JOINT SPACE POINT-TO-POINT MOTION PLANNING FOR ROBOTS. AN INDUSTRIAL IMPLEMENTATION

Gianluca Antonelli * **Stefano Chiaverini** *
Marco Palladino ** **Gian Paolo Gerio** *** **Gerardo Renga** ***

* *DAEIMI, Università degli Studi di Cassino*
Via G. Di Biasio 43, 03043 Cassino (FR), Italy
{antonelli,chiaverini}@unicas.it
** *Consorzio C.R.E.A.T.E., Università degli Studi di Reggio Calabria*
Via Graziella, Loc. Feo di Vito, 89128 Reggio Calabria, Italy.
mpall@katamail.com
*** *Comau S.p.A.*
Strada Orbassano 20/22, 10092 Beinasco (TO), Italy.
{gian_paolo.gerio, grenga}@comau.com

Abstract: A path-following algorithm for industrial robots is presented in this paper. The algorithm generates off-line a joint space point-to-point trajectory that, by exploiting knowledge of the dynamic model, takes into account the actuators' torque limits while preserving the geometric path. Due to the software characteristic of the specific industrial architecture, the trajectories adopt a trapezoidal velocity profile. The algorithm has been designed, implemented and extensively tested on a Comau SMART H4 robot, a closed-chain six-degree-of-freedom industrial manipulator.

Keywords: Path Planning, Industrial Robots

1. INTRODUCTION

Minimization of the travelling time and preservation of the geometric path are critic constraints when programming the robot's trajectory. In fact, in case of a too demanding trajectory, saturation of the torques generally results in a path error that may cause serious damages. On the other side, programming of a slow trajectory is safe but does not exploit the machine at its maximum performance. These counteracting requirements naturally pose a minimum-time path-following motion control problem.

A wide literature on the minimum-time path-following control problem exists. In off-line implementations the trajectory is computed before the movement starts, whereas in on-line implementations the trajectory is modified during the robot motion based on the current sensor readings.

Among the off-line methods, in Bobrow *et al.* (1985) an iterative procedure is proposed. A substantially similar solution is given by Shin and Mckay (1985). Slotine and yang (1989) describe a more efficient algorithm. The trajectory obtained from the previous algorithm is the optimal one as it has been demonstrated by Chen and Desrochers (1989). An on-line closed-loop approach is devised in Dahl and Nielsen (1990) and the experimental results are presented in Dahl (1994), where a secondary controller modifies the nominal trajectory during the motion to cope with uncertainties about dynamic model and torque limits. The output of these methods are bang-bang-like trajectories that achieve the limit torque at least in one joint at every time instant thus allowing to exploit the actuators at their maximum performance.

One common drawback of both off-line and on-line above methods is that the obtained velocity profile of the optimal trajectory does not have a predictable behavior, which is not always allowable in industrial tasks, e.g., glue spreading or painting.

In this paper, with reference to the experimental setting constituted by Comau Robotics manipulators, an off-

line path planning algorithm is developed that takes into account the strict requirements of many industrial applications. In particular, the trajectory has to be composed of linear-parabolic polynomial sequences, i.e., constant speed and constant acceleration segments; this approach combines simplicity of the reference signal construction with sub-optimality of the trapezoidal velocity profile. According to the guidelines in Hollerbach (1980), the proposed algorithm does not guarantee that one joint reaches its torque limit at every instant, as in Dahl and Nielsen (1990); however, at least one actuator reaches the torque limit during the trajectory. The computational burden is kept limited by computing the dynamic model only in a very small number of *significant* points along the trajectory, moreover the solution is not iterative. In Antonelli *et al.* (2004, 2004b) the extension of the fly joint motion and the point-to-point and fly Cartesian motion are proposed.

