Online Trajectory Planner with Constraints on Velocity, Acceleration and Torque

Luigi Biagiotti and Roberto Zanasi
DII, University of Modena and Reggio Emilia
Modena, Italy
{*luigi.biagiotti, roberto.zanasi*}*@unimore.it*

Abstract—**A filter for trajectories smoothing is presented. The filter provides an output reproducing the input trajectory if this is compliant with given constraints on the velocity, the acceleration and the torque required for tracking it. Otherwise, the filter approximates the input profile guaranteeing that the output trajectory satisfies all the kinematic and dynamic limits. The tracking of the input signal is optimal in the sense that at each time instant the limit value of one among velocity, acceleration and torque is reached. The filter, based on a variable structure controller, is designed in the continuous-time domain but can be implemented by discretization as a sampled system. It can therefore used in mechatronic and robotic applications driven by digital controllers in order to filter trajectories planned without considering the above mentioned constraints or to generate in real-time smooth trajectories by simply providing basic inputs such as step or ramp functions.**

I. INTRODUCTION

In recent years, because of the need of more flexibility and the increase of the automation in a number of industrial processes, online trajectories planning has become a very important issues for a number of applications. As a matter of fact, in may cases it is impossible to compute offline the motion profiles to be tracked by mechatronic and robotic systems. For instance, when the required motions are not repetitive or not predictable in advance it is necessary to generate proper motion profiles, generally starting from basic input reference commands such as step or ramp functions. Consider the case of wheeled mobile robots [1], [2] or unmanned air vehicles [3], for which the target is generally given or modified in real-time according to the task to be performed, the presence of obstacles in the workspace, etc. Online trajectory generation is a relevant problem also in the field of automatic machines and industrial robots, e.g. when command inputs are directly provided by the human operator or when it is necessary to synchronize different subsystems. A typical example is given by a robot which must track objects (to be grasped or machined) on a conveyor. Since objects may arrive unordered and with a varying rate it is necessary to generate online the trajectory in order to correctly track the objects of interest [4]. In all the above mentioned applications, it is necessary to plan trajectories compliant with the kinematic constraints (i.e. limits of velocity, acceleration, jerk, etc.) and dynamic constraints (i.e. torque and possibly torque-derivative bounds) imposed by the actuation system, by the mechanical structure of the plant and by the specific application, starting from

the rough commands provided by the operator or more often by a supervisory system. Moreover, the trajectories must be generally optimized according to a some criterion, usually the minimization of the execution time. For this purpose, several approaches are available in the literature: in [5], [6], [7] trajectory filters dealing with the kinematic constraints for one-dimensional profiles are proposed, in [8] the trajectories generation for robotic manipulators with bounded torques and torque-derivatives is considered.

The proposed trajectory generator belongs to the same research line of the filters presented in [7], [9], [10] and [11], that are all based on the phase plane analysis [12] and the variable structure control technique [13], both in the continuous and the discrete-time domain. In this case, besides the constraints on velocity and acceleration, also the torque necessary to track the trajectory is taken into account. This issues is very important when the best trajectory feasible with a given actuation system (characterized by precise torque limits) is aimed.

II. PROBLEM FORMULATION

In many applications, the constraints on the admissible reference signals, used to drive systems with electric or pneumatic actuation, depend on the physical limits of the motors in terms of maximum velocity, acceleration but also maximum torque they can provide. Therefore during the planning of trajectories for such actuators it is necessary to meet the following constraints:

1) velocity constraint

$$\dot{x}_m \leq \dot{x} \leq \dot{x}_M, \qquad (1)$$

2) acceleration constraint

$$\ddot{x}_m \leq \ddot{x} \leq \ddot{x}_M, \qquad (2)$$

3) torque/force constraint

$$\tau_m \leq \tau \leq \tau_M. \qquad (3)$$

In order to take into account the constraint on the torque during trajectory planning it is necessary to translate it in a constraint on the velocity and/or the acceleration, by assuming a well-defined model of the load to be driven. For the sake of simplicity, we assume a simple but quite common load model represented by an inertia plus some frictional phenomena (modeled with a damper):

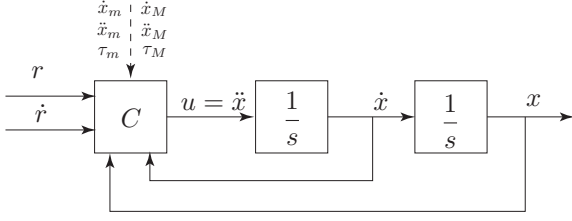$$\tau = J\ddot{x} + b\dot{x}, \qquad (4)$$

Fig. 1. Basic structure of the trajectory filter.

where J is the total inertia (including the motor inertia) and b is the damping coefficient. In this case, (3) can be rewritten as

$$J^{-1}(\tau_m - b\dot{x}) \le \ddot{x} \le J^{-1}(\tau_M - b\dot{x}).$$

This expression is only a particular case of the more general expression

$$a_m(\dot{x}) \le \ddot{x} \le a_M(\dot{x}). \tag{5}$$

where the acceleration is bounded by two generic functions of the speed.

