Real-time prioritized kinematic control under inequality constraints for redundant manipulators

Oussama Kanoun
Department of Mechano-Informatics
The University of Tokyo, 7-3-1 Hongo Bunkyo-Ku, Tokyo 113-8656, Japan
Email: okanoun@ynl.t.u-tokyo.ac.jp

***Abstract*—This paper describes a fast algorithm for the prioritized kinematic control of redundant manipulators. Building on the classical prioritized task framework, the focus is set on efficient computation and handling of inequality constraints throughout priority levels. Classical approaches that tend to account for inequality constraints through potential fields are computationally competitive but have quality issues. Formulating the same control problems with a Quadratic Program (QP) removes these issues but is known to be costly. The following work revisits the formulation of a hierarchy of QPs for the prioritized control of redundant manipulators and proposes an algorithm that can meet real time requirements for current humanoid robots. Because lower control objectives often become infeasible, a particular point of focus is the numerical stability, hereby addressed with Tikhonov regularization. The method was tested in simulation for the control of the humanoid robot HRP-2.**

I. Introduction

The programmer of a humanoid robot must often specify at least three layers of controls: the first to ensure critical constraints, such as to maintain balance and avoid collision, the second to try and achieve some goal task, like reaching for an object, and the third to satisfy a secondary task of less importance, for instance to stay as close to a reference posture as possible.

The *task-priority* algorithm [14, 16, 18, 1] offers the means to calculate controls respecting a strict prioritization. The principle of this algorithm is to constrain the control for each layer within the set that does not induce perturbations for higher priority layers. For regulation tasks, the control for each layer is defined as the solution to a set of linear equations [19], thus the algorithm consists in a sequence of fast pseudo-inversions, each within the affine subspace defined by higher layers.

The problem with the classical task-priority framework is its inadequacy to handle inequality constraints, which arise from collision avoidance imperatives, limited operational range for actuators, etc. A workaround solution consists in transforming the inequality constraints into strict equality constraints through artificial potential fields [12]. Although computationally attractive, this was shown to have undesired effects on the control [8, 13].

It has been proposed in [11, 5] to solve the prioritized control problem with Quadratic Programs (QP), one per priority layer, thus leaving the constraints for the numerical solver to handle. Although the algorithm was shown to be effective, the potential of the QP was not fully used. At each priority level, the involved QP was in fact a linearly-constrained least squares problem. This meant that each new QP could be expressed with fewer parameters as the priority level decreased. An algorithm recently developed in [7] took advantage of this structure.

A good implementation of a prioritized control algorithm should be numerically stable, given that lower-priority objectives become infeasible. Numerical stability in this framework is dependent on the good conditioning of the task Jacobians. Addressing this problem is known as regularization. One way is to carry out the Singular Value Decomposition on the Jacobian and truncate the offending singular directions. However, this was shown in [4] to induce discontinuities on the control. The algorithm presented in [7] resorts to a Complete Orthogonal Decomposition, which suffer from the same issue. A second category of techniques address the problem by artificially boosting the singular values of the ill-conditioned matrix, ensuring an upper bound on the condition number and a smooth transition to it. Such techniques have been used in the context of kinematic control where they yielded the desirable regularity properties [4], for instance with the Tikhonov regularization [17] and SVD filtering [15].

The contribution of the paper is a new algorithm relying on a hierarchy of QPs to solve the prioritized kinematic control algorithm. The primary goal is a faster and more stable alternative to the algorithm by [7], that is reached through the use of regularized QR factorizations. The secondary goal is to show that using a stack of QPs for the kinematic control of a humanoid robot is an affordable upgrade from previous techniques, which is illustrated through simulations of kinematic control for HRP-2.

The first part of the paper revisits the classical algorithm excluding inequalities, rewrites it with QR factorizations and tests its performance. The algorithm is then extended to handle inequality constraints through an active set algorithm, as was suggested in [7].

II. The classical task-priority framework with QR factorization

This section gives a quick overview of the task priority framework and then proposes an efficient implementation using QR factorizations with Tikhonov regularization.

A. Background

Consider the problem of regulating kinematic tasks that are configuration-dependent. Such tasks are written as

$$f(q) = 0 \tag{1}$$

with $q \in \mathbb{R}^n$ being the n-dimensioned configuration vector of the robot and f an m-valued differentiable function of q. In a differential inverse kinematics scheme [19], $f(q)$ is regulated to the target value 0 through the differential equation

$$\frac{\partial f}{\partial q}\,\dot{q} = -\lambda_f f(q), \quad \lambda_f > 0. \tag{2}$$

Equation 2 can be rank-deficient but a solution control is always defined by the least squares formulation

$$\min_{\dot{q}} \frac{1}{2}\|\frac{\partial f}{\partial q}\,\dot{q} + \lambda_f f(q)\|^2. \tag{3}$$

The set of admissible controls is an affine subspace $\mathcal{F} \in \mathbb{R}^n$ whose dimension is greater than or equal to $n - m$. In the presence of multiple regulation tasks of equal importance, the corresponding objective functions are summed and minimized. Prioritizing the tasks requires their objective functions be minimized one after the other, each within the optimal set of the previous layers. For instance, regulating a function $g(q) \in \mathbb{R}^l$ to 0 in second priority defines the additional minimization problem

$$\min_{\dot{q}\in\mathcal{F}} \frac{1}{2}\|\frac{\partial g}{\partial q}\,\dot{q} + \lambda_g g(q)\|^2, \quad \lambda_g > 0. \tag{4}$$

