



HAL
open science

On inverse kinematics with inequality constraints: new insights into minimum jerk trajectory generation

Wael Suleiman

► **To cite this version:**

Wael Suleiman. On inverse kinematics with inequality constraints: new insights into minimum jerk trajectory generation. *Advanced Robotics*, 2016, 30 (17-18), pp.1164-1172. 10.1080/01691864.2016.1202136 . hal-01355640

HAL Id: hal-01355640

<https://hal.science/hal-01355640>

Submitted on 24 Aug 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Regular Paper

On Inverse Kinematics with Inequality Constraints: New Insights Into Minimum Jerk Trajectory Generation

Wael Suleiman*

*Electrical and Computer Engineering Department,
Faculty of Engineering, University of Sherbrooke,
2500, boul. Universite, Sherbrooke, Quebec (Canada) J1K 2R1.
(Received 00 Month 201X; accepted 00 Month 201X)*

The problem of inverse kinematics is revisited in the present paper. The paper is focusing on the problem of solving the inverse kinematics problem while minimizing the jerk of the joint trajectories. Even-though the conventional inverse kinematics algorithms have been proven to be efficient in many applications, it has been proven that constraints on the accelerations or the jerk cannot be guaranteed, and even yields to divergence or makes the problem unsolvable. The proposed algorithm yields smooth velocity and acceleration trajectories, which are highly desired features for industrial robots. The algorithm uses the joint jerk as the control parameter instead of the classical use of the joint velocity, as a result constraints on the jerk function can be easily incorporated.

To validate the proposed approach, we have conducted several simulations scenarios. The simulation results have revealed that the proposed method can efficiently solve the inverse kinematics problem while considering constraints on the joint acceleration and jerk.

Keywords: Inverse kinematics; Acceleration limits; Smooth motions; Minimum jerk; Quadratic Programming

1. Introduction

The inverse kinematics problem is one of the most studied problems in robotics, and many methods to solve it have been proposed in the literature. Those methods are extensively used in robotics, they can be also combined with other techniques, e.g. force control, to cope with the imperfect modelling of the robot and its dynamic parameters.

Generally speaking, the objective is to find a vector in the configuration space that satisfies constraints in the operational space, in other words the values of the robot's joints to make the end-effector reaches a desired goal (position and orientation) in the Cartesian space. Although in some special cases the problem can be solved analytically [1], [2], [3] it is generally solved numerically [4, 5]. Many efficient and robust numerical algorithms have been proposed for solving the inverse kinematics problem [6], [7], [8], [4], [5], [9], [10], only to cite few.

2. Inverse Kinematics Problem: Revisited

The inverse kinematics problem was originally formulated as follows:

$$\min_{\dot{\mathbf{q}}_t} \dot{\mathbf{q}}_t^T Q \dot{\mathbf{q}}_t$$

* A preliminary version of the paper has been presented at the Workshop on Whole-Body Control for Robots in the Real World, IROS 2014. Corresponding author. Email: Wael.Suleiman@USherbrooke.ca

subject to

$$J\dot{\mathbf{q}}_t = \dot{r}_t \tag{1}$$

where Q is a diagonal and positive semi-definite matrix, J is the Jacobian matrix, $\dot{\mathbf{q}} \in \mathbb{R}^n$ is the joint velocity and \dot{r}_t is the linear and angular velocity of the end-effector.

The optimization problem (1) can be efficiently solved using the pseudo-inverse technique [4, 5] as follows:

$$\dot{\mathbf{q}}_t = \hat{J}^\dagger \dot{r}_t + (I - \hat{J}^\dagger J)z \tag{2}$$

where $\hat{J}^\dagger = Q^{-1} J^T (J Q^{-1} J^T)^{-1}$ and z is an arbitrary vector. If Q is the identity matrix, \hat{J}^\dagger becomes the well-known Moore-Penrose pseudo-inverse. It is worth mentioning that the optimization problem (1) is time-invariant as the time derivative can be simply replaced by the difference. Thus, we obtain the following equivalent optimization problem:

$$\min_{\Delta \mathbf{q}} \Delta \mathbf{q}^T Q \Delta \mathbf{q}$$

subject to

$$J \Delta \mathbf{q} = \Delta r \tag{3}$$

The solution of the above problem is given by:

$$\Delta \mathbf{q}_t = \hat{J}^\dagger \Delta r_t + (I - \hat{J}^\dagger J)\hat{z} \tag{4}$$

However, in order to consider the velocity and joints limits as well as to avoid obstacles, inequality constraints should be added. There, many research in robotics have been conducted to solve the following general inverse kinematics problem:

$$\min_{\dot{\mathbf{q}}_t} \dot{\mathbf{q}}_t^T Q \dot{\mathbf{q}}_t$$

subject to

$$\begin{aligned} J\dot{\mathbf{q}}_t &= \dot{r}_t \\ b^- &\leq A\dot{\mathbf{q}}_t \leq b^+ \\ \dot{\mathbf{q}}^- &\leq \dot{\mathbf{q}}_t \leq \dot{\mathbf{q}}^+ \end{aligned} \tag{5}$$

$\dot{\mathbf{q}}^-$ and $\dot{\mathbf{q}}^+$ are respectively the lower and upper limits of the velocity. A, b^- and b^+ represent the additional constraints that result from considering geometric constraints, such as collision avoidance (for more details, the reader is referred to the Appendix and Scenario 4 in Section 8), or physical constraints, such as joint limits. For more details and examples regarding the above-mentioned general formulation, the reader is referred to [11],[12], [13], [14] [15].

The main differences between (1) and (5) are:

- There is no closed-form expression for the solution of (5). However, it is a convex optimization problem if Q is a positive semi-definite matrix, and in this case, the problem can be efficiently

solved by a Quadratic Programming (QP) solver [12], [13],[14],[15].

- Problem (5) is time-variant.

3. Motivation

The integration of acceleration limits in the inverse kinematics problem of an industrial robot manipulator is motivated by the following reasons:

- Accelerations are proportional to the exerted torques by the joints actuators, therefore ensuring acceleration limits prevents that high torques are applied on the joints and thus prevents damaging the actuators.
- Minimizing the jerk of joints yields to smooth and human-like motions, that is mainly interesting while interacting with humans.