The structure of the proposed trajectory filter is shown in Fig. 1. The goal of the controller C is to nullify in minimum time both position and velocity errors with respect to the input $r(t)$, defined as $y = x - r$ and $\dot{y} = \dot{x} - \dot{r}$, guaranteing at the same time the compliance with the constraints (1), (2) and (5). Since $u = \ddot{x}$, from (2) and (5) one achieves

$$u_m(\dot{x}) \le u \le u_M(\dot{x}) \tag{6}$$

where

$$u_m(\dot{x}) = \max\{\ddot{x}_m, a_m(\dot{x})\}$$
$$u_M(\dot{x}) = \min\{\ddot{x}_M, a_M(\dot{x})\}$$

with, in the general case, $u_m(\dot{x}) \ne -u_M(\dot{x})$, see Fig. 2.

Additionally, we suppose that in the interval $[\dot{x}_m, \dot{x}_M]$ the two functions $u_m(\dot{x})$ and $u_M(\dot{x})$ never cross the axis $u = 0$. In this way, in all the domain $[\dot{x}_m, \dot{x}_M]$, it is guaranteed that $u_M(\dot{x}) > 0$ and $u_m(\dot{x}) < 0$. These conditions have important implications on the shape of the system trajectories in the phase plane.

Finally, the bound on the torque (and on the acceleration) has become a constraint on the control action $u(t)$ and the control problem consists in steering the state of the error dynamics $(y, \dot{y})$ to the origin of the phase space in minimum time with the constraints (6) and (1) for any admissible initial condition $(y_0, \dot{y}_0)$.
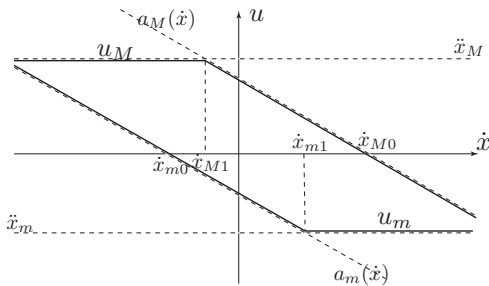

Fig. 2. Constraints on the control signal.

III. BASIC TRAJECTORIES OF THE SYSTEM

For a given control signal $\hat{u}(\dot{x})$, the expression of a trajectory of the system starting from a generic point $(y_0, \dot{y}_0)$ can be readily deduced. Let $y = h_{\hat{u}, \dot{y}_0}(\dot{y}) + y_0$ be the explicit expression of the system trajectory through $(y_0, \dot{y}_0)$, that for the system composed by the two integrators can be always found when the sign of $\hat{u}(\dot{x})$ does not change. Clearly $h_{\hat{u}, \dot{y}_0}(\dot{y}_0) = 0$. Deriving with respect to time, one obtains

$$\dot{y} = \frac{dh_{\hat{u}, \dot{y}_0}(\dot{y})}{d\dot{y}} \ddot{y}.$$

By considering the additional condition on the reference input $\ddot{r}(t) = 0$, it follows that $\ddot{y} = \ddot{x} = \hat{u}(\dot{x})$ and the differential equation becomes

$$\dot{y} = \frac{dh_{\hat{u}, \dot{y}_0}(\dot{y})}{d\dot{y}} \hat{u}(\dot{y} + \dot{r}). \tag{7}$$

By solving this differential equation one obtains the expression of the function $h_{\hat{u}, \dot{y}_0}(\dot{y})$ and therefore of a generic trajectory:

$$y = h_{\hat{u}, \dot{y}_0}(\dot{y}) + y_0 = \int_{\dot{y}_0}^{\dot{y}} \frac{\xi}{\hat{u}(\xi + \dot{r})} d\xi + y_0 \tag{8}$$

In particular, the trajectory starting from the origin $(\dot{y}, \dot{y}_0) = (0, 0)$ is

$$y = h_{\hat{u}, 0}(\dot{y}) = \int_0^{\dot{y}} \frac{\xi}{\hat{u}(\xi + \dot{r})} d\xi.$$

It is possible to find the trajectories through $(\dot{y}, \dot{y}_0) = (0, 0)$ corresponding to the control inputs $\hat{u} = u_m(\dot{x})$ and $\hat{u} = u_M(\dot{x})$, called respectively $h_{m,0}(\dot{y})$, and $h_{M,0}(\dot{y})$. The adoption of the load model (4) is mainly due to the possibility of obtaining a solution of (7) in a closed form, given in (9) and (10). In any case, the theory here reported remains still valid also with different models (possibly nonlinear) but it is necessary to solve (7). In more complex cases, a numerical solution of this differential equation is conceivable.

