Time-Optimal Path Tracking for Robots
a Convex Optimization Approach

Diederik Verscheure, Dr. B. Demeulenaere,
Prof. J. Swevers, Prof. J. De Schutter, Prof. M. Diehl

Katholieke Universiteit Leuven,
Department of Mechanical Engineering,
PMA Division

19 februari 2008

Outline

1. Introduction
   - Time-optimal motion planning
   - Convex optimization
2. Time-optimal path tracking
   - Problem formulation
   - Existing solution methods
3. Reformulation and solution method
   - Convex reformulation
   - Direct transcription
   - SOCP formulation
   - Implementation
4. Examples
5. Conclusions

Outline

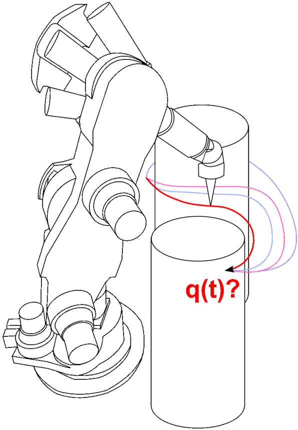
Time-optimal motion planning

Time-optimal motion planning

- important for maximizing the productivity of robot systems.
- reduces need for time-consuming manual programming.

Two approaches

- direct approach.
- decoupled approach (Choset et al. 2005).



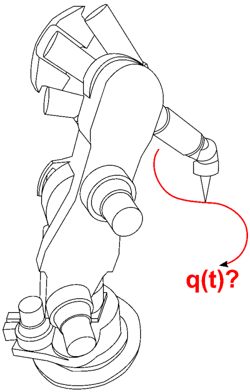
Time-optimal motion planning

The direct approach

- solve motion planning directly in system's state space ($2 \times n$ states).
- complex because of nonlinear robot dynamics.

The decoupled approach

- solve in two stages:
  - path planning: geometric.
  - path tracking: dynamic (2 states).
- easier and computationally cheaper.



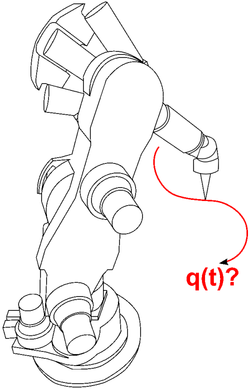
Time-optimal motion planning

The direct approach

- solve motion planning directly in system's state space ($2 \times n$ states).
- complex because of nonlinear robot dynamics.

The decoupled approach

- solve in two stages:
  - path planning: geometric.
  - path tracking: dynamic (2 states).
- easier and computationally cheaper.



Optimization problems

Optimization problems in engineering

- nonlinear.
- large number of variables.
- optimality of the solution?

Convex optimization problems

"In fact the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity."
- Rockafellar

Convex optimization problems

Convex optimization problems

Can be formulated as (Boyd and Vandenberghe 2004):

$$\min_{\mathbf{x}} \; f_o(\mathbf{x}), \tag{1}$$

$$\text{subject to } \mathbf{F}\mathbf{x} = \mathbf{g}, \tag{2}$$

$$f_j(\mathbf{x}) \leq 0, \tag{3}$$

$$\text{for } j = 1 \ldots m.$$

- convex objective function $f_o(\mathbf{x})$.
- linear equality constraints!
- convex nonlinear inequality constraints $f_j(\mathbf{x})$.

Convex optimization problems

Why should we care?

- any local optimum is also globally optimal.
- convergence to optimal value using any nonlinear solver.
- efficient solvers for particular types of convex problems.
- let numerical experts worry about numerical issues.

There is one catch: formulation as a particular type of convex problem is not always obvious.

Types of convex optimization problems

Types of convex optimization problems

- linear program (LP).
- quadratic program (QP).
- second-order cone program (SOCP).
- semidefinite program (SDP).

For numerical efficiency, it is important to use the least general formulation.

Types of convex optimization problems

Types of convex optimization problems

- linear program (LP).
- quadratic program (QP).
- second-order cone program (SOCP).
- semidefinite program (SDP).

For numerical efficiency, it is important to use the least general formulation.

Outline

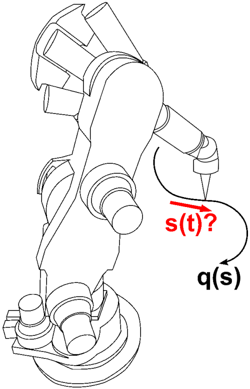
Time-optimal path tracking

Time-optimal path tracking

- equations of motion

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_s(\mathbf{q})\mathrm{sgn}(\dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}),$$

- path $\mathbf{q}(s)$, function of path coordinate s.
- lower and upper bounds $\underline{\boldsymbol{\tau}}$ and $\overline{\boldsymbol{\tau}}$ on torques.