Letting $A = \frac{\partial f}{\partial q} \in \mathbb{R}^{m\times n}$ and $b = -\lambda_f f(q)$, the particular solution of problem (3) having the minimal L_2 norm is given by the pseudo-inverse $A^\dagger$ of A,

$$\dot{q} = A^\dagger b. \tag{5}$$

The general solution is written

$$\dot{q} = A^\dagger b + (I - A^\dagger A)z, \quad z \in \mathbb{R}^n. \tag{6}$$

The pseudo-inverse is usually calculated with the singular value decomposition (SVD) of A (see for instance [4]). The operator $(I - A^\dagger A)$ is the orthogonal projector on the null space of A, hence any secondary control z does not interfere with the first control. This was observed and used in [14] for the control of a robot arm, with a first task to position the end effector and a secondary control deriving from an artificial potential field pushing the arm away from obstacles. The framework was extended by [16, 18, 1] to any number of priority levels.

The cost of the prioritization algorithm is that of the involved pseudo-inversions. The SVD being computationally expensive, the alternative expression of the pseudo-inverse in the under-determined case,

$$\dot{q} = A^T(AA^T)^{-1}b. \tag{7}$$

was often used instead. This is the fastest way to calculate the pseudo-inverse solution and also the least numerically stable. The expression is valid for a non singular A, therefore $\dot{q}$ grows unbounded when A approaches singularity. It was proposed in [17] to calculate an approximate control using the *damped least squares* solution

$$\dot{q} = A^T(AA^T + k^2 I)^{-1}b \equiv A^* b, \quad k \neq 0. \tag{8}$$

The added diagonal term $k^2 I$ ensures that $AA^T + k^2 I$ remains well conditioned near singularities of A although it introduces a systematic error [16]. The main problem of this approach is that the operator $(I - A^* A)$ is no longer an orthogonal projector on the null space of A [6].

Rather than falling back to the use of a SVD, the research by [7] suggested the use of a complete orthogonal decomposition identified as a cheaper alternative. The following sections describe an alternative formulation that makes use of regularized QR factorizations, motivated both by better numerical stability and lower cost.

B. Algorithm with QR factorizations

Consider again the linear system

$$A\dot{q} = b. \tag{9}$$

If A has full rank, there exists an orthogonal matrix $Q \in \mathbb{R}^{n\times n}$ and an upper triangular matrix $R \in \mathbb{R}^{m\times m}$ such that

$$A^T = Q \begin{bmatrix} R \\ 0 \end{bmatrix}. \tag{10}$$

The importance of the triangular form lies in the observation that the lower triangular system $\begin{bmatrix} R^T & 0 \end{bmatrix} y = b$ is easily solved in $y \in \mathbb{R}^n$ by forward substitution. Partitioning the matrix Q into $\begin{bmatrix} Y & Z \end{bmatrix}$ with $Y \in \mathbb{R}^{n\times m}$ and $Z \in \mathbb{R}^{n\times(n-m)}$ gives the orthonormal bases for the range space and the null space of A respectively. Therefore, the particular vector y^* where all components starting from rank $m+1$ are taken null gives the solution of minimum norm to (9),

$$\dot{q} = Qy^* = YR^{-T}b. \tag{11}$$

Several methods can be used to compute a QR factorization of A. An efficient one makes use of Householder transformations (see for instance [3]).

The algorithm for prioritization is as follows. Let

$$A_1\dot{q} = b_1 \tag{12}$$

be a linear system of size m_1 where A_1 is assumed to be of full rank. Obtain the orthogonal and upper triangular factors Q_1 and R_1 and partition Q_1 into $\begin{bmatrix} Y_1 & Z_1 \end{bmatrix}$ with $Y_1 \in \mathbb{R}^{n\times m_1}$ and $Z_1 \in \mathbb{R}^{n\times(n-m_1)}$. The minimum-norm solution to (12) is written

$$\dot{q}_1 = Y_1 R_1^{-T} b_1 \tag{13}$$

and the general solution is

$$\dot{q} = \dot{q}_1 + Z_1 z_1,\ z_1 \in \mathbb{R}^{n-m_1}. \tag{14}$$

The next layer of priority should be efficiently written with fewer parameters. If

$$A_2\dot{q} = b_2, \quad A_2 \in \mathbb{R}^{m_2\times n}, b_2 \in \mathbb{R}^{m_2} \tag{15}$$

is a secondary control objective, the control is defined as a solution to

$$\min_{z_1 \in \mathbb{R}^{n-m_1}} \frac{1}{2}\|A_2 Z_1 z_1 - (b_2 - A_2 \dot{q}_1)\|^2, \qquad (16)$$

obtained by substituting $\dot{q}$ in equation (15) with its expression in (14). Let $A_2' = A_2 Z_1$ and $b_2' = b_2 - A_2\dot{q}_1$, such that the problem is re-written

$$\min_{z_1} \frac{1}{2}\|A_2' z_1 - b_2'\|^2. \qquad (17)$$

Supposing again that A_2' has full rank, factorize ${A_2'}^T$ to obtain

$${A_2'}^T = Q_2 \begin{bmatrix} R_2 \\ 0 \end{bmatrix}, \qquad (18)$$

where Q_2 is $(n-m_1) \times (n-m_1)$ orthogonal and R_2 is $m_2 \times m_2$ upper triangular. As previously, it is convenient to define a partition $\begin{bmatrix} Y_2 & Z_2 \end{bmatrix}$ of Q_2 such that the general solution for z_1 may be written

$$z_1 = Y_2 R_2^{-T} b_2' + Z_2 z_2, \quad z_2 \in \mathbb{R}^{n-m_1-m_2}. \qquad (19)$$

The same steps can be followed to define the solutions of all stages.