2. BACKGROUND

The joint vector is represented by

$$\boldsymbol{q} = [\, q_1 \quad \cdots \quad q_n \,]^{\mathrm{T}}, \tag{1}$$

where $\boldsymbol{q} \in \mathbb{R}^n$, being n the number of manipulator's joints. The rigid body dynamic model of the manipulator in the joint space can be written as

$$\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{F}_v\dot{\boldsymbol{q}} + \boldsymbol{F}_s\mathbf{sign}(\dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}\,, \tag{2}$$

where $\boldsymbol{B}(\boldsymbol{q})$ is the $n \times n$ inertia matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ is the vector of Coriolis and centrifugal forces, $\boldsymbol{F}_v\dot{\boldsymbol{q}}$ is the vector of viscous friction, $\boldsymbol{F}_s\mathbf{sign}(\dot{\boldsymbol{q}})$ is the vector of static friction, $\boldsymbol{g}(\boldsymbol{q})$ is the vector of gravitational forces and $\boldsymbol{\tau}$ is the vector of joint torques, being all the vectors of dimension n. In order to identify the dynamic parameters, different techniques can be implemented starting, e.g., from the works of Guatier and Khalil (1988), Presse and Gautier (1993), Swevers *et al.* (1997); for the Comau's robots a systematic procedure has recently been implemented and validated in the industrial environment by Antonelli *et al.* (1999).

Starting from the well known results of dynamic scaling of robot trajectories (see, e.g., Sciavicco and Siciliano (2000)), in the following some useful equations are computed including the viscous and static friction contributions. Let define the vector $\overline{\boldsymbol{q}}(r(t))$ that verifies the equation

$$\boldsymbol{q}(t) = \overline{\boldsymbol{q}}\big(r(t)\big)\,, \tag{3}$$

where $r(t)$ is a scalar strictly-increasing time function that satisfies $r(0) = 0$ and $r(t_f) = \overline{t}_f$. Choosing a linear scaling function $r(t)$, it is $r(t) = ct$ with c positive constant; it is worth noticing that $c > 1$ implies a longer duration of the trajectory. Following the guidelines in Sciavicco and Siciliano (2000), it can be obtained that

$$\overline{\boldsymbol{\tau}}(ct) = \frac{\boldsymbol{\tau}_s(t)}{c^2} + \frac{\boldsymbol{\tau}_v(t)}{c} + \boldsymbol{g}\big(\boldsymbol{q}(t)\big)\,, \tag{4}$$

where the vector $\boldsymbol{\tau}_s(t)$ contains the inertia and Coriolis contributions and the vector $\boldsymbol{\tau}_v(t)$ the friction effects.

3. DYNAMIC CONSIDERATIONS FOR TRAPEZOIDAL JOINT VELOCITY PROFILE

Given an initial configuration $\boldsymbol{q}_i$ and a final configuration $\boldsymbol{q}_f$ in the joint space, the trajectory is obtained by resorting to a parameter function $s \in [0,1]$ with a trapezoidal velocity profile . The velocity profile has three different phases, defined in the following *start*, *cruise*, and *arrival*; it can have different acceleration values in the start and in the arrival phase.

From s, the position vector for every joint is generated as follows:

$$\boldsymbol{q} = (\boldsymbol{q}_f - \boldsymbol{q}_i)s + \boldsymbol{q}_i\,,$$

and the velocity and acceleration vectors are generated correspondingly from $\dot{s}$, $\ddot{s}$.

It is possible to notice that, given the trapezoidal velocity profile, there are some points where the possibility to reach the maximum joint torque is high. These points are denoted in the following as *significant points* and are represented in Figure 1.

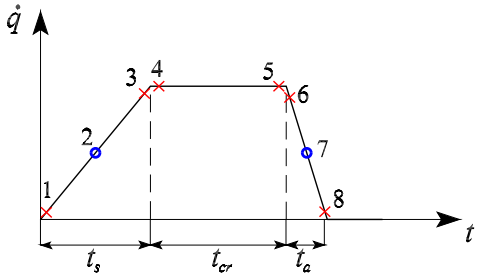

Fig. 1. *Significant points* for a trapezoidal velocity profile in the joint space.