A first attempt to integrate constraints on the accelerations would be using the following finite difference approach:

$$\ddot{\mathbf{q}}_t = \frac{\dot{\mathbf{q}}_t - \dot{\mathbf{q}}_{t-1}}{T} \quad (6)$$

where T is the sampling period.

Then the optimization problem (5) can be modified as follows:

$$\min_{\dot{\mathbf{q}}_t} \dot{\mathbf{q}}_t^T Q \dot{\mathbf{q}}_t$$

subject to

$$\begin{aligned} J\dot{\mathbf{q}}_t &= \dot{r}_t \\ b^- &\leq A\dot{\mathbf{q}}_t \leq b^+ \\ \ddot{\mathbf{q}}^- &\leq \ddot{\mathbf{q}}_t \leq \ddot{\mathbf{q}}^+ \\ T\ddot{\mathbf{q}}^- + \dot{\mathbf{q}}_{t-1} &\leq \dot{\mathbf{q}}_t \leq T\ddot{\mathbf{q}}^+ + \dot{\mathbf{q}}_{t-1} \end{aligned} \quad (7)$$

where $\ddot{\mathbf{q}}^-$ and $\ddot{\mathbf{q}}^+$ are respectively the lower and upper limits of the accelerations. Recall that $\dot{\mathbf{q}}_{t-1}$ is a constant known vector resulting from solving the inverse kinematics problem at the instant $t - 1$.

It is clear that when $T \rightarrow 0$, the additional constraint on the acceleration leads to $\dot{\mathbf{q}}_t = \dot{\mathbf{q}}_{t-1}$, which yields to an infeasible optimization problem. In practice, even when the sampling frequency is 1 *KHz*, that means $T = 10^{-3}$ s, the optimization problem (7) often becomes infeasible.

A second attempt would be to differentiate the equality constraint in (5), thus:

$$J(\mathbf{q}_t)\ddot{\mathbf{q}}_t + \dot{J}(\mathbf{q}_t, \dot{\mathbf{q}}_t)\dot{\mathbf{q}}_t = \ddot{r}_t \quad (8)$$

The generalized inverse kinematics problem becomes:

$$\min_{\ddot{\mathbf{q}}_t} \ddot{\mathbf{q}}_t^T Q \ddot{\mathbf{q}}_t$$

subject to

$$\begin{aligned}
 J(\mathbf{q}_t)\ddot{\mathbf{q}}_t &= \ddot{\mathbf{r}}_t - \dot{J}(\mathbf{q}_t, \dot{\mathbf{q}}_t)\dot{\mathbf{q}}_t \\
 b^- &\leq A\ddot{\mathbf{q}}_t \leq b^+ \\
 \ddot{\mathbf{q}}^- &\leq \ddot{\mathbf{q}}_t \leq \ddot{\mathbf{q}}^+
 \end{aligned} \tag{9}$$

The following assumptions are usually made:

- **Assumption 1:** As the problem is formulated in function of the acceleration $\ddot{\mathbf{q}}_t$, the velocity $\dot{\mathbf{q}}_t$ is then supposed to be constant and equals to $\dot{\mathbf{q}}_{t-1}$.
- **Assumption 2:** As it is very complex to compute $\dot{J}(\mathbf{q}_t, \dot{\mathbf{q}}_t)$ in a closed form, an approximation using finite difference is deployed.

Even-though the Jacobian matrix $J(\mathbf{q}_t)$ can be computed in an efficient way using only the generalized coordinates vector \mathbf{q}_t , using a finite difference approach to compute $\dot{J}(\mathbf{q}_t, \dot{\mathbf{q}}_t)$ (**Assumption 2**) leads, in practice, to ill-conditioning and numerical instability, this is because $J(\mathbf{q}_t)$ is a complex and highly nonlinear function, that yields to divergence or infeasibility of the optimization problem (9).

The main contribution of the paper is providing a general inverse kinematics framework that can efficiently handle constraints on the acceleration or the jerk.

4. Problem Formulation

In order to efficiently consider constraints on the accelerations, one would integrate the acceleration variation (commonly called jerk) into (5). The main idea is to introduce the jerk as the control parameter of the inverse kinematics problem instead of the joint velocity.

The relationship between the position, velocity, acceleration and the jerk can be expressed by the following equation:

$$\begin{bmatrix} \mathbf{q}_t \\ \dot{\mathbf{q}}_t \\ \ddot{\mathbf{q}}_t \end{bmatrix} = \begin{bmatrix} I_n & T & I_n & \frac{T^2}{2} & I_n \\ 0 & I_n & T & I_n \\ 0 & 0 & I_n \end{bmatrix} \begin{bmatrix} \mathbf{q}_{t-1} \\ \dot{\mathbf{q}}_{t-1} \\ \ddot{\mathbf{q}}_{t-1} \end{bmatrix} + \begin{bmatrix} \frac{T^3}{6} I_n \\ \frac{T^2}{2} I_n \\ T I_n \end{bmatrix} \ddot{\mathbf{q}}_t \tag{10}$$

Where $X_t = [\mathbf{q}_t^T \ \dot{\mathbf{q}}_t^T \ \ddot{\mathbf{q}}_t^T]^T \in \mathbb{R}^{3n}$ is the vector that we need to compute, $X_{t-1} = [\mathbf{q}_{t-1}^T \ \dot{\mathbf{q}}_{t-1}^T \ \ddot{\mathbf{q}}_{t-1}^T]^T$ is the actual (known) configuration values. T is the sampling control period, and I_n is identity matrix of size n (number of degrees of freedom). The unknown parameter in (10) is the jerk vector $\ddot{\mathbf{q}}_t$. It is worth to mention that most of modern manipulators have a fixed sampling control period, therefore the inverse kinematics problem should be discretized in consequence.