Problem

Find relation $s(t)$ that minimizes trajectory duration T and satisfies torque constraints.

Time-optimal path tracking

Path $\mathbf{q}(s)$ function of the path coordinate s.

$$\dot{\mathbf{q}}(s) = \mathbf{q}'(s)\dot{s}, \tag{4}$$

$$\ddot{\mathbf{q}}(s) = \mathbf{q}'(s)\ddot{s} + \mathbf{q}''(s)\dot{s}^2, \tag{5}$$

Equations of motion can be rewritten as

$$\boldsymbol{\tau}(s) = \mathbf{m}(s)\ddot{s} + \mathbf{c}(s)\dot{s}^2 + \mathbf{g}(s), \tag{6}$$

where

$$\mathbf{m}(s) = \mathbf{M}(\mathbf{q}(s))\mathbf{q}'(s), \tag{7}$$

$$\mathbf{c}(s) = \mathbf{M}(\mathbf{q}(s))\mathbf{q}''(s) + \mathbf{C}(\mathbf{q}(s), \mathbf{q}'(s))\mathbf{q}'(s), \tag{8}$$

$$\mathbf{g}(s) = \mathbf{F}_s(\mathbf{q}(s))\mathrm{sgn}(\mathbf{q}'(s)) + \mathbf{G}(\mathbf{q}(s)), \tag{9}$$

Time-optimal path tracking

Time-optimal path tracking subject to actuator constraints

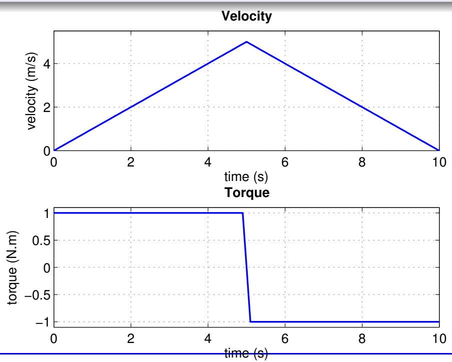
$$
\begin{aligned}
\min_{T, s(\cdot), \boldsymbol{\tau}(\cdot)} \quad & T, && (10)\\
\text{subject to } \quad & \boldsymbol{\tau}(t) = \mathbf{m}(s(t))\ddot{s}(t) + \mathbf{c}(s(t))\dot{s}(t)^2 + \mathbf{g}(s(t)), \\
& s(0) = 0 \text{ and } s(T) = 1, && (11)\\
& \dot{s}(0) = \dot{s}_0 \text{ and } \dot{s}(T) = \dot{s}_T, && (12)\\
& \dot{s}(t) \geq 0, && (13)\\
& \underline{\boldsymbol{\tau}}(s(t)) \leq \boldsymbol{\tau}(t) \leq \overline{\boldsymbol{\tau}}(s(t)), && (14)\\
& \text{for } t \in [0, T].
\end{aligned}
$$

Optimal control problem with 2 states s and $\dot{s}$ instead of $2 \times n$.

Time-optimal path tracking

Structure of the solution

- velocity along the path $\dot{s}$ as high as possible.
- at least one actuator saturated at each point on the path .
- fully determined by "switching points" along the path.



Existing solution methods

Existing solution methods:

Indirect methods

Numerical searches and forward and backward integration to determine "switching points"

- tedious implementation.
- exhaustive searches for candidate switching points.
- not very flexible.

Other methods

- Dynamic programming.
- Direct methods.

Outline

Reformulation

Nonlinear optimal control problem

$\Downarrow$

Convex optimal control problem

$\Downarrow$

Discrete large scale nonlinear optimization problem

$\Downarrow$

Discrete large scale SOCP

Convex reformulation

Convex reformulation:

- objective function can be rewritten as

$$T = \int_0^T 1 dt = \int_{s(0)}^{s(T)} \frac{1}{\dot{s}} ds = \int_0^1 \frac{1}{\dot{s}} ds. \tag{15}$$

