Smart changes of the end-effector orientation for the automatic handling of singular configurations

Fabio Ghilardelli, Corrado Guarino Lo Bianco*, *Member, IEEE,* and Marco Locatelli

***Abstract*—Trajectories in the operational space could easily cause feasibility issues when they incur in kinematic singularities. In proximity of singular configurations, as known, joint velocities and accelerations could indeed assume unfeasible values, thus worsening the tracking performances of controllers: In case of strong violations of the assigned bounds, manipulators could even be stopped in order to prevent control losses. Problems are especially evident when trajectories are online defined for non-repetitive tasks. This paper proposes an algorithm for the real-time handling of wrist singularities for non-redundant manipulators. The proposed method makes it possible to execute otherwise unfeasible trajectories by simultaneously preserving an accurate path tracking, by guaranteeing the fulfillment of the assigned longitudinal time law and by fulfilling a given set of bounds on the joint velocities and accelerations. This result is achieved by acting on the orientation of the end-effector: Only minor changes are admitted in order to minimize their impact on the process quality. The proposed method has been verified by means of simulations and real experiments. Comparisons have been made with the performances of an analogous commercial system.**

***Index Terms*—Kinematic singularities, kinematics constraints, real-time trajectory planning.**

I. Introduction

Possible problems deriving from the physical limits of electromechanical systems can be mitigated by means of proper reference signals which keep those systems far from unfeasible configurations. In robotic contexts, undesired behaviors typically arise because of the velocity and the acceleration limits of the electric actuators. Problems become especially evident in case of trajectories planned in the operational space since, as known, even slow Cartesian motions could require unfeasible velocities and accelerations in the neighborhoods of singular configurations.

Early works on this topic date back to the mid '90s and were basically focused on redundant manipulators. Critical configurations were avoided by properly acting on the null space of the Jacobian matrix and by exploiting the redundant Degrees of Freedom (DOF) [1], [2] in order to find feasible solutions that were able to preserve the end-effector path.

For non-redundant manipulators the problem is even more complex. It can be managed by designing systems which singularities are located outside the workspace [3]–[5], or by planning trajectories which do not cross singularities [6], [7], or by admitting that planned trajectories could be modified, for example, by altering the path geometry in order to skip the singular points. To this purpose offline and online approaches have both been proposed [8]–[12]. Unfortunately, in many industrial applications modifications of the Cartesian path are not allowed, so that the problem solution is typically based on one of the following two alternative methods: The velocity of the end-effector is properly slowed down in order to preserve its Cartesian path and orientation or, alternatively, small changes of the tool orientation are admitted in order to maintain the assigned execution time and the Cartesian path.

The scientific literature proposes many works which handle the problem through the first approach. Proposed solutions concern both trajectories in the configuration space [13]–[19] or trajectories in the operational space [20]–[23].

On the contrary, strategies based on the second approach are seldom considered, despite there are many industrial processes which require preserving both paths and time-laws. Let us mention, for example, application fields like automatic painting [24]–[27], gluing, or welding [28], in which changes of the tool velocity or of the Cartesian path would lead to quality losses, while small changes of the tool orientation ($\pm$20 deg) can be admitted [11], [12], [29]. For the above mentioned applications, the problem is typically handled by means of offline constrained optimization algorithms, but in case of non-repetitive tasks, appropriate online techniques need to be developed. As previously recalled, only a few online algorithms [30]–[32] have been proposed in the literature to this purpose (actually, the iTaSC algorithm proposed in [31] does not specifically address the singularity management, but it could be adapted to that target). They are based on different strategies but, due to their computational burden or to the adopted design choices, they are all suited to slow motions, like those that can be obtained through the use of manual teaching-devices.

The Singularity Avoidance System (SAS) proposed in this paper is conceived to handle in real time the wrist singularities which represent, as known, the main cause of problems in case of trajectories in the operational space for 6R anthropomorphic, non-redundant manipulators. Like in [30]–[32], this result is achieved by admitting minor changes of the tool-frame orientation while accurately preserving the assigned Cartesian path and time-law but, differently from those approaches, which are specifically suited for manually operated manipulators, the novel system is conceived to operate at normal working speeds, i.e., it can be used in real industrial contexts in order to preserve the robot productivity even in presence of singularities. This result is possible by adopting a totally different strategy: The system core is indeed represented by an array of nonlinear filters, which permits reduced evaluation

*Corresponding author

C. Guarino Lo Bianco and M. Locatelli are with the Dipartimento di Ingegneria dell'Informazione, University of Parma, Italy, email:{guarino,locatell}@ce.unipr.it

F. Ghilardelli is with COMAU S.p.a (FCA Group), Torino, Italy, email:fabio.ghilardelli@comau.com

This work was supported in part by MIUR (Ministero dell'Istruzione, dell'Università e della Ricerca) in the framework of a PRIN2010-11 project (MARIS - Marine Autonomous Interventions)

times.

The paper is structured as follows. The problem statement and the proposed solution are formulated in Section II. Sections III, IV, and V describe each single block which composes the SAS. Simulation results are reported and commented in Section VI, while Section VII is totally devoted to the experimental tests executed on a Comau Smart SiX 6-1.4 manipulator. Finally, the conclusions are drawn in Section VIII.

NOTATION

The following notation is assumed along the paper. ${}^{C}\mathbf{p}_{A,B}$ is the vector which individuates the position of point A w.r.t. point B described w.r.t. frame C. A compact notation is used any time a vector is described w.r.t. an inertial frame (typically represented by the manipulator frame 0) so that $\mathbf{p}_A$ is the position of A w.r.t. 0 described w.r.t. frame 0. An analogous notation is used for velocities and accelerations.

II. PROBLEM DEFINITION AND PROPOSED SOLUTION

Trajectories in the operational space are typically specified by defining proper time functions which describe position and orientation of tool frame T. For example, they can be expressed through the following signals described w.r.t. an inertial frame

${}^{0}_{T}\mathbf{R}(t)$	Rotation matrix: Orientation of frame T;
$\mathbf{p}_T(t)$	Position of frame T;
$\boldsymbol{\omega}_T(t)$	Angular velocity of frame T;
$\mathbf{v}_T(t)$	Linear velocity of frame T;
$\boldsymbol{\alpha}_T(t)$	Angular acceleration of frame T;
$\mathbf{a}_T(t)$	Linear acceleration of frame T.

According to the scheme shown in Fig. 1, the Cartesian planner is normally followed by an inverse kinematics block which returns an equivalent trajectory in the configuration space that, in turn, is used to drive the joint actuators.

When trajectories are planned in real-time, their feasibility w.r.t. the physical limits of the system can not be guaranteed in advance. In particular, joint velocities could exceed the given bounds or available motor torques could be insufficient to guarantee the required joint accelerations. If $\mathbf{q} \in \mathbb{R}^N$ is the vector of the joint generalized variables, the following requirements must be fulfilled:

$$\dot{\mathbf{q}}^- \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}^+, \tag{1}$$

$$\ddot{\mathbf{q}}^- \leq \ddot{\mathbf{q}} \leq \ddot{\mathbf{q}}^+, \tag{2}$$

where $\dot{\mathbf{q}}^-, \ddot{\mathbf{q}}^- \in (\mathbb{R}^-)^N$, and $\dot{\mathbf{q}}^+, \ddot{\mathbf{q}}^+ \in (\mathbb{R}^+)^N$ represent proper bounds for joint velocities and accelerations. The given limits can be assumed variable thus permitting, as shown in Section V, the possible management of torque constraints.

The fulfillment of (1) and (2) is obtained by modifying the standard planning scheme according to Fig. 1. In particular, while the position signals are sent unchanged to the inverse kinematics block, the orientation signals are preprocessed by the SAS which slightly modifies the tool frame orientation in order to preserve the feasibility. The basic assumption that is made for the synthesis of the system is that trajectory equations are unknown to the SAS: Reference

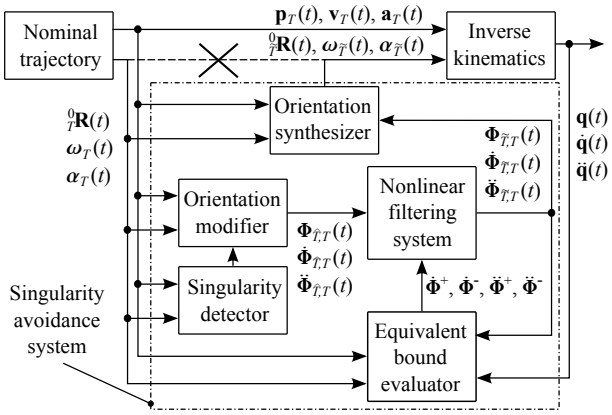

Fig. 1. A schematic representation of the singularity avoidance system: The orientation signals are not directly sent to the inverse kinematic block but, conversely, they are processed by the SAS.

signals ${}^{0}_{T}\mathbf{R}(t), \boldsymbol{\omega}_T(t), \boldsymbol{\alpha}_T(t), \mathbf{p}_T(t), \mathbf{v}_T(t), \mathbf{a}_T(t)$ are acquired and processed by the SAS at each sample time in order to promptly react to unforeseen singularities, thus permitting the real-time management of trajectories in the operational space.

The first block of the SAS is the Singularity Detector (SD). It detects possible singular configurations along the nominal trajectory and consequently raises an alarm in order to activate the SAS. Such alarm is then read by the Orientation Modifier (OM), which specifies, through an auxiliary frame $\hat{T}$, a new candidate orientation for the tool frame in order to skip the singularity. The orientation displacement between nominal reference frame T and $\hat{T}$ is described through vector $\boldsymbol{\Phi}_{\hat{T},T} := [\alpha \ \ \beta \ \ \gamma]^T$, which corresponds to a Roll-Pitch-Yaw (RPY) minimal notation. Since only minor trajectory modifications are admitted, $\boldsymbol{\Phi}_{\hat{T},T}$ is generally close to zero: As shown later in the paper this eliminates the appearance of singularity problems deriving from the adopted notation. The SD and the OM blocks are described in Section III.