A first set of significant points are found in correspondence of the changes of phase, where there is a discontinuity in the acceleration. These significant points, marked by a $\times$ sign in Figure 1, are:

- *Point 1*: immediately after the starting time instant. Since the velocity of all the joints is close to zero, the predominant dynamic contributions are inertia and gravity, leading to:

$$\boldsymbol{\tau} \approx \boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q})\,. \tag{5}$$

- *Point 3*: immediately before the switch between the start and the cruise phases. At Point 3, every dynamic contribution is significant in forming the total torque; in particular, the viscous friction reaches its maximum value. Since the velocity of all the joints is close to the cruise value, the torque can be approximated as:

$$\begin{aligned}\boldsymbol{\tau} \approx \boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}_{cr})\dot{\boldsymbol{q}}_{cr} + \boldsymbol{F}_v\dot{\boldsymbol{q}}_{cr} \\ + \boldsymbol{F}_s\mathbf{sign}(\dot{\boldsymbol{q}}_{cr}) + \boldsymbol{g}(\boldsymbol{q})\,.\end{aligned} \tag{6}$$

- *Point 4*: immediately after the switch between the start and the cruise phases. At Point 4, the acceleration is zero and the velocity is at the cruise value; then, the torque can be computed as:

$$\boldsymbol{\tau} \approx \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}_{cr})\dot{\boldsymbol{q}}_{cr} + \boldsymbol{F}_v\dot{\boldsymbol{q}}_{cr} + \boldsymbol{F}_s\mathbf{sgn}(\dot{\boldsymbol{q}}_{cr}) + \boldsymbol{g}(\boldsymbol{q})\,. \quad (7)$$

- *Point 5*: immediately before the switch between the cruise and the arrival phases. At Point 5 the inertial contribution is zero and the approximation (7) holds.
- *Point 6*: immediately after the switch between the cruise and the arrival phases. At Point 6 the approximation (6) holds.
- *Point 8*: immediately before the final time instant. Since the robot is going to stop the motion, the predominant dynamic contributions are inertia and gravity, and the approximation (5) holds.

Another set of significant points are in the middle of the start and the arrival phase, namely, *Points 2 and 7*, marked by a $\circ$ sign in Figure 1. In fact, very long trajectories most described at maximum cruise velocity exhibit the highest torque demand at the relatively short acceleration/deceleration transient. It must be remarked, however, that it is not common to ask for a large movement at the maximum velocity with path tracking requirement.

At Points 2 and 7 the joint velocity is half its cruise value; then, the following approximation holds:

$$\begin{aligned}\boldsymbol{\tau} \approx \boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \frac{\dot{\boldsymbol{q}}_{cr}}{2})\frac{\dot{\boldsymbol{q}}_{cr}}{2} + \boldsymbol{F}_v\frac{\dot{\boldsymbol{q}}_{cr}}{2} \\ +\boldsymbol{F}_s\mathbf{sign}(\dot{\boldsymbol{q}}_{cr}) + \boldsymbol{g}(\boldsymbol{q})\,, \end{aligned} \quad (8)$$

Moreover, it must be noticed that each motor, together with a torque limit, also has to respect a velocity limit. The value of this limit usually allows the robot to execute long movements at a cruise velocity without exceeding the torque limit. For this reason there is no need to add any significant point between Points 4 and 5.

The above concept of significant points can be exploited to reduce the computational burden of the planning algorithm by computing the dynamic model only in a small set of points rather than along the whole trajectory. As a matter of fact, intensive simulations using the 6-dof Comau SMART H4 dynamic model and experiments run on the real robot, confirm validity of these intuitive dynamic observations.

4. PROPOSED PLANNING ALGORITHM

In accordance with the Comau's control architecture several constraints need to be satisfied, namely:

- off-line implementation: the algorithm has to generate the trajectory before the movement starts;
- low computational burden: to reduce the time that lasts from the trajectory data input and the robot motion; avoid iterative solutions;
- trapezoidal velocity profile.

4.1 Joint space point-to-point motion

The proposed algorithm is divided in two different scaling techniques that output two different final times both compatible with the torques limits, the smaller is then used to compute the trajectory. The two techniques, detailed in the following, share most of the computations, thus avoiding to increase the computational burden. In particular, they both need a trial trajectory computed as detailed in the following.

