (21) can be further reduced to

$$S_i t_i = S_i J_i R_i R_i^T \dot{\theta}_i \tag{22}$$

or

$$t_{is} = W_i \dot{\theta}_{ir} \tag{23}$$

where $W_i = S_i J_i R_i$ is the selected and reduced Jacobian matrix and $t_{is} = S_i t_i$.

Eq. (3) shows that the Jacobian matrix can be partitioned into two submatrices, A and B, the first for the rotational velocities and the second for the translational velocities. Since both submatrices use different units and different scales, the selected and reduced Jacobian should be partitioned, as

$$W_i = \begin{bmatrix} W_{ir} \\ W_{it} \end{bmatrix} \tag{24}$$

where W_{ir} corresponds to the rotations and W_{it}, to the translations. Vector t_{is} is partitioned as well to yield

$$t_{isr} = W_{ir} \dot{\theta}_{irr} \tag{25}$$

$$t_{ist} = W_{it} \dot{\theta}_{irt} \tag{26}$$

where $\boldsymbol{t}_{isr}$ represents the rotational components of $\boldsymbol{t}_{is}$ and $\boldsymbol{t}_{ist}$, the translational components.

Finally,

$$\dot{\boldsymbol{\theta}}_{ir} = \dot{\boldsymbol{\theta}}_{irr} + \dot{\boldsymbol{\theta}}_{irt} \tag{27}$$

which is equivalent to considering shell i as two distinct sensors, one for the rotations and the other for the translations.

In order to solve Eqs. (25) and (26) for the joint velocities, $\boldsymbol{W}_{ir}$ and $\boldsymbol{W}_{it}$ must be inverted. These matrices may not be square, which means that the simple matrix inverse (Section 3.1) cannot be used in this context. Also, the method should be applicable in singular configurations, which rules out the straightforward use of the pseudo-inverse method (Section 3.2). Therefore, the damped pseudo-inverse (Section 3.3) is used.

For the singular value decomposition (Section 3.5), a different approach can be used. Since this method treats all singular values individually, different options are presented. One can choose to damp all of them as in Eq. (19), whether or not they are near a singularity (meaning the singular value is close enough to zero). On the other hand, one can also decide to damp only those near singularities or even not to consider them at all, that i, equating their contribution to zero in the summation of Eq. (18), since the robot should not be able to move in the singular desired direction.

5.1. Direct rotations

The architecture of the custom-built serial manipulator used in this work is such that the rotational input of the shells is aligned with their respective joint axis. Therefore, one could map the rotational input directly to the current joint only with a proportional function and the two translational inputs to the preceding joints using one of the aforementioned methods. Mathematically, this method could be written as

$$\dot{\boldsymbol{\theta}}_{ir} = \eta r_i \boldsymbol{e}_i + \dot{\boldsymbol{\theta}}_{irt} \tag{28}$$

where η is simply a scaling factor, r_i is the rotational input, and $\boldsymbol{e}_i$ is a vector whose components are all zero except for the i-th component which is equal to 1. In this case, $\boldsymbol{t}_i$ becomes a two-component vector of the translational inputs.

6. Simulation and comparison of the trajectory planning schemes

To summarize, the method proposed in ref. [23] – and summarized in Section 5 – to map the inputs from the link shells onto a serial manipulator joint velocities can be implemented using any of the five approaches described in Section 3. From these approaches, this paper studies particularly the effectiveness of the Damped Pseudo-Inverse (Section 3.3), the Jacobian transpose (Section 3.4), and the Singular Value Decomposition (Section 3.5) inverse kinematics strategies in order to assess their capability to provide an intuitive response to the user inputs. To this end, simulation results based on an assumed input trajectory are presented in this section.

It can be shown that Eqs. (9) and (19) yield the same results. Indeed, the singular value decomposition is just another algorithm to compute the general solution of a linear system of equations. For this reason, comparing the simple damped singular value decomposition grants no additional information. Rather, a variable damping coefficient λ^2 is used in Eq. (19), given by

$$\lambda_i^2 = \begin{cases} -\lambda^2(3\sigma_i^2 - 2\sigma_i^3) - \lambda^2 & , \sigma_i < 1 \\ 0 & , \sigma_i \geq 1 \end{cases} \tag{29}$$

where λ represents the maximum damping coefficient when $\sigma_i = 0$.