In Fig. 3 the trajectories of the system obtained with $\hat{u} = u_m(\dot{x})$ and $\hat{u} = u_M(\dot{x})$ for different values of the initial conditions are reported. In this case, only the constraints on the acceleration and on the torque have been considered, while the limits on the speed have not been taken into account yet.
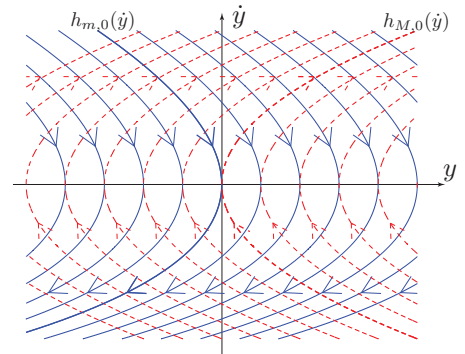

Fig. 3. Phase portrait of the "double integrator" with control input $u(t) = u_M$ (dashed lines) and $u(t) = u_m$ (solid lines).

$$h_{M,0}(\dot{y}) = \begin{cases} -\dfrac{J}{b}\dot{y} + \dfrac{J(b\dot{r}-\tau_M)}{b^2}\log\left(\dfrac{b(\dot{y}+\dot{r})-\tau_M}{b\dot{r}-\tau_M}\right), & \text{if } (\dot{r} \geq \dot{x}_{M1}) \\[3mm] -\dfrac{J}{b}(\dot{x}_{M1}-\dot{r}) + \dfrac{J(b\dot{r}-\tau_M)}{b^2}\log\left(\dfrac{b\dot{x}_{M1}-\tau_M}{b\dot{r}-\tau_M}\right) + \dfrac{\dot{y}^2-(\dot{x}_{M1}-\dot{r})^2}{2\ddot{x}_M}, & \text{if } (\dot{r} < \dot{x}_{M1}) \text{ and } (\dot{r}+\dot{y} > \dot{x}_{M1}) \\[3mm] \dfrac{\dot{y}^2}{2\ddot{x}_M}, & \text{otherwise} \end{cases} \quad (9)$$

$$h_{m,0}(\dot{y}) = \begin{cases} \dfrac{\dot{y}^2}{2\ddot{x}_m}, & \text{if } (\dot{r} \geq \dot{x}_{m1}) \\[3mm] -\dfrac{J}{b}(\dot{x}_{m1}-\dot{r}) + \dfrac{J(b\dot{r}-\tau_m)}{b^2}\log\left(\dfrac{b\dot{x}_{m1}-\tau_m}{b\dot{r}-\tau_m}\right) + \dfrac{\dot{y}^2-(\dot{x}_{m1}-\dot{r})^2}{2\ddot{x}_m}, & \text{if } (\dot{r} < \dot{x}_{m1}) \text{ and } (\dot{r}+\dot{y} > \dot{x}_{m1}) \\[3mm] -\dfrac{J}{b}\dot{y} + \dfrac{J(b\dot{r}-\tau_m)}{b^2}\log\left(\dfrac{b(\dot{y}+\dot{r})-\tau_m}{b\dot{r}-\tau_m}\right), & \text{otherwise.} \end{cases} \quad (10)$$
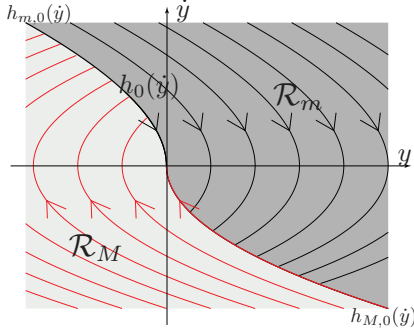

Fig. 4. Regions of the phase plane of the "double integrator" defined by the curve $y = h_0(\dot{y})$.

Fig. 5. Output of the system with the controller C_{opt} (a) and related trajectory in the phase plane (b).

All the trajectories $y = h_{u_m,\dot{y}_0}(\dot{y}) + y_0$ passing through the point $(\hat{y}, 0)$, $\hat{y} > 0$, intersect $h_{M,0}(\dot{y})$, and likewise all the trajectories $y = h_{u_M,\dot{y}_0}(\dot{y}) + y_0$ passing through the point $(\hat{y}, 0)$, $\hat{y} < 0$, intersect $h_{m,0}(\dot{y})$. In particular, the curve $y = h_0(\dot{y})$ with

$$h_0(\dot{y}) = \begin{cases} h_{M,0}(\dot{y}) & \text{if } \dot{y} \leq 0 \\ h_{m,0}(\dot{y}) & \text{if } \dot{y} > 0 \end{cases}$$

splits the phase plane into two regions, see Fig. 4. In both regions, with a proper control all the trajectories are forced towards $h_0(\dot{y})$:

- in $\mathcal{R}_m$ the trajectories obtained by applying to the system the control input $u = u_m(\dot{y} + \dot{r})$ go towards the curve $h_{M,0}(\dot{y})$;
- in $\mathcal{R}_M$ the trajectories corresponding to the control input $u = u_M(\dot{y} + \dot{r})$ tend to the curve $h_{m,0}(\dot{y})$.