Trial trajectory The trial trajectory is that used as input for the two algorithms detailed in the next Subsections. Since its accuracy can affect the maximum velocity scaling technique, the trial trajectory needs to be carefully designed. Without entering in the details, the trial trajectory has a trapezoidal velocity profile whose initial and final accelerations are computed by resorting to the equations:

$$\ddot{\boldsymbol{q}}_i = \mathrm{diag}\left\{\frac{1}{B_{1,1}(\boldsymbol{q}_i)}, \ldots, \frac{1}{B_{n,n}(\boldsymbol{q}_i)}\right\}\left(\boldsymbol{\tau}_{max} - \boldsymbol{g}(\boldsymbol{q}_i)\right),$$

$$\ddot{\boldsymbol{q}}_f = \mathrm{diag}\left\{\frac{1}{B_{1,1}(\boldsymbol{q}_f)}, \ldots, \frac{1}{B_{n,n}(\boldsymbol{q}_f)}\right\}\left(\boldsymbol{\tau}_{max} - \boldsymbol{g}(\boldsymbol{q}_f)\right),$$

where $B_{j,j}(\cdot)$ is the j-th diagonal element of the inertia matrix; therefore, these expressions are a cost-effective approximation of the inversion of the whole matrix $\boldsymbol{B}(\cdot)$.

Maximum acceleration scaling The basic idea of the maximum acceleration scaling is to compute the maximum acceleration achievable in the start and arrival phases. Since the overall joint displacement has been fixed, the cruise velocity is also determined.

Firstly, the trial trajectory is planned. For the i-th joint, at each time instant t_s where a significant point is attained, by computing the dynamic effects (see Section 3) a scaling coefficient can be obtained as:

$$\begin{aligned} c_{i,s} = &-\frac{\tau_{v,i}(t_s)}{2(g_i(\boldsymbol{q}(t_s)) - \tau_{max,i})} \\ &\pm\sqrt{\left(\frac{\tau_{v,i}(t_s)}{2(g_i(\boldsymbol{q}(t_s)) - \tau_{max,i})}\right)^2 - \frac{\tau_{s,i}(t_s)}{g_i(\boldsymbol{q}(t_s)) - \tau_{max,i}}} \\ &\text{for} \quad i = 1, \ldots, n\,, \\ &\qquad s = 1, \ldots, 8\,, \end{aligned} \quad (9)$$

where $\tau_{max,i}$ is the torque limit of the joint i; notice that equation (9) represents two positive real solutions, of which the smallest is considered. The overall scaling factor c is selected by considering the most conservative $c_{i,s}$ among those obtained for all the joints at all the significant points.

Remarkably, it is not necessary to re-plan the trajectory since the time vector is only scaled by the c factor; the new velocities are simply divided by the same factor.

Since the start phase is scaled independently from the arrival phase the maximum torque is achieved in, at least, two points of the trajectory. On the other side, the cruise velocity is not optimized and it can result in too conservative values (see, e.g., Figure 2). It is intuitively clear that for very large movement this approach may results in a *slow* overall movement.

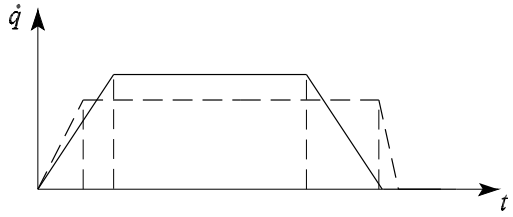

Fig. 2. Maximum acceleration scaling; the acceleration are optimized and the resulting cruise velocity is thus obtained. The dashed line represents a case with $c > 1$.

Is it then possible to develop a scaling that preserves the maximum cruise velocity by accepting in the start and arrival phases an acceleration lower than the maximum.

Maximum velocity scaling The maximum velocity scaling technique plans a trapezoidal velocity profile by first imposing that the cruise velocity is the maximum achievable by the motors at the joints and then finding congruent acceleration values for the start and arrival phases. Since the cruise phase satisfies the torque and velocity limits by construction, only the start and arrival phases must undergo proper time scaling to ensure feasible torque values; remarkably, the two phases can be independently scaled leading to different scaling coefficients. A graphical representation of this process is given in Figure 3, while the mathematical development is detailed in the following.