The trajectory of frame $\hat{T}$ can still be unfeasible. In fact, the avoidance of singular points, by itself, is not sufficient to guarantee that the modified trajectory is feasible w.r.t. (1) and (2). On the contrary, any trajectory change that is made in configurations which are close to singularities could potentially require additional joint velocities and accelerations, thus worsening an already critical situation. For this reason, the OM is followed by a Nonlinear Filtering System (NFS) which constrains the dynamics of $\boldsymbol{\Phi}_{\hat{T},T}$ between proper bounds. The NSF is made by three independent scalar filters, each of them acting on a single component of $\boldsymbol{\Phi}_{\hat{T},T}$. The NFS returns a signal $\boldsymbol{\Phi}_{\tilde{T},T}$ which represents the best possible approximation of $\boldsymbol{\Phi}_{\hat{T},T}$ satisfying the following inequalities

$$\dot{\boldsymbol{\Phi}}^- \leq \dot{\boldsymbol{\Phi}}_{\tilde{T},T} \leq \dot{\boldsymbol{\Phi}}^+, \tag{3}$$

$$\ddot{\boldsymbol{\Phi}}^- \leq \ddot{\boldsymbol{\Phi}}_{\tilde{T},T} \leq \ddot{\boldsymbol{\Phi}}^+, \tag{4}$$

where $\dot{\boldsymbol{\Phi}}^-, \ddot{\boldsymbol{\Phi}}^- \in (\mathbb{R}^-)^3$ and $\dot{\boldsymbol{\Phi}}^+, \ddot{\boldsymbol{\Phi}}^+ \in (\mathbb{R}^+)^3$ are proper limits that are devised by the Equivalent Bound Evaluator (EBE) starting from the actual constraints of the actuators and from the manipulator status of motion. As shown in Section V, if (3) and (4) are satisfied, then (1) and (2) hold with certainty

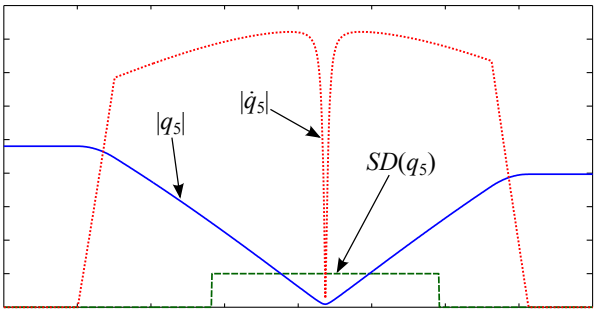

Fig. 2. Signal $|q_5|$ (solid line) and $|\dot{q}_5|$ (dotted line) that are used for the evaluation of alarm signal $SD(q_5)$ (dashed line).

and the trajectory is feasible. The internal structure of the NFS filters has been extensively described in [33], so that interested readers can refer to that paper for details.

The last block shown in Fig. 1, i.e., the Orientation Synthesizer (OS), returns ${}^{0}_{\widetilde{T}}\mathbf{R}(t)$, $\boldsymbol{\omega}_{\widetilde{T}}(t)$, and $\boldsymbol{\alpha}_{\widetilde{T}}(t)$ of the modified trajectory, which are obtained, according to the procedure described in Section IV, from ${}^{0}_{T}\mathbf{R}(t)$, $\boldsymbol{\omega}_T(t)$, and $\boldsymbol{\alpha}_T(t)$ and from the knowledge of $\boldsymbol{\Phi}_{\widetilde{T},T}$, $\dot{\boldsymbol{\Phi}}_{\widetilde{T},T}$, and $\ddot{\boldsymbol{\Phi}}_{\widetilde{T},T}$.

III. Singularity Detector and Orientation Modifier blocks

As specified in Section II, the SD detects the insurgence of singular configurations and raises an alarm that activates the OM. Several techniques have been tested for the implementation of the SD and the OM blocks. The method described in the following was finally chosen because of its very light computational burden, but it is worth to mention that techniques based on the use of the manipulability index have shown similar performances.

The alarm signal must be raised sufficiently in advance in order to have enough time to smoothly modify the tool orientation. Since the focus of this paper is on wrist singularities, the SD constantly controls the fifth joint variable, i.e., q_5. In particular, good results have been achieved by adopting the following alarm function

$$SD(q_5) := \begin{cases} \text{true} & \text{if} \quad \left\{ \left[|q_5| \leq \overline{a}_1 \right] \text{and} \left[|\dot{q}_5| \geq \overline{b} \right] \right\} \\ & \quad \text{or} \left[|q_5| \leq \overline{a}_2 \right], \\ \text{false} & \text{otherwise}, \end{cases} \tag{5}$$

where $\overline{a}_1 > \overline{a}_2$ and $\overline{b}$ represent appropriate thresholds. In particular, $\overline{a}_1$ is a "far" threshold which announces that the robot is approaching a singularity. It activates the alarm only if the $\dot{q}_5$ is higher than threshold $\overline{b}$: If the velocity is not too high the alarm can be delayed. Conversely, closer to the singularity, i.e., when $|q_5| \leq \overline{a}_2$, the alarm is raised independently from $\dot{q}_5$. Fig. 2 shows an example case taken from one of the test trajectories proposed in Section VII.

Once the alarm has been raised, the OM block synthesizes the new candidate orientation for the tool frame, which is expressed through an auxiliary frame $\hat{T}$. The orientation displacement between nominal reference frame T and modified reference frame $\hat{T}$ is represented by means of vector $\boldsymbol{\Phi}_{\hat{T},T}(\mathbf{q})$ which is associated to an RPY notation. Vector $\boldsymbol{\Phi}_{\hat{T},T}(\mathbf{q})$ is obtained by performing a sort of local search in the surroundings of neutral configuration $\boldsymbol{\Phi}_0 := [0\ 0\ 0]^T$. In particular, by defining the search space as follows $\left\{ \boldsymbol{\sigma} := [\sigma_1\ \sigma_2\ \sigma_3]^T \in \{-1, 0, 1\}^3 \setminus [0\ 0\ 0]^T \right\}$, it is possible to associate to each one of the resulting 26 vectors $\boldsymbol{\sigma}$ an angular displacement according to the following equation

$$\hat{\boldsymbol{\Phi}}_{\hat{T},T} := \left[\frac{\sigma_1 r}{\|\boldsymbol{\sigma}\|} \ \frac{\sigma_2 r}{\|\boldsymbol{\sigma}\|} \ \frac{\sigma_3 r}{\|\boldsymbol{\sigma}\|} \right]^T , \tag{6}$$

where $r = 10^{-3}$ defines the radius of the search space. Then, named $\hat{\boldsymbol{\Phi}}^*_{\hat{T},T}$ the displacement which causes the highest increment of $|q_5|$ w.r.t. the value detected for $\boldsymbol{\Phi}_0$, $\boldsymbol{\Phi}_{\hat{T},T}(\mathbf{q})$ is obtained as follows

$$\boldsymbol{\Phi}_{\hat{T},T}(\mathbf{q}) := c(\mathbf{q})\ \hat{\boldsymbol{\Phi}}^*_{\hat{T},T} , \tag{7}$$

where $c(\mathbf{q})$ is given by the following scalar

$$c(\mathbf{q}) := \frac{c_1 \left\| \mathbf{v}_T(\mathbf{q}) \right\| + c_2}{c_3 + |q_5|} . \tag{8}$$

Practically, $c(\mathbf{q})$ is chosen such as to guarantee higher angular displacements when close to critical configurations and for higher speeds of the tool-frame. Coefficient c_1, c_2, and c_3 can be selected on the basis of some simple considerations. It is possible to impose, for example, that close to the singularity, i.e., when $q_5 \simeq 0$, the maximum and the minimum angular deviations w.r.t. the nominal trajectory do not exceed some given value. To this purpose, the following equations can be used

$$c_{max} = \frac{c_1 \left\| \mathbf{v}_{T_{max}} \right\| + c_2}{c_3} , \tag{9}$$

$$c_{min} = \frac{c_2}{c_3} , \tag{10}$$

where $\mathbf{v}_{T_{max}}$ is the maximum Cartesian velocity for the application considered, c_{max} establishes the maximum deviation at full speed, and c_{min} fixes the minimum deviation for speeds close to zero.

The choice of c_3 determines which term in (8), among $\|\mathbf{v}_T\|$ and $|q_5|$, has more influence on $c(\mathbf{q})$. Close to singularity $|q_5|$ is very small so that, for large values of c_3, the variability of $c(\mathbf{q})$ mainly depends on $\|\mathbf{v}_T\|$, i.e., the tool-frame orientation basically changes on the basis of the Cartesian velocity. Conversely, for small values of c_3, the variability of $c(\mathbf{q})$ is mainly function of $|q_5|$, i.e., it depends on the closeness to singular points. With a few iterations on c_3, it is possible to devise an appropriate tuning for the system since c_1 and c_2 can be evaluated through (9) and (10).

IV. The Orientation Synthesizer block

The OS is the SAS output block which combines the orientation of the nominal trajectory with $\boldsymbol{\Phi}_{\widetilde{T},T} := [\alpha\ \beta\ \gamma]^T$, $\dot{\boldsymbol{\Phi}}_{\widetilde{T},T} := [\dot{\alpha}\ \dot{\beta}\ \dot{\gamma}]^T$, and $\ddot{\boldsymbol{\Phi}}_{\widetilde{T},T} := [\ddot{\alpha}\ \ddot{\beta}\ \ddot{\gamma}]^T$, i.e., with the signals that are provided by the NFS and whose synthesis will be later discussed in Section V. Changes only affect the tool frame orientation and, consequently, its angular velocity and acceleration while, according to the premises, signals $\mathbf{p}_T(t)$, $\mathbf{v}_T(t)$, and $\mathbf{a}_T(t)$ are left unchanged in order to preserve the geometry of the Cartesian path.

The modified orientation of the tool frame can be evaluated according to the following expression

$$ {}^{0}_{\widetilde{T}}\mathbf{R} = {}^{0}_{T}\mathbf{R}\ {}^{T}_{\widetilde{T}}\mathbf{R}(\mathbf{\Phi}_{\widetilde{T},T}) \,, $$

where ${}^{T}_{\widetilde{T}}\mathbf{R}(\mathbf{\Phi}_{\widetilde{T},T})$ is the rotation matrix corresponding to $\mathbf{\Phi}_{\widetilde{T},T}$.

As known, the relative angular velocity between frame $\widetilde{T}$ and frame T can always be expressed as follows

$$ {}^{T}\boldsymbol{\omega}_{\widetilde{T},T} = \mathbf{T}(\mathbf{\Phi}_{\widetilde{T},T})\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} \,, \tag{11} $$

where $\mathbf{T}(\mathbf{\Phi}_{\widetilde{T},T})$ is a proper matrix which correlates angular velocities to $\dot{\mathbf{\Phi}}_{\widetilde{T},T}$. The structure of $\mathbf{T}(\mathbf{\Phi}_{\widetilde{T},T})$ depends on the minimal notation. For the RPY notation, it can be easily proved that it is given by

$$ \mathbf{T}(\mathbf{\Phi}_{\widetilde{T},T}) = \begin{bmatrix} 1 & 0 & \sin(\beta) \\ 0 & \cos(\alpha) & -\sin(\alpha)\cos(\beta) \\ 0 & \sin(\alpha) & \cos(\alpha)\cos(\beta) \end{bmatrix} . \tag{12} $$

In the following, the argument of $\mathbf{T}(\cdot)$ will be omitted for conciseness.