IV. Optimal variable-structure controller

On the basis of considerations of Sec. III, it results that the optimal controller which allows the system error to reach the origin in minimum time is given by

$$C_{\text{opt}}: \quad u = \begin{cases} u_M(\dot{y}+\dot{r}), & \text{if } (y,\dot{y}) \in \mathcal{R}_M \\ u_m(\dot{y}+\dot{r}), & \text{if } (y,\dot{y}) \in \mathcal{R}_m \end{cases}$$

or, with a notation tied to the controller implementation

$$u = u_M(\dot{y}+\dot{r})\frac{1-\text{sign}(\sigma)}{2} + u_m(\dot{y}+\dot{r})\frac{1+\text{sign}(\sigma)}{2} \quad (11)$$

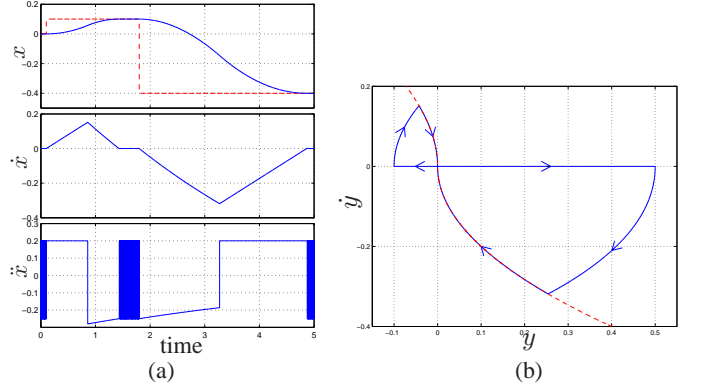being $\sigma = y - h_0(\dot{y})$. The controlled system is globally stable. As a matter of fact, for any possible value of the initial condition $(y_0, \dot{y}_0)$, the state of the system is driven on the "switching surface" $\sigma = 0$ (that is $y = h_0(\dot{y})$) in finite time and then it ideally moves along this curve in minimum time. If the first phase the control signal is positive and equal to the maximum allowed value, in the second phase it assumes a negative value and vice-versa, originating the classical "acceleration" and "deceleration" phases typical of constant acceleration trajectories. In Fig. 5 the trajectories obtained by filtering an input signal $r(t)$ composed by two steps is reported, along with the corresponding trajectories of the system error in the phase plane. It is worth noticing that the control signal/acceleration is not constant since it is constrained by a velocity dependent limit. Moreover when the final position is reached, and accordingly the error $(y, \dot{y})$ is practically coincident with the origin, the control signal starts switching at a very high frequency (see the acceleration profiles $\ddot{x}$ in Fig. 5.(a)). This is due to the fact that the simulation of the system, and therefore the numerical solution of the differential equations describing it, is performed with a finite integration step size. As a consequence in the first phase of the trajectory the controller does not steer the system state exactly on $y = h_0(\dot{y})$, but it switches the value of $u(t)$ only when such a curve is crossed. Therefore, the real trajectory will overcome the optimal one of a quantity that depends on the integration step of the algorithm used to compute the system
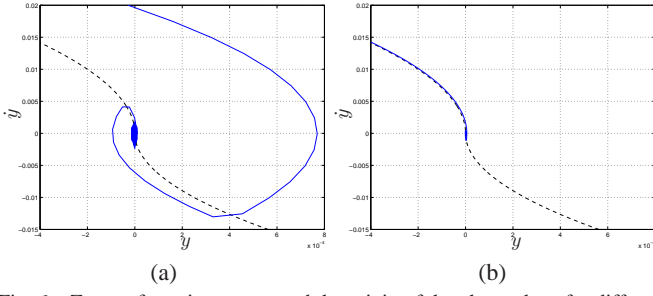
Fig. 6. Zoom of a trajectory around the origin of the phase plane for different values of maximum integration steps: 0.01 (a) and 0.001 (b).

dynamics: the smaller the maximum step size is, the closer the real trajectory to $y = h_0(\dot{y})$ will be, see Fig. 6. Then, the state of the system moves along a trajectory "parallel" to the ideal one until it reaches again the curve $y = h_0(\dot{y})$ and at this point the control action switch again, producing a sort of spiral around the origin. The control action will start switching at high (ideally infinite) frequency between the two values $u = u_m(\dot{r})$ and $u = u_M(\dot{r})$ producing chattering, typical of variable structure controllers, see Fig. 6.