First, a trial trajectory is planned setting the cruise velocity at the maximum achievable motor speed; to find a proper start acceleration value then proceed as follows.

Let define as $\boldsymbol{q}_c$ the joint configuration at which the peak torque value of the start phase of the trial trajectory is attained. As by their definition, this is likely to be in the neighborhood of one of the significant points 1–3; therefore, we only need to compute the joint torques at the three points by using the approximations (5), (8), and (6), respectively, and select $\boldsymbol{q}_c$ accordingly.

After the time scaling to be yet applied, the same significant point corresponding to $\boldsymbol{q}_c$ will correspond to the different joint configuration $\overline{\boldsymbol{q}}_c$. One problem to be solved is that, in order to compute the scaling factor c^*, the value of $\overline{\boldsymbol{q}}_c$ is needed which in turn requires c^* to be known. To overcome this problem, it is suggested to assume

$$\overline{\boldsymbol{q}}_c \approx \boldsymbol{q}_c \,, \tag{10}$$

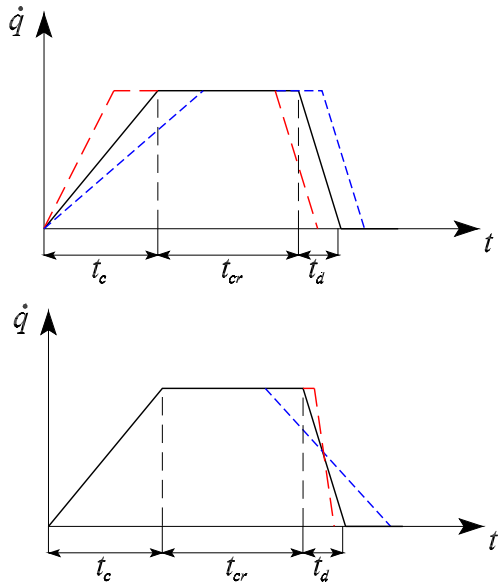

Fig. 3. Maximum velocity scaling. The cruise velocity is imposed and the acceleration/deceleration are computed accordingly.

meaning that the time scaling will not affect much the configuration at which the peak torque is attained. A consequence of (10) is therefore that the trial trajectory is assumed to be close to the final one, that results in a valid c^* being close to unity. Of course, the approximation (10) is very accurate in more general conditions when the maximum torque is reached at the significant point 1 (or Point 8 when dealing with the arrival phase).

Moreover, in view of the linear time scaling of the start phase, a fixed cruise velocity results in

$$\dot{\overline{\boldsymbol{q}}}_c = \dot{\boldsymbol{q}}_c \,; \tag{11}$$

in particular, this implies that the friction effects are unchanged at corresponding configurations before and after time scaling, i.e.,

$$\boldsymbol{\tau}_v(t) = \overline{\boldsymbol{\tau}}_v(c^* t) \,. \tag{12}$$

From a practical point of view, the approximation (10) can be considered valid as long as the resulting error in computing the torque is sufficiently small. This in turn depends on the validity of the following relationships:

$$\|\boldsymbol{B}(\boldsymbol{q}_c) - \boldsymbol{B}(\overline{\boldsymbol{q}}_c)\| \approx 0 \,, \tag{13}$$

$$\|\boldsymbol{g}(\boldsymbol{q}_c) - \boldsymbol{g}(\overline{\boldsymbol{q}}_c)\| \approx 0 \,, \tag{14}$$

$$\|\boldsymbol{C}(\boldsymbol{q}_c, \dot{\boldsymbol{q}}_c) - \boldsymbol{C}(\overline{\boldsymbol{q}}_c, \dot{\boldsymbol{q}}_c)\| \approx 0 \,, \tag{15}$$

that relax the need for accurate holding of (10) in the much frequent cases in which the dependence of the dynamic terms from the configuration is somewhat weak.