C. Robustness to algorithmic singularities

Prioritizing control objectives will often make lower priority tasks infeasible. In the above development, if the secondary objective becomes infeasible, the equation $A_2' z_1 = b_2'$ becomes rank-deficient and this phenomenon is known as an occurrence of *algorithmic singularity* [2]. The QR factorization of ${A_2'}^T$ would give an ill-conditioned triangular part R_2 whose inversion would yield a large displacement z_1 in the null space of A_1'.

Several ways exist to deal with algorithmic singularities. A solution consists in truncating the SVD of the ill-conditioned matrix according to a desired maximum condition number, but this appears to generate an undesirable discontinuity in the control [4]. A Complete Orthogonal Decomposition, by definition of the algorithm, shares the same problem.

The solutions that rely on smooth regularization, such as the damped least-squares solution (8), are free of this issue. A similar property is desired here.

Considering again the linear system $A\dot{q} = b$ representing equation (2), the standard problem

$$\min_{\dot{q}} \frac{1}{2}\|A\dot{q} - b\|^2 \qquad (20)$$

is replaced with the regularized alternative

$$\min_{u \in \mathbb{R}^{m+n}} \frac{1}{2}\left\| \begin{pmatrix} A & k^2 I \end{pmatrix} u - b \right\|^2, \qquad (21)$$

with $k \neq 0$ and I denoting the identity matrix of rank m. In the present context of kinematic control, appending the diagonal matrix $k^2 I$ to the Jacobian A can be thought of as the addition of m virtual degrees of freedom, each useful for a unique row of A. The parameter vector u in problem (21) is the vector $[\dot{q}^T w^T]^T$ where $w \in \mathbb{R}^m$ captures the contribution of the m virtual degrees of freedom to the solution. The vector w can be used to measure the error introduced by the regularization, which is $\sum_1^m k^2 w_i$.

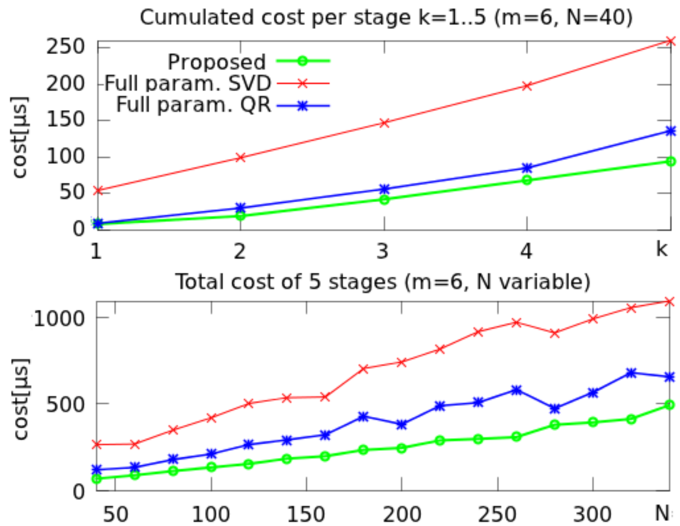

Fig. 1. Upper window: for a small number of parameters ($N = 40$), the implementations do not show significant performance difference between a constantly full-sized QR algorithm and the proposed algorithm. Lower window: the gain in computation time is more noticeable as N takes higher values.

The proposed solution has a problem at this point: the factorization of $\begin{pmatrix} A & k^2 I \end{pmatrix}^T$ does not yield a basis of the null space of A, therefore the results of the factorization cannot be used for the subsequent priority levels as shown through (12)-(19). This is essentially the same issue as what was discussed for formula (8).

A simple workaround to this problem is to perform the factorization of A^T as well. This is based on the observation that factorizing A^T still gives the orthonormal basis needed for the subsequent priority levels, the ill-conditioned triangular factor being simply discarded. Factorizing both A^T and its regularized counterpart seems inefficient but these factorizations can be made in parallel, such that the effective cost remains the same.

D. Performance comparison

The algorithm is compared in theoretical cost with a Complete Orthogonal Decomposition(COD)-based approach and in practice to a SVD-based approach.

In the algorithm [7] which was the first to reduce the problem size with every priority stage, the use of CODs was demonstrated to be more efficient than carrying SVDs on constantly full-sized priority levels. For a matrix $A \in \mathbb{R}^{m \times n}$ with $m \leq n$, the reported COD cost was $6nm^2 - m^3$, n decreasing with each level. A Householder QR factorization of A^T, used for the present algorithm, costs $2nm^2 - \frac{2}{3}m^3$ [3], with n also decreasing in the same manner. Even with a naive factorization of the regularized matrix $\begin{pmatrix} A & k^2 I \end{pmatrix}^T$ (i.e. without taking advantage of the sparsity of the regularizing block), the cost actually increases to $2nm^2 + \frac{4}{3}m^3$ and remains lower than that of a COD.