Bearing in mind (11) and assuming the compact notation for vectors described w.r.t. frame 0, it is immediately possible to derive the angular velocity of frame $\widetilde{T}$ by means of the following equation

$$ \boldsymbol{\omega}_{\widetilde{T}} = \boldsymbol{\omega}_{T} + \boldsymbol{\omega}_{\widetilde{T},T} = \boldsymbol{\omega}_{T} + {}^{0}_{T}\mathbf{R}\ {}^{T}\boldsymbol{\omega}_{\widetilde{T},T} = \boldsymbol{\omega}_{T} + {}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} \,. \tag{13} $$

The angular acceleration of $\widetilde{T}$ can be evidently obtained by deriving (13). A few algebraic manipulations lead to the following expression

$$ \begin{aligned} \boldsymbol{\alpha}_{\widetilde{T}} &= \boldsymbol{\alpha}_{T} + {}^{0}_{T}\dot{\mathbf{R}}\,\mathbf{T}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} + {}^{0}_{T}\mathbf{R}\,\dot{\mathbf{T}}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} + {}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\ddot{\mathbf{\Phi}}_{\widetilde{T},T} \\ &= \boldsymbol{\alpha}_{T} + \boldsymbol{\omega}_{T} \times {}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} + {}^{0}_{T}\mathbf{R}\left[\dot{\overline{\mathbf{T}}}\,\dot{\overline{\mathbf{\Phi}}}_{\widetilde{T},T} + \mathbf{T}\,\ddot{\mathbf{\Phi}}_{\widetilde{T},T}\right], \end{aligned} \tag{14} $$

where $\dot{\overline{\mathbf{\Phi}}}_{\widetilde{T},T} := [\dot{\alpha}\dot{\beta}\ \dot{\gamma}\dot{\alpha}\ \dot{\beta}\dot{\gamma}]^{T}$ and

$$ \dot{\overline{\mathbf{T}}} := \begin{bmatrix} 0 & 0 & \cos(\beta) \\ -\sin(\alpha) & -\cos(\alpha)\cos(\beta) & \sin(\alpha)\sin(\beta) \\ \cos(\alpha) & -\sin(\alpha)\cos(\beta) & -\cos(\alpha)\sin(\beta) \end{bmatrix} . $$

V. The Equivalent Bounds Evaluator block

As specified in Section II, the SAS scheme requires that bounds $\dot{\mathbf{q}}^{-}, \dot{\mathbf{q}}^{+}, \ddot{\mathbf{q}}^{-}$, and $\ddot{\mathbf{q}}^{+}$ must be converted into equivalent bounds on $\dot{\mathbf{\Phi}}^{-}, \dot{\mathbf{\Phi}}^{-}, \dot{\mathbf{\Phi}}^{+}$, and $\ddot{\mathbf{\Phi}}^{+}$. Such conversion depends on the current position of tool frame $\widetilde{T}$, assumed to coincide with that of T (i.e., $\mathbf{p}_{\widetilde{T}} = \mathbf{p}_{T}$), and on its current orientation, assumed to be given by ${}^{0}_{\widetilde{T}}\mathbf{R}$. The solution of the inverse kinematic problem for ${}^{0}_{\widetilde{T}}\mathbf{R}$ and $\mathbf{p}_{\widetilde{T}}$ returns the current value of $\mathbf{q}$, which can be used for the evaluation of the Jacobian matrix $\mathbf{J}_{\widetilde{T}}$. For this reason, from now on, $\mathbf{J}_{\widetilde{T}}$ will be supposed to be known.

In non-singular configurations – recall that the proposed approach makes it possible to avoid such configurations – it is possible to write

$$ \dot{\mathbf{q}} = \mathbf{J}^{-1}_{\widetilde{T}}\,\overline{\mathbf{v}}_{\widetilde{T}} \,, \tag{15} $$

where $\overline{\mathbf{v}}_{\widetilde{T}} := [\mathbf{v}^{T}_{\widetilde{T}}\ \boldsymbol{\omega}^{T}_{\widetilde{T}}]^{T}$ represents the generalized velocity of $\widetilde{T}$. It is always possible to partition $\mathbf{J}^{-1}_{\widetilde{T}}$ into two 6x3 submatrices respectively named $\overline{\mathbf{J}}_{v_{\widetilde{T}}}$ and $\overline{\mathbf{J}}_{\omega_{\widetilde{T}}}$ and defined such that the following equality applies

$$ \mathbf{J}^{-1}_{\widetilde{T}} =: \left[\overline{\mathbf{J}}_{v_{\widetilde{T}}} \mid \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\right], \tag{16} $$

By virtue of (16), it is possible to rewrite (15) as follows

$$ \dot{\mathbf{q}} = \overline{\mathbf{J}}_{v_{\widetilde{T}}}\,\mathbf{v}_{\widetilde{T}} + \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\,\boldsymbol{\omega}_{\widetilde{T}} \,. $$

Velocity $\mathbf{v}_{\widetilde{T}}$ coincides with $\mathbf{v}_{T}$, while $\boldsymbol{\omega}_{\widetilde{T}}$ is given by (13), so that constraint (1) can be posed in the following form

$$ \dot{\mathbf{q}}^{-} \le \overline{\mathbf{J}}_{v_{\widetilde{T}}}\,\mathbf{v}_{T} + \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\left[\boldsymbol{\omega}_{T} + {}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T}\right] \le \dot{\mathbf{q}}^{+} \,, $$

or, analogously,

$$ \dot{\tilde{\mathbf{q}}}^{-} \le \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\,{}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} \le \dot{\tilde{\mathbf{q}}}^{+} \,, \tag{17} $$

where

$$ \dot{\tilde{\mathbf{q}}}^{-} := \dot{\mathbf{q}}^{-} - \mathbf{J}^{-1}_{\widetilde{T}}\,\overline{\mathbf{v}}_{T}, \tag{18} $$

$$ \dot{\tilde{\mathbf{q}}}^{+} := \dot{\mathbf{q}}^{+} - \mathbf{J}^{-1}_{\widetilde{T}}\,\overline{\mathbf{v}}_{T}. \tag{19} $$

Any feasible $\dot{\mathbf{\Phi}}_{\widetilde{T},T}$ must fulfill (17). All the terms in (18) and (19) are known: $\dot{\mathbf{q}}^{-}$ and $\dot{\mathbf{q}}^{+}$ are user-defined bounds, $\mathbf{J}^{-1}_{\widetilde{T}}$ is the inverse Jacobian associated to the modified tool frame $\widetilde{T}$, $\overline{\mathbf{v}}_{T}$ is the generalized velocity of frame T. In the same way, many of the terms in (17) are known: $\overline{\mathbf{J}}_{\omega_{\widetilde{T}}}$ derives from (16), ${}^{0}_{T}\mathbf{R}$ is the rotation matrix of frame T, and $\mathbf{T}$ is given by (12).

The acceleration constraints can be similarly handled. It is known that

$$ \overline{\mathbf{a}}_{\widetilde{T}} = \dot{\mathbf{J}}_{\widetilde{T}}\,\dot{\mathbf{q}} + \mathbf{J}_{\widetilde{T}}\,\ddot{\mathbf{q}} \,, $$

which can be solved for $\ddot{\mathbf{q}}$, thus leading to

$$ \ddot{\mathbf{q}} = \mathbf{J}^{-1}_{\widetilde{T}}\left[\overline{\mathbf{a}}_{\widetilde{T}} - \dot{\mathbf{J}}_{\widetilde{T}}\dot{\mathbf{q}}\right] = \mathbf{J}^{-1}_{\widetilde{T}}\left[\overline{\mathbf{a}}_{\widetilde{T}} - \dot{\mathbf{J}}_{\widetilde{T}}\,\mathbf{J}^{-1}_{\widetilde{T}}\,\overline{\mathbf{v}}_{\widetilde{T}}\right] . \tag{20} $$

By using the partitioning scheme suggested in (16), it is possible to rewrite (20) as follows

$$ \ddot{\mathbf{q}} = \overline{\mathbf{J}}_{v_{\widetilde{T}}}\,\mathbf{a}_{\widetilde{T}} + \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\,\boldsymbol{\alpha}_{\widetilde{T}} - \mathbf{J}^{-1}_{\widetilde{T}}\,\dot{\mathbf{J}}_{\widetilde{T}}\,\mathbf{J}^{-1}_{\widetilde{T}}\,\overline{\mathbf{v}}_{\widetilde{T}} \,, $$

which, in turn, can be written, because of (14) and since $\mathbf{a}_{\widetilde{T}} = \mathbf{a}_{T}$, as follows

$$ \begin{aligned} \ddot{\mathbf{q}} = \overline{\mathbf{J}}_{v_{\widetilde{T}}}\,\mathbf{a}_{T} + \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\Big\{&\boldsymbol{\alpha}_{T} + \boldsymbol{\omega}_{T} \times {}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} \\ &+ {}^{0}_{T}\mathbf{R}\left[\dot{\overline{\mathbf{T}}}\,\dot{\overline{\mathbf{\Phi}}}_{\widetilde{T},T} + \mathbf{T}\,\ddot{\mathbf{\Phi}}_{\widetilde{T},T}\right]\Big\} - \mathbf{J}^{-1}_{\widetilde{T}}\,\dot{\mathbf{J}}_{\widetilde{T}}\,\mathbf{J}^{-1}_{\widetilde{T}}\,\overline{\mathbf{v}}_{\widetilde{T}} \,. \end{aligned} \tag{21} $$