At this point, the scaling factor c^* for the start phase can be derived. In fact, from the equation

$$\overline{\boldsymbol{\tau}}(c^* t) = \frac{\boldsymbol{B}(\boldsymbol{q}_c)\ddot{\boldsymbol{q}}_c}{c^{*2}} + \boldsymbol{C}(\boldsymbol{q}_c, \dot{\boldsymbol{q}}_c)\dot{\boldsymbol{q}}_c + \boldsymbol{\tau}_v(\dot{\boldsymbol{q}}_c) + \boldsymbol{g}(\boldsymbol{q}_c) \,,$$

for each joint i the torque limit is consistent with the factor c_i^* satisfying the relationship

$$\overline{\tau}_{max,i} = \frac{(\boldsymbol{B}(\boldsymbol{q}_c)\ddot{\boldsymbol{q}}_c)_i}{c_i^{*2}} + (\boldsymbol{C}(\boldsymbol{q}_c, \dot{\boldsymbol{q}}_c)\dot{\boldsymbol{q}}_c)_i + \tau_{v,i}(\dot{\boldsymbol{q}}_c) + g_i(\boldsymbol{q}_c) \tag{16}$$

This easily gives

$$c_i^* = \sqrt{\frac{(\boldsymbol{B}(\boldsymbol{q}_c)\ddot{\boldsymbol{q}}_c)_i}{\tau_{max,i} - \tau_{v,i}(\dot{\boldsymbol{q}}_c) - (\boldsymbol{C}(\boldsymbol{q}_c, \dot{\boldsymbol{q}}_c)\dot{\boldsymbol{q}}_c)_i - g_i(\boldsymbol{q}_c)}} \quad \text{for} \quad i = 1, \ldots, n\,. \tag{17}$$

Among the n factors computed, the value of c^* is taken as the most conservative c_i^*.

The arrival phase can be handled following the same guidelines by working instead on the significant points 6–8.

Procedure The steps to be implemented are summarized in the following:

(1) planning of the trial trajectory;
(2) computation of the dynamic contributions at the 8 significant points;
(3) implementation of the maximum acceleration technique;
(4) implementation of the maximum velocity technique;
(5) choice of the trajectory corresponding to the smallest t_f obtained.

It must be remarked that the computational load of the two methods is not heavy since they share most of the equations; in particular, the computation of the dynamic model at the significant points is the same.

Notice that the maximum velocity method is suited to large movements, while the maximum acceleration method suits short ones.

There are different sources of approximations to be taken into account when computing the scaled trajectory: the accuracy of the dynamic model; the accuracy of (10) and (13)–(15); and the presence of the feedback motion control in the on-line implementation. These inaccuracies are taken into account simply considering a more conservative value for the torques' limits. Intensive simulations and on-field tests have shown the effectiveness of the above presented algorithm.

4.2 Comparison between the proposed and an on-line algorithm

It is expected that on-line, full-model based motion planners show better performance than the off-line proposed method based on reduced model information. In order to verify the differences with the proposed approach, a comparison with the algorithm proposed in Dahl and Nielsen (1990) has been implemented. In the simulation of the latter the following hold:

- the input trajectories have trapezoidal velocity profile and they are not obtained by means of minimum time optimization in $(s, \dot{s})$ space;
- the trajectories are planned in the joint space and include the motor speed limits;
- the joint servos have not been considered.

The results of the simulations show that the travelling times in joint space obtained from the on-line algorithm are about 20% shorter than the ones obtained with the proposed algorithm; however, the longer is the cruise phase, the smaller is the percentage time improvement. From a computational point of view, notice that, in case of a $2\,\text{ms}$ sampling time, for a trajectory of $1\,\text{s}$ duration the dynamic model is computed 500 times by the on-line algorithm; remarkably, the proposed approach requires computation of the dynamic model only 8 times. Therefore, one main advantage of the proposed algorithm relies in a significant computational lightness; moreover, it can be easily included into the preexisting planning algorithm.

5. CASE STUDIES

In this Section the implementation of the proposed technique on the Comau Smart H4 robot is presented. The algorithm is currently under implementation for all the robots manufactured by Comau. The SMART H-4 is a 6-dof robot manufactured by Comau shown in Figures 4 and 5. It exhibits a closed-chain structure and a non-spherical wrist that allows to insert the power cables into the wrist link.

