Solution for Exercise 10: ACADO code generation for Nonlinear MPC

TEMPO Summer School on Numerical Optimal Control and Embedded Optimization
University of Freiburg, July 27 - August 7, 2015
Rien Quirynen, Dimitris Kouzoupis, Joachim Ferreau and Moritz Diehl

Contents

Swing-up of an inverted pendulum

```
clc;
clear all;
close all;

Ts = 0.05;
EXPORT = 1;

DifferentialState x1 theta v1 dtheta;
Control F;

M = 1;
m = 0.1;
g = 9.81;
l = 0.8;

% Differential Equation

f_expl = [   dot(x1) == v1; ...
             dot(theta) == dtheta; ...
             dot(v1) == (- l*m*sin(theta)*dtheta^2 + F + g*m*cos(theta)*sin(theta))/(M
+ m - m*cos(theta)^2); ...
             dot(dtheta) == (- l*m*cos(theta)*sin(theta)*dtheta^2 + F*cos(theta) + g*m*
sin(theta) + M*g*sin(theta))/(l*(M + m - m*cos(theta)^2)) ];

% SIMexport
acadoSet('problemname', 'sim');

numSteps = 2;
sim = acado.SIMexport( Ts );
sim.setModel(f_expl);
sim.set( 'INTEGRATOR_TYPE',             'INT_IRK_GL2' );
sim.set( 'NUM_INTEGRATOR_STEPS',        numSteps        );

if EXPORT
    sim.exportCode( 'export_SIM' );

    cd export_SIM
    make_acado_integrator('../integrate_pendulum')
    cd ..
end

% MPCexport
acadoSet('problemname', 'mpc');

N = 40;
```

```
ocp = acado.OCP( 0.0, N*Ts, N );

h = [x1 theta v1 dtheta F];
hN = [x1 theta v1 dtheta];

W = acado.BMatrix(eye(length(h)));
WN = acado.BMatrix(eye(length(hN)));

ocp.minimizeLSQ( W, h );
ocp.minimizeLSQEndTerm( WN, hN );

xmin = -2; xmax = 2;
Fmin = -20; Fmax = 20;

ocp.subjectTo( xmin <= x1 <= xmax );
ocp.subjectTo( Fmin <= F <= Fmax );
% ocp.subjectTo( 'AT_END', [x1 theta v1 dtheta] == 0 );

ocp.setModel(f_expl);

mpc = acado.OCPexport( ocp );
mpc.set( 'HESSIAN_APPROXIMATION',       'GAUSS_NEWTON'      );
mpc.set( 'DISCRETIZATION_TYPE',         'MULTIPLE_SHOOTING' );
mpc.set( 'SPARSE_QP_SOLUTION',          'FULL_CONDENSING_N2');
mpc.set( 'LEVENBERG_MARQUARDT',          1e-5               );
mpc.set( 'INTEGRATOR_TYPE',             'INT_RK4'           );
mpc.set( 'NUM_INTEGRATOR_STEPS',        2*N                 );
mpc.set( 'QP_SOLVER',                   'QP_QPOASES'            );

if EXPORT
    mpc.exportCode( 'export_MPC' );

    global ACADO_;
    copyfile([ACADO_.pwd '/../../external_packages/qpoases'], 'export_MPC/qpoases')

    cd export_MPC
    make_acado_solver('../acado_MPCstep')
    cd ..
