IMA Conference on Mathematics of Robotics
9 – 11 September 2015, St Anne's College, University of Oxford

Online Minimum-Jerk Trajectory Generation

By M. M. Ghazaei Ardakani, A. Robertsson, R. Johansson†

The Department of Automatic Control, LTH, Lund University, SE–221 00 Lund, Sweden
{mahdi.ghazaei, anders.robertsson, rolf.johansson}@control.lth.se

Abstract

Robotic trajectory generation is reformulated as a controller design problem. For minimum-jerk trajectories, an optimal controller using the Hamilton-Jacobi-Bellman equation is derived. The controller instantaneously updates the trajectory in a closed-loop system as a result of changes in the reference signal. The resulting trajectories coincide with piece-wise fifth-order polynomial trajectories for piece-wise constant target states. Since having hard constraints on the final time poses certain robustness issues, a smooth transition between the finite-horizon and an infinite-horizon problem is developed. This enables to switch softly to a tracking mode when a moving target is reached.

1. Introduction

A fundamental problem in robotics is planning the motion for a task. At the lowest level, a movement is described by a trajectory, *i.e.*, a mapping from the time to the position. An important class of motion planning problems is concerned with point-to-point trajectory generation. In this case, the objective can for instance be to reach a final state in the minimum time given certain constraints (Macfarlane & Croft 2003; Haschke *et al.* 2008) or in a fixed time. The fixed-time problems are of importance when a less aggressive strategy than a minimum-time solution is sufficient. Moreover, fixed-time motions lend themselves to the coordination between several degrees of freedom or entities.

Several approaches to dealing with fixed-time problems have been suggested. A common approach is fitting a piece-wise polynomial between the starting point and a final point (Taylor 1979; Lin *et al.* 1983). As suggested by von Stryk and Schlemmer (1994), the energy or the power consumption can be optimized and minimizing the effort was proposed by Martin and Bobrow (1997). The solutions were obtained by numerical methods either by discretization or parameter optimization over a set of basis functions. A sub-optimal solution to the fixed-time trajectory planning considering a more generic cost function was derived by Dulçba (1997).

For trajectory generation, polynomials play an important role. They show up for example as partial solutions to the minimum-time problems or they are used as the basis functions. Specially, fifth-order polynomials result from a *minimum-jerk* model suggested by Flash and Hogan (1985). An important feature of this model is that it provides a kinematic description of the motion of the human hand in planar scenarios. The model is able to predict the bell-shaped velocity profiles in point-to-point movements as well as the characteristics of the curvature in via-point movements. According to this model, the

† The authors are members of the LCCC Linnaeus Center and the ELLIIT Excellence Center at Lund University.

cost functional to be minimized is:

$$C = \frac{1}{2} \int\limits_{0}^{d_0} \left(\dddot{X}^2 + \dddot{Y}^2 \right) dt, \qquad \dddot{X} := \frac{d^3X}{dt^3}, \; \dddot{Y} := \frac{d^3Y}{dt^3} \tag{1.1}$$

where X and Y represent the coordinates and d_0 the duration of the movement. Assuming that the X and the Y coordinates are decoupled, it is possible to break (1.1) into two one-dimensional optimization problems. Using the variational principle, the solution is shown to be a fifth-order polynomial (Shadmehr 2005).

As a motivating example for this article, consider a quadcopter task which is to follow/catch flying objects as soon as they cross a border. A camera system detects the objects and estimates of the current position, velocity, and acceleration as well as an estimate of the arrival of the objects are provided. We choose a minimum-jerk trajectory profile to generate smooth motion. This application falls under the fixed-time trajectory planning. Moreover, it requires the replanning of the trajectory online, *i.e.*, as soon as new estimates are obtained.