- introduce optimization variables

$$a(s) = \ddot{s}, \tag{16}$$

$$b(s) = \dot{s}^2 \tag{17}$$

- additional linear constraint

$$b'(s) = 2a(s), \tag{18}$$

Convex reformulation

Convex reformulation

$$
\begin{aligned}
&\min_{a(\cdot),b(\cdot),\boldsymbol{\tau}(\cdot)} \int_0^1 \frac{1}{\sqrt{b(s)}} ds, && (19)\\
&\text{subject to } \boldsymbol{\tau}(s) = \mathbf{m}(s)a(s) + \mathbf{c}(s)b(s) + \mathbf{g}(s), && (20)\\
&\qquad b(0) = \dot{s}_0^2 \text{ and } b(1) = \dot{s}_T^2, && (21)\\
&\qquad b'(s) = 2a(s), && (22)\\
&\qquad b(s) \geq 0, && (23)\\
&\qquad \underline{\boldsymbol{\tau}}(s) \leq \boldsymbol{\tau}(s) \leq \overline{\boldsymbol{\tau}}(s), && (24)\\
&\qquad \text{for } s \in [0, 1].
\end{aligned}
$$

Convex objective function, linear constraints and time is eliminated.

Reformulation

Nonlinear optimal control problem

$\Downarrow$

Convex optimal control problem

$\Downarrow$

Discrete large scale nonlinear optimization problem

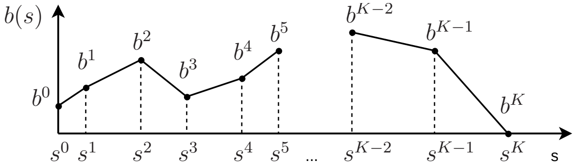
$\Downarrow$

Discrete large scale SOCP

Direct transcription

Direct transcription:

- discretize the path coordinate s on $[0, 1]$.
- discretize $b(s)$, $a(s)$ and $\tau_i(s)$.
- $b(s) = \dot{s}^2$ is assumed to be piecewise linear.



Direct transcription

- objective function can be approximated as

$$\int_0^1 \frac{1}{\sqrt{b(s)}} ds \approx \sum_{k=0}^{K-1} \frac{2\Delta s^k}{\sqrt{b^{k+1}} + \sqrt{b^k}}. \tag{25}$$

- differential constraint $b'(s) = 2a(s)$ is rewritten as

$$(b^{k+1} - b^k) = 2a^k \Delta s^k. \tag{26}$$

- ...

Direct transcription

Large-scale optimization problem

$$\min_{a^k, b^k, \boldsymbol{\tau}^k} \sum_{k=0}^{K-1} \frac{2\Delta s^k}{\sqrt{b^{k+1}} + \sqrt{b^k}},$$

$$\text{subject to } \boldsymbol{\tau}^k = \mathbf{m}(s^{k+1/2})a^k + \mathbf{c}(s^{k+1/2})b^{k+1/2} + \mathbf{g}(s^{k+1/2}),$$

$$b^0 = \dot{s}_0^2 \text{ and } b^K = \dot{s}_T^2, \tag{27}$$

$$(b^{k+1} - b^k) = 2a^k \Delta s^k, \tag{28}$$

$$b^k \geq 0 \text{ and } b^K \geq 0, \tag{29}$$

$$\underline{\boldsymbol{\tau}}(s^{k+1/2}) \leq \boldsymbol{\tau}^k \leq \overline{\boldsymbol{\tau}}(s^{k+1/2}), \tag{30}$$

$$\text{for } k = 0 \dots K-1.$$

Reformulation

Nonlinear optimal control problem

$\Downarrow$

Convex optimal control problem

$\Downarrow$

Discrete large scale nonlinear optimization problem

$\Downarrow$

Discrete large scale SOCP

SOCP formulation

Second-order cone program

Standard form (Boyd and Vandenberghe 2004):

$$\min_{\mathbf{x}} \ \mathbf{f}^T\mathbf{x}, \tag{31}$$

$$\text{subject to } \mathbf{F}\mathbf{x} = \mathbf{g}, \tag{32}$$

$$\|\mathbf{M}_j\mathbf{x} + \mathbf{n}_j\|_2 \leq \mathbf{p}_j^T\mathbf{x} + q_j, \tag{33}$$

$$\text{for } j = 1 \ldots m.$$

Linear objective and equality constraints, and second-order cone inequality constraints.

SOCP formulation

SOCP formulation:

- introduce variables d^k.
- new linear objective function

$$\sum_{k=0}^{K-1} 2\Delta s^k d^k. \tag{34}$$

- additional inequality constraints

$$\frac{1}{\sqrt{b^{k+1}} + \sqrt{b^k}} \leq d^k. \tag{35}$$

- can be replaced by two equivalent second-order cone constraints by introducing variables c^k.