The objective is therefore to transform the inverse kinematic problem (5) into a function of $\mathbf{u}_t = \ddot{\mathbf{q}}_t$ instead of $\dot{\mathbf{q}}_t$. To this end, we can use the following formula:

$$\dot{\mathbf{q}}_t = \dot{\mathbf{q}}_{t-1} + T \ddot{\mathbf{q}}_{t-1} + \frac{T^2}{2} \mathbf{u}_t \tag{11}$$

The inverse kinematic problem becomes:

$$\min_{\mathbf{u}_t} \mathbf{u}_t^T Q \mathbf{u}_t$$

subject to

$$\begin{aligned} J\mathbf{u}_t &= \tilde{\mathbf{r}}_t \\ \tilde{\mathbf{b}}^- &\leq A\mathbf{u}_t \leq \tilde{\mathbf{b}}^+ \\ u_{\dot{\mathbf{q}}}^- &\leq \mathbf{u}_t \leq u_{\dot{\mathbf{q}}}^+ \end{aligned} \quad (12)$$

where:

$$\begin{aligned} \tilde{\mathbf{r}}_t &= \frac{2}{T^2} (\dot{\mathbf{r}}_t - J\dot{\mathbf{q}}_{t-1} - JT\ddot{\mathbf{q}}_{t-1}) \\ \tilde{\mathbf{b}}^- &= \frac{2}{T^2} (\mathbf{b}^- - A\dot{\mathbf{q}}_{t-1} - AT\ddot{\mathbf{q}}_{t-1}) \\ \tilde{\mathbf{b}}^+ &= \frac{2}{T^2} (\mathbf{b}^+ - A\dot{\mathbf{q}}_{t-1} - AT\ddot{\mathbf{q}}_{t-1}) \\ u_{\dot{\mathbf{q}}}^- &= \frac{2}{T^2} (\dot{\mathbf{q}}^- - \dot{\mathbf{q}}_{t-1} - T\ddot{\mathbf{q}}_{t-1}) \\ u_{\dot{\mathbf{q}}}^+ &= \frac{2}{T^2} (\dot{\mathbf{q}}^+ - \dot{\mathbf{q}}_{t-1} - T\ddot{\mathbf{q}}_{t-1}) \end{aligned} \quad (13)$$

4.1 Incorporating Acceleration and Jerk Constraints

Let us consider the following constraint on the acceleration:

$$\ddot{\mathbf{q}}^- \leq \ddot{\mathbf{q}}_t \leq \ddot{\mathbf{q}}^+ \quad (14)$$

Using the following equation:

$$\ddot{\mathbf{q}}_t = \ddot{\mathbf{q}}_{t-1} + T\mathbf{u}_t \quad (15)$$

The constraint (14) becomes:

$$\frac{1}{T} (\ddot{\mathbf{q}}^- - \ddot{\mathbf{q}}_{t-1}) \leq \mathbf{u}_t \leq \frac{1}{T} (\ddot{\mathbf{q}}^+ - \ddot{\mathbf{q}}_{t-1}) \quad (16)$$

Contrary to the acceleration constraints in (7), the above constraints will not yield to infeasibility as the limits become unbounded ($-\infty \leq \mathbf{u}_t \leq +\infty$) when $T \rightarrow 0$.

Moreover, in order to generate smooth acceleration trajectories, limits can be added on the jerk as follows:

$$u^- \leq \mathbf{u}_t \leq u^+ \quad (17)$$

where u^- and u^+ are the lower and upper limits of the jerk vector.

4.2 Incorporating Joint Limits

A compact and efficient way for dealing with both of velocity and joint limits has been originally proposed in [11]. The constraints on the joint and velocity constraints are replaced by the following con-

straints:

$$\hat{\mathbf{q}}^- \leq \dot{\mathbf{q}}_t \leq \hat{\mathbf{q}}^+ \quad (18)$$

where $\hat{\mathbf{q}}^-$ and $\hat{\mathbf{q}}^+$ are the generalized joint velocity limits defined as follows:

$$\hat{\mathbf{q}}_j^+ = \begin{cases} \dot{\mathbf{q}}_j^+ \frac{(\mathbf{q}_j^+ - \mathbf{q}_j) - \rho_s}{\rho_i - \rho_s} & \text{if } \mathbf{q}_j^+ - \mathbf{q}_j \leq \rho_i \\ \dot{\mathbf{q}}_j^+ & \text{otherwise} \end{cases} \quad (19)$$

$$\hat{\mathbf{q}}_j^- = \begin{cases} \dot{\mathbf{q}}_j^- \frac{(\mathbf{q}_j - \mathbf{q}_j^-) - \rho_s}{\rho_i - \rho_s} & \text{if } \mathbf{q}_j - \mathbf{q}_j^- \leq \rho_i \\ \dot{\mathbf{q}}_j^- & \text{otherwise} \end{cases}$$

$\hat{\mathbf{q}}_j^*$ is the j element of the vector $\hat{\mathbf{q}}^*$, \mathbf{q}_j is the value of joint j , \mathbf{q}_j^+ and \mathbf{q}_j^- are the upper and lower limits of the joint j and $\dot{\mathbf{q}}_j^+$ and $\dot{\mathbf{q}}_j^-$ are the upper and lower velocity limits of the joint j .

ρ_i and ρ_s are user-defined positive constants, and ρ_i is usually called the influence distance. It can be easily proven that the equalities constraints in (19), not only yield a motion within the humanoid's velocity limits, but also the joints limits are respected as well with a safety margin equals to ρ_s :

$$\mathbf{q}_j^- + \rho_s \leq \mathbf{q}_j \leq \mathbf{q}_j^+ - \rho_s$$

5. Practical Implementation

An issue that has been arisen in practice is that the limits on the jerk ($u^- \leq \mathbf{u}_t \leq u^+$) might yield in some special cases to divergence, a way to solve this issue is introducing a slack variable in order to relax the equality constraint, thus the optimization problem becomes:

$$\min_{\mathbf{u}_t, \delta} \mathbf{u}_t^T Q \mathbf{u}_t + \delta^T Q_\delta \delta$$

subject to

$$J \mathbf{u}_t + \delta = \tilde{\mathbf{r}}_t \quad (20)$$

$$\tilde{\mathbf{b}}^- \leq A \mathbf{u}_t \leq \tilde{\mathbf{b}}^+ \quad (21)$$

$$u_{\hat{\mathbf{q}}}^- \leq \mathbf{u}_t \leq u_{\hat{\mathbf{q}}}^+ \quad (22)$$

$$u_{\ddot{\mathbf{q}}}^- \leq \mathbf{u}_t \leq u_{\ddot{\mathbf{q}}}^+ \quad (23)$$

$$u^- \leq \mathbf{u}_t \leq u^+ \quad (24)$$

where:

- δ is a slack variable that has the same dimension as $\tilde{\mathbf{r}}_t$. The matrix Q_δ is a positive matrix defined in such a way that $\|Q\| \ll \|Q_\delta\|$.
- (22) designs the generalized velocity constraints, where:

$$u_{\hat{\mathbf{q}}}^- = \frac{2}{T^2} \left(\hat{\mathbf{q}}^- - \dot{\mathbf{q}}_{t-1} - T \ddot{\mathbf{q}}_{t-1} \right)$$

$$u_{\hat{\mathbf{q}}}^+ = \frac{2}{T^2} \left(\hat{\mathbf{q}}^+ - \dot{\mathbf{q}}_{t-1} - T \ddot{\mathbf{q}}_{t-1} \right)$$

- (23) designs the accelerations constraints, where:

$$u_{\ddot{q}}^- = \frac{1}{T} (\ddot{q}^- - \ddot{q}_{t-1})$$

$$u_{\ddot{q}}^+ = \frac{1}{T} (\ddot{q}^+ - \ddot{q}_{t-1})$$

To reduce the required time to solve the above-mentioned optimization problem, one might consider reducing the number of constraints. In our case, it can be easily noticed that the constraints (22), (23) and (24) are lower and upper bounds on the variable \mathbf{u}_t , they can be therefore transformed into a more compact form as follows:

$$\hat{u}^- \leq \mathbf{u}_t \leq \hat{u}^+$$

where:

$$\hat{u}^- = \max \left\{ u_{\hat{q}}^-, u_{\ddot{q}}^-, u^- \right\}$$

$$\hat{u}^+ = \min \left\{ u_{\hat{q}}^+, u_{\ddot{q}}^+, u^+ \right\}$$
(25)

The above-mentioned operators min and max are defined over the rows of the vectors.

Let us introduce the following parameters:

$$\mathcal{Z} = \begin{bmatrix} \mathbf{u}_t \\ \delta \end{bmatrix}, \quad Q_{\mathcal{Z}} = \begin{bmatrix} Q & \mathbf{0} \\ \mathbf{0} & Q_{\delta} \end{bmatrix}, \quad \mathcal{J} = [J \ I]$$

$$\mathcal{A} = [A \ \mathbf{0}], \quad \mathcal{Z}^+ = \begin{bmatrix} \hat{u}^+ \\ \delta^+ \end{bmatrix}, \quad \mathcal{Z}^- = \begin{bmatrix} \hat{u}^- \\ \delta^- \end{bmatrix}$$
(26)

where δ^- and δ^+ are user-defined constants as lower and upper limits of the slack variable δ .

Thus the optimization problem is transformed into the following classical QP problem:

$$\min_{\mathcal{Z}} \mathcal{Z}^T Q_{\mathcal{Z}} \mathcal{Z}$$

subject to

$$\mathcal{J} \mathcal{Z} = \tilde{r}_t$$

$$\tilde{b}^- \leq \mathcal{A} \mathcal{Z} \leq \tilde{b}^+$$

$$\mathcal{Z}^- \leq \mathcal{Z} \leq \mathcal{Z}^+$$
(27)

6. Complexity Analysis

In order to compare the complexity of the new algorithm with the conventional general inverse kinematics algorithms, one can compare the QP problems (5) and (27):

- On one hand, the dimension of the optimization variable in the proposed algorithm (\mathcal{Z} in (27)) is bigger than the conventional velocity variable (\dot{q}_t in (5)). On the other hand, adding the slack variable in (27) has the advantage of increasing the robustness of the QP problem by relaxing the hard equality constraint.
- The number of equality and inequality constraints is the same.

As a result, the complexity of the QP problems is theoretically quite similar. The numerical simulations (Scenario 3) in Section 8 also confirmed the above conclusion.

However, it is worth to note that the new algorithm can efficiently handle constraints on the acceleration and the jerk that conventional inverse kinematics algorithms simply cannot handle.

7. Implementation Algorithm

The implementation algorithm can be summarized as follows:

- (1) Initial values: $X_0 = [q_0 \ 0 \ 0]^T$ and $t = T$
- (2) Compute the jacobian matrix using the actual configuration (X_{t-T})
- (3) Compute the upper and lower limits of the jerk (25)
- (4) Solve the optimization problem (27) and obtain $u_t = \ddot{q}_t$
- (5) Compute the configuration vector X_t using (10)
- (6) $t=t+T$
- (7) Return to step 2

8. Experimental Results

In order to validate the proposed method, we have conducted three simulations scenarios:

8.1 Scenario 1: acceleration limits

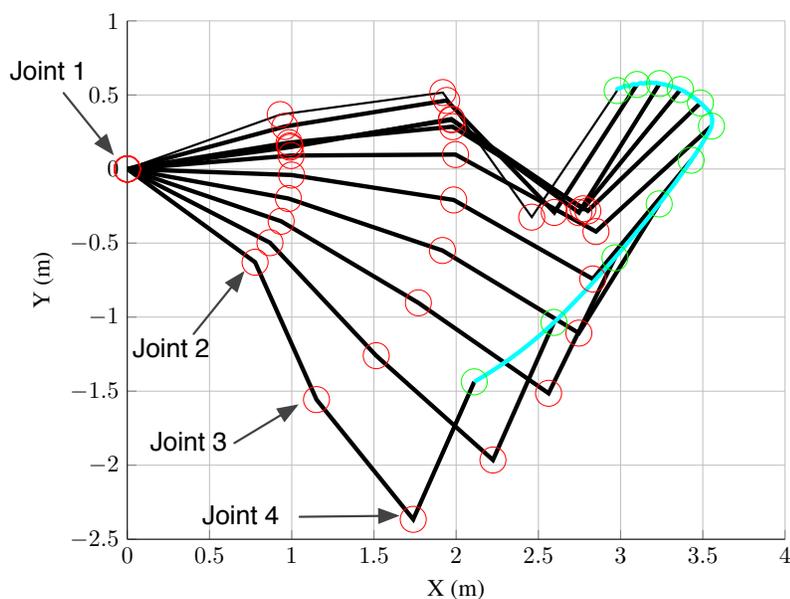


Figure 1. Planar redundant robot: the end-effector is designed by green circle and the robot joints by red circles, and the end-effector trajectory by solid cyan line.