Earlier works have focused on computationally efficient solutions to time-optimal problems for online trajectory generation (Kröger and Wahl 2010; Hehn and D'Andrea 2011). A generic way to deal with the online trajectory generation is to buffer each segment of the trajectory (or its parameters) and implement switching between the pieces. However, this approach becomes inefficient if the update rate of the trajectory is high. Thus, rather than considering trajectories as solely a function of time, we propose an alternative view based on dynamical systems. This allows for a fully reactive trajectory generation method with continuous reactions to the changes in the target.

2. Problem formulation

This article concerns the one-dimensional minimum-jerk trajectory-generation problem given a fixed time. The main motivation is to update trajectories immediately as a result of changes in the (moving) target while ensuring the continuity of the position, velocity and acceleration of the trajectory. Moreover, we require a smooth transition between trajectory planning and tracking modes. In contrast to mathematically designed or optimal trajectories purely as a function of time, we regard a trajectory as an output of a dynamical system. The exogenous input signal defines the set-point for the trajectory generator.

To minimize the jerk, each decoupled degree of freedom can be represented by a triple integrator. Let u denote the jerk and $y(t)$ the trajectory, then

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \tag{2.1}$$

$$y = x_1. \tag{2.2}$$

We define the reference signal $r(t) \in \mathbb{R}^3$ such that $r_1(t)$, $r_2(t)$, $r_3(t)$ denote the current position, velocity and acceleration of the target, respectively. Given the reference signal $r(t)$ and the desired time t_f, we wish to design $u(x, r, t)$ that produces the solution to

$$\underset{u}{\text{minimize}} \quad \int_{t_0}^{t_f} u^2 \, dt \tag{2.3}$$

subject to (2.1), $x(t_0) = x_0$, and $x(t) = r(t), \text{ for } t \geq t_f$.

3. Optimal Controller

Starting from model (2.1), we have

$$\dot{x} = \begin{pmatrix} x_2 & x_3 & u \end{pmatrix}^T := f(t, x, u). \tag{3.1}$$

According to the Pontryagin maximum principle (Pontryagin *et al.* 1962; Liberzon 2011),

$$\dot{x}^* = H_p(x^*, u^*, p, p_0) \tag{3.2}$$
$$\dot{p} = -H_x(x^*, u^*, p, p_0) \tag{3.3}$$
$$x^*(t_0) = x_0, \quad x^*(t_f) = r(t_f) \tag{3.4}$$

Here, H denotes the Hamiltonian, the subscripts denote partial derivatives with respect to the given variable, x and p are the states and the costates, respectively, t_0 and t_f denote the initial and final time, respectively, and variables with star correspond to the optimal solution. The optimal control maximizes the Hamiltonian, that is

$$H(x^*(t), u^*(t), p(t), p_0) \leq H(x^*(t), u, p(t), p_0) \tag{3.5}$$

Denoting the running cost by $L(x, u)$, for problem (2.3), the Hamiltonian is

$$H(x, u, p, p_0) = \langle p, f(x, u)\rangle + p_0 L(x, u) \tag{3.6}$$
$$= \begin{pmatrix} p_1 & p_2 & p_3 \end{pmatrix} \begin{pmatrix} x_2 \\ x_3 \\ u \end{pmatrix} + p_0 u^2. \tag{3.7}$$

By the partial differentiation of H with respect to u, we find the extremum to be

$$\frac{\partial H}{\partial u} = 2p_0 u + p_3 = 0 \Rightarrow u^* = -\frac{p_3}{2p_0}. \tag{3.8}$$

Consequently, the Hamiltonian along the optimal trajectory is

$$H(x^*, u^*, p, p_0) = p_1 x_2^* + p_2 x_3^* - \frac{p_3^2}{4p_0}, \tag{3.9}$$

where

$$\dot{p}_1 = -H_{x_1} = 0 \tag{3.10}$$
$$\dot{p}_2 = -H_{x_2} = -p_1 \tag{3.11}$$
$$\dot{p}_3 = -H_{x_3} = -p_2. \tag{3.12}$$