The algorithm was tested against a classical SVD-based algorithm and a simpler QR-based algorithm that solves each stage in full coordinates. The testing platform was a notebook computer with the Intel 540M 2.52Ghz processor. The results are reported in Figure 1.

With its equality-only version covered, the algorithm is extended in the next section to handle inequality constraints.

III. Priority under inequality constraints

This section revisits the previous algorithm to account for linear inequality constraints at each priority level. First, the algorithm is tied to an existing formulation.

A. Background

A formulation of a hierarchy of inequality-constrained least-squares problems was shown in [11] to allow prioritization of inequalities themselves. The present work considers inequalities in the constraints only. Note that that the algorithm described in [7] handles the general case.

Let $\mathcal{I}$ be a set of n_i linear inequality constraints acting on the control $\dot{q}$. Denote the inequalities by

$$c_i^T \dot{q} \leq d_i \tag{22}$$

with $c_i \in \mathbb{R}^n$, $d_i \in \mathbb{R}$ for $1 \leq i \leq n_i$. Consider the general case where initial equality constraints must also be satisfied for all control objectives and let m_0 such constraints be denoted

$$A_0 \dot{q} = b_0, \qquad A_0 \in \mathbb{R}^{m_0 \times n}, b_0 \in \mathbb{R}^{m_0}. \tag{23}$$

The first control objective $A_1 \dot{q} = b_1$ is solved through the minimization

$$\min_{\dot{q}} \ \frac{1}{2} \|A_1 \dot{q} - b_1\|^2 \tag{24}$$

$$\text{s.t. } A_0 \dot{q} = b_0 \tag{25}$$

$$c_i^T \dot{q} \leq d_i \quad \forall i. \tag{26}$$

According to the algorithm described in [11], a solution $\dot{q}_1^*$ to problem (24) is used to define the minimization of the next priority layer as

$$\min_{\dot{q}} \ \frac{1}{2} \|A_2 \dot{q} - b_2\|^2 \tag{27}$$

$$\text{s.t. } c_i^T \dot{q} \leq d_i \quad \forall i \tag{28}$$

$$A_0 \dot{q} = b_0 \tag{29}$$

$$A_1 \dot{q} = A_1 \dot{q}_1^*, \tag{30}$$

which says that the solution control for the secondary objective is within the set optimizing the first.

Although sufficient, the description of the new set of constraints may be too loose. A more efficient description should follow from examining the Lagrange multipliers at the optimal point $\dot{q}_1^*$. The multipliers which are canceled at point $\dot{q}_1^*$ correspond to the inequality constraints that should be forwarded to the following stage. The remaining inequality constraints, if any, are said *saturated* and would indicate that the optimal solution $\dot{q}_1^*$ does not satisfy the equation $A_1 \dot{q} = b_1$. Therefore, following the same analysis in [11], the saturated constraints should be transformed into equalities for all following priority layers, saving the work of re-verifying them.

Consequently, by regrouping the saturated inequality constraints in a subset $\mathcal{I}_s \subset \mathcal{I}$, the secondary minimization (27) is re-written

$$\min_{\dot{q}} \ \frac{1}{2} \|A_2 \dot{q} - b_2\|^2 \tag{31}$$

$$\text{s.t. } c_i^T \dot{q} \leq d_i \ \ i \notin \mathcal{I}_s \tag{32}$$

$$c_j^T \dot{q} = d_j \ \ j \in \mathcal{I}_s \tag{33}$$

$$A_0 \dot{q} = b_0 \tag{34}$$

$$A_1 \dot{q} = A_1 \dot{q}_1^*. \tag{35}$$

The third priority layer would have an additional equality constraint $A_2 \dot{q} = A_2 \dot{q}_2^*$ and possibly a larger set $\mathcal{I}_s$. The process is similar for the rest of the layers but solving each of these problems in $\dot{q}$ is not efficient.

B. Algorithm with QR factorizations

The aim of this section is to describe the resolution of the above sequence of problems with QR factorizations and a standard active set algorithm. Björck [3] previously described a similar method for the particular case where the problem has a single over-constrained objective. The algorithm given here has the same inner workings but is written for the case of prioritized under-determined linear equations.

The algorithm must start from a feasible control $\dot{q}_0$ strictly satisfying all constraints (by opposition to control objectives). To find a feasible point, first the equalities $A_0 \dot{q} = b_0$ are solved using a QR factorization of A_0^T as shown in (12) and (13). The result is then taken inside the convex polytope defined by the inequality constraints. This can be done in several ways, see for instance [9].