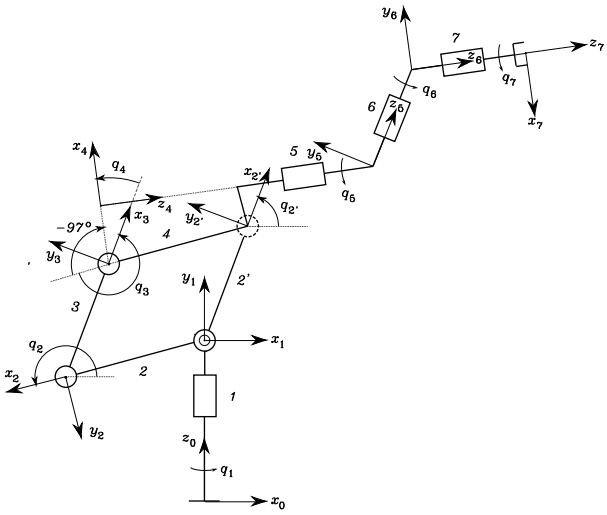

Fig. 4. Kinematic sketch of the Comau Smart H-4.

The motor torque limits, expressed in Nm, are given by

$$\boldsymbol{\tau}_{max} = [\,9844 \quad 9494 \quad 9494 \quad 3890 \quad 3890 \quad 2502\,]^{\mathrm{T}}\,.$$

The effectiveness of the proposed motion planner has been verified in a large number of testing trajectories. These have been selected so as to represent all the possible working conditions in which the robot will operate; in particular, some motions have been extracted from a spot welding operation currently used during car

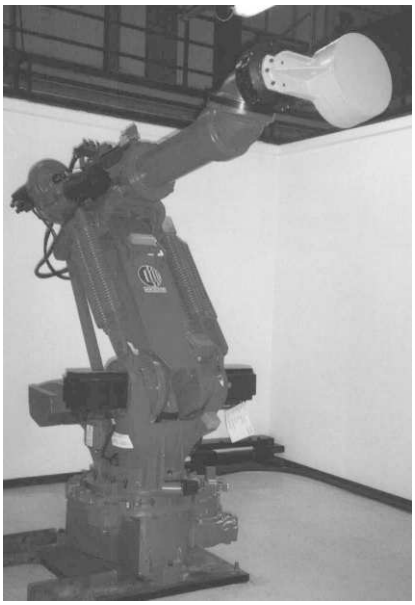

Fig. 5. The Comau Smart H4.

assembling. The full set of testing trajectories include large and small movements, with and without a load of 180 kg, starting from different configurations that are representative of the workspace. Notice that the presence of a payload does not represent an additional computational charge; in fact, the structure of the dynamic model is the same with and without load, the only difference is in the use of a different set of identified dynamic parameters. The overall performance of the proposed algorithm always improved the previous planner; in particular, the improvement obtained in short movements is significant (about 10–20%). This can be explained by observing that, with respect to the previous planner, the proposed algorithm gives a good improvement on the start and arrival phases, whereas they both achieve the maximum velocity in the cruise phase. Figure 6 presents the time history of the velocities and the torques of one planned motion.

6. CONCLUSIONS

A path-following algorithm for industrial robots has been presented in this paper. Due to the given industrial constraints, the algorithm is implemented off-line and may use knowledge of the dynamic model under severe computational limits. To comply with the software characteristic of the specific industrial architecture, the trajectories adopt a trapezoidal velocity profile. The algorithm has been developed and extensively tested on a Comau SMART H-4, a closed-chain six-degree-of-freedom industrial manipulator. The obtained results show improved performance with respect to the currently implemented planner.