These equations combined with (3.8) give us

$$u^* = k_1 t^2 + k_2 t + k_3, \tag{3.13}$$

with coefficients k_1, k_2, and k_3 to be determined. Integrating the control signal three times results in x_1, which is apparently a fifth-order polynomial. For the sake of simplicity, we assume $t_0 = 0$ and $x(t_f) = 0$. By matching the initial and final conditions, we obtain

$$\begin{aligned} y = x_1^* &= y_0(1 - 10t_n^3 + 15t_n^4 - 6t_n^5) + v_0 d_0 t_n(1 - 6t_n^2 + 8t_n^3 - 3t_n^4) \qquad (3.14) \\ &+ \frac{a_0}{2} d_0^2 t_n^2 (1 - 3t_n + 3t_n^2 - t_n^3) \\ \dot{y} = x_2^* &= \frac{y_0}{d_0}(-30t_n^2 + 60t_n^3 - 30t_n^4) + v_0(1 - 18t_n^2 + 32t_n^3 - 15t_n^4) \qquad (3.15) \\ &+ a_0 d_0 t_n (1 - \frac{9}{2} t_n + 6t_n^2 - \frac{5}{2} t_n^3) \\ \ddot{y} = x_3^* &= \frac{y_0}{d_0^2}(-60t_n + 180t_n^2 - 120t_n^3) + \frac{v_0}{d_0}(-36t_n + 96t_n^2 - 60t_n^3) \qquad (3.16) \\ &+ a_0(1 - 9t_n + 18t_n^2 - 10t_n^3), \end{aligned}$$

and the optimal control signal is

$$\begin{aligned} \dddot{y} = u^* &= \frac{y_0}{d_0^3}(-60 + 360t_n - 360t_n^2) + \frac{v_0}{d_0^2}(-36 + 192t_n - 180t_n^2) \\ &+ \frac{a_0}{d_0}(-9 + 36t_n - 30t_n^2) \qquad (3.17) \end{aligned}$$

where $t_n := (t - t_0)/d_0$ is the normalized elapsed time with respect to the duration $d_0 = t_f - t_0$ and y_0, v_0, and a_0 are initial position, velocity, and acceleration, respectively.

Let us now consider the Hamilton-Jacobi-Bellman (HJB) equation (Bellman & Kalaba 1965; Liberzon 2011)

$$-V_t(t, x) = \inf_{u \in U} \{L(t, x, u) + \langle V_x(t, x), f(t, x, u)\rangle\}, \qquad (3.18)$$

with the cost function J and the value function V defined as

$$\begin{aligned} J(t, x, u) &:= \int_t^{t_f} L(s, x(s), u(s))\, ds + K(x(t_f)) \qquad (3.19) \\ V(t, x) &:= \inf_{u[t, t_f]} J(t, x, u). \qquad (3.20) \end{aligned}$$

Here, $U \subseteq \mathbb{R}$ defines the control set, $L(\cdot)$ and $K(\cdot)$ denote the running cost and the terminal cost, respectively. For the minimum-jerk problem, we have

$$\begin{aligned} L(t, x, u) &= u^2, \qquad K(x(t_f)) = 0 \qquad (3.21) \\ -V_t(t, x) &= \inf_{u \in U} \{u^2 + \langle V_x(t, x), f(t, x, u)\rangle\} \qquad (3.22) \\ &= \min_{u \in \mathbb{R}} \{u^2 + V_{x_1} x_2 + V_{x_2} x_3 + V_{x_3} u\}. \qquad (3.23) \end{aligned}$$

The optimum is achieved for

$$\frac{\partial(u^2 + V_{x_1} x_2 + V_{x_2} x_3 + V_{x_3} u)}{\partial u} = 0 \Rightarrow u^* = -\frac{V_{x_3}}{2}. \qquad (3.24)$$