The factorization of A_0^T gives the upper triangular matrix R_0 and the orthogonal matrix $Q_0 = [Y_0 \ Z_0]$ as seen before. The optimization of the first priority stage can then be written

$$\min_{z_0 \in \mathbb{R}^{dim(Z_0)}} \ \frac{1}{2} \|A_1 Z_0 z_0 - (b_1 - A_1 \dot{q}_0)\|^2 \tag{36}$$

$$\text{s.t. } c_j^T \dot{q} \leq d_j, \quad j \in \mathcal{I}. \tag{37}$$

The general principle of an active set algorithm is to step towards the optimum while *sticking* to the boundary of the admissible region when encountered. The first step z_0 is found by solving the unconstrained version of the problem (36), written

$$\min_{z_0 \in \mathbb{R}^{dim(Z_0)}} \ \frac{1}{2} \|A_1 Z_0 z_0 - (b_1 - A_1 \dot{q}_0)\|^2. \tag{38}$$

The displacement z_0 may lead the control $\dot{q} = \dot{q}_0 + Z_0 z_0$ out of the admissible region, that is, the convex polytope defined by all inequality constraints. In that case, the control is trimmed back to the boundary and the remainder of the algorithm computes successive displacements on the boundary until a point achieving the global minimum is reached. Moving on the boundary means that some inequality constraints are added to or removed from the problem as equality constraints. The addition or the removal of a constraint changes the null space where the unconstrained optimization (38) is conducted and this change is reflected in the null space basis Z_0. An important feature of the QR decomposition appears in the fact that the factorization of the augmented matrix $\left[A_0^T \ c_i\right]$ can be obtained from that of A_0^T without added cost with respect to factorizing $\left[A_0^T \ c_i\right]$. Also, updating the QR factorization to discard a constraint may be conducted more efficiently than recomputing from scratch, depending on the rank of the constraint to remove. Details of these techniques can be found in [3].

When the active set algorithm terminates, the outcomes of interest are (a) an optimal point $\dot{q}_1$, (b) the set $\mathcal{I}_1$ referencing the saturated inequality constraints and (c) the QR factorization of $\left[\ A_0^T \ C_1 \ A_1^T\ \right]$ where C_1 is defined as a matrix regrouping the column vectors c_i whose indices are in $\mathcal{I}_1$. Call Q_1 the orthogonal matrix in this factorization and separate Q_1 into Y_1 and Z_1 as done previously. Then, the general solution of the first control objective is written

$$\dot{q} = \ \dot{q}_1 + Z_1 z_1 \text{ such that } c_j^T \dot{q} \le d_j, \ j \notin \mathcal{I}_1 \tag{39}$$

with z_1 any vector in $\mathbb{R}^{dim(Z_1)}$. This gives the efficient expression of the secondary optimization problem

$$\min_{z_1 \in \mathbb{R}^{dim(Z_1)}} \ \frac{1}{2}\|A_2 Z_1 z_1 - (b_2 - A_2 \dot{q}_1)\|^2 \tag{40}$$

$$\text{s.t. } c_j^T \dot{q} \le d_j, \ j \notin \mathcal{I}_1. \tag{41}$$

which has the same shape as (36).

The third stage is detailed for the generalization. Call $\dot{q}_2$ an optimal control from solving (40) and $\mathcal{I}_2$ the set of new inequality constraints saturated in the second stage at point $\dot{q}_2$. As previously, define $C_2 = [\{c_i\}_{i \in \mathcal{I}_2}]$ the matrix regrouping the inequality constraints in $\mathcal{I}_2$. Recalling that the QR factorization (Q_2, R_2) of $\left[\ A_0^T \ C_1 \ A_1^T \ C_2 \ A_2^T\ \right]$ follows from the active set algorithm, partition Q_2 into Y_2 and Z_2 to write the third stage

$$\min_{z_2 \in \mathbb{R}^{dim(Z_2)}} \ \frac{1}{2}\|A_3 Z_2 z_2 - (b_3 - A_3 \dot{q}_2)\|^2 \tag{42}$$

$$\text{s.t. } c_j^T \dot{q} \le d_j, \ j \notin \{\mathcal{I}_1 \cup \mathcal{I}_2\}. \tag{43}$$

The problem at rank p is defined and solved in the same manner:

$$\min_{z_{p-1}} \frac{1}{2}\|A_p Z_{p-1} z_{p-1} - (b_p - A_p \dot{q}_{p-1})\|^2 \tag{44}$$

$$\text{s.t. } c_j^T \dot{q} \le d_j, \ j \notin \bigcup_{t=1}^{p} \mathcal{I}_t. \tag{45}$$

This algorithm was implemented using the Householder QR factorizations. It was tested in simulation for the kinematic control of a humanoid robot and the results are reported in the following section.

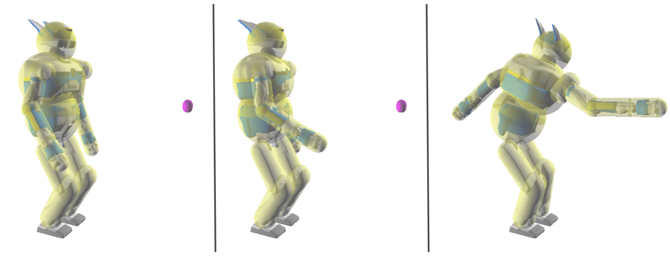

Fig. 2. Basic control scenario:8 equality constraints, 72 inequality constraints, the first task layer has 3 equalities. The yellow cylinders represent the simplified model of the geometry useful for efficient avoidance of collision.