V. VELOCITY CONSTRAINT

In order to take into account the constraint on the velocity the control action must be properly modified with the purpose of forcing the control signal to zero when the maximum (minimum) admissible value of the velocity is reached and the signal $u(t)$ is still positive (negative). In order to impose the bounds (1), the control signal can be chosen as

$$C_{\mathrm{v}}: \quad u'(t) = \begin{cases} \min\{0, u(t)\}, & \text{if } \dot{x} \geq \dot{x}_M \\ u(t), & \text{if } \dot{x}_m < \dot{x} < \dot{x}_M \\ \max\{0, u(t)\}, & \text{if } \dot{x} \leq \dot{x}_m \end{cases}$$

where $u(t)$ is the signal provided by the controller C_{opt}, as reported in (11). In this way, when $\dot{y}_M = \dot{x}_M - \dot{r}$ is reached, the control signal becomes zero and the state $(y, \dot{y})$ moves at the maximum allowed value until $\sigma \leq 0$. It the initial condition lies outside the admissible velocity region (that is with $\dot{x} \geq \dot{x}_M$), $u'(t)$ remains equal to zero and the velocity is constant until $\sigma \leq 0$. As soon as the state crosses the curve $y = h_0(\dot{y})$, the control signal becomes negative and the trajectory moves towards the origin. Similar considerations hold when the trajectory in the phase plane reaches the minimum admissible value $\dot{y}_m = \dot{x}_m - \dot{r}$. If the initial conditions satisfy the constraints on the velocity and these limits do not change during system operations, the filter output will ever satisfy all the constraints reported in Sec. II.

VI. CHATTERING SUPPRESSION

In order to remove the chattering due to the numerical approximation of the continuous-time filter, which is unavoidable when the trajectory is generated, the controller C_{opt} has been modified by considering an additional region around the origin. In this way the normal behavior of the filter remains unchanged, but when the error $(y, \dot{y})$ is very close to the origin it is imposed to the system a linear behavior with an exponential decay. Therefore the structure of the controller results
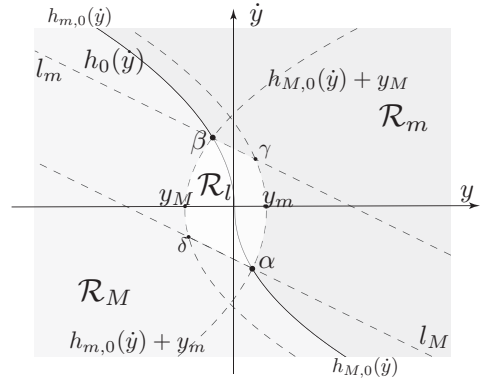

Fig. 7. Regions of the phase plane of the "double integrator" defined for the controller C_{nc}.

$$C_{\mathrm{nc}}: \quad u = \begin{cases} u_M(\dot{y} + \dot{r}), & \text{if } (y, \dot{y}) \in \mathcal{R}_M \\ -p^2 y - 2p\dot{y}, & \text{if } (y, \dot{y}) \in \mathcal{R}_l \\ u_m(\dot{y} + \dot{r}), & \text{if } (y, \dot{y}) \in \mathcal{R}_m \end{cases}$$

where $p > 0$ is a free parameter defining the decay rate of the error dynamics and indirectly the width of the linear region $\mathcal{R}_l$, whose shape is shown in Fig. 7. The most critical point of the proposed controller concerns the choice of this linear region. Obviously the control action in this part of the phase plane cannot exceed the limits imposed by the constraint on the accelerations, therefore the control $u(t)$ in the linear region must meet the constraints

$$u_m(\dot{y} + \dot{r}) \leq \underbrace{-p^2 y - 2p\dot{y}}_{u(t)} \leq u_M(\dot{y} + \dot{r}). \qquad (12)$$

The inequalities in (12) define a region in the phase plane (which includes the origin) bounded by the the two curves

$$y = l_m(\dot{y}, \dot{r}) \qquad \text{with} \qquad l_m(\dot{y}, \dot{r}) = \frac{2}{p}\dot{y} - \frac{u_m(\dot{y} + \dot{r})}{p^2},$$

$$y = l_M(\dot{y}, \dot{r}) \qquad \text{with} \qquad l_M(\dot{y}, \dot{r}) = \frac{2}{p}\dot{y} - \frac{u_M(\dot{y} + \dot{r})}{p^2}.$$

Therefore the linear region $\mathcal{R}_l$ must be contained between this two curves. In particular, they are assumed as boundaries of $\mathcal{R}_l$, guaranteeing that when the the trajectory enters in this region the control signal changes continuously. In order to completely define $\mathcal{R}_l$ it is necessary to consider two additional sides: for this purpose two trajectories of the system passing through $(y_M, 0)$ and $(y_m, 0)$ and corresponding to the control action $u_M(\dot{y} + \dot{r})$ and $u_m(\dot{y} + \dot{r})$ (namely $y = h_{M,0}(\dot{y}) + y_M$ and $y = h_{m,0}(\dot{y}) + y_m$) are used. The two points y_M and y_m are determined by imposing that the curve $y = h_{M,0}(\dot{y}) + y_M$ crosses point β while $y = h_{m,0}(\dot{y}) + y_m$ crosses point α, see Fig. 7. Points $\alpha = (y_\alpha, \dot{y}_\alpha)$ and $\beta = (y_\beta, \dot{y}_\beta)$ are located where the curve $y = h_0(\dot{y})$ intersects $y = l_M(\dot{y}, \dot{r})$ and $y = l_m(\dot{y}, \dot{r})$, respectively. Therefore it is necessary before to solve the systems