Therefore,

$$-V_t(t, x) = -\frac{V_{x_3}^2}{4} + V_{x_1} x_2 + V_{x_2} x_3. \qquad (3.25)$$

As a result of the application of the maximum principle, (3.17) gives us an expression for $\dddot{y}$ along the optimal path. Now, considering the value function in (3.20), *i.e.*, the cost to go, and using the definition of the cost function in (3.19), we conclude

$$V(t, x) = \int_t^{t_f} \dddot{y}^2(s)\, ds. \qquad (3.26)$$

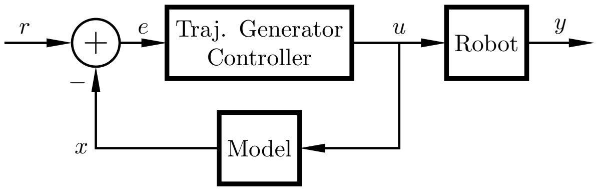

FIGURE 1. Block diagram of trajectory generation, where r, x, e, u, and y denote the reference signal, the states of the model, the error signal, the control signal and the robot position, respectively.

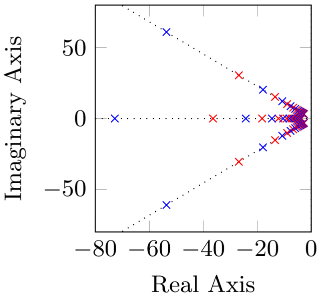

FIGURE 2. Closed-loop poles for various values of the remaining time $d = [1 : -0.05 : 0.05]$.

Note that to calculate (3.26), the initial state is the current state, the duration d_0 is equal to the remaining time $t_f - t$, and the normalized elapsed time t_n is $(s-t)/(t_f-t)$. It is straightforward to verify that the resulting value function satisfies the HJB equation. Accordingly, from (3.24) we derive

$$u^* = -\frac{V_{x_3}}{2} = -\left(60\frac{x_1}{(t_f-t)^3} + 36\frac{x_2}{(t_f-t)^2} + 9\frac{x_3}{t_f-t}\right). \tag{3.27}$$

This reformulation gives us a control law for generating the trajectory. The feedback signal is linear in the states, but nonlinear with respect to the time.

4. Servo Problem

The previous section outlined the optimal solution to the problem with the final state located at the origin. However, to solve the original problem (2.3), it is necessary to generalize the final state. This corresponds to the servo problem for following an arbitrary reference (Glad & Ljung 2000).

Assuming that our best estimate of the value of the target in d seconds depends only on the current state of the target, it is possible to use the transformation $e(t) = r(t) - x(t)$ depicted in Fig. 1. In this case, the reference signal can be absorbed into the initial conditions of the model. Hence, the new variable $e_1(t)$ generated by the controller fulfills the minimum-jerk criterion and $e(t_f) = 0$. Since $r_1(t)$ does not impact the jerk (it is a polynomial in time of maximum order two) and $x_1(t) = r_1(t) - e_1(t)$, the cost functional in this special case is invariant under this transformation. Therefore, the solution after the transformation is optimal for the new boundary conditions.

5. Switching to Tracking

An obvious issue with (3.27) is that it is sensitive to the errors in the states when the time approaches t_f. Without loss of generality, assume there is some noise $\epsilon(t)$ in the velocity measurement. In this case, the closed-loop system obeys the differential equation

$$\dddot{y} = -\left(60\frac{y}{(t_f-t)^3} + 36\frac{\dot{y}}{(t_f-t)^2} + 9\frac{\ddot{y}}{t_f-t}\right) - 36\frac{\epsilon(t)}{(t_f-t)^2}. \tag{5.1}$$

Thus, as t approaches t_f, a very small noise level can blow up the control signal. A remedy to this problem is to switch to an infinite-horizon problem when $d = t_f - t$ becomes small. In the following, we show that this transition can be done smoothly.

Consider the plot of the closed-loop poles of the system for fixed values of d in Fig. 2.