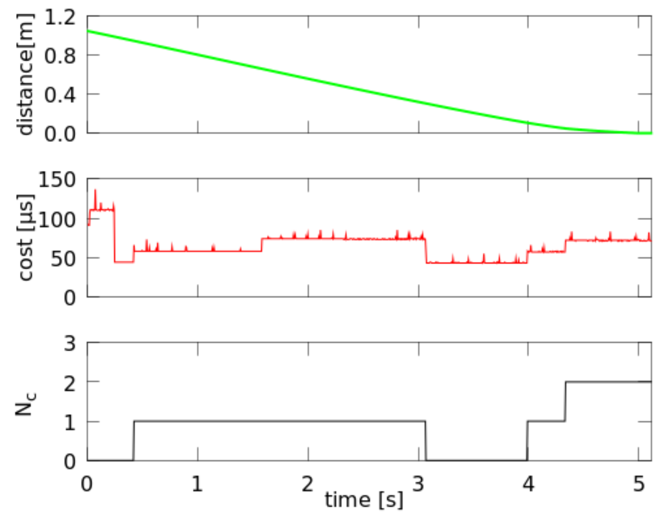

Fig. 3. Simulation results for the basic control scenario. From top to bottom: distance of hand to target, calculation time per iteration, number of inequality constraints activated per iteration. The calculation time accounts for the preparation of the linear systems.

IV. Simulation Results

This section describes three test scenarios for the control of the humanoid robot HRP-2 with increasing difficulty.

A. Setting

The humanoid robot, HRP-2, is modeled by a 28-degree-of-freedom kinematic tree: 6 in each limb, 2 in the trunk and 2 in the neck. The robot has additional degrees of freedom for opening/closing its hands but they are irrelevant to the present tests and not accounted for in the computations.

B. Basic control

First, the model of HRP-2 is programmed to reach for a static target point while standing on two feet and maintaining balance (Figure 2). The algorithm is required to solve a single control objective (or task) under equality and inequality constraints. The task is written

$$p(q) = p_t. \tag{46}$$

$p(.)$ is the Cartesian position of a point belonging to the reaching hand, referenced in a fixed global frame, and p_t is the position of the target point. Following equation (2), the regulation equation for this task is written

$$\frac{\partial p}{\partial q}\,\dot{q} = -\lambda(p(q) - p_t), \quad \lambda > 0. \tag{47}$$

The gain λ is taken such that $\|\lambda(p(q) - p_t)\|_2$, which is homogeneous to a linear velocity, is bounded above by $0.25\mathrm{m/s}$.

The first equality constraints maintain the balance of the robot for quasi-static motion. The motion of the COM is authorized vertically above the support polygon. The regulation equations are derived as above, but the gain λ is set at the high value of $1\mathrm{s}^{-1}$. For the computation of all jacobians, a foot is conveniently considered as the base link for the kinematic model, such that only one foot is constrained. There is a total of 8 equality constraints for this scenario.

A first set of inequality constraints is derived from the joint limits. As for equalities, a function $f(.)$ of the configuration can be regulated to satisfy an inequality $f(q) \leq 0$ following the linear differential inequality

$$\frac{\partial f}{\partial q}\,\dot{q} \leq -\lambda_f f(q) \tag{48}$$

where λ_f is a positive number scaling the feedback. The joint limits are expressed as

$$q_{min} \preceq q \preceq q_{max}. \tag{49}$$

Applying the regulation (48) for upper limits gives the velocity bounds

$$\dot{q} \preceq -\lambda(q - q_{max}) \tag{50}$$

and the lower joint limits are obtained similarly, for a total of 56 inequality constraints. The gain λ is taken equal to $0.5\mathrm{s}^{-1}$. In a simple Euler integration scheme, this means that a joint angle update cannot exceed half the gap separating its current value from its bounds. For a control on the real robot, the bounds are usually more conservative to satisfy the limits on the velocities and accelerations. This will be used in the last scenario to stress-test the implementation.

The second set of inequality constraints is for self collision avoidance. The geometrical model of HRP-2 is wrapped link by link in sphere-capped cylinders. Calling d the distance between the cores of two cylinders of radii $r1$ and $r2$ and letting $d_{min} = r_1 + r_2$ be the minimal distance required between the cylinders, the constraint is expressed

$$d \geq d_{min} \tag{51}$$

and regulated by

$$-\dot{d} \leq -\lambda(d_{min} - d). \tag{52}$$

Consider a pair of cylinders checked for collision, called C_{R1} and C_{R2}. Call P_1 the point on C_{R1} that is closest to C_{R2} and let P_2 be its counterpart on C_{R2}. Define the mutual normal vector $\vec{n} = \overrightarrow{P_1P_2}/\|\overrightarrow{P_1P_2}\|$. As previously shown in [8], the inequality (52) expands to

$$-\langle J_2 - J_1 | \vec{n} \rangle \dot{q} \leq -\lambda(d_{min} - d), \tag{53}$$

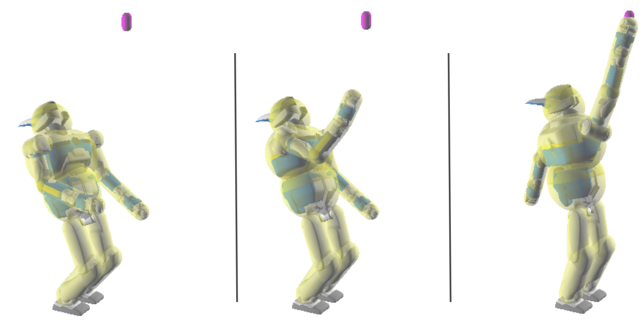

Fig. 4. Prioritized control scenario. The task of of hand positioning is prioritized over keeping the vision field centered at the target.

with J_1 and J_2 being the jacobians of the positions of points P_1 and P_2 respectively. 16 such constraints are set in the solver to avoid self collision.