By substituting (21) into (2), after a few manipulations it is possible to obtain

$$ \ddot{\tilde{\mathbf{q}}}^{-} \le \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\,{}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\ddot{\mathbf{\Phi}}_{\widetilde{T},T} \le \ddot{\tilde{\mathbf{q}}}^{+} \,, \tag{22} $$

where

$$ \begin{aligned} \ddot{\tilde{\mathbf{q}}}^{-} :=\ & \ddot{\mathbf{q}}^{-} - \mathbf{J}^{-1}_{\widetilde{T}}\left[\overline{\mathbf{a}}_{T} - \dot{\mathbf{J}}_{\widetilde{T}}\,\mathbf{J}^{-1}_{\widetilde{T}}\,\overline{\mathbf{v}}_{\widetilde{T}}\right] \\ & - \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\left\{\boldsymbol{\omega}_{T} \times {}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} + {}^{0}_{T}\mathbf{R}\,\dot{\overline{\mathbf{T}}}\,\dot{\overline{\mathbf{\Phi}}}_{\widetilde{T},T}\right\} , \\ \ddot{\tilde{\mathbf{q}}}^{+} :=\ & \ddot{\mathbf{q}}^{+} - \mathbf{J}^{-1}_{\widetilde{T}}\left[\overline{\mathbf{a}}_{T} - \dot{\mathbf{J}}_{\widetilde{T}}\,\mathbf{J}^{-1}_{\widetilde{T}}\,\overline{\mathbf{v}}_{\widetilde{T}}\right] \\ & - \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\left\{\boldsymbol{\omega}_{T} \times {}^{0}_{T}\mathbf{R}\,\mathbf{T}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} + {}^{0}_{T}\mathbf{R}\,\dot{\overline{\mathbf{T}}}\,\dot{\overline{\mathbf{\Phi}}}_{\widetilde{T},T}\right\} . \end{aligned} $$

Vectors $\dddot{\tilde{\mathbf{q}}}^-$ and $\dddot{\tilde{\mathbf{q}}}^+$ can be evaluated from the knowledge of the nominal trajectory, of the current values of $\mathbf{\Phi}_{\widetilde{T},T}$ and $\dot{\mathbf{\Phi}}_{\widetilde{T},T}$, and of the user-defined acceleration bounds, i.e., $\ddot{\mathbf{q}}^-$ and $\ddot{\mathbf{q}}^+$. Evidently, any feasible $\ddot{\mathbf{\Phi}}_{\widetilde{T},T}$ must fulfill (22).

It is worth to mention that in this paper $\ddot{\mathbf{q}}^-$ and $\ddot{\mathbf{q}}^+$ are assumed to be constant for the sake of simplicity, but the whole approach does not actually require such assumption. This potentiality can be used, e.g., to manage the torque constraints. To this purpose, given proper torque bounds $\boldsymbol{\tau}^- \in (\mathbb{R}^-)^N$ and $\boldsymbol{\tau}^+ \in (\mathbb{R}^+)^N$ it is possible to convert them into the following equivalent acceleration bounds

$$\ddot{\mathbf{q}}^+ = \mathbf{M}^{-1}(\mathbf{q})[\boldsymbol{\tau}^+ - \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) - \mathbf{f}(\mathbf{q},\dot{\mathbf{q}})], \quad (23)$$

$$\ddot{\mathbf{q}}^- = \mathbf{M}^{-1}(\mathbf{q})[\boldsymbol{\tau}^- - \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) - \mathbf{f}(\mathbf{q},\dot{\mathbf{q}})], \quad (24)$$

which clearly depend on $\mathbf{q}$ and $\dot{\mathbf{q}}$. Equations (23) and (24) directly descend from the usual manipulator model

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) \ ,$$

where $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ is the matrix of Coriolis and centripetal terms, $\mathbf{g}(\mathbf{q})$ is the vector of gravitational terms, and $\mathbf{f}(\mathbf{q},\dot{\mathbf{q}})$ is the vector associated to friction. Similarly, the actuators' dynamics can be accounted for by means of techniques analogous to those derived for [34].

Equations (17) and (22) represent the starting point for converting the original bounds on $\dot{\mathbf{q}}$ and on $\ddot{\mathbf{q}}$ into equivalent bounds on $\dot{\mathbf{\Phi}}_{\widetilde{T},T}$ and on $\ddot{\mathbf{\Phi}}_{\widetilde{T},T}$. However, such conversion is not straightforward since $\mathbf{q}$ and $\mathbf{\Phi}_{\widetilde{T},T}$ are dimensionally different. More in details, (17) and (22) represent a set of $2N$ independent constraints on $\dot{\mathbf{q}}$ and on $\ddot{\mathbf{q}}$ that must be converted into 6 equivalent constraint equations on $\dot{\mathbf{\Phi}}_{\widetilde{T},T}$ and on $\ddot{\mathbf{\Phi}}_{\widetilde{T},T}$. The conversion clearly admits some degrees of freedom that can be used to obtain the best possible performances from the trajectory modification strategy. This is the reason why the equivalent bounds are obtained by solving a linear programming (LP) problem which returns a set of proper equivalent limits $\dot{\mathbf{\Phi}}^- := [\dot{\alpha}^-\ \dot{\beta}^-\ \dot{\gamma}^-]^T, \ddot{\mathbf{\Phi}}^- := [\ddot{\alpha}^-\ \ddot{\beta}^-\ \ddot{\gamma}^-]^T$ and $\dot{\mathbf{\Phi}}^+ := [\dot{\alpha}^+\ \dot{\beta}^+\ \dot{\gamma}^+]^T, \ddot{\mathbf{\Phi}}^+ := [\ddot{\alpha}^+\ \ddot{\beta}^+\ \ddot{\gamma}^+]^T$ for $\mathbf{\Phi}_{\widetilde{T},T}$ and $\ddot{\mathbf{\Phi}}_{\widetilde{T},T}$ such that if (3) and (4) hold, then (17) and (22) – and, in turn, (1) and (2) – are certainly satisfied.

The best SAS performances can clearly be obtained when bounds on $\dot{\mathbf{\Phi}}_{\widetilde{T},T}$ and $\ddot{\mathbf{\Phi}}_{\widetilde{T},T}$ are kept wide: In fact, this permits a good reactivity of the trajectory modifier which can rapidly react to critical situations. As a consequence, by defining $\mathbf{A} := \overline{\mathbf{J}}_{\omega_{\widetilde{T}}}\,{}^0_T\mathbf{R}\,\mathbf{T}$ and bearing in mind (17) and (22), the optimal bounds can be obtained by solving the following semi-infinite minimax problem

$$\max_{\substack{\dot{\mathbf{\Phi}}^+,\,\ddot{\mathbf{\Phi}}^+ \in (\mathbb{R}^+)^3 \\ \dot{\mathbf{\Phi}}^-,\,\ddot{\mathbf{\Phi}}^- \in (\mathbb{R}^-)^3}} \ \min_{i=1,\ldots,6}\{\Gamma_i^+ - \Gamma_i^-\} \quad (25)$$

subject to

$$\dot{\tilde{\mathbf{q}}}^- \le \mathbf{A}\,\dot{\mathbf{\Phi}}_{\widetilde{T},T} \le \dot{\tilde{\mathbf{q}}}^+ \quad \forall \dot{\mathbf{\Phi}}_{\widetilde{T},T} \in [\dot{\mathbf{\Phi}}^-, \dot{\mathbf{\Phi}}^+], \quad (26)$$

$$\ddot{\tilde{\mathbf{q}}}^- \le \mathbf{A}\,\ddot{\mathbf{\Phi}}_{\widetilde{T},T} \le \ddot{\tilde{\mathbf{q}}}^+ \quad \forall \ddot{\mathbf{\Phi}}_{\widetilde{T},T} \in [\ddot{\mathbf{\Phi}}^-, \ddot{\mathbf{\Phi}}^+], \quad (27)$$

where Γ_i^+ and Γ_i^- are, respectively, the components of vectors $\mathbf{\Gamma}^+ := [k\,\dot{\alpha}^+\ k\,\dot{\beta}^+\ k\,\dot{\gamma}^+\ \ddot{\alpha}^+\ \ddot{\beta}^+\ \ddot{\gamma}^+]^T$ and $\mathbf{\Gamma}^- := [k\,\dot{\alpha}^-\ k\,\dot{\beta}^-\ k\,\dot{\gamma}^-\ \ddot{\alpha}^-\ \ddot{\beta}^-\ \ddot{\gamma}^-]^T$ which contain the desired bounds, while k is used to correctly weight the velocity and acceleration bounds. For the problem at hand, in particular, $k = 10$ guarantees acceleration bounds 10 times larger than velocity bounds.

Matrix $\mathbf{A}$ is certainly full-rank. As stated in Section III, indeed, $\mathbf{\Phi}_{\widetilde{T},T}$ is always kept close to zero, so that $\mathbf{T}(\mathbf{\Phi}_{\widetilde{T},T})$, owing to (12), is almost an identity matrix. Furthermore, the avoidance of singular configurations also guarantees that $\overline{\mathbf{J}}_{\omega_{\widetilde{T}}} \in \mathbb{R}^{N\times 3}$, i.e., the rightmost part of the inverse Jacobian, is certainly full rank. Finally, since ${}^0_T\mathbf{R} \in \mathbb{R}^{3\times 3}$ is always non-singular, it is possible to conclude that $\mathbf{A}$ is a full rank $N \times 3$ real matrix.

Equations (26) and (27) are clearly linear. This property can be used to convert both semi-infinite constraints into a set of finite constraints. For example, the feasibility of (26) can be checked by ignoring the interior points of box $[\dot{\mathbf{\Phi}}^-, \dot{\mathbf{\Phi}}^+]$ and by only inspecting its vertexes. More precisely, the feasibility of (26) is guaranteed if the following vertex points are feasible

$$\dot{\mathbf{\Phi}}_1 := [\dot{\alpha}^-\ \dot{\beta}^-\ \dot{\gamma}^-]^T, \quad \dot{\mathbf{\Phi}}_2 := [\dot{\alpha}^-\ \dot{\beta}^-\ \dot{\gamma}^+]^T,$$
$$\dot{\mathbf{\Phi}}_3 := [\dot{\alpha}^-\ \dot{\beta}^+\ \dot{\gamma}^-]^T, \quad \dot{\mathbf{\Phi}}_4 := [\dot{\alpha}^-\ \dot{\beta}^+\ \dot{\gamma}^+]^T,$$
$$\dot{\mathbf{\Phi}}_5 := [\dot{\alpha}^+\ \dot{\beta}^-\ \dot{\gamma}^-]^T, \quad \dot{\mathbf{\Phi}}_6 := [\dot{\alpha}^+\ \dot{\beta}^-\ \dot{\gamma}^+]^T,$$
$$\dot{\mathbf{\Phi}}_7 := [\dot{\alpha}^+\ \dot{\beta}^+\ \dot{\gamma}^-]^T, \quad \dot{\mathbf{\Phi}}_8 := [\dot{\alpha}^+\ \dot{\beta}^+\ \dot{\gamma}^+]^T.$$