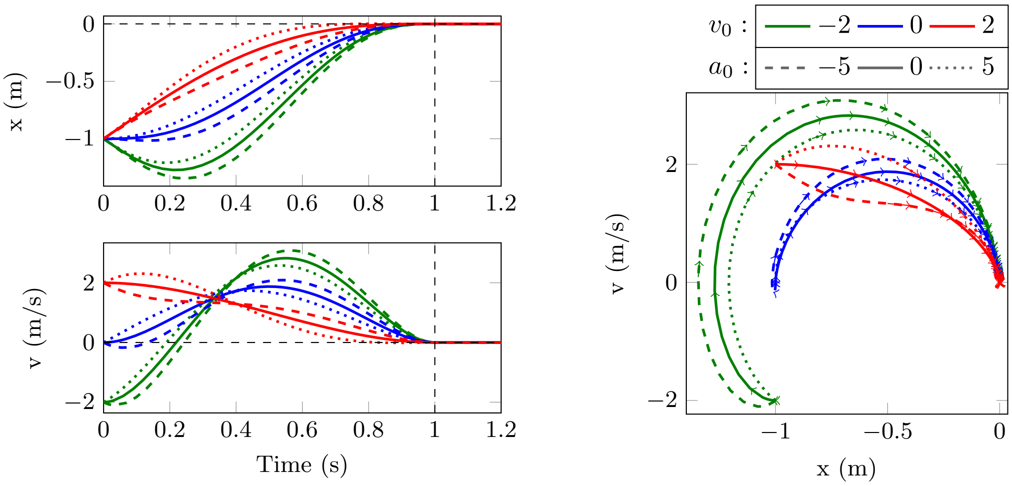

FIGURE 3. Experiment 1: on the left, curves resulting from the control law (6.1), starting from $x_0 = -1$, $v_0 = \{-2, 0, 2\}$, and $a_0 = \{-5, 0, 5\}$ with $t_f = 1$ s and $r(t) = 0$. On the right, the corresponding phase portrait.

The characteristic equation of the closed-loop system for a fixed remaining time d is

$$d^3 s^3 + 9d^2 s^2 + 36ds + 60 = 0. \tag{5.2}$$

If p is a solution of (5.2) for $d = 1$, then p/d is a solution for any given d. Thus, $\arg(p)$ is independent of the remaining time while the poles move toward infinity as the remaining time approaches zero. Furthermore, we can rewrite the characteristic equation as

$$(s + \alpha\omega)(s^2 + 2\zeta\omega s + \omega^2) = 0, \tag{5.3}$$

where $(\omega d)^2 = 12 - 2\sqrt[3]{3^2} + 6\sqrt[3]{3} \approx 16.493$, $\zeta = (6 - \sqrt[3]{3^2} + \sqrt[3]{3})/2\omega d \approx 0.66$, and $\alpha = (3 + \sqrt[3]{3^2} - \sqrt[3]{3})/\omega \approx 0.896d$. Considering these values and the knowledge of the noise in the system, it is possible to find a minimum acceptable value for d. Accordingly, a smooth transition is achieved by limiting d from below.

6. Simulations

In this section, we present the simulation results of the closed-loop trajectory generation. The results are generated using a modified control law in (3.27)

$$u = -\left(60\frac{e_1}{d^3} + 36\frac{e_2}{d^2} + 9\frac{e_3}{d}\right), \quad d = \max(t_f - t, 0.06). \tag{6.1}$$

In the first experiment, the reference is set to zero and the initial position to -1. The initial velocity and the initial acceleration are varied. With $t_f = 1$ s, the control law is expected to result in fifth-order polynomials with zero velocity and acceleration at the origin in one unit of time. Figure 3 visualizes the evolution of the states in a phase portrait. Note that the trajectories in the phase plane cross each other since the state space has a higher dimension than two and the system is time-variant.