We first considered a planar redundant manipulator (Fig. 1). The objective is to move the end-effector from an initial to a goal position while following a predefined Bezier trajectory.

The inequality constraints that have been considered are:

- (1) Velocity limits (22).
- (2) Acceleration constraints (23).

For this scenario, we chose the following parameters:

- $\dot{q}^+ = -\dot{q}^- = 0.4 \times [1 \ 1 \ 1 \ 1]^T$
- $\ddot{q}^+ = -\ddot{q}^- = 5 \times [1 \ 1 \ 1 \ 1]^T$
- $T = 10^{-3}$

The trajectories of the joints, joint velocity and joint acceleration are given in Fig. 2. The tracking error between the desired trajectory and the executed trajectory is given in Fig. 2(d).

Fig. 2(b) and Fig. 2(c) point out that the constraints on the joint velocity and acceleration are fully respected, and Fig. 2(d) shows that the error between the executed trajectory and the desired one is less than 1.5×10^{-3} m. It is worth to mention that the tracking error is less than 10^{-4} m if we do not consider the acceleration limits, however by adding the acceleration constraints the trajectory is followed as much as possible while respecting the velocity and acceleration constraints.

Fig. 2(a) shows that the joint trajectories are smooth, mathematically speaking they are C^2 functions as a result of considering constraints on the acceleration. It is worth to point out that the conventional formulation Eq. (7) has failed to find a solution after few iterations, this is because the optimization problem became infeasible.

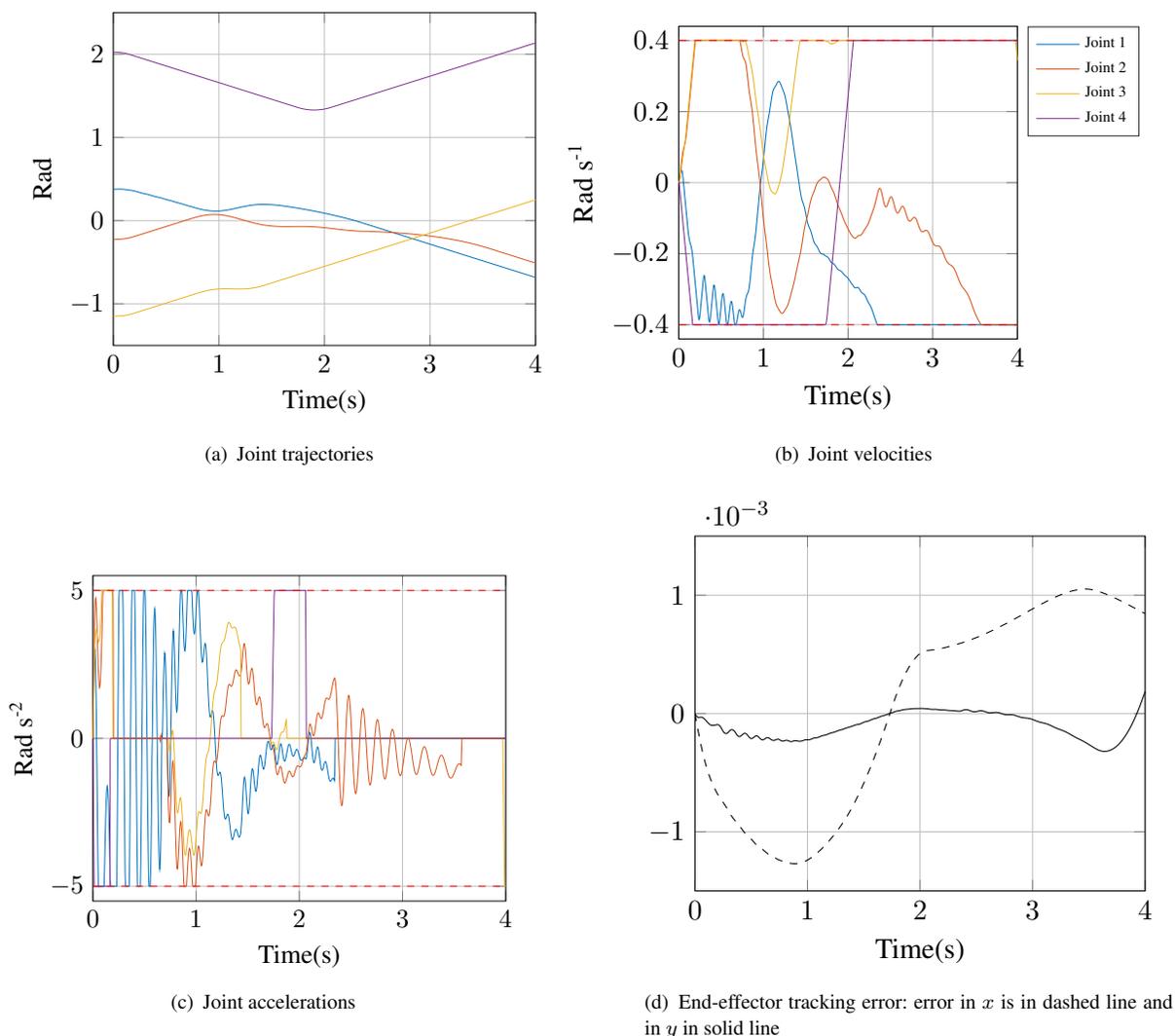


Figure 2. Scenario 1

8.2 Scenario 2: jerk limits

In this scenario, we considered the same planar redundant manipulator of Scenario 1 with the following inequality constraints:

- (1) Velocity limits (22).
- (2) jerk constraints (24).