Table I. *DH parameters of the 5-DOF serial manipulator.*

i	a_i [m]	b_i [m]	α_i [rad]	θ_i [rad]
1	0	0.2905	$\pi/2$	θ_1
2	0	0.1885	$\pi/2$	θ_2
3	0	0.3835	$\pi/2$	θ_3
4	0	0.1745	$\pi/2$	θ_4
5	0	0.1900	$\pi/2$	θ_5

Table II. *Maximum speed and acceleration for the robot's joints.*

| j	$\dot{\theta}_{j,max}$ [rad/s]	$\ddot{\theta}_{j,max}$ [rad/s^2]
1	$4\pi/3$	22
2	$4\pi/3$	22
3	$4\pi/3$	54
4	$4\pi/3$	54
5	2π	60
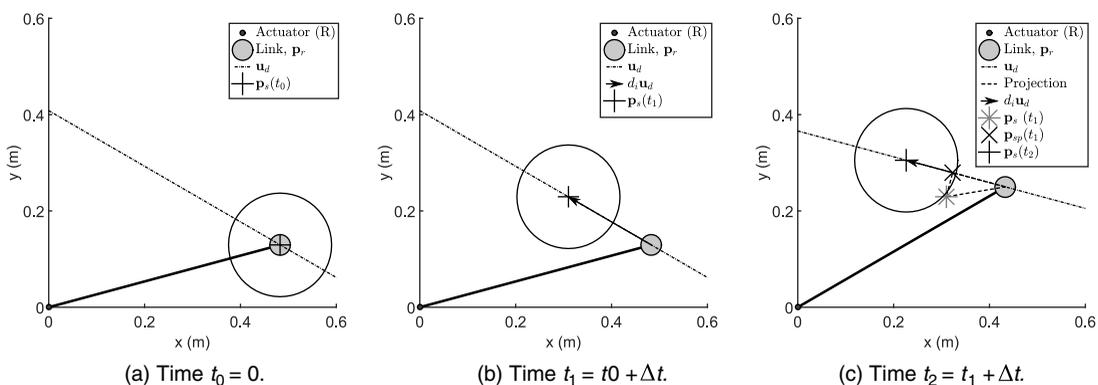
Figure 4. *Schematic representation of the shell trajectory for a 1-DOF shell mounted on a link.*

6.1. Robot parameters

The robot parameters used for the simulation correspond to the serial manipulator developed in ref. [22], as given in Tables I and II.

6.2. Trajectory

A trajectory for the low-impedance shell is needed to compare the different strategies studied. However, since the robot's response may differ depending on the strategy used, it is not possible to specify a shell trajectory in the fixed reference frame. Moreover, the shell's location in space is constrained by the link's pose and cannot be prescribed arbitrarily. For these reasons, determining a Cartesian shell trajectory *a priori* is difficult. A quasi-Cartesian trajectory is used instead, as described here. The procedure is illustrated in Fig. 4 considering a 1-DOF shell for simplicity.

First, a general harmonic motion is determined, given by

$$d = D\sin(2\pi f t) \tag{30}$$

as well as a feasible direction of motion $\boldsymbol{u}_d$, expressed in the link's space. For instance, $\boldsymbol{u}_d$ must be in the plane perpendicular to the link's axis for the 3-DOF shell presented earlier. For the 1-DOF example, $\boldsymbol{u}_d$ is defined by the only possible direction of motion, shown by the dotted line in Fig. 4.

At $t_0 = 0$, the shell is at rest, meaning it is at the origin of the link's coordinate system located in the neutral configuration.

$$\boldsymbol{p}_s(t_0) = \boldsymbol{p}_{ref}(t_0). \tag{31}$$

At $t_1 = t_0 + \Delta t$, where Δt is given by the sampling frequency, the shell is moved in the desired direction a distance $d_1 = d(t_1) - d(t_0)$ based on the harmonic function given in Eq. (30)

$$\boldsymbol{p}_s(t_1) = \boldsymbol{p}_s(t_0) + d_1 \boldsymbol{Q}_l \boldsymbol{u}_d \tag{32}$$

where the matrix $\boldsymbol{Q}_l$ is used to express $\boldsymbol{u}_d$ in the Cartesian space, rather than the link space. The robot moves according to the method chosen for solving the inverse kinematics.