The second experiment illustrates the result of the trajectory generation for a moving target. Every second, a new target is activated. The objective is to intercept the target in 0.8 s and to continue tracking it until a new target is detected. In Fig. 4, the solid blue and dashed green curves correspond to the robot and the active target, respectively. As seen in the figure, there is a smooth transition to a tracking mode when the target

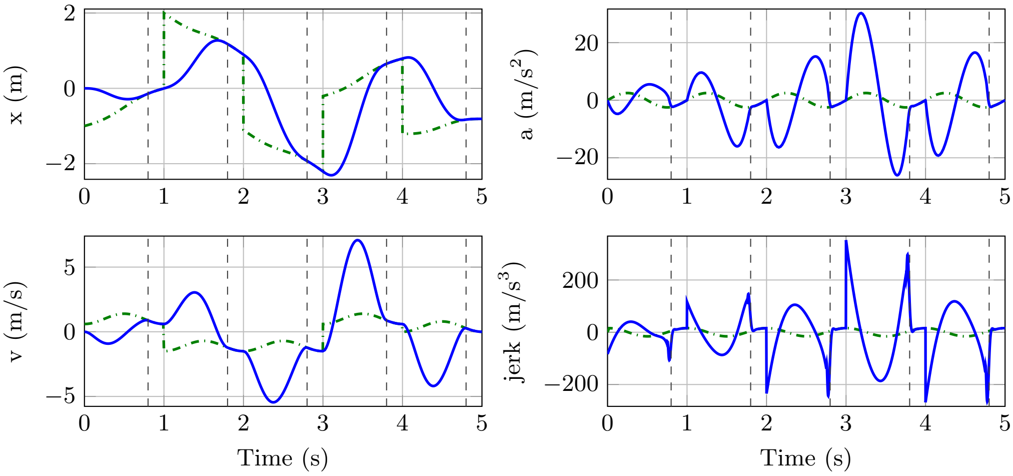

FIGURE 4. Experiment 2: intercepting a moving target in 0.8 s: positions, velocities vs. time on the left and acceleration and jerk vs. time on the right. The solid blue and the dashed green curves correspond to the robot and the target, respectively. Every second, a new target is activated. The trajectory generation softly switches to a tracking mode before the deadline. As expected, x, v, and a are continuous.

is almost reached and the position, velocity, and acceleration of the generated trajectory are continuous.

7. Discussion

Considering the online trajectory generation for a moving target, there are at least two strategies. One can estimate a time $\hat{t}_f$ and a desired *future value* for the states $\hat{r}_f$, such that the generated trajectory meets this target state, *i.e.*, $x(\hat{t}_f) = \hat{r}_f$. The procedure is repeated as soon as a better estimate is obtained. The other strategy is that the generated trajectory tracks the *current value* of the target but additionally superimposes a motion that eliminates the initial offset. These strategies do not necessarily lead to the same solution. The first strategy works better if an accurate estimation of the target's final state is possible, while the second strategy naturally leads to a smooth transition between trajectory planning and tracking modes. Since the states in many physical systems cannot change discontinuously, the second strategy is advantageous when the time horizon is short. This justifies the developed method in this article, where the trajectory generation depends only on the error $r(t) - y(t)$. More importantly, in case of the minimum-jerk trajectories the second strategy does not affect the optimality of the solution if at any time instant $r(t)$ and t_f provide the best estimate of the target state.

It is necessary to pay attention to the source of changes when designing a closed-loop trajectory generator. Variations in the target state do not necessarily have to impact the trajectory in the same way as the disturbances on the robot. This requires a controller with two degrees of freedom. For this reason, we did not include the state feedback from the robot in Fig. 1. However, when the deviation between the states of the internal model and the actual system becomes large, it is reasonable to update the state of the model.

Minimum-jerk trajectories can be time scaled to accommodate limitations on the kinematic variables similarly to the approach by Dahl and Nielsen (1990). If the constraints are constant, the solution to the minimum-time problem provides the lowest allowable t_f (Hehn and D'Andrea 2011), *i.e.*, any trajectory generated with a larger time interval than the minimum time satisfies the constraints. For more discussion about handling state constraints by adjusting the time, see (Ghazaei A. 2015).