The results of the simulation are reported in Figure 3. The fast computation times are due to the rare activations of inequality constraints.

C. Prioritized control

In this scenario where the effect of strict priority is shown, the robot has an additional task layer to keep the gaze on the target point. The target point is positioned high over the robot as shown in Figure 4). This constraints and the type of the first task are identical to those seen in the first scenario. The vision field regulation task is defined as

$$\overrightarrow{ov} \times \overrightarrow{op_t} = \vec{0} \qquad \text{for } \overrightarrow{ov}.\overrightarrow{op_t} \geq 0 \tag{54}$$

where o is a point on the optical axis and $\overrightarrow{ov}$ is a vector lying on the optical axis ahead of o. The regulation equation is derived as

$$[\overrightarrow{op_t} \times (\overrightarrow{ov} \times J_w) - \overrightarrow{ov} \times J_o]\dot{q} = -\lambda(\overrightarrow{ov} \times \overrightarrow{op_t}) \tag{55}$$

where J_w and J_o respectively stand for the orientation Jacobian and position Jacobian at point o (this is a point attached to the head). The scaling factor λ is chosen to bound $\|\lambda_2(\overrightarrow{ov} \times \overrightarrow{op_t})\|$ above by $2.10^{-4}\mathrm{m}^2/\mathrm{s}$.

The results as reported in Figure 5.

D. A stress test

The final scenario stresses the algorithm by strongly constraining the velocity limits and adding external collision avoidance constraints. The Figure 6 shows the setup where a few square-shaped obstacles are placed in front of HRP-2 to further constrain the motion of the hand towards its target. The inner width of the smallest of the squares is only $1mm$ larger than the diameter of the cylinders surrounding the hands and the forearms. This is a typical case of narrow passage where an approach based on potential fields would fail (see [13] for details).

The first and second control objectives are similar to those in the previous scenario. The regulation inequalities due to the collision avoidance constraints are identical to (53) except for the fact that in every pair of cylinders checked, one is not

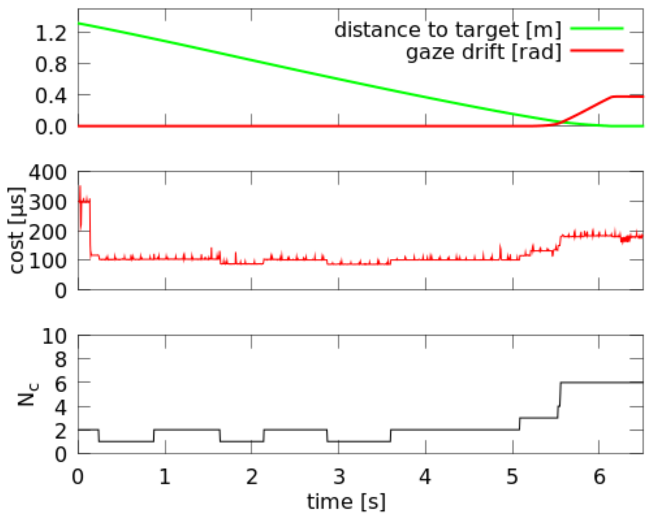

Fig. 5. Prioritized control scenario: 8 equality constraints, 72 inequality constraints, 3 equalities for the first task layer and 3 more for the second. The gaze drifts at the end of the motion letting the higher hand positioning task be satisfied.

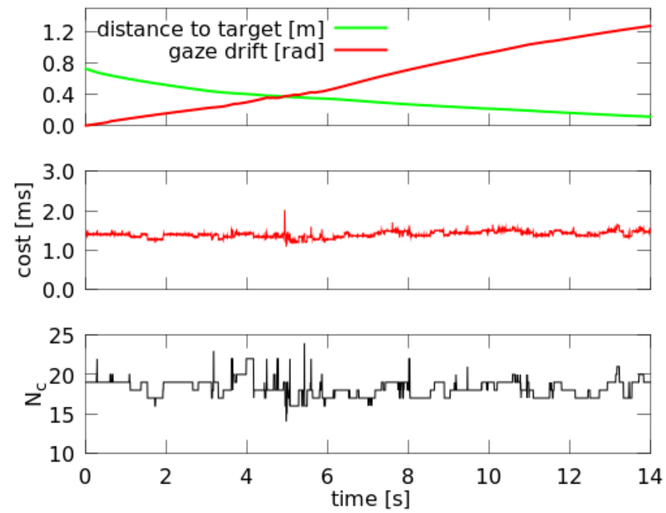

Fig. 7. Stress test scenario: 8 equality constraints, 160 inequality constraints, the first task layer has 3 equalities and the second layer has also 3. The joint velocities have small bounds while the tasks are set with large steps, such that the active set algorithm is tested for the worst case.