SOCP formulation

SOCP formulation

$$\min_{a^k, b^k, \boldsymbol{\tau}^k, c^k, d^k} \sum_{k=0}^{K-1} 2\Delta s^k d^k, \tag{36}$$

$$\text{subject to } \boldsymbol{\tau}^k = \mathbf{m}(s^{k+1/2})a^k + \mathbf{c}(s^{k+1/2})b^{k+1/2} + \mathbf{g}(s^{k+1/2}),$$

$$b^0 = \dot{s}_0^2 \text{ and } b^K = \dot{s}_T^2, \tag{37}$$

$$(b^{k+1} - b^k) = 2a^k \Delta s^k, \tag{38}$$

$$\underline{\boldsymbol{\tau}}(s^{k+1/2}) \leq \boldsymbol{\tau}^k \leq \overline{\boldsymbol{\tau}}(s^{k+1/2}), \tag{39}$$

$$\left\| \begin{matrix} 2 \\ c^{k+1} + c^k - d^k \end{matrix} \right\|_2 \leq c^{k+1} + c^k + d^k, \tag{40}$$

$$\left\| \begin{matrix} 2c^k \\ b^k - 1 \end{matrix} \right\|_2 \leq b^k + 1. \tag{41}$$

SOCP formulation

OK, but...

- this sounds all very complicated...
- and why should I care?

Easy and efficient

Once the (complicated) formulation is done, implementation is

- very easy.
- automatically efficient.
- requires almost no optimization knowledge.

Matlab implementation

Matlab implementation using YALMIP (Löfberg 2001):

- objective function $\sum_{k=0}^{K-1} 2\Delta s^k d^k$
  t = sum(2*dvar.*ds);
- constraints

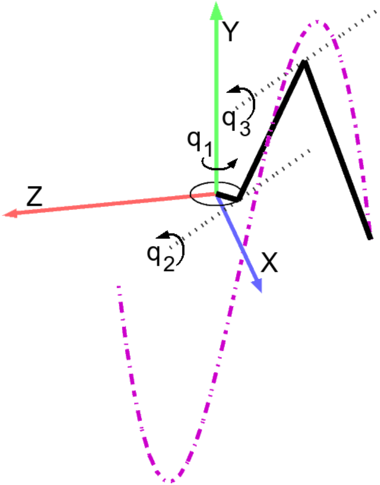
$$\left\| \begin{array}{c} 2c^k \\ b^k - 1 \end{array} \right\|_2 \leq b^k + 1 \qquad (42)$$

  F = F + set(cone([2*cvar(1,k); bvar(1,k)-1], bvar(1,k)+1));
- solve with SeDuMi (Sturm 1999)
  diagnostics = solvesdp(F,t);
- no numerical "hassle": tolerances, scaling, ...
- implementation straightforward, solution automatically efficient.

Outline

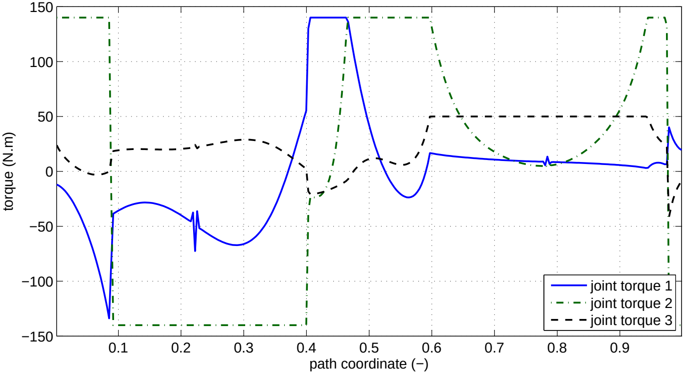
Three-DOF elbow manipulator

Example in (Pfeiffer and Johanni 1987):



Three-DOF elbow manipulator

Torques:



Three-DOF elbow manipulator

Numerical jitter

- because solution is not uniquely defined (singular point).
- major problem for indirect methods. Singular arcs can result in infinitely many "switching points".
- not a problem for the solver.

Undesirable in practice

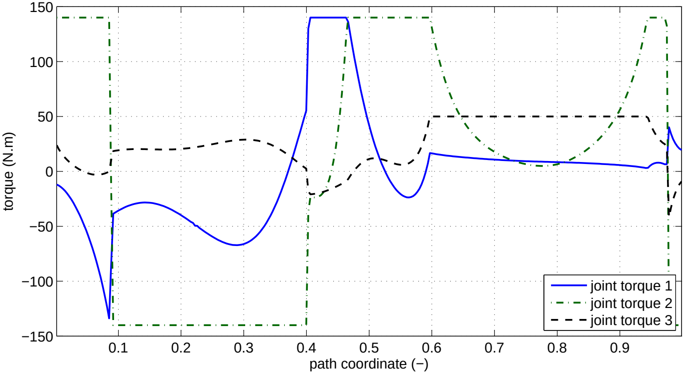
- just add a suitable penalty term to the objective function

$$\int_0^T |\dot{\tau}_i(s)|\, dt = \int_0^1 \frac{|\tau_i'(s)\dot{s}|}{\dot{s}} ds = \int_0^1 \left|\tau_i'(s)\right| ds, \qquad (43)$$

- still SOCP.