For this scenario, we chose the following parameters:

- $\dot{q}^+ = -\dot{q}^- = 0.4 \times [1 \ 1 \ 1 \ 1]^T$
- $\ddot{q}^+ = -\ddot{q}^- = 500 \times [1 \ 1 \ 1 \ 1]^T$
- $T = 10^{-3}$

Fig. 3(b) and Fig. 3(c) point out that the constraints on the joint velocity and jerk are fully respected. On one hand, Fig. 3(d) shows that the tracking has been slightly increased in comparison with Scenario 1, this is mainly a result of respecting the jerk constraints. On the other hand, Fig. 3(a) shows that joint trajectories are smoother than those of Scenario 1, this is because they are C^3 functions in this case. It is important to mention that the obtained results cannot be compared with the conventional inverse kinematics methods, this is because those methods simply cannot handle constraints on the jerk.

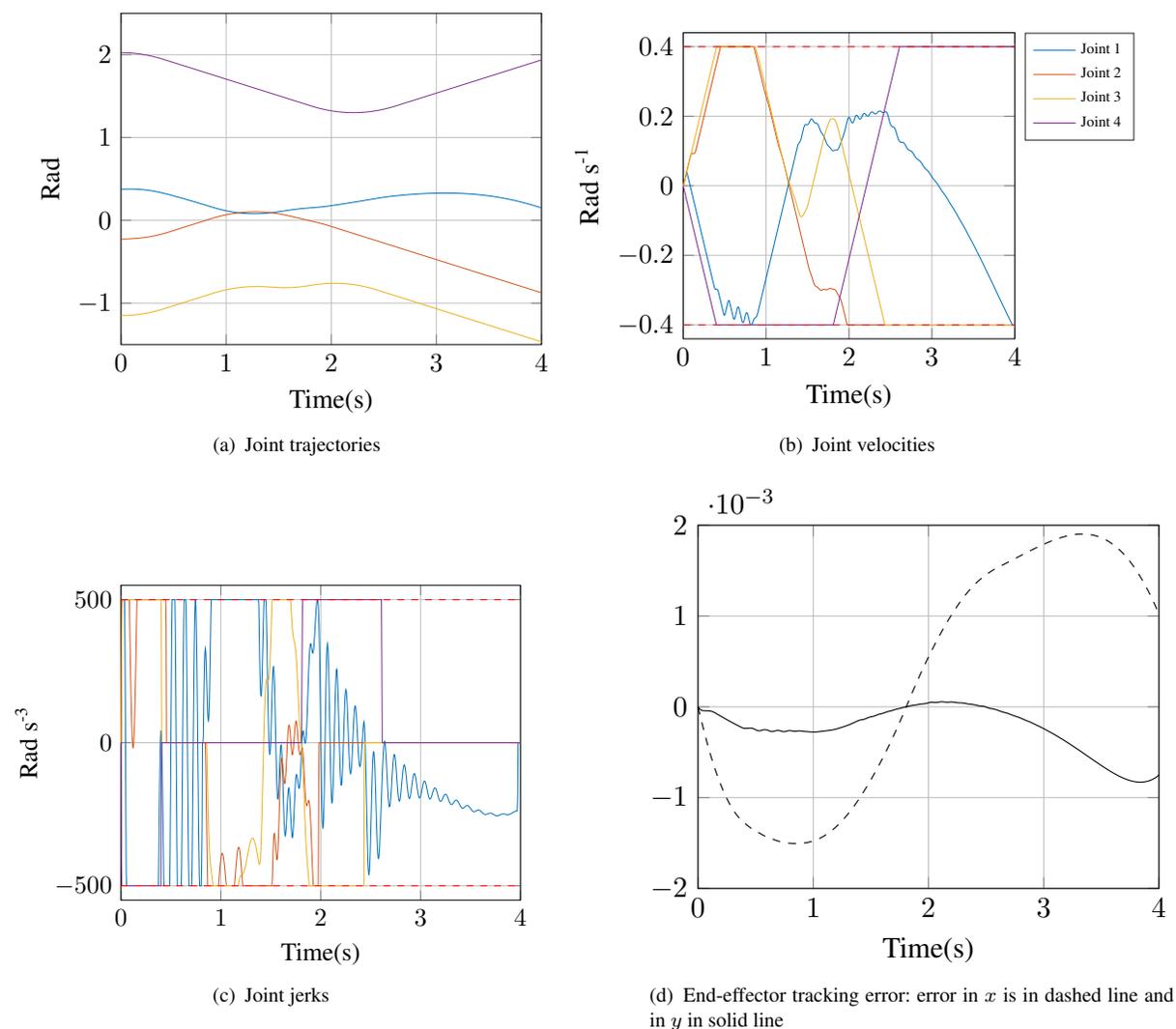


Figure 3. Scenario 2

8.3 Scenario 3: Computational Complexity

A simulated scenario of a Baxter research robot in which the robot should reach a randomized goal pose (position and orientation) from a randomized initial pose, both the initial and goal poses are in the reachable space of the robot (Fig. 4).