The same concept applies for (27), so that (25) can be easily reconverted into the following LP problem

$$\max_{\lambda,\,\dot{\mathbf{\Phi}}^+,\ddot{\mathbf{\Phi}}^+,\dot{\mathbf{\Phi}}^-,\ddot{\mathbf{\Phi}}^-} \{\lambda\}$$

subject to

$$\lambda \le \Gamma_i^+ - \Gamma_i^- \quad i = 1,2,\ldots,6 \quad (28)$$

$$\dot{\mathbf{\Phi}}^+, \ddot{\mathbf{\Phi}}^+ \ge 0 \quad (29)$$

$$\dot{\mathbf{\Phi}}^-, \ddot{\mathbf{\Phi}}^- \le 0 \quad (30)$$

$$\dot{\tilde{\mathbf{q}}}^- \le \mathbf{A}\,\dot{\mathbf{\Phi}}_i \le \dot{\tilde{\mathbf{q}}}^+ \quad i = 1,2,\ldots,8, \quad (31)$$

$$\ddot{\tilde{\mathbf{q}}}^- \le \mathbf{A}\,\ddot{\mathbf{\Phi}}_i \le \ddot{\tilde{\mathbf{q}}}^+ \quad i = 1,2,\ldots,8. \quad (32)$$

For the application at hand, each linear problem must be solved within the sampling time of the process, i.e., in less than one millisecond. Existing LP solvers, like `CPLEX` are extremely fast but waste a considerable amount of time – at least, compatibly with the sampling time – for the preparation of their environment. For this reason, instead of `CPLEX`, a custom LP solver has been implemented by posing a great care in choosing the right strategies in order to reduce computational times. The proposed solution is based on the well known simplex algorithm (see, e.g., [35]). While such algorithm has an exponential worst-case complexity, it works very well in practice. It has been observed that the number of iterations required by the simplex algorithm is of the same order of magnitude of the number of constraints of the problem. Problem (28)-(32) has more than 200 constraints but only 13 variables. Thus, rather than applying the simplex algorithm to (28)-(32), it is convenient to apply it to its dual, where the number of constraints is equal to 13, i.e., the number of variables of problem (28)-(32). As known, the optimal value of the original problem is equal to the optimal value of the

dual problem, while the solution of the original problem can be derived from the dual solution, e.g., by the complementarity conditions (please, refer again to [35] for details).

An important feature of the simplex algorithm, which has been specifically exploited, is the so-called "warm-start", i.e., the possibility of solving an LP problem by starting from the solution of another LP problem, when this latter is just a slight perturbation of the former one. This is indeed our case: Each new LP problem (28)-(32) differs from the preceding one for what concerns matrix $\mathbf{A}$ and vectors $\dot{\tilde{\mathbf{q}}}^+, \dot{\tilde{\mathbf{q}}}^-, \ddot{\tilde{\mathbf{q}}}^+, \ddot{\tilde{\mathbf{q}}}^-$, but the new entries are just slight perturbations of the old ones. Let B_i^* be the optimal basis for problem (28)-(32) at the ith sampling instant, from which we can easily derive the primal and dual optimal solutions and their common optimal value (see again [35] for the details). In most cases (99.56% in our experiments) B_i^* is still an optimal basis at sampling instant $i+1$. In few cases (0.40% in our experiments) B_i^* is still primal or dual feasible at sampling instant $i+1$, so that it can be used as a warm starting point for the primal or dual simplex algorithm, respectively. In very few cases (0.04% in our experiments) B_i^* is neither primal nor dual feasible at sampling instant $i+1$, so that we need to apply the *two-phase* simplex method, in order to recover (dual) feasibility and start with the (dual) simplex algorithm (notice that this is also needed at the first iteration). Test experiments executed on a Intel Core2 Duo PC running at 3.0GHz have shown that the computational times for the LP problems are in the range $6.5 \cdot 10^{-5}$ s up to $5.2 \cdot 10^{-4}$ s, with an average time of $7.0 \cdot 10^{-5}$ s, i.e., they are perfectly compatible with the sample time of the SAS which is equal to $2 \cdot 10^{-3}$ s.

Equivalent bounds $\dot{\mathbf{\Phi}}^-, \ddot{\mathbf{\Phi}}^-, \dot{\mathbf{\Phi}}^+$, and $\ddot{\mathbf{\Phi}}^+$, once evaluated by the custom LP solver, are sent to the NFS for the generation of the smooth signals that are used to change the tool orientation. The description of the NFS is omitted for brevity, but its structure and behavior have been proposed and extensively discussed in [33].

VI. Simulation Results

The SAS has been simulated by considering the kinematics of a Comau Smart SiX 6-1.4 manipulator. In the following, the MKS system of units is adopted any time units are not explicitly specified. The components of constraint vectors $\dot{\mathbf{q}}^-, \dot{\mathbf{q}}^+, \ddot{\mathbf{q}}^-$, and $\ddot{\mathbf{q}}^+$ have been posed equal to ($i = 1, 2, \ldots, N$): $\dot{q}_i^- = -10$ rad s^{-1}, $\dot{q}_i^+ = 10$ rad s^{-1}, $\ddot{q}_i^- = -100$ rad s^{-2} and $\ddot{q}_i^+ = 100$ rad s^{-2}. Velocity limits coincide with those of the actual robot, while acceleration bounds, since the dynamic model of the manipulator was not available, have been tentatively chosen such as to obtain smooth transients. Thresholds in (5) and coefficients of (8) have been respectively posed equal to $\overline{a}_1 = 0.25$, $\overline{a}_2 = 0.16$, $\overline{b} = 0.65$, $c_1 = 7/450$, $c_2 = 2/4500$, and $c_3 = 17/1000$. Constants have been tuned such as to guarantee orientation changes in the range ± 10 deg at the maximum Cartesian speed considered for the experiments ($\|\mathbf{v}_T\| = 0.4$ ms^{-1}). The above settings have been adopted for all the experiments. The same tests have been also repeated by considering the Comau Singularity Avoidance System (CSAS), i.e., the algorithm for the singularity avoidance implemented in the Comau controllers. Simulations have been executed on the Comau ROBOSim, a software environment which is used by the Comau Robotics for the simulation of its manipulators. When the user indicates that a trajectory is critical the CSAS, which does not automatically detect singularities, plans three feasible point-to-point joint trajectories for the wrist axes such that initial and final orientations of the tool-frame coincide with those assigned, but intermediate orientations are left free. Then, the trajectories of the first three axes are planned such as to guarantee the accurate path tracking and the fulfillment of the time-law. Practically, similarly to what happens with the strategy proposed in this paper, the feasibility is preserved by modifying the tool-frame orientation.

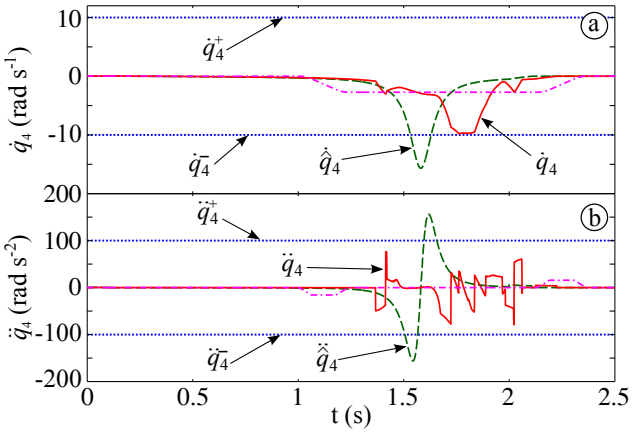

Fig. 3. (a) Velocities and (b) accelerations of joint 4 obtained with the SAS (solid lines) compared with those obtained without the SAS (dashed lines) and with the CSAS (dash-dotted lines). Dotted lines highlight the assigned bounds.

In the first test case proposed here, the trajectory is given by a straight line from $\mathbf{p}_A = [0.65\ 0.83\ 1.12]^T$ to $\mathbf{p}_B = [-0.2\ 0.83\ 1.12]^T$. The nominal orientation of the tool-frame is kept constant and equal to $\mathbf{\Phi}_{T,0} = [\pi/2\ 0\ \pi/2]^T$ (the RPY notation has been assumed). The trajectory passes close to a wrist singularity located at $\overline{\mathbf{p}}_T = [0\ 0.83\ 1.077]^T$ with $\|\mathbf{v}_T\| = 0.4$ ms^{-1}: If the SAS is not used, velocities and accelerations of joints 4 and 6 become unfeasible. Fig. 3 shows the behavior of joint 4: Without the SAS (dashed lines) the velocity and the acceleration limits are violated while, conversely, feasibility is preserved when the SAS is active (solid lines). The same figure also shows what happens when the CSAS is used (dash-dotted lines): Velocities and accelerations are evidently lower, but this result is achieved by modifying the tool orientation far from the singularity, i.e., when it is not necessary. Similar transients are obtained for joint 6. According to the theory, the SAS achieves this result with the aid of the nonlinear filters which keep the first and the second time derivatives of $\mathbf{\Phi}_{\tilde{T},T}$ between the equivalent bounds evaluated by the EBE. This assertion can be verified through Figs. 4–6, which compare components α, β, and γ of $\mathbf{\Phi}_{\tilde{T},T}$ (solid lines) with the homologous components $\hat{\alpha}$, $\hat{\beta}$, and $\hat{\gamma}$ of $\mathbf{\Phi}_{\hat{T},T}$ (dashed lines) generated by the OM block. In non critical regions, $\mathbf{\Phi}_{\hat{T},T}$ is kept equal to zero, so that the

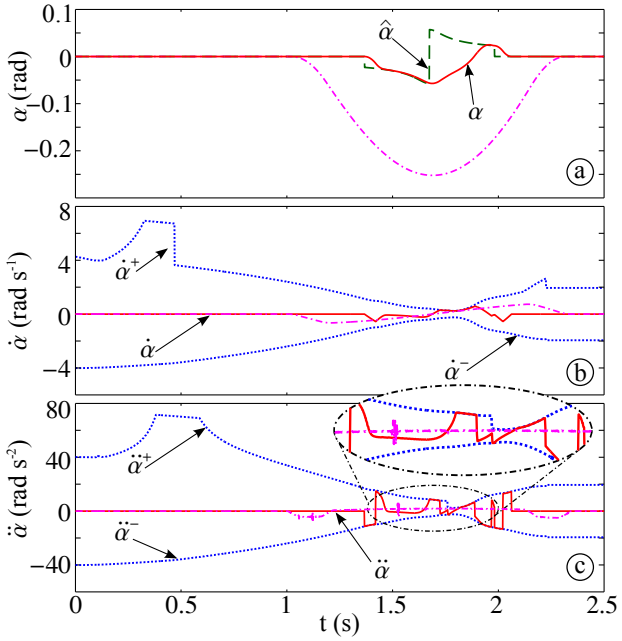

Fig. 4. Time shapes of α, $\dot{\alpha}$, and $\ddot{\alpha}$ as obtained by mean of the SAS (solid lines) and the CSAS (dash-dotted lines). Dotted lines represent the equivalent bounds on $\dot{\alpha}$ and on $\ddot{\alpha}$. Dashed line $\hat{\alpha}$ represents the non-smooth signal provided by the OM.