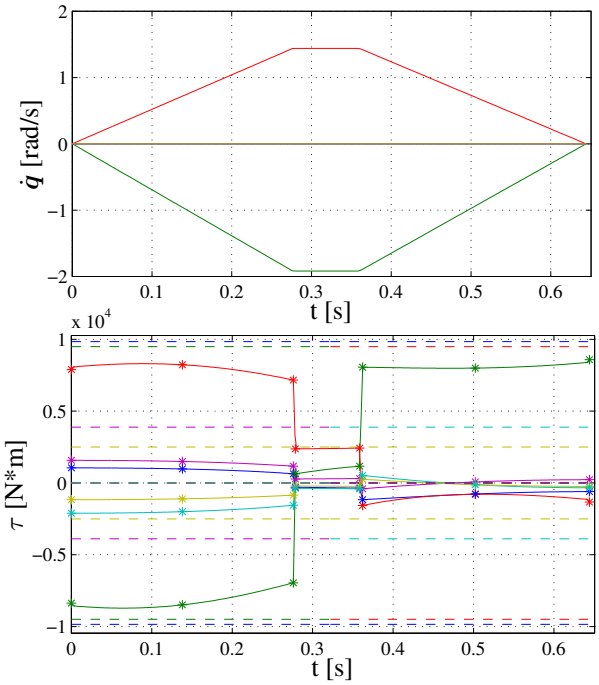

Fig. 6. Velocity and torque profiles for one planned motion.

REFERENCES

Antonelli G., Caccavale C., Chiacchio P., "A systematic procedure for the identification of dynamic parameters of robot manipulators", *Robotica*, vol. 17, pp. 427–435, 1999.

Antonelli G., Chiaverini C., Palladino M., Gerio G., Renga G., "Cartesian Space Motion Planning for Robots. An Industrial Implementation", *4th Int. Workshop on Robot Motion and Control*, Puszczykowo, PL, pp 279–284, 2004.

Antonelli G., Chiaverini C., Palladino M., Gerio G., Renga G., "Joint Space Fly Motion Planning for Robots. An Industrial Implementation", *12th Mediterranean Conference on Control and Automation*, Kusadasi, T, 2004b.

Bobrow J.E., Dubowsky S., and Gibson J.S., "Time-optimal control of robotic manipulators along specified paths", *The Int. J. of Robotics Research*, vol. 4, n. 3, pp. 3–17, 1985.

Chen Y. and Desrochers A.A., "Structure of mimimum-time control law for robotic manipulator with constrained path", *Proc. 1989 IEEE Int. Conf. on Robotics and Automation*, Scottsdale, AZ, pp. 971–976, 1989.

Gautier M. and Khalil W., "On the identification of inertial parameters of robots," *Proc. 27th IEEE Conf. on Decision and Control*, pp. 2264–2269, Austin, TX, 1988.

Dahl O., Nielsen L., "Torque-limited path following by on-line trajectory time scaling", *IEEE Trans. on Robotics and Automation*, vol. 6, pp. 554–561, October 1990.

Dahl O., "Path-constrained robot control with limited torque—Experimental Evaluation", *IEEE Trans. on Robotics and Automation*, vol. 10, pp. 658–669, October 1994.

Hollerbach J.M., "Dynamic scaling of manipulator trajectories", *Trans. ASME—J. of Dynamic Systems Measurement and Control*, vol. 106, pp. 102–106, 1984.

Presse C. and Gautier M., "New criteria of exciting trajectories for robot identification," *Proc. 1993 IEEE Int. Conf. on Robotics and Automation*, Atlanta, GA, pp. 907–912, 1993.

Sciavicco L., Siciliano B., *Modeling and Control of Robot Manipulators*, Springer-Verlag, London, UK, 2000.

Shin K.G. and McKay N.D., "Minimum time control of robotic manipulators along specified paths", *IEEE Trans. on Automatic Control*, vol. 30, pp. 531–541, 1985.

Slotine J.-J.E., Yang J.-J.E. "Improving the efficiency of time-optimal path-following algorithms", *IEEE Trans. on Robotics and Automation*, vol. 5, pp. 118–124, 1989.

Swevers J., Ganseman C., Tükel D.B., De Schutter J., and Van Brussel H., "Optimal robot excitation and identification", *IEEE Trans. on Robotics and Automation*, vol. 13, pp. 730–740, 1997.