At $t_2 = t_1 + \Delta t$, the shell might not be in a feasible location relative to its associated link due to the reaction of the robot. The shell position is then projected onto its feasible space to simulate its compliance to the constraints imposed by its architecture during motion. For example, Fig. 4(c) shows that $\boldsymbol{p}_s(t_1)$ is no longer on the dotted line representing its feasible locations. However, it should be kept on this line during motion, hence the projection $\boldsymbol{p}_{sp}(t_1)$. Then, the shell is moved once more in the desired direction a distance $d_2 = d(t_2) - d(t_1)$

$$\boldsymbol{p}_s(t_2) = \boldsymbol{p}_{sp}(t_1) + d_2 \boldsymbol{Q}_l \boldsymbol{u}_d \tag{33}$$

and the cycle starts anew.

The maximum range of the shell relative to the link is verified for each time step, $\left\| [\boldsymbol{p}_s(t_i)]_l \right\| \leq p_{max}$, where $[\boldsymbol{p}_s(t_i)]_l$ represents the shell's position relative to its link and p_{max} is the maximum range of motion of the shell. The joints' acceleration and speed are also limited to the values given in Table II, and $\lambda^2 = 0.1$ is used for each of the methods using a damping coefficient.

6.3. Results

In order to compare the different methods for solving the inverse kinematics, the angle ψ between the link's Cartesian velocity and the shell's motion is studied. Indeed, a smaller angle ψ means that the robot better follows the user's inputs.

As previously mentioned, two low-impedance shells are attached to the robot, mounted on the third and fifth links. Since the architecture of the joint upstream from the shell, relative to the shell, is the same for both shells, one can study the motion of either one. The shell on the third link is chosen here.

The results obtained vary depending on the direction of $\boldsymbol{u}_d$ relative to the robot's configuration. For this reason, a direction referred to as *optimal* is chosen. The initial configuration is chosen as the one shown in Fig. 2 and $\boldsymbol{u}_d$ is parallel to the fifth link, in this configuration. This direction is called *optimal* for two reasons. First, a force in this direction would produce a pure moment around the second joint's axis, regardless of the robot's configuration. Second, the same force would produce a pure moment around the first joint's axis for the robot's initial configuration. Choosing such a direction should yield $\psi = 0$ in the initial configuration, that is, the direction of motion coincides with the motion of the shell, hence the optimal direction.

Three trajectories are studied. The parameters of Eq. (30) are given in Table III for each of the trajectories. Trajectory 1 represents a normal trajectory where both the robot and the shell remain within their limits. This means that the joints do not reach their maximum speed and the shell does not reach its physical limits.

Trajectories 2 and 3 represent trajectories where both the robot and the shell reach their limits. The former uses a high amplitude while the latter uses a high frequency to achieve this.

Table III. *Trajectory parameters.*

Trajectory	D [mm]	f [Hz]
1	100	1
2	500	1
3	100	3
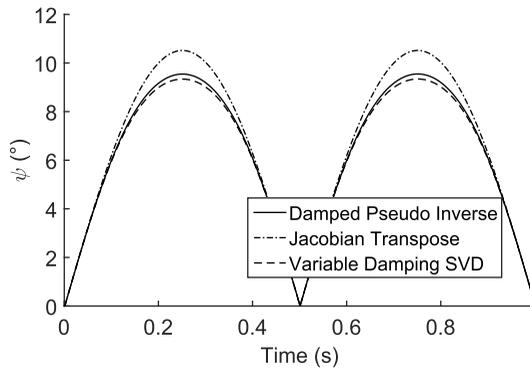
***Figure 5.*** *Trajectory 1: angle ψ between link speed and shell speed.*
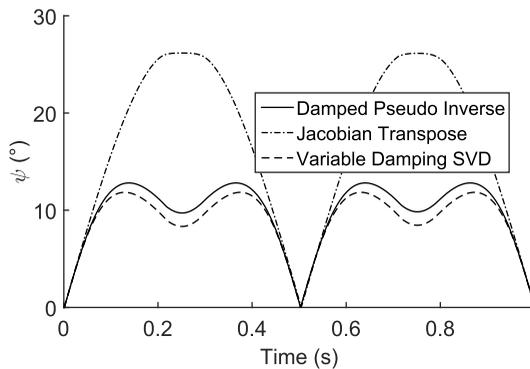
Figure 6. *Trajectory 2: angle ψ between link speed and shell speed.*