end

% PARAMETERS SIMULATION
X0 = [0 pi 0 0];
Xref = [0 0 0 0];
input.x = [repmat(X0,N/2,1); repmat(Xref,N/2+1,1)];
input.od = [];

Uref = zeros(N,1);
input.u = Uref;

input.y = [repmat(Xref,N,1) Uref];
input.yN = Xref.';

input.W = diag([5e-1 1 2e-3 2e-3 1e-4]);
input.WN = diag([5e-1 1 2e-3 2e-3]);

input.shifting.strategy = 1;

% SIMULATION LOOP
display('------------------------------------------------------------------')
display('               Simulation Loop'                                    )
display('------------------------------------------------------------------')
```

```
iter = 0; time = 0;
Tf = 4;
KKT_MPC = []; INFO_MPC = [];
controls_MPC = [];
state_sim = X0;

while time(end) < Tf
    tic
    % Solve NMPC OCP
    input.x0 = state_sim(end,:).';
    output = acado_MPCstep(input);

    % Save the MPC step
    INFO_MPC = [INFO_MPC; output.info];
    KKT_MPC = [KKT_MPC; output.info.kktValue];
    controls_MPC = [controls_MPC; output.u(1,:)];

    input.x = output.x;
    input.u = output.u;

    % Simulate system
    sim_input.x = state_sim(end,:).';
    sim_input.u = output.u(1,:).';
    sim_input.od = 0.2;
    states = integrate_pendulum(sim_input);
    state_sim = [state_sim; states.value'];

    iter = iter+1;
    nextTime = iter*Ts;
    disp(['current time: ' num2str(nextTime) '    ' char(9) ' (RTI step -- QP error: ' n
um2str(output.info.status) ',' ' ' char(2) ' KKT val: ' num2str(output.info.kktValue,'%
1.2e') ',' ' ' char(2) ' CPU time: ' num2str(round(output.info.cpuTime*1e6)) ' µs)'])
    time = [time nextTime];

    visualize(time, state_sim, Xref, xmin, xmax, l);
end

figure;
subplot(2,2,1);
plot(time, state_sim(:,1)); hold on;
plot([0 time(end)], [0 0], 'r:');
plot([0 time(end)], [xmin xmin], 'g--');
plot([0 time(end)], [xmax xmax], 'g--');
xlabel('time(s)');
ylabel('x');
ylim([1.5*xmin 1.5*xmax])

subplot(2,2,2);
plot(time, state_sim(:,2)); hold on;
plot([0 time(end)], [0 0], 'r:');
xlabel('time(s)');
ylabel('theta');

subplot(2,2,[3 4]);
stairs(time(1:end-1), controls_MPC); hold on;
plot([0 time(end)], [0 0], 'r:');
plot([0 time(end)], [Fmin Fmin], 'g--');
plot([0 time(end)], [Fmax Fmax], 'g--');
xlabel('time(s)');
ylabel('F');
```

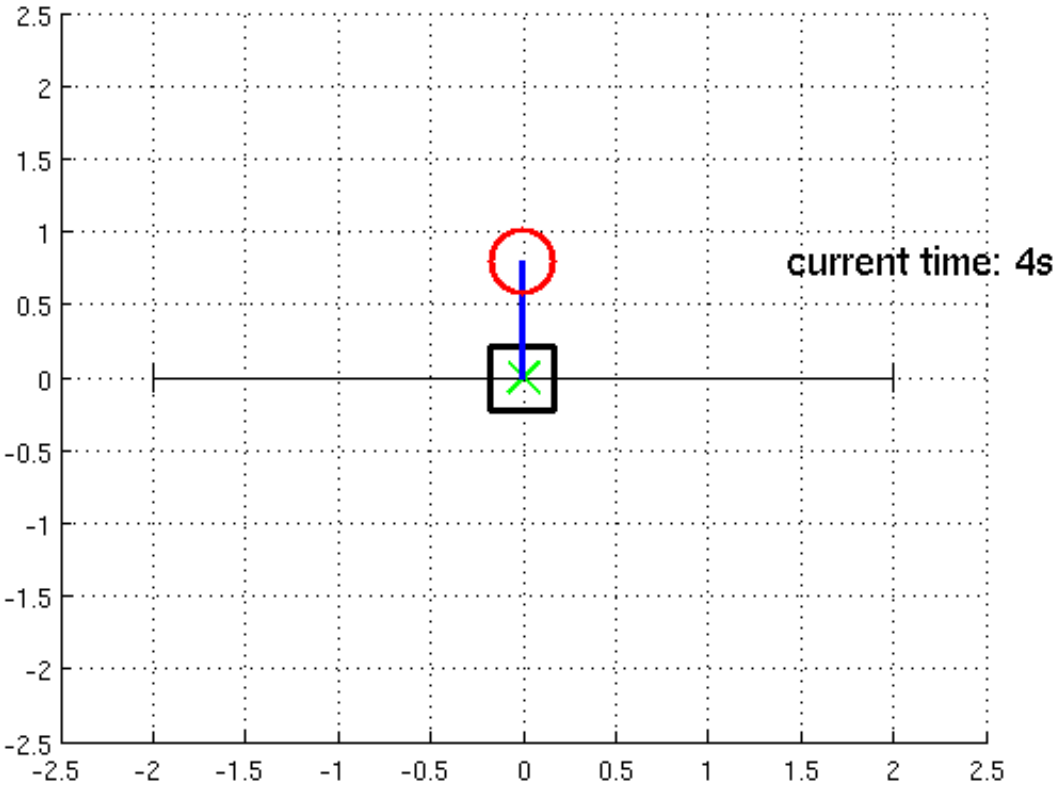
```
ylim([1.5*Fmin 1.5*Fmax])


figure;
semilogy(time(1:end-1), KKT_MPC, ':bx');
xlabel('time(s)')
ylabel('KKT')
```

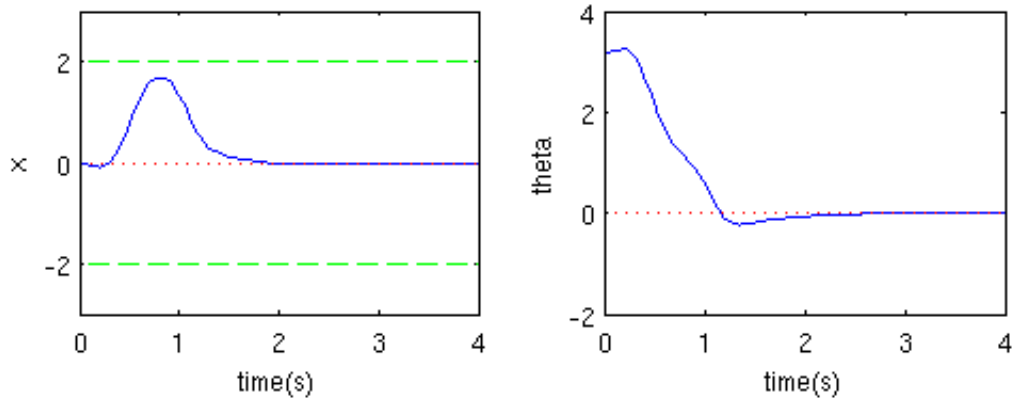