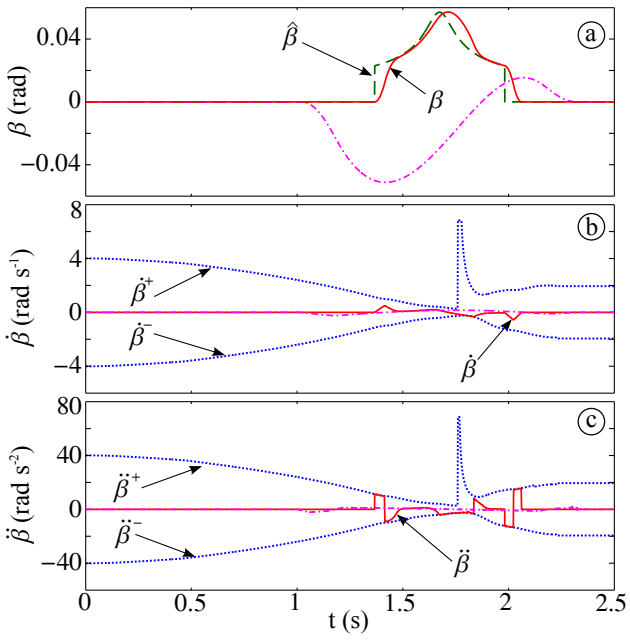

Fig. 5. Time shapes of β, $\dot{\beta}$, and $\ddot{\beta}$ as obtained by mean of the SAS (solid lines) and the CSAS (dash-dotted lines). Dotted lines represent the equivalent bounds on $\dot{\beta}$ and on $\ddot{\beta}$. Dashed line $\hat{\beta}$ represents the non-smooth signal provided by the OM of the SAS.

manipulator is driven by the nominal trajectory. Notice that, for the considered path, the OM block always imposes $\gamma = 0$. When a singularity is detected, the OM block modifies the tool orientation by generating a proper signal $\mathbf{\Phi}_{\hat{T},T}$. Such signal is generally discontinuous and unfeasible w.r.t. the given joint limits. For this reason, it is filtered by the NFS which generates a smooth signal $\mathbf{\Phi}_{\tilde{T},T}$ which first and second time derivatives fulfill equivalent bounds $\dot{\mathbf{\Phi}}^-, \ddot{\mathbf{\Phi}}^-$ and $\dot{\mathbf{\Phi}}^+, \ddot{\mathbf{\Phi}}^+$ (dotted lines), thus preserving the trajectory feasibility. The same Figs. 4–6 also show the values of α, β, and γ which are obtained with the CSAS (dash-dotted lines): Orientation changes are evidently higher – especially for α – and last longer, thus possibly worsening the process quality.

The effectiveness of the method proposed in this paper depends on the distance between the trajectory and the singular point. Consequently, the second set of tests has been conceived to verify the system performances for trajectories passing at different distances from $\overline{\mathbf{p}}_T$. To this purpose a set of linear paths from $\mathbf{p}_A = [0.65\ 0.83\ z]^T$ to $\mathbf{p}_B = [-0.2\ 0.83\ z]^T$ for $z \in [1.05, 1.12]$ has been generated and executed with $\|\mathbf{v}_T\| = 0.4$ ms^{-1}. Fig. 7 shows the shapes for α, β, and γ achieved with the SAS (solid lines) and with the CSAS (dash-dotted lines). As expected, the SAS changes the orientation only when the tool is close to the singularity. Moreover, the deviation depends on the distance between the path and $\overline{\mathbf{p}}_T$: Higher changes are introduced when the path passes closer to the singularity. For the considered $\mathbf{v}_T$, interval $z \in [1.065, 1.092]$ is still precluded (remember that singularity occurs at $z = 1.077$) but for lower velocities (0.1 ms^{-1}) $\overline{\mathbf{p}}_T$ can even be crossed. The CSAS performances are totally different: The orientation is modified when the tool is far

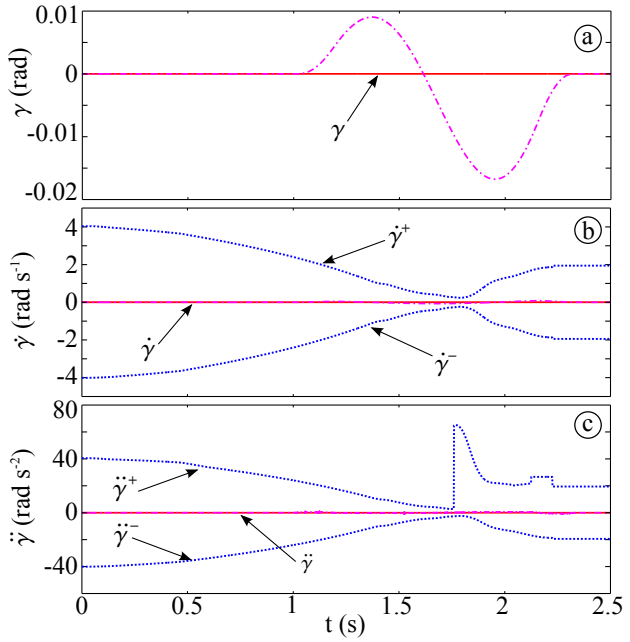

Fig. 6. Time shapes of γ, $\dot{\gamma}$, and $\ddot{\gamma}$ as obtained by mean of the SAS (solid lines) and of the CSAS (dash-dotted lines). . Dotted lines represent the equivalent bounds on $\dot{\gamma}$ and on $\ddot{\gamma}$. Notice that the SAS always imposes $\gamma = 0$

from the singularity, changes are more relevant (see, e.g., the maximum amplitude for α) and they are less affected by the distance between path and singular point. These are clearly undesired behaviors, since modifications of the nominal trajectory should be kept at minimum.

The last set of tests verify the SAS performances for tra-

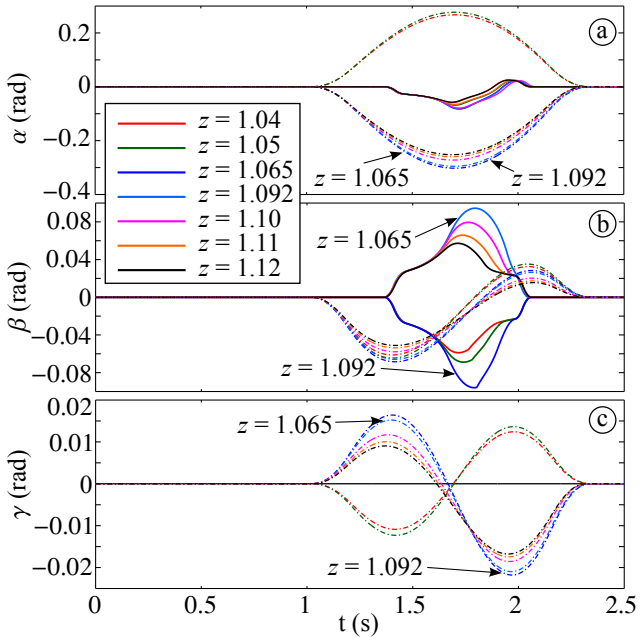

Fig. 7. Orientation modifications generated by the SAS (solid lines) and by the CSAS (dash-dotted lines) drawn in function of the distance from the singular point. Notice that the SAS does not modify γ.

jectories approaching $\overline{\mathbf{p}}_T$ from different directions, according to the procedure schematically shown in Fig. 8 and in Video1 of the graphical abstract. In particular, for each direction i, a sequence of parallel trajectories lying in the plane $y = 0.83$ and converging toward $\overline{\mathbf{p}}_T$ has been executed. The distance between two of them is equal to 10^{-3} m (in Video 1 such distance has been posed equal to 10^{-2} m in order to shorten its length). The scanning process ends when feasibility is lost: The minimum distance d_i from $\overline{\mathbf{p}}_T$ of the last feasible trajectory is stored and the scanning process is then repeated for the $(i+1)$-th direction. Four different velocities $\mathbf{v}_T$ have been considered and, for each of them, 20 scanning approaches have been performed for a total of 10^3 trajectories.

Distances d_i are compared in Fig. 9 with those obtained through an homologous scanning process executed without the SAS aid. The results highlight that the SAS permits, in any case, closer approaches to $\overline{\mathbf{p}}_T$. Improvements are especially evident for vertical ($m = \infty$) and horizontal trajectories ($m = 0$) since at the lowest velocities $\overline{\mathbf{p}}_T$ can even be crossed.

VII. Experimental Results

The SAS performances have been further checked by replicating the same experiments of Section VI on a real Comau Smart SiX 6-1.4 manipulator. The experimental setup is shown in Fig. 10 and it is composed by the manipulator itself, its control system, and a Personal Computer (PC) which processor is an Intel Core2 Duo @3.0GHz, running an Ubuntu operative system with an RTAI patched kernel. Some functions of the Comau control unit can be disabled and demanded to the external PC: For the experiments proposed in the following, the manipulator is driven by the SAS algorithm which runs on the PC. The communication between PC and controller is

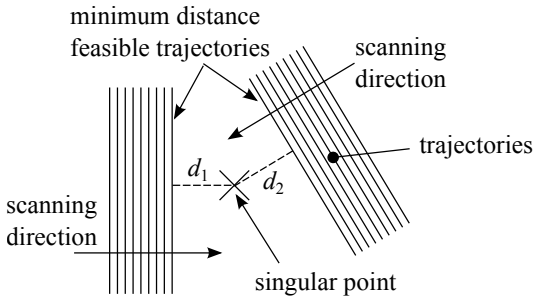

Fig. 8. Scanning procedure: The singular point is approached from several directions by means of parallel trajectories. For each direction, minimum distance d_i corresponding to the last feasible segment is stored.