8. Conclusion

A controller model for trajectory generation with continuous reactions to the changes in the target is proposed. We solve the Hamilton-Jacobi-Bellman equation in order to find the optimal minimum-jerk controller. The result is a time-varying linear feedback law, which produces fifth-order polynomials for piece-wise constant target states. For this controller, we show that limiting the remaining time from below naturally leads to a smooth transition between trajectory planning and tracking modes. Thus, we have obtained a fully reactive trajectory generation method for possibly moving targets with the desirable properties of minimum-jerk trajectories.

REFERENCES

Bellman, R. and Kalaba, R. E. (1965). *Dynamic programming and modern control theory.* Academic Press, New York.

Dahl, O. and Nielsen, L. (1990). Torque limited path following by on-line trajectory time scaling. *IEEE Trans. Rob. and Autom.*, 6:554–561.

Dulçba, I. (1997). Minimum cost, fixed time trajectory planning in robot manipulators. A suboptimal solution. *Robotica*, 15(5):555–562.

Flash, T. and Hogan, N. (1985). The coordination of arm movements: an experimentally confirmed mathematical model. *J. Neuroscience*, 5(7):1688–1703.

Ghazaei Ardakani, M. M. (2015). Topics in trajectory generation for robots. Licentiate Thesis TFRT--3265--SE, Dept. Automatic Control, Lund University, Sweden.

Glad, T. and Ljung, L. (2000). *Control Theory: Multivariable and Nonlinear Methods.* Taylor & Francis, London.

Haschke, R., Weitnauer, E., and Ritter, H. (2008). On-line planning of time-optimal, jerk-limited trajectories. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS), Sep. 22-26, 2008*, pages 3248–3253, Nice, France.

Hehn, M. and D'Andrea, R. (2011). Quadrocopter trajectory generation and control. In *Proc. 18th IFAC World Cong., Aug. 28–Sep. 2, 2011*, volume 18, pages 1485–1491, Milano, Italy.

Kröger, T. and Wahl, F. M. (2010). On-line trajectory generation: Basic concepts for instantaneous reactions to unforeseen events. *IEEE Trans. Robot.*, 26(1):94–111.

Liberzon, D. (2011). *Calculus of Variations and Optimal Control Theory: A Concise Introduction.* Princeton University Press, Princeton, New Jersey.

Lin, C.-S., Chang, P.-R., and Luh, J. (1983). Formulation and optimization of cubic polynomial joint trajectories for industrial robots. *IEEE Trans. Automatic Control*, 28(12):1066–1074.

Macfarlane, S. and Croft, E. A. (2003). Jerk-bounded manipulator trajectory planning: design for real-time applications. *IEEE Trans. Rob. Autom.*, 19(1):42–52.

Martin, B. J. and Bobrow, J. E. (1997). Minimum effort motions for open chain manipulators with task-dependent end-effector constraints. In *Proc. IEEE Int. Conf. Rob. and Autom., Apr. 20–25, 1997*, volume 3, pages 2044–2049, Albuquerque, NM.

Pontryagin, L. S., Boltyanskii, V., Gamkrelidze, R., and Mishchenko, E. (1962). *The Mathematical Theory of Optimal Processes.* Interscience, New York.

Shadmehr, R. (2005). *The computational neurobiology of reaching and pointing: a foundation for motor learning.* MIT press, Cambridge, MA.

Taylor, R. H. (1979). Planning and execution of straight line manipulator trajectories. *IBM J. Research and Development*, 23(4):424–436.

von Stryk, O. and Schlemmer, M. (1994). In Bulirsch, R. and Kraft, D., editors, *Computational Optimal Control*, chapter Optimal Control of the Industrial Robot Manutec R3, pages 367–382. Birkhauser Verlag, Basel, Switzerland.