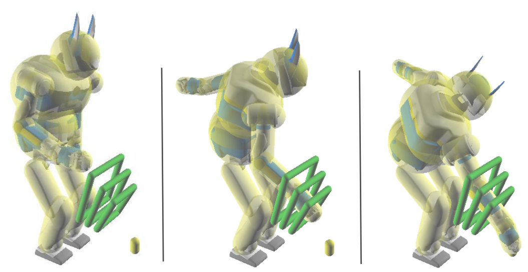

Fig. 6. Stress test scenario. The robot must reach for an object through some obstacles while its joint velocities are significantly reduced.

moving. The velocity limits are significantly shrunk by setting the maximum absolute velocity to 3.10^{-4}rad/s for each joint parameter. Moreover, the regulations of the first and second tasks are set such that the authorized steps were 0.1m/s and $0.1\text{m}^2/\text{s}$ respectively. These values are guaranteed to make the objectives for each instance of the algorithm infeasible, pushing the control $\dot{q}$ to the edges of the polytope defined by the inequality constraints, thus activating many of them.

The results are as reported in Figure 7. It should be noted that the time spent in the calculation of the involved linear systems was below $10\mu s$ for every iteration, thus non significant given the results. As expected, the active set algorithm spends a lot of time stepping on the boundary of the admissible control region. The calculation times can be improved in the future by including techniques for efficient downdate of QR factorizations, not implemented at the time of testing. Nonetheless, this heavily-constrained scenario shows that the algorithm is suitable for real time control on HRP-2 at the comfortable rate of 200Hz.

V. Conclusion and Future Works

This paper presented a fast algorithm for the prioritized kinematic control of redundant manipulators. The method followed the choice of enforcing inequality constraints on the manipulator by means of inequality-constrained optimization. The obtained results demonstrate that the method is effective for use in real time kinematic control. The algorithm relied on standard QR factorizations and an active set method to enforce linear inequality constraints at every priority stage. A smooth regularization scheme was adopted to make the proposed algorithm numerically stable.

In future works, the integration with general purpose search-based algorithms may be considered, such as to evaluate the efficiency that a more powerful inverse kinematics algorithm may add in solving motion planning problems, especially for manipulation objectives involving narrow passages.

Acknowledgments

This work was supported by the JSPS Postdoctoral Fellowship for Foreign Researchers. The author would like to thank the reviewers of this paper for their helpful comments. The SVD and Householder QR factorization algorithms used for the tests were readily available from the efficient Eigen[10] software.

References

[1] P. Baerlocher and R. Boulic. Task-priority formulations for the kinematic control of highlyredundant articulated structures. In *IEEE-RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 323–329, 1998.

[2] J. Baillieul. Kinematic programming alternatives for redundant manipulators. In *IEEE International Conference on Robotics and Automation*, pages 722–728, St. Louis, MI, 1985.

[3] Å. Björck. *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia, 1996.

[4] S. Chiaverini. Singularity-Robust Task-Priority Redundancy Resolution for Real-Time Kinematic Control of Robot Manipulators. *IEEE Transactions on Robotics and Automation*, 13(3), 1997.

[5] M. de Lasa, I. Mordatch, and A. Hertzmann. Feature-based locomotion controllers. In *ACM SIGGRAPH*, pages 1–10, 2010.

[6] A.S. Deo and I.D. Walker. Robot Subtask Performance with Singularity Robustness using Optimal Damped Least-Squares. In *IEEE International Conference on Robotics and Automation*, pages 434–441, Nice, France, 1992.

[7] A. Escande, N. Mansard, and P.B. Wieber. Fast resolution of hierarchized inverse kinematics with inequality constraints. In *IEEE International Conference on Robotics and Automation*, pages 3733–3738, 2010.

[8] B. Faverjon and P. Tournassoud. A local based approach for path planning of manipulators with a high number of degrees of freedom. In *IEEE International Conference on Robotics and Automation*, volume 4, pages 1152–1159, 1987.

[9] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, New York, second edition, 1987.

[10] Gaël Guennebaud, Benoît Jacob, et al. Eigen v3. http://eigen.tuxfamily.org, 2010.

[11] O. Kanoun, F. Lamiraux, P.B. Wieber, F. Kanehiro, E. Yoshida, and J.P. Laumond. Prioritizing linear equality and inequality systems: application to local motion planning for redundant robots. In *IEEE International Conference on Robotics and Automation*, pages 2939–2944, 2009.

[12] O. Khatib. The potential field approach and operational space formulation in robot control. *Adaptive and Learning Systems: Theory and Applications*, pages 367–377, 1986.

[13] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. In *IEEE International Conference on Robotics and Automation*, pages 1398–1404, 1991.

[14] A. Liégeois. Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms. *IEEE Transactions on Systems, Man, and Cybernetics*, 7 (12):868–871, 1977.

[15] A.A. Maciejewski and C.A. Klein. Numerical filtering for the operation of robotic manipulators through kinematically singular configurations. *Journal of Robotic Systems*, 5(6):527552, 1988.

[16] Y. Nakamura. *Advanced robotics: redundancy and optimization*. Addison-Wesley Longman Publishing, 1990.

[17] Y. Nakamura and H. Hanafusa. Inverse kinematic solutions with singularity robustness for robot manipulator control. *Journal of dynamic systems, measurement, and control*, 108(3):163–171, 1986.

[18] B. Siciliano and J.J.E. Slotine. A general framework for managing multiple tasks in highly redundantrobotic systems. In *International Conference on Advanced Robotics*, pages 1211–1216, 1991.

[19] D. E. Whitney. Resolved motion rate control of manipulators and human prostheses. *IEEE Transactions on Man-Machine Systems*, 10:47–53, 1969.