$$\begin{cases} y = h_{m,0}(\dot{y}, \dot{r}) \\ y = l_M(\dot{y}, \dot{r}) \end{cases} \Rightarrow (y_\alpha, \dot{y}_\alpha), \qquad (13)$$

and

$$\begin{cases} y = h_{M,0}(\dot{y}, \dot{r}) \\ y = l_m(\dot{y}, \dot{r}) \end{cases} \Rightarrow (y_\beta, \dot{y}_\beta). \qquad (14)$$
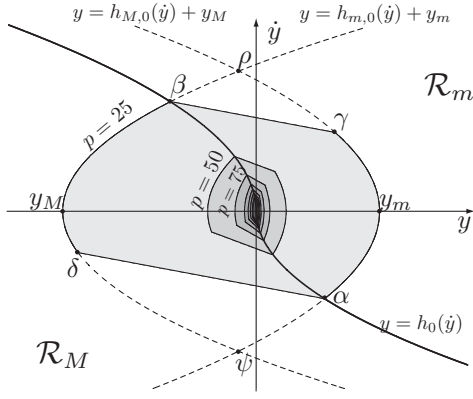
Fig. 8. Linear region of the controller C_{nc} as a function of the parameter p.

In general, the solution can be found numerically. Note that the equations which define the curves depend on $\dot{r}$ (also $h_{m,0}$ and $h_{M,0}$ are functions of $\dot{r}$ although in the rest of paper this variable is left out) and therefore also the solutions are functions of $\dot{r}$. As a consequence when the set point $(r, \dot{r})$ is changed, one should recompute points α, β. This is possible in principle even if the filter is used for planning the trajectory online since the state of the system will enter in the linear region only at the end of the trajectory, and therefore in the first part of the motion one can solve (13) and (14). Otherwise, if one considers an input signal composed only by step functions, $\dot{r} = 0$ and therefore α and β do not change when a new set point is applied.

Once α and β are known it is possible to obtain y_m and y_M:

$$y_M = h_{m,0}(\dot{y}_\beta) - h_{M,0}(\dot{y}_\beta),$$
$$y_m = h_{M,0}(\dot{y}_\alpha) - h_{m,0}(\dot{y}_\alpha).$$

This choice allows to simplify the definition of the control action in the linear region, whose expression results

if $(y < h_{m,0}(\dot{y}) + y_m)$ and $(y > h_{M,0}(\dot{y}) + y_M)$

$$u = \text{sat}(-p^2 y - 2p\dot{y}, u_m(\dot{y} + \dot{r}), u_M(\dot{y} + \dot{r})) \quad (15)$$

where the saturation function is defined as

$$\text{sat}(u, u_{min}, u_{max}) = \begin{cases} u_{max}, & \text{if } u \geq u_{max} \\ u, & \text{if } u_{min} \leq u \leq u_{max} \\ u_{min}, & \text{if } u \leq u_{min}. \end{cases}$$

Therefore, the linear control is defined in the whole region between the two curves $y = h_{m,0}(\dot{y}) + y_m$ and $y = h_{M,0}(\dot{y}) + y_M$, but, due to the saturation function, in the two "triangles" $\beta\gamma\rho$ and $\alpha\psi\delta$ (see Fig. 8), the control is equal to u_m and u_M respectively and therefore is consistent with the control action in the contiguous areas $\mathcal{R}_m$ and $\mathcal{R}_M$, where the controller C_{opt} is still applied. Finally, the controller C_{nc} defined by (15) and (11), in cascade with the controller C_{v}, guarantees the optimal solution of the trajectory planning problem with constraints on velocity, acceleration and torque, compatibly with the presence of the linear control region in the neighborhood of the origin which eliminates undesired oscillations in the control action. In the linear region, the system dynamics is described by the differential equation
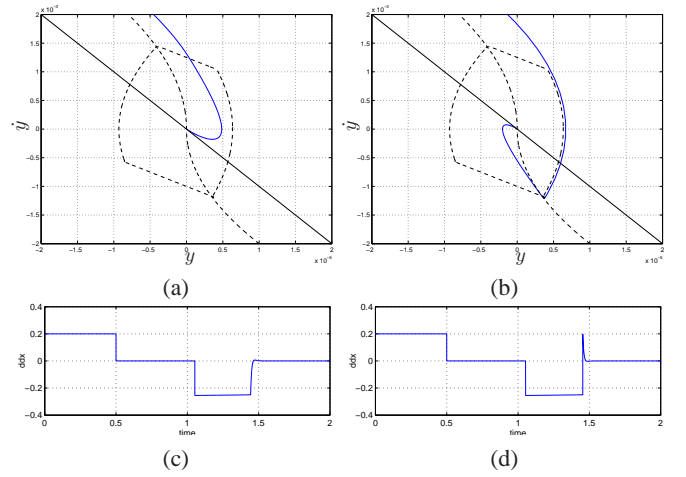
$$\ddot{y} = -p^2 y - 2p\dot{y} \quad (16)$$