The results obtained for angle ψ, that is, the angle between the direction of motion of the shell and the link, are presented in Figs. 5, 6, and 7 for the three trajectories and the three methods for inverse kinematics.

The results show little difference (a few degrees) between using the damped pseudo-inverse Jacobian strategy and the singular value decomposition using a variable damping coefficient, as expected. For all three figures, the angles ψ for the Jacobian transpose method are larger than those for the other two strategies, only a few degrees for Figs. 5 and 7, and about double for Fig. 6.

Figures 5 and 7 illustrate similar results if the difference in frequency is omitted. Figure 6 shows a different behavior. Compared to trajectory 1, trajectory 2 corresponds to increasing the velocity and acceleration by a factor of 5 while trajectory 3 corresponds to increasing the velocity by a factor of 3 and the acceleration by a factor of 9. It should be pointed out that the key factors in characterizing the motion are the velocity and acceleration, whose effect is clearly seen here.
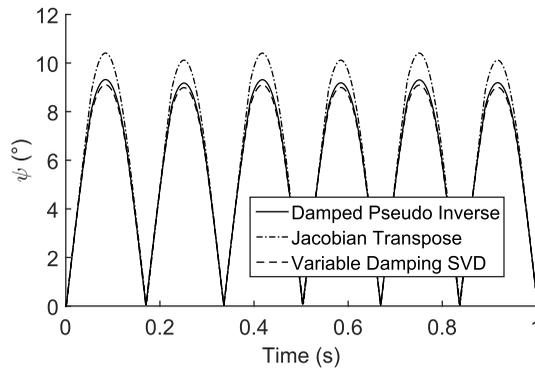
Figure 7. *Trajectory 3: angle ψ between link speed and shell speed.*

Figure 6 illustrates two different phenomena, mainly the large values of angle ψ for the Jacobian transpose strategy and the angle oscillation near the change of direction of the movement for the other two methods. Both can be explained similarly.

As previously mentioned, Fig. 2 shows the robot's initial configuration. Link 3 is parallel to link 1. The joint axes are aligned with their respective link. Vector $\boldsymbol{u}_d$ is in the direction of link 5. When shell 3 is moved, joints 1 and 2 move according to the right-hand rule. As the position of joint 2 (θ_2) increases, $\boldsymbol{u}_d$ aligns itself more and more with the axis of joint 1. In other words, when θ_2 has moved by 90°, $\boldsymbol{u}_d$ is parallel to link 1. This means that at this point, joint 1 can only produce a motion of the link that is perpendicular to $\boldsymbol{u}_d$, a purely parasitic motion. The Jacobian would be singular for this particular direction and configuration. The robot could then be reduced to one DOF when considering link 3. Only joint 2 would be active at this point.

During the motion, θ_2 increases and the angle between $\boldsymbol{u}_d$ and link 1 decreases, which means that the parasitic nature of the motion of joint 1 increases. The Jacobian matrix accounts for this and reduces the involvement of joint 1. At some point, the reduced motion of joint 1 equates and then overcompensates its parasitic motion. Then, the system approaches the one-DOF robot mentioned earlier. If θ_2 were to reach 90°, the angles shown in Fig. 6 would drop back to zero. They would then rise again as θ_2 moves away from 90°, which explains the oscillation.

This phenomenon is also visible for the Jacobian transpose strategy, although much less drastically. The point at which the reduced motion of joint 1 equates the parasitic link motion it induces happens at a higher value of θ_1. This point is closer to the change in direction of the trajectory, which decreases greatly the oscillations. This means that joint 1 moves more than for the other two strategies, incurring a larger parasitic motion. Hence, the angles between the link's Cartesian velocity and the shell's Cartesian motion are larger.