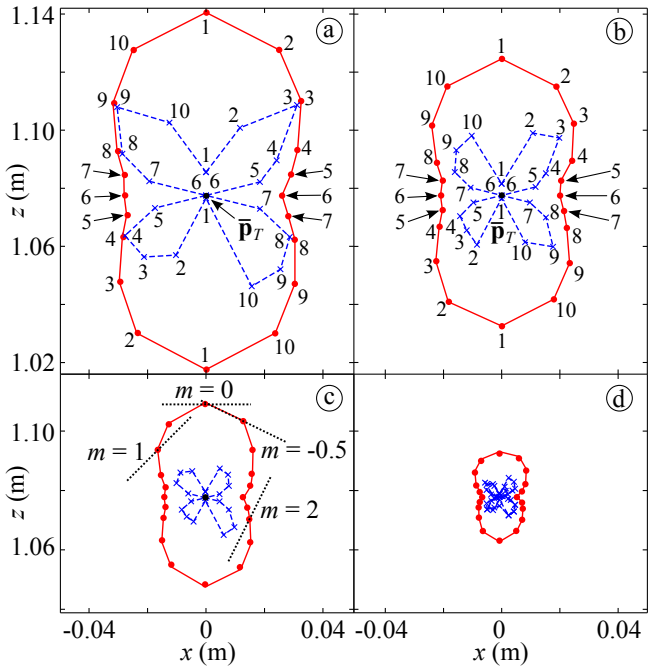

Fig. 9. Minimum achievable feasible distance from the singularity point with the SAS (dashed lines) or without it (solid lines). Results achieved for a longitudinal velocity equal to: (a) 0.4 ms^{-1}; (b) 0.3 ms^{-1}; (c) 0.2 ms^{-1}; (d) 0.1 ms^{-1}. Slopes of the linear segments: (1) m=0; (2) m=-1/2; (3) m=-1; (4) m=-2; (5) m=-4; (6) $m = \infty$; (7) m=4; (8) m=2; (9) m=1; (10) m=1/2.

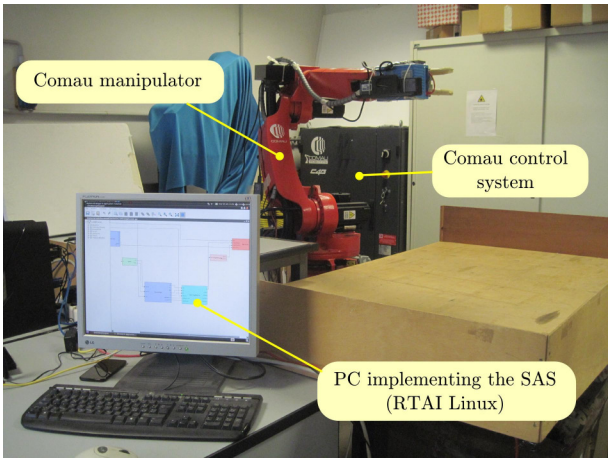

Fig. 10. The experimental setup.

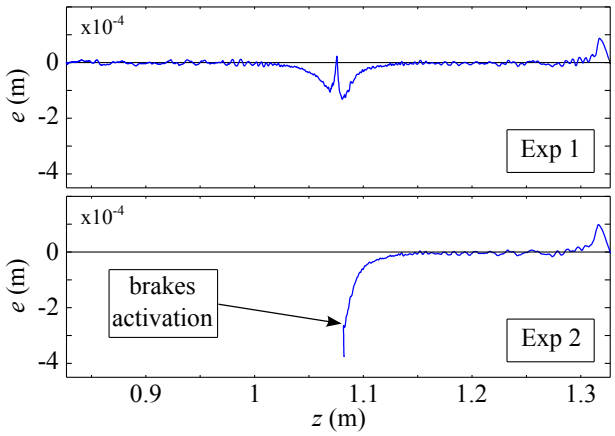

Fig. 11. Path tracking errors for vertical trajectories (SAS disabled) for Experiments 1 and 2.

obtained through a dedicated, real-time, Ethernet connection. The computational time for the whole algorithm spans in the interval from $9.2 \cdot 10^{-5}$ s to $7.0 \cdot 10^{-4}$ s, and it is on average equal to $1.3 \cdot 10^{-5}$ s, i.e., it is perfectly compatible with the process sample time which is equal to $2 \cdot 10^{-3}$ s.

The number of experimental results described in this section is limited for space reasons, but many other experiments have been executed. For example, as shown in Video 1 of the graphical abstract, the last set of tests proposed in Section VI has also been replicated on the real manipulator, pointing out a perfect agreement between simulated and experimental results.

The validation tests proposed in the following concern the execution of vertical and horizontal trajectories lying in the plane $y = 0.83$ and passing close to a wrist singularity. In particular, vertical trajectories pass $5{\cdot}10^{-3}$ m far from the singularity, while horizontal ones pass at $3.5{\cdot}10^{-3}$ m: Both trajectories are the closest ones to the singularity that can be safely executed with the SAS assistance for $\|\mathbf{v}_T\| = 0.4$ ms^{-1}. Performances are evaluated on the basis of the path tracking errors and results are also documented by Video 2 of the graphical abstract. In Experiment 1 a vertical trajectory is executed by disabling the SAS and by assuming $\|\mathbf{v}_T\| = 0.07$ ms^{-1}, which is the maximum velocity that can be feasibly maintained without the SAS or the CSAS assistance. Due to the extremely low velocities, dynamic errors are negligible, as can be evinced form Fig. 11 which shows that the maximum path tracking errors, as calculated from the motors' encoders through the forward kinematics, are roughly equal to 10^{-4} m. Then, in Experiment 2, still keeping the SAS and the CSAS disabled, speed is increased to 0.1 ms^{-1}: Joint velocities become unfeasible, so that the manipulator is stopped by the controller. In Experiments 3, 4, and 5 the SAS has been activated. For each experiment two alternative sets of bounds have been considered ($i = 1, 2, \ldots, 6$):

B1) $\dot{q}_i^- = -8$, $\dot{q}_i^+ = 8$, $\ddot{q}_i^- = -40$ and $\ddot{q}_i^+ = 40$;

B2) $\dot{q}_i^- = -10$, $\dot{q}_i^+ = 10$, $\ddot{q}_i^- = -25$ and $\ddot{q}_i^+ = 25$.

Specifically, B2 is characterized by higher velocity constraints and lower acceleration limits. The use of two alternative sets of bounds makes it possible to point out that the controller tracking performances can be improved through a proper choice of the SAS constraints. Practically, when B2 is used, trajectories require lower torques, so that closed-loop controllers are less solicited and tracking errors reduce. This characteristic is particularly evident at high speeds as shown in Fig. 12 and by the data reported in Table I. It is worth to mention that the feasibility has been preserved in all the experiments. In Experiment 6, the singularity has been managed through the CSAS. Path tracking errors are lower than those achieved in Experiment 5 because joint speeds are kept lower by admitting much higher orientation errors (see also Table I and Video 2).

Similar performances, as can be evinced from Table I and from Video 2, have been obtained for Experiments 7–12 which concern the horizontal trajectories.

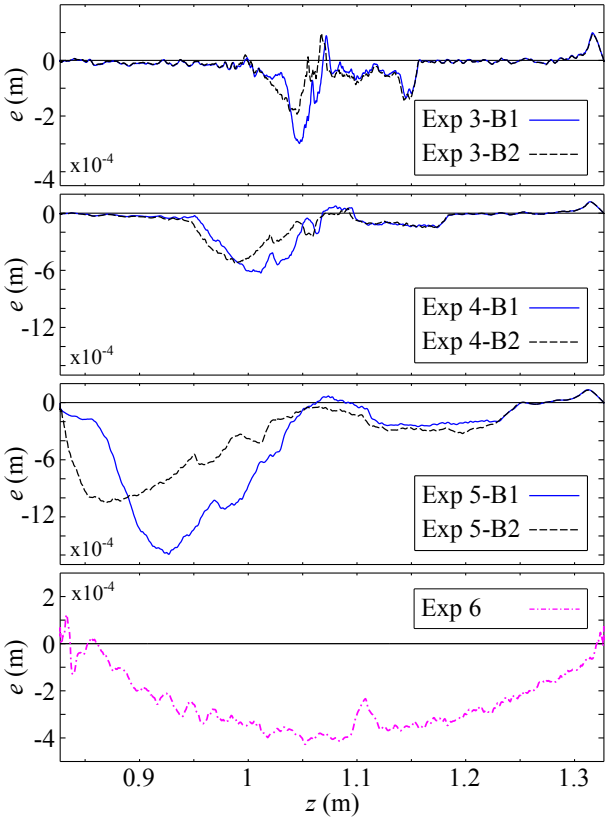

Fig. 12. Path tracking errors for Experiments 3, 4, 5, and 6.

TABLE I
EXPERIMENTALLY MEASURED MAXIMUM PATH TRACKING ERRORS AND MAXIMUM ANGULAR DEVIATIONS FROM THE NOMINAL PATH.

	vertical trajectories				horizontal trajectories			
	Exp.	$\|\mathbf{v}_T\|$ [ms^{-1}]	max error $\times 10^{-4}$ [m]	max $\|\alpha\|$, $\|\beta\|$, $\|\gamma\|$ [deg]	Exp.	$\|\mathbf{v}_T\|$ [ms^{-1}]	max error $\times 10^{-4}$ [m]	max $\|\alpha\|$, $\|\beta\|$, $\|\gamma\|$ [deg]
SAS off	1	0.07	1.31	–	7	0.02	1.32	–
	2	0.1	2.76	–	8	0.03	1.93	–
SAS on	3-B1	0.1	2.99	1.04	9-B1	0.03	4.65	0.391
	3-B2	0.1	1.93	1.04	9-B2	0.03	5.28	0.391
	4-B1	0.2	6.28	3.00	10-B1	0.2	13.5	2.86
	4-B2	0.2	5.26	2.99	10-B2	0.2	10.6	3.02
	5-B1	0.4	15.9	6.58	11-B1	0.4	21.9	4.57
	5-B2	0.4	10.5	6.22	11-B2	0.4	16.4	5.07
CSAS on	6	0.4	4.23	15.61	12	0.4	2.54	17.33

VIII. Conclusions and future works.

The approach proposed in this paper for the automatic handling of kinematic singularity can be used any time a slight modification of the tool orientation can be reasonably admitted in order to preserve an accurate tracking of the Cartesian trajectory. The achievable distances from singular points are evidently influenced by the longitudinal velocity, but they are also correlated to the approaching direction. For this reason, the research activity is currently concentrated on the study of strategies which could modify the SAS behavior depending on the motion direction. Another research direction concerns the combination of the SAS with the trajectory scaling system proposed in [22]: The hybrid implementation of the two methods could potentially preserve the feasibility even with trajectories passing closer to singular points. Obviously, such combined strategy is only meaningful for applications in which the Cartesian speed of the tool frame can be modified. Finally, as stated in the paper, preliminary tests have shown that the manipulability index can validly be used in the SD and in the OM blocks at the cost of a higher computational burden. This possibility opens new perspectives since, in some cases, small changes of the tool frame orientation can also be used to manage the shoulder singularities, provided that they could be detected by means of a general purpose method.