(a)   (b)

(c)   (d)

Fig. 9. Zoom of a trajectory of the system with the controller C_{nc} around the origin of the phase plane for different values of maximum integration steps and related acceleration profiles: 0.0001 (a)-(c) and 0.001 (b)-(d).

whose eigenvalues are real and coincident: $s_{1,2} = -p$. The parameter p defines the position of poles and therefore the settling time of the system. The size of the linear region is a function of the free parameters p, as illustrated in Fig. 8. Therefore the choice of p results from a tradeoff between a fast decay of the error when the system state approaches the origin and a dimension of $\mathcal{R}_l$ sufficient large to avoid oscillations. For this latter reason, the choice of p is tied to the selection of the integration step in the solution of the differential equation describing the dynamic behavior of the proposed filter. The smaller the maximum integration step is, the smaller the width of the linear region can be and accordingly the higher p can be. For instance, the trajectory profiles computed with the same boundary conditions and constraints and with $p = 100$ but with a maximum step size of 0.0001 and 0.001 respectively leads to quite different results, as reported in Fig. 9. In the former case, the state of the filter enters in the linear region, and then it approach the origin tangent to a line parallel to the eigenvector of the system (16): $\dot{y} = -py$, see Fig. 9.(a) and (c). In the latter case, the trajectory in the phase state enters in $\mathcal{R}_l$ only after a further switch of the control action from u_m to u_M, and therefore the acceleration profile is characterized by a noticeable overshoot, see Fig. 9.(b)-(d).

VII. PRACTICAL IMPLEMENTATION AND SIMULATIVE EXAMPLES

The controllers C_{opt} and C_{nc} are based on the computation of the phase portrait trajectories for the continuous-time system of Fig. 1, but the implementation of the trajectory planner on digital controllers requires a discrete-time system. For this aim it is sufficient to apply the same controllers to a chain of two discrete-time integrators, see Fig. 10. In any case, also with a fixed sampling time T_s, the algorithm for the chattering suppression allows to avoid oscillations on the output trajectories. In particular, the choice of the unique free parameters of the controller, that is p, is related to the sampling time T_s. Smaller values of the period T_s allow higher values of p and therefore fast dynamics of the error. In
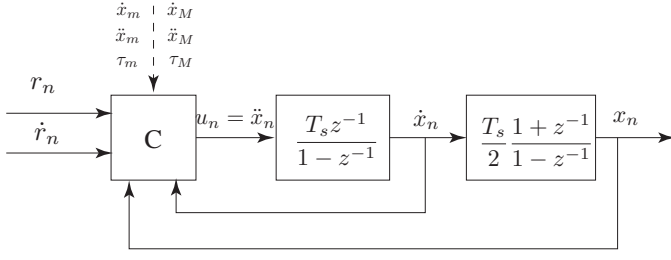
Fig. 10. Basic structure of the trajectory planner in the discrete-time domain.
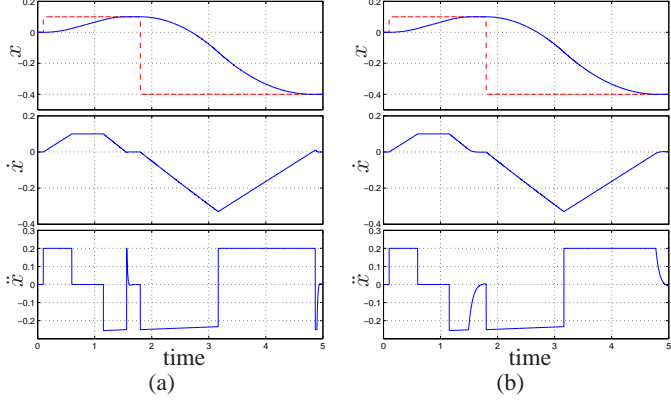

Fig. 11. Trajectory profiles obtained with the discrete system for different values of the parameter p and the same sampling time $T_s = 0.001$s: $p = 50$ (b), $p = 10$ and $T_s = 0.001$s (c).

Fig. 11.(a) and Fig. 11.(b) the trajectories obtained with the same sampling time $T_s = 0.001$s and different values of p are compared. For the sake of generality, asymmetric constraints are assumed: $\dot{x}_m = -0.4$, $\dot{x}_M = 0.1$, $\ddot{x}_m = -0.3$, $\ddot{x}_M = 0.2$, $\tau_m = -0.1$ and $\tau_M = 0.1$. Note the presence in the case (a), corresponding to $p = 50$, of an undershoot/overshoot in the acceleration profile when the system state approaches the origin and correspondingly the filter output reaches the input reference, while in the case (b), with $p = 10$, the spikes in the acceleration profile are eliminated but the resulting dynamics at the end of the motion is very slow.