The results show that the Jacobian transpose strategy yields larger values of angle ψ for all three figures. Figure 6 shows a maximum value of angle ψ slightly over 25° where the other two strategies yield about 13°. Experimentation would be required to determine if a user can perceive the difference of about 12° as the Jacobian transpose strategy confers other advantages which are non-negligible. The Jacobian transpose strategy does not require a matrix inversion and the oscillation of alignment are greatly reduced.

7. Conclusion and future work

As previously mentioned, this paper's objective is to quantitatively compare different strategies to solve the inverse kinematics when used in conjunction with the low-impedance sensitive shells developed in refs. [22] and [23]. To this end, the different strategies are presented and then adapted to fit the current application. The manipulator and the low-impedance sensitive shell are described as well. Then, a simulation is conducted to compare quantitatively the different strategies.

Since the relative motion between the shell and its associated link is taken as the desired link Cartesian velocities, the instantaneous angle between the shell's motion and the link's velocity is studied. Also, since a Cartesian trajectory for the shell cannot be determined beforehand, a procedure is developed to generate such a trajectory while taking into account the architecture of the robot and the shells.

The results from the simulations show that the performances vary greatly depending on the robot's configuration and the direction of the shell's motion. Furthermore, the Jacobian transpose strategy tends to yield the worst results while the damped pseudo-inverse and the singular value decomposition achieve similar results, as expected. Nevertheless, the advantage that the Jacobian transpose grants by avoiding the matrix inversion is not to be dismissed.

In the future, experimentation should be conducted to assert whether a human user can discriminate between the different results obtained in the simulations. If it is not the case, then the Jacobian transpose strategy may be preferable to the other two because of its computational simplicity.

References

[1] T. S. Tadele, T. de Vries and S. Stramigioli, "The safety of domestic robotics: A survey of various safety-related publications," *IEEE Robot. Automat. Mag.* **21**(3), 134–142 (2014).

[2] J. Krüger, T. Lien and A. Verl, "Cooperation of human and machines in assembly lines," *CIRP Ann.* **58**(2), 628–646 (2009).

[3] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier and P. Fraisse, "Collaborative manufacturing with physical human–robot interaction," *Robot Comput-Integr. Manuf.* **40**(2), 1—13 (2016).

[4] S. Haddadin, A. Albu-Schäffer and G. Hirzinger, "Requirements for safe robots: Measurements, analysis and new insights," *Int. J. Robot. Res.* **28**(11-12), 1507–1527 (2009).

[5] A. Izadbakhsh, P. Kheirkhahan and S. Khorashadizadeh, "Fat-based robust adaptive control of electrically driven robots in interaction with environment," *Robotica* **37**(5), 779–800 (2019).

[6] A. Izadbakhsh, "Fat-based robust adaptive control of electrically driven robots without velocity measurements," *Nonlinear Dynam.* **89**(1), 289–304 (2017).

[7] V. Duchaine and C. M. Gosselin, "General Model of Human-Robot Cooperation Using A Novel Velocity Based Variable Impedance Control," **In:** *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)* (March 2007) pp. 446–451.

[8] A. Lecours, B. Mayer-St-Onge and C. Gosselin, "Variable Admittance Control of A Four-Degree-of-freedom Intelligent Assist Device," **In:** *2012 IEEE International Conference on Robotics and Automation* (May 2012) pp. 3903–3908.

[9] P. D. Labrecque and C. Gosselin, "Variable admittance for pHRI: from intuitive unilateral interaction to optimal bilateral force amplification," *Robot Comput.-Integr. Manuf.* **52**(8), 1–8 (2018).

[10] V. Duchaine and C. Gosselin, "Safe, Stable and Intuitive Control for Physical Human-Robot Interaction," **In:** *2009 IEEE International Conference on Robotics and Automation* (May 2009) pp. 3383–3388.