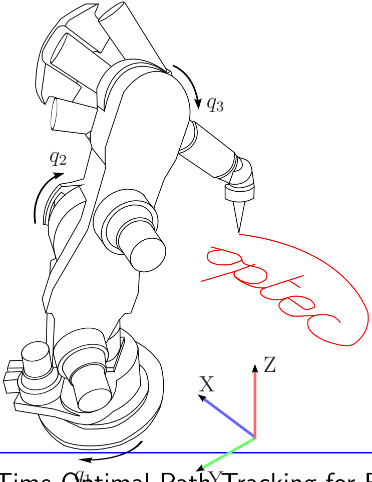
Three-DOF elbow manipulator

Torques with penalization of torque jumps:



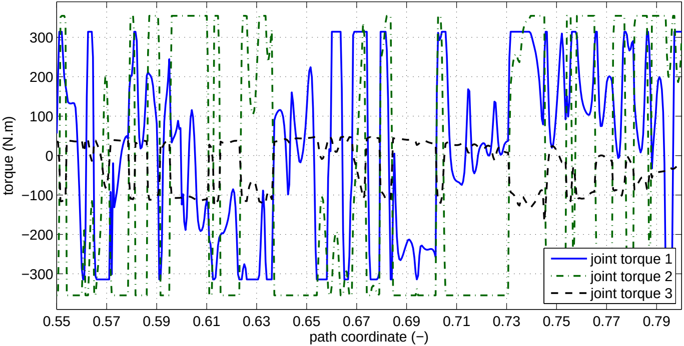
Complicated example

A more complicated example: a six-DOF manipulator carrying out an non-smooth path.



Complicated example

Torques: complicated and a lot of switching!



2000 grid points. Solver time 2.87 s on a Pentium 4 running at 3.60 GHz. (Video)

Complicated example

Time-optimal path tracking

In practice, the pure time-optimal solution is too agressive.

Smoothing

- easy to incorporate penalty terms, such as actuator energy (integral of torques squared).
- still SOCP.
- other constraints and objective functions also possible (though not necessarily SOCP or convex).

Outline

1. Introduction
   - Time-optimal motion planning
   - Convex optimization
2. Time-optimal path tracking
   - Problem formulation
   - Existing solution methods
3. Reformulation and solution method
   - Convex reformulation
   - Direct transcription
   - SOCP formulation
   - Implementation
4. Examples
5. Conclusions

Conclusions

Conclusions:

- convex optimization approach to time-optimal path tracking.
- easy to implement.
- very efficient.
- flexible.

Future work:

- constraints on the rate of change of the torques.
- explore various applications, such as minimization of reaction forces at the robot base.
- integrate with path planning (iterative/sequential)

References

Carried out in the context of OPTEC:

- Optimization in Engineering Center: interdisciplinary research on development of efficient optimization algorithms for engineering applications.
- free matlab implementation available: `http://homes.esat.kuleuven.be/~optec/software/timeopt/`
- D. Verscheure, B. Demeulenaere, J. Swevers, J. De Schutter, M. Diehl, "Practical Time-Optimal Trajectory Planning for Robots: a Convex Optimization Approach", IEEE Transactions on Automatic Control, submitted

Thank you for your attention

- Thank you for your attention!
- Comments/Suggestions/Questions?

Boyd, S. and L. Vandenberghe (2004).
Convex Optimization.
Cambridge University Press.

Choset, H., W. Burgard, S. Hutchinson, G. Kantor, L. E. Kavraki, K. Lynch, and S. Thrun (2005).
Principles of Robot Motion: Theory, Algorithms, and Implementation.
MIT Press.

Löfberg, J. (2001).
YALMIP, Yet another LMI parser.
Sweden: University of Linköping.
Available: `http://www.control.isy.liu.se/$\sim$johanl`.

Pfeiffer, F. and R. Johanni (1987).
A concept for manipulator trajectory planning.
IEEE Journal of Robotics and Automation RA-3(2), 115–123.

Sturm, J. (1999).

Using SeDuMi: a Matlab toolbox for optimization over symmetric cones.
Optimization Methods and Software, 11–12.