Finally, in Fig. 12.(a) the profiles of position, velocity and acceleration of the trajectory obtained with $p = 50$ and $T_s = 0.0001$ are shown. In order to evaluate the planned trajectory it is necessary to consider the torque resulting from the motion, see Fig. 12.(b). Obviously, the dynamic characteristics (inertia and damping coefficient) of the load have been considered in (9) and (10) when implementing the trajectory filter. In this case it is assumed $J = 0.2$ and $b = 0.01$. It is worth noticing that for time $\in [1.15, 1.54]$ and time $\in [1.80, 3.16]$, that is during the phases with negative acceleration, the torque reaches the limit value and therefore the acceleration is modified in order to meet this constraint. Conversely, during the phases with positive acceleration the maximum torque is not reached and the acceleration is saturated to its limit value. In any case, the trajectory is optimal in the sense that at each time instant at least a constraint is reached and therefore it is not possible to further reduce its duration.
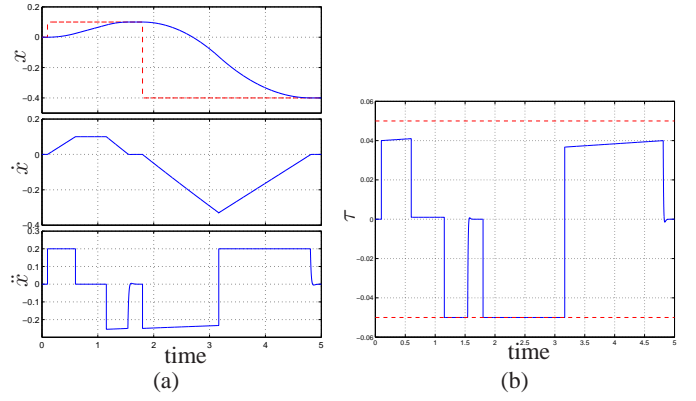

Fig. 12. Trajectory profiles obtained with the discrete system for $p = 50$ and $T_s = 0.0001$ (a) and resistive torque produced by a load (with $J = 0.2$ and $b = 0.01$) following this motion profile (b).

VIII. CONCLUSIONS

In this paper, a filter for online trajectory planning is presented. Given a rough reference signal, such as a sequence of step functions, the filter produces an output profile compliant not only with desired kinematic constraints on velocity, acceleration but also with a bound on the torque that is necessary to actuate the load (the case of a inertial/frictional load is considered). The proposed filters is designed in the continuous-time domain, but the extension to the discrete-time domain is illustrated and the effectiveness of the trajectory generator is proved by means of numerical examples.

REFERENCES

[1] T. M. Howard and A. Kelly, "Optimal rough terrain trajectory generation for wheeled mobile robots," *International Journal of Robotics Research*, vol. 26 (2), pp. 141–166, 2007.

[2] A. Sahraei, M. Manzuri, M. R. Razvan, M. Tajfard, and S. Khoshbakht, "Real-time trajectory generation for mobile robots," in *Proceedings of the 10th Congress of the Italian Association for Artificial Intelligence on AI*IA 2007: Artificial Intelligence and Human-Oriented Computing*, 2007.

[3] E. Anderson, R. Beard, and T. McLain, "Real-time dynamic trajectory smoothing for unmanned air vehicles," *IEEE Trans. on Control Systems Technology*, vol. 13(3), pp. 471–477, 2005.

[4] M. Bonfé, M. Vignali, and M. Fiorini, "Plc-based control of a robot manipulator with closed kinematic chain," in *2009 IEEE International Conference on Robotics and Automation*, 2009.

[5] R. Zanasi, C. G. L. Bianco, and A. Tonielli, "Nonlinear filter for the generation of smooth trajectories," *Automatica*, vol. 36, 2000.

[6] C. Zheng, Y. Su, and P. Muller, "Simple online smooth trajectory generations for industrial systems," *Mechatronics*, vol. 19, 2009.

[7] R. Zanasi and R. Morselli, "Third order trajectory generator satisfying velocity, acceleration and jerk constraints," in *Control Applications, 2002. Proceedings of the 2002 International Conference on*, 2002.

[8] O. Gerelli and C. G. lo Bianco, "Real-time path tracking control of robotic manipulators with bounded torques and torque-derivatives," in *IEEE-RSJ Int. Conf. on Intelligent Robots and Systems, IROS 2008*, Nice, France, 2008.

[9] C. G. L. Bianco, A. Tonielli, and R. Zanasi, "Nonlinear filters for the generation of smooth trajectories," *Automatica*, vol. 36, 2000.

[10] R. Morselli and R. Zanasi, "Position trajectory generator with nonlinear constraints," in *Proceedings of the 2002 IEEE International Conference on Control Applications*, 2002.

[11] R. Zanasi and R. Morselli, "Discrete minimum time tracking problem for a chain of three integrators with bounded input," *Automatica*, vol. 39 (9), pp. 1643–1649, 2003.

[12] J. Slotine, *Applied Nonlinear Control*. Prentice Hall, 1991.

[13] V. Utkin, "Variable structure systems with sliding modes," *IEEE Trans. on automatic Control*, vol. 22, pp. 212–222, 1977.