```
Writing c++ files...
Compiling c++ files...
Building with 'g++'.
MEX completed successfully.

ACADO Toolkit for Matlab - Developed by David Ariens and Rien Quirynen, 2009-2013
Support available at http://www.acadotoolkit.org/matlab

[ACADO] Information: Code generation successful


compiling... Building with 'gcc'.
MEX completed successfully.
done! --> ../integrate_pendulum.mexa64
Writing c++ files...
Compiling c++ files...
Building with 'g++'.
MEX completed successfully.

ACADO Toolkit for Matlab - Developed by David Ariens and Rien Quirynen, 2009-2013
Support available at http://www.acadotoolkit.org/matlab


ACADO Toolkit::Code Generation Tool
Copyright (C) 2008-2014 by Boris Houska, Hans Joachim Ferreau,
Milan Vukov, Rien Quirynen, KU Leuven.
Developed within the Optimization in Engineering Center (OPTEC)
under supervision of Moritz Diehl. All rights reserved.

ACADO Toolkit is distributed under the terms of the GNU Lesser
General Public License 3 in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU Lesser General Public License for more details.

compiling... Building with 'g++'.
/mnt/HDD-home/rien/Google Drive/GIT/tempo-noc/exercises/e10_ACADO/solution/export_MPC/a
cado_solver_mex.c: In function ‘int getArray(unsigned int, const mxArray*, int, const c
har*, real_t*, unsigned int, unsigned int)’:
/mnt/HDD-home/rien/Google Drive/GIT/tempo-noc/exercises/e10_ACADO/solution/export_MPC/a
cado_solver_mex.c:181:47: warning: deprecated conversion from string constant to ‘char*
’ [-Wwrite-strings]
    mexErrMsgTxtAdv("Field %s not found.", name);
                                           ^
/mnt/HDD-home/rien/Google Drive/GIT/tempo-noc/exercises/e10_ACADO/solution/export_MPC/a
cado_solver_mex.c:185:64: warning: deprecated conversion from string constant to ‘char*
’ [-Wwrite-strings]
   mexErrMsgTxtAdv("Field %s must be an array of doubles.", name);
                                                            ^
/mnt/HDD-home/rien/Google Drive/GIT/tempo-noc/exercises/e10_ACADO/solution/export_MPC/a
cado_solver_mex.c:188:75: warning: deprecated conversion from string constant to ‘char*
```

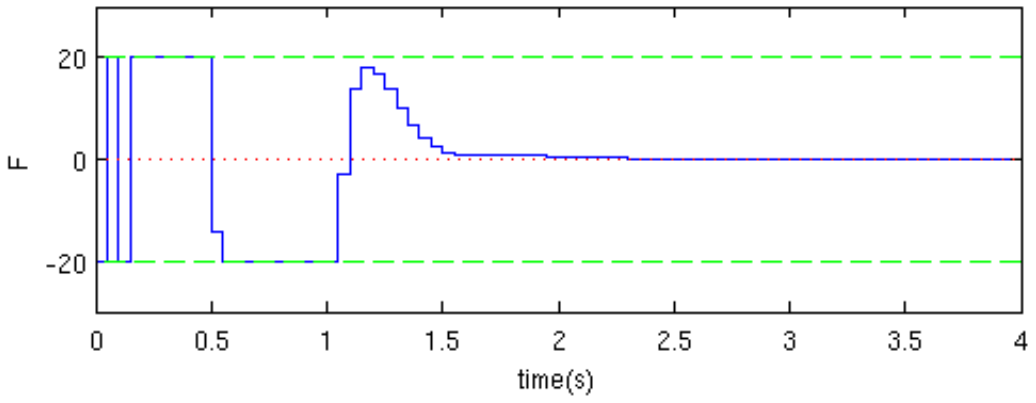
```
’ [-Wwrite-strings]
   mexErrMsgTxtAdv("Field %s must be of size: %d x %d.", name, nRows, nCols);
                                                                          ^


MEX completed successfully.