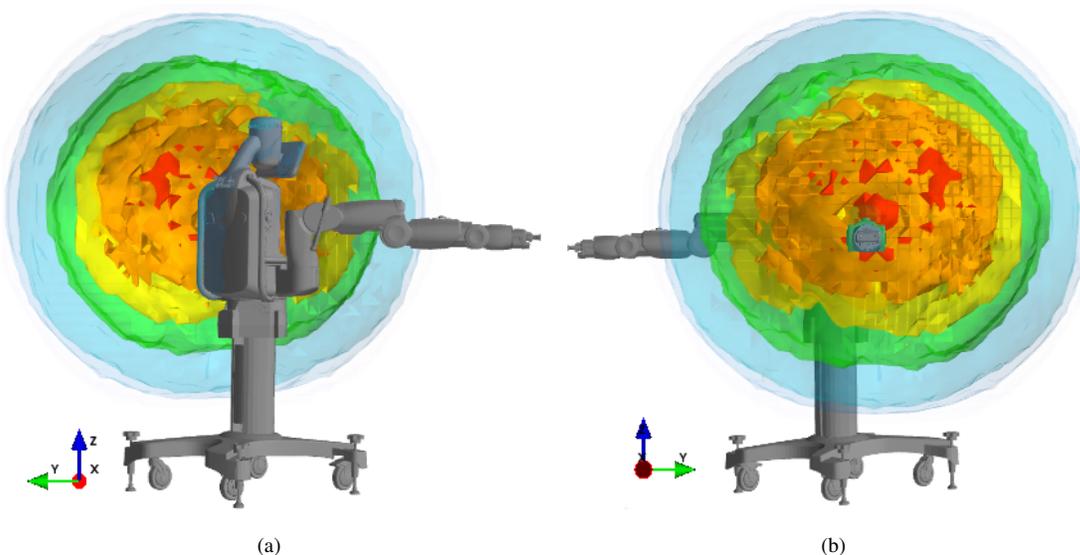


Figure 4. 6D Kinematic reachability space of Baxter’s left arm. The color density increases according to the number of possible configurations.

The reaching motion has been repeated 5000 times and only the joint velocity constraints have been considered. The computational time of the conventional method and the proposed one are reported in Table 1. The experiments were performed on Intel(R) Core(TM) i5-3470 CPU @ 3.20 GHz PC with 8 GB RAM.

Table 1. Computational complexity

Method	Average comput. time	Max comput. time	Min comput. time
Conventional method	542 μ s	621 μ s	485 μ s
Proposed method	583 μ s	638 μ s	512 μ s

From Table 1, we can observe that the computational time of the proposed method and of the conventional one are quite similar. However, it is worth to mention that we have only considered the velocity constraints because the conventional method cannot handle the jerk or the acceleration constraints.

8.4 Scenario 4: Smooth and Collision-free Motions

A simulated scenario of a Baxter research robot and an obstacle that consists of a sphere attached to a thin cylindrical rod is given in Fig. 5. The objective is to reach a goal pose from an initial pose while avoiding the collision with the obstacle.

As mentioned in Section 2, the collision avoidance problem can be formulated as kinematics inequality constraints [16], an overview of the formulation can be found in Appendix A.

In this scenario, the following constraints have been considered:

- (1) Collision avoidance: the Baxter robot is approximated by its collision geometry model from the Unified Robot Description Format (URDF) file, where the arms are approximated by cylinders and boxes. The collision avoidance is formulated as inequality constraints having the following form: $A\dot{q}_t \leq B$.
- (2) Joint limits of the robot joints, these values are given in the URDF file.

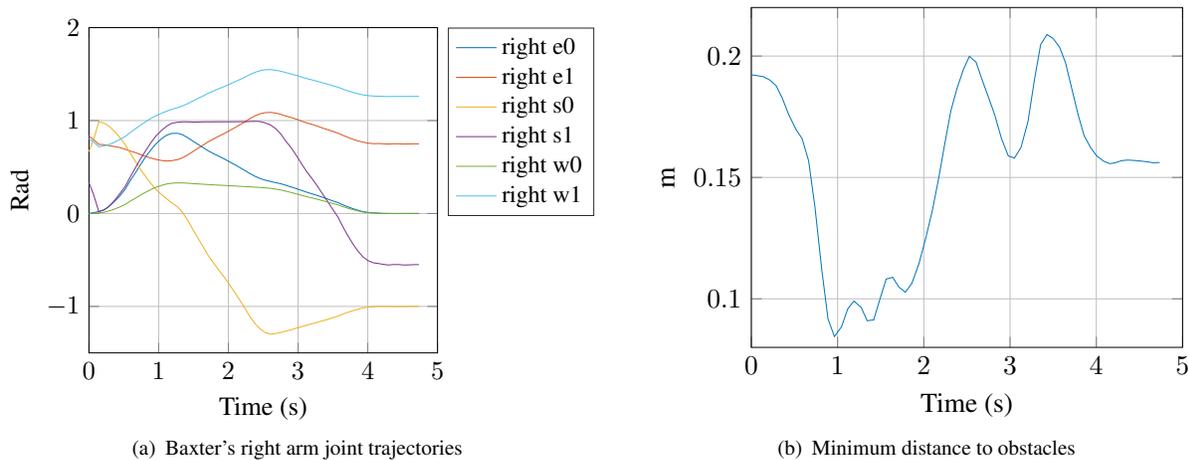
- (3) Velocity limits of the robot's joints, these values are also given in the URDF file.
- (4) Desired joint acceleration and jerk limits.

The end-effector tries to reach the goal pose at each iteration via a direct straight line, at the same time the collision avoidance inequality constraints repulse the arm to keep a minimum clearance to the obstacles.

Snapshots of the simulated motion are given in Fig. 5. The right arm's joint trajectories are shown in Fig. 6(a) and the minimum distance to the obstacles is given in Fig. 6(b).



Figure 5. Reaching scenario while avoiding collision



(a) Baxter's right arm joint trajectories

(b) Minimum distance to obstacles

Figure 6. Scenario 3

9. Conclusion and Future Work

In this paper, a new method for inverse kinematics while considering jerk limits has been proposed. The method is simple to implement, yet efficient to handle acceleration and jerk constraints. As discussed in the paper, several attempts have been proposed in the literature to integrate the acceleration limits into the conventional inverse kinematics algorithms, however they are not numerically efficient and yield often to ill-conditioned optimization problems. Moreover, the proposed algorithm guarantees that the accelerations of the joints are smooth by defining a threshold on the jerk.

Simulation results have been conducted and the efficiency of the algorithm has been pointed out. Future work will focus on the realtime implementation of the proposed algorithm and the validation on the real Baxter research robot.