[11] A. Sharon, N. Hogan and D. E. Hardt, "High Bandwidth Force Regulation and Inertia Reduction Using A macro/micro Manipulator System," **In:** *Proceedings 1988 IEEE International Conference on Robotics and Automation*, vol. **1** (1988) pp. 126–132.

[12] O. Khatib, "Augmented Object and Reduced Effective Inertia in Robot Systems," **In:** *1988 American Control Conference* (1988) pp. 2140–2147.

[13] P. D. Labrecque, J. Haché, M. Abdallah and C. Gosselin, "Low-impedance physical human-robot interaction using an active–passive dynamics decoupling," *IEEE Robot. Automat. Lett.* **1**(2), 938–945 (2016).

[14] P. D. Labrecque, T. Laliberté, S. Foucault, M. E. Abdallah and C. Gosselin, "uMan: a low-impedance manipulator for human–robot cooperation based on underactuated redundancy," *IEEE/ASME Trans. Mechatron.* **22**(3), 1401–1411 (2017).

[15] N. Badeau, C. Gosselin, S. Foucault, T. Laliberté and M. E. Abdallah, "Intuitive physical human-robot interaction: using a passive parallel mechanism," *IEEE Robot. Automat. Mag*. **25**(2), 28–38 (2018).

[16] C. M. Gosselin, M. T. Masouleh, V. Duchaine, P.-L. Richard, S. Foucault and X. Kong, "Parallel Mechanisms of the Multipteron Family: Kinematic Architectures and Benchmarking," **In:** *Proceedings 2007 IEEE International Conference on Robotics and Automation* (2007) pp. 555–560.

[17] Z. Ma, H.-H. See, G.-S. Hong, M. H. Ang, A.-N. Poo, W. Lin, P.-Y. Tao and J. S. Short, "Control and Modeling of an End-Effector in A Macro-Mini Manipulator System for Industrial Applications," **In:** *2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)* (2017) pp. 676–681.

[18] J. Li, Y. Guan, H. Chen, B. Wang, T. Zhang, X. Liu, J. Hong, D. Wang and H. Zhang, "A high-bandwidth end-effector with active force control for robotic polishing," *IEEE Access* **8**, 169122–169135 (2020).

[19] A. Vysocky and P. Novak, "Human - robot collaboration in industry," *MM Sci. J.* **2016**, 903–906 (06 2016).

[20] V. Villani, F. Pini, F. Leali and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics* **55**(2), 248–266 (2018).

[21] A. Hentout, A. Mustapha, A. Maoudi and A. Isma, "Human-robot interaction in industrial collaborative robotics: a literature review of the decade 2008-2017," *Adv. Robot.* **33**(15-16), 764–799 (07 2019).

[22] G. Boucher, T. Laliberté and C. Gosselin, "A Parallel Low-Impedance Sensing Approach for Highly Responsive Physical Human-Robot Interaction," **In:** *2019 International Conference on Robotics and Automation (ICRA)* (2019) pp. 3754–3760.

[23] T. Laliberté and C. Gosselin, "Low-impedance displacement sensors for intuitive physical human-robot interaction: motion guidance, design, and prototyping," *IEEE Trans. Robot.* **38**(3), 1518–1530 (2021).

[24] R. Penrose, "A generalized inverse for matrices," *Math. Proc. Cambridge* **51**(3), 406–413 (1955).

[25] C. W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods," *IEEE Trans. Syst. Man Cybern*. **16**(1), 93–101 (1986).

[26] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *J. Dyn. Syst. Meas. Control* **108**(3), 163–171 (09 1986).

[27] A. Balestrino, G. De Maria and L. Sciavicco, "Robust Control of Robotic Manipulators," **In:** *Proceedings of the 9th IFAC World Congress*, vol. 6 (1984) p. 07.

[28] W. A. Wolovich and H. Elliott, "A Computational Technique for Inverse Kinematics," **In:** *The 23rd IEEE Conference on Decision and Control* (1984) pp. 1359–1363.

[29] G. H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," *Numer. Math.* **14**(5), 403–420 (1970).

[30] S. R. Buss and J.-S. Kim, "Selectively damped least squares for inverse kinematics," *J. Graph. Tools* **10**(3), 37–49 (2005).