References

[1] P. Chiacchio, S. Chiaverini, L. Sciavicco, and B. Siciliano, "Closed-loop inverse kinematics schemes for constrained redundant manipulators with task space augmentation and task priority strategy," *Int. J. on Robot. Res.*, vol. 10, no. 4, pp. 410–425, 1991.

[2] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators ," *IEEE Tran. on Rob. and Autom.*, vol. 13, no. 3, pp. 398–410, 1997.

[3] A. Gupta and M. O'Malley, "Design of a haptic arm exoskeleton for training and rehabilitation," *IEEE/ASME Trans. on Mech.*, vol. 11, no. 3, pp. 280–289, June 2006.

[4] J. Perry, J. Rosen, and S. Burns, "Upper-Limb Powered Exoskeleton Design," *IEEE/ASME Trans. on Mech.*, vol. 12, no. 4, pp. 408–417, Aug 2007.

[5] Y. Jung and J. Bae, "Kinematic Analysis of a 5-DOF Upper-Limb Exoskeleton With a Tilted and Vertically Translating Shoulder Joint," *IEEE/ASME Trans. on Mech.*, in press. DOI: 10.1109/TMECH.2014.2346767.

[6] X. Ding and C. Fang, "A Novel Method of Motion Planning for an Anthropomorphic Arm Based on Movement Primitives," *IEEE/ASME Trans. on Mech.*, vol. 18, no. 2, pp. 624–636, April 2013.

[7] F. Bourbonnais, P. Bigras, and I. Bonev, "Minimum-Time Trajectory Planning and Control of a Pick-and-Place Five-Bar Parallel Robot," *IEEE/ASME Trans. on Mech.*, vol. 20, no. 2, pp. 740–749, April 2015.

[8] L. Everett, J. Colson, and B. Mooring, "Automatic singularity avoidance using joint variations in robot task modification," *IEEE Rob. Autom. Mag.*, vol. 1, no. 3, pp. 13–19, Sept 1994.

[9] G. Marani, J. Kim, J. Yuh, and W. Chung, "A real-time approach for singularity avoidance in resolved motion rate control of robotic manipulators ," in *Proc. 2002 IEEE Int. Conf. on Rob. & Autom., ICRA 2002*, Washington, DC, May 2002, pp. 1973–1978.

[10] C. Qiu, Q. Cao, and S. Miao, "An on-line task modification method for singularity avoidance of robot manipulators," *Robotica*, vol. 27, pp. 539–546, 2009.

[11] P. From and J. Gravdahl, "A Real-Time Algorithm for Determining the Optimal Paint Gun Orientation in Spray Paint Applications," *IEEE Trans. on Autom. Science and Eng.*, vol. 7, no. 4, pp. 803–816, 2010.

[12] P. From, J. Gunnar, and J. Gravdahl, "Optimal Paint Gun Orientation in Spray Paint Applications - Experimental Results," *IEEE Trans. on Autom. Science and Eng.*, vol. 8, no. 2, pp. 438–442, Apr. 2011.

[13] O. Dahl and L. Nielsen, "Torque-limited path following by online trajectory time scaling," *IEEE Trans Robot Automat*, vol. 6, no. 5, pp. 554–561, 1990.

[14] O. Dahl, "Path-constrained robot control with limited torques-experimental evaluation," *IEEE Trans Robot Automat*, vol. 10, no. 5, pp. 658–669, 1994.

[15] D. Constantinescu and E. A. Croft, "Smooth and time-optimal trajectory planning for industrial manipulators along specified paths," *J. Robot. Syst.*, vol. 17, no. 5, pp. 233–249, 2000.

[16] J. Moreno-Valenzuela and E. Oronzco-Manríquez, "A new approach to motion control of torque-constrained manipulators by using time-scaling of reference trajectories," *J Mech Sci Technol*, vol. 23, no. 12, pp. 3221–3231, Dec. 2009.

[17] I. Pietsch, M. Krefft, O. Becker, C. Bier, and J. Hesselbach, "How to reach the dynamic limits of parallel robots? An autonomous control approach," *IEEE Trans. on Autom. Science and Eng.*, vol. 2, no. 4, pp. 369–380, Oct. 2005.

[18] C. Guarino Lo Bianco and O. Gerelli, "Trajectory scaling for a manipulator inverse dynamics control subject to generalized force derivative constraints," in *2009 IEEE/RSJ Int. Conf. on Intell. Robots and Systems, IROS 2009*, St. Louis, MO, Oct. 2009, pp. 5749–5754.

[19] ——, "Online trajectory scaling for manipulators subject to high-order kinematic and dynamic constraints," *IEEE Trans. on Rob.*, vol. 27, no. 6, pp. 1144–1152, Dec. 2011.

[20] J. E. Bobrow, S. Dubowsky, and J. S. Gibson, "Time-optimal control of robotics manipulators along specified paths," *Int. J. Robot. Res.*, vol. 4, no. 3, pp. 3–17, 1985.

[21] A. Bemporad, T.-J. Tarn, and N. Xi, "Predictive path parameterization for constrained robot control ," *IEEE Trans. on Contr. Sys. Tech.*, vol. 7, no. 6, pp. 648–656, Nov. 1999.

[22] C. Guarino Lo Bianco and F. Ghilardelli, "Real-Time Planner in the Operational Space for the Automatic Handling of Kinematic Constraints," *IEEE Trans. on Autom. Sci. and Eng.*, vol. 11, no. 3, pp. 730–739, 2014.

[23] F. Lange and M. Suppa, "Predictive path-accurate scaling of a sensor-based defined trajectory," in *IEEE Int. Conf. on Rob. and Autom, ICRA'14*, May 2014, pp. 754–759.

[24] P. Hertling, L. Hog, R. Larsen, J. Perram, and H. Petersen, "Task curve planning for painting robots. I. Process modeling and calibration," *IEEE Trans. on Rob. and Autom.*, vol. 12, no. 2, pp. 324–330, Apr 1996.

[25] R. Ramabhadran and J. K. Antonio, "Fast solution techniques for a class of optimal trajectory planning problems with applications to automated spray coating," *IEEE Trans. on rob. and aut.*, vol. 13, no. 4, pp. 519–530, 1997.

[26] V. Potkonjaka, G. S. Dordević, D. Kostić, and M. Rašić, "Dynamics of anthropomorphic painting robot: Quality analysis and cost reduction," *Robotics and Autonomous Systems*, vol. 32, no. 1, pp. 17–38, 2000.

[27] D. Conner, A. Greenfield, P. Atkar, A. Rizzi, and H. Choset, "Paint deposition modeling for trajectory planning on automotive surfaces," *IEEE Trans. on Autom. Sci. and Eng.*, vol. 2, no. 4, pp. 381–392, Oct 2005.

[28] J. Peng, Q. Chen, J. Lu, J. Jin, and C. van Luttervelt, "Real time optimization of robotic arc welding based on machine vision and neural networks," in *The 1998 IEEE Int. Conf. on Industrial Electronics, Control, and Instrumentation, IECON'98*, 1998, pp. 1279–1283.

[29] W. Tillmann, E. Vogli, and B. Krebs, "Influence of the spray angle on the characteristics of atmospheric plasma sprayed hard material based coatings," *J. of Ther. Spray Tech.*, vol. 17, no. 5-6, pp. 948–955, 2008.

[30] G. Schreiber, M. Otter, and G. Hirzinger, "Solving the singularity problem of non-redundant manipulators by constraint optimization," in *IEEE/RSJ Int. Conf. on Intel. Rob. and Sys., IROS'99*, vol. 3, 1999, pp. 1482–1488.

[31] W. Decre, H. Bruyninckx, and J. De Schutter, "Extending the iTaSC constraint-based robot task specification framework to time-independent trajectories and user-configurable task horizons," in *IEEE Int. Conf. on Rob. and Autom., ICRA'13*, May 2013, pp. 1941–1948.

[32] Y. Huang, Y. Yong, Y. Chiba, T. Arai, T. Ueyama, and J. Ota, "Kinematic control with singularity avoidance for teaching-playback robot manipulator system," *IEEE Trans. on Autom. Sci and Eng.*, In Press, DOI: 10.1109/TASE.2015.2392095.

[33] C. Guarino Lo Bianco and F. Wahl, "A novel second order filter for the real-time trajectory scaling," in *IEEE Int. Conf. on Rob. and Autom., ICRA 2011*, Shanghai, China, May 2011, pp. 5813–5818.

[34] C. Guarino Lo Bianco, F. Ghilardelli, and D. Kubus, "Experimental validation of a time scaling algorithm for robotics systems," in *IEEE Int. Conf. on Robot. and Bio., ROBIO 2012*, Dec 2012, pp. 2044–2049.

[35] R. Vanderbei, *Linear Programming: Foundations and Extensions (fourth edition)*, ser. International Series in Operations Research & Management Science. New York: Springer, 2014.

Fabio Ghilardelli received the M.S. degree with honors in computer engineering and the Ph.D. degree in information technology from the University of Parma, Italy, in 2011 and 2015, respectively. Currently, he is employed in COMAU S.p.a (FCA Group), Torino, Italy in the Robotic Business Unit as a software engineer. His main research topics are represented by real-time motion planning, real-time managing of manipulators' singularities, motion and control software architectures.

Corrado Guarino Lo Bianco graduated with honors in Electronic Engineering and received the Ph.D. degree in Control System Engineering from the University of Bologna, Italy, in 1989 and 1994, respectively. Currently, he is with the Dipartimento di Ingegneria dell'Informazione of the University of Parma as Associate Professor on Industrial Robotics. He is involved in researches concerning mobile and industrial robotics. In particular, he is mainly interested in topics concerning the smooth and optimal trajectory generation, robot kinematics and dynamics, robot control.

Marco Locatelli is Full Professor of Operations Research by the Department of Ingegneria dell'Informazione at the University of Parma. His research interests are mainly in the field of nonlinear optimization and, more specifically, he is interested in the theoretical and practical aspects of global optimization. Together with Prof. F. Schoen, he has recently published the monograph Global Optimization: Theory, Algorithms, and Applications in the MOS-SIAM Series on Optimization. He is in the editorial board of the Journal of Global Optimization and of Computational Optimization and Applications.