Acknowledgment

This research is partially supported by a discovery grant (Prof. Wael Suleiman) from the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

- [1] Raghaven M, Roth B. Kinematic analysis of the 6r manipulator of general geometry. In: The Fifth International symposium on robotics research. 1990. p. 263–269.
- [2] Raghavan M, Roth B. Solving polynomial systems for the kinematic analysis and synthesis of mechanisms and robot manipulators. *Journal of Mechanical Design*. 1995 06;117(B):71–79.
- [3] Neppalli S, Csencsits MA, Jones BA, Walker ID. Closed-form inverse kinematics for continuum manipulators. *Advanced Robotics*. 2009;23(15):2077–2091.
- [4] Nakamura Y. *Advanced Robotics: Redundancy and Optimization*. Addison-Wesley. 1991.
- [5] Siciliano B, Sciavicco L, Villani L, Oriolo G. *Robotics: Modelling, planning and control*. Springer. 2009.
- [6] Whitney DE. The mathematics of coordinated control of prosthetic arms and manipulators. *Journal of Dynamic Systems, Measurement, and Control*. 1972 12;94(4):303–309.
- [7] Nakamura Y, Yamane K. Dynamics computation of structure-varying kinematic chains and its application to human figures. *IEEE Transactions on Robotics and Automation*. 2000 Apr;16(2):124–134.
- [8] Chan S, Lawrence P. General inverse kinematics with the error damped pseudoinverse. In: *IEEE International Conference on Robotics and Automation (ICRA)*. 1988 Apr. p. 834–839 vol.2.
- [9] Flacco F, De Luca A. Optimal redundancy resolution with task scaling under hard bounds in the robot joint space. In: *IEEE International Conference on Robotics and Automation (ICRA)*. 2013. p. 3969–3975.
- [10] Siciliano B. Kinematic control of redundant robot manipulators: A tutorial. *Journal of Intelligent and Robotic Systems*. 1990;3(3):201–212.
- [11] Kanehiro F, Lamiroux F, Kanoun O, Yoshida E, Laumond J-P. A Local Collision Avoidance Method for Non-strictly Convex Objects. In: *Proceedings of Robotics: Science and Systems*. Zurich, Switzerland. 2008 June.
- [12] Kanehiro F, Morisawa M, Suleiman W, Kaneko K, Yoshida E. Integrating geometric constraints into reactive leg motion generation. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2010. p. 4069–4076.
- [13] Kanoun O. Real-time prioritized kinematic control under inequality constraints for redundant manipulators. In: *Proceedings of Robotics: Science and Systems*. Los Angeles, CA, USA. 2011 June.
- [14] Escande A, Mansard N, Wieber PB. Fast resolution of hierarchized inverse kinematics with inequality constraints. In: *IEEE International Conference on Robotics and Automation (ICRA)*. 2010. p. 3733–3738.
- [15] Escande A, Mansard N, Wieber PB. Hierarchical quadratic programming: Fast online humanoid-robot motion generation. *The International Journal of Robotics Research*. 2014;33:1006–1028.
- [16] Faverjon B, Tournassoud P. A local based approach for path planning of manipulators with a high number of degrees of freedom. *IEEE International Conference on Robotics and Automation (ICRA)*. 1987. p. 1152–1159.

Appendix A. Collision avoidance as kinematics constraint

Let us consider the two strictly convex objects \mathcal{O}_1 and \mathcal{O}_2 in Fig. A1, where \mathcal{O}_2 is supposed to be fixed obstacle and \mathcal{O}_1 is a part of the robot. Let d be the distance between the pair of closest points p_1 and p_2 belonging to \mathcal{O}_1 and \mathcal{O}_2 respectively. Since \mathcal{O}_1 and \mathcal{O}_1 are strictly convex objects, d is \mathcal{C}^1 function.

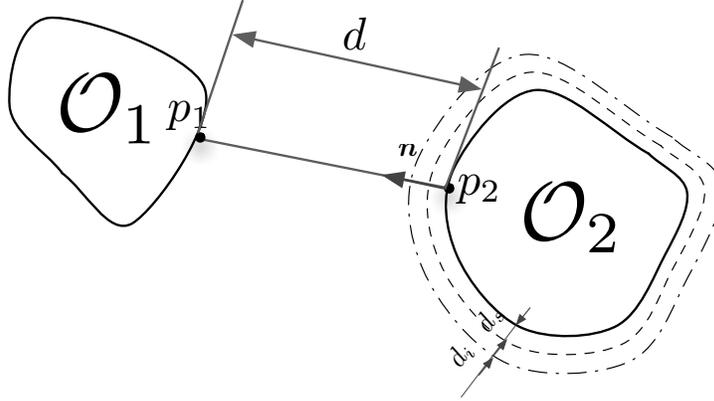


Figure A1. Collision avoidance as kinematics inequality constraints

In order to incorporate the collision avoidance as an inequality constraint, the derivative of distance with respect to time, \dot{d} , is bounded as follows:

$$-\dot{d} \leq \zeta \frac{d - d_s}{d_i - d_s}; \quad d_i > d_s \quad (\text{A1})$$

where ζ is a positive coefficient that regulates the convergence speed, and d_i and d_s are called respectively influence and security distances. As \mathcal{O}_2 is fixed, \dot{d} can be expressed as follows:

$$\dot{d} = \mathbf{n}^T \dot{p}_1 \quad (\text{A2})$$

where \mathbf{n} is a unit vector defined as: $\mathbf{n} = \frac{(p_1 - p_2)}{d}$.

As \mathcal{O}_1 is a part of the robot, \dot{p}_1 can be expressed as a function of the robot generalized coordinates, q , its derivatives, \dot{q} , and the jacobian matrix at the point p_1 , $J(q, p_1)$, as follows:

$$\dot{p}_1 = J(q, p_1) \dot{q} \quad (\text{A3})$$

By replacing A2 and A3 in A1, we obtain:

$$-\mathbf{n}^T J(q, p_1) \dot{q} \leq \zeta \frac{d - d_s}{d_i - d_s} \quad (\text{A4})$$

As it can be easily seen that (A4) is a linear inequality constraint with respect to the velocity vector \dot{q} . Moreover, the above method has been extended to the case of non-strictly convex objects [11].