done! --> ../acado_MPCstep.mexa64
-----------------------------------------------------------------
               Simulation Loop
-----------------------------------------------------------------
current time: 0.05       (RTI step -- QP error: 0,  KKT val: 1.29e+04,  CPU time: 1144
µs)
current time: 0.1        (RTI step -- QP error: 0,  KKT val: 2.38e+03,  CPU time: 636 µ
s)
current time: 0.15       (RTI step -- QP error: 0,  KKT val: 5.23e+03,  CPU time: 566 µ
s)
current time: 0.2        (RTI step -- QP error: 0,  KKT val: 6.09e+03,  CPU time: 577 µ
s)
current time: 0.25       (RTI step -- QP error: 0,  KKT val: 3.63e+04,  CPU time: 549 µ
s)
current time: 0.3        (RTI step -- QP error: 0,  KKT val: 1.18e+03,  CPU time: 531 µ
s)
current time: 0.35       (RTI step -- QP error: 0,  KKT val: 9.45e-02,  CPU time: 518 µ
s)
current time: 0.4        (RTI step -- QP error: 0,  KKT val: 8.35e-02,  CPU time: 464 µ
s)
current time: 0.45       (RTI step -- QP error: 0,  KKT val: 1.83e-01,  CPU time: 440 µ
s)
current time: 0.5        (RTI step -- QP error: 0,  KKT val: 7.29e-01,  CPU time: 416 µ
s)
current time: 0.55       (RTI step -- QP error: 0,  KKT val: 2.00e+00,  CPU time: 383 µ
s)
current time: 0.6        (RTI step -- QP error: 0,  KKT val: 9.39e-02,  CPU time: 395 µ
s)
current time: 0.65       (RTI step -- QP error: 0,  KKT val: 2.10e-03,  CPU time: 369 µ
s)
current time: 0.7        (RTI step -- QP error: 0,  KKT val: 9.82e-03,  CPU time: 350 µ
s)
current time: 0.75       (RTI step -- QP error: 0,  KKT val: 3.29e-02,  CPU time: 341 µ
s)
current time: 0.8        (RTI step -- QP error: 0,  KKT val: 6.75e-02,  CPU time: 316 µ
s)
current time: 0.85       (RTI step -- QP error: 0,  KKT val: 1.19e-01,  CPU time: 296 µ
s)
current time: 0.9        (RTI step -- QP error: 0,  KKT val: 1.95e-01,  CPU time: 444 µ
s)
current time: 0.95       (RTI step -- QP error: 0,  KKT val: 3.06e-01,  CPU time: 263 µ
s)
current time: 1          (RTI step -- QP error: 0,  KKT val: 4.71e-01,  CPU time: 250 µ
s)
current time: 1.05       (RTI step -- QP error: 0,  KKT val: 7.15e-01,  CPU time: 220 µ
s)
current time: 1.1        (RTI step -- QP error: 0,  KKT val: 1.07e+00,  CPU time: 208 µ
s)
current time: 1.15       (RTI step -- QP error: 0,  KKT val: 4.45e-02,  CPU time: 206 µ
s)
current time: 1.2        (RTI step -- QP error: 0,  KKT val: 8.84e-02,  CPU time: 204 µ
s)
current time: 1.25       (RTI step -- QP error: 0,  KKT val: 1.11e-01,  CPU time: 198 µ
s)
current time: 1.3        (RTI step -- QP error: 0,  KKT val: 7.87e-02,  CPU time: 287 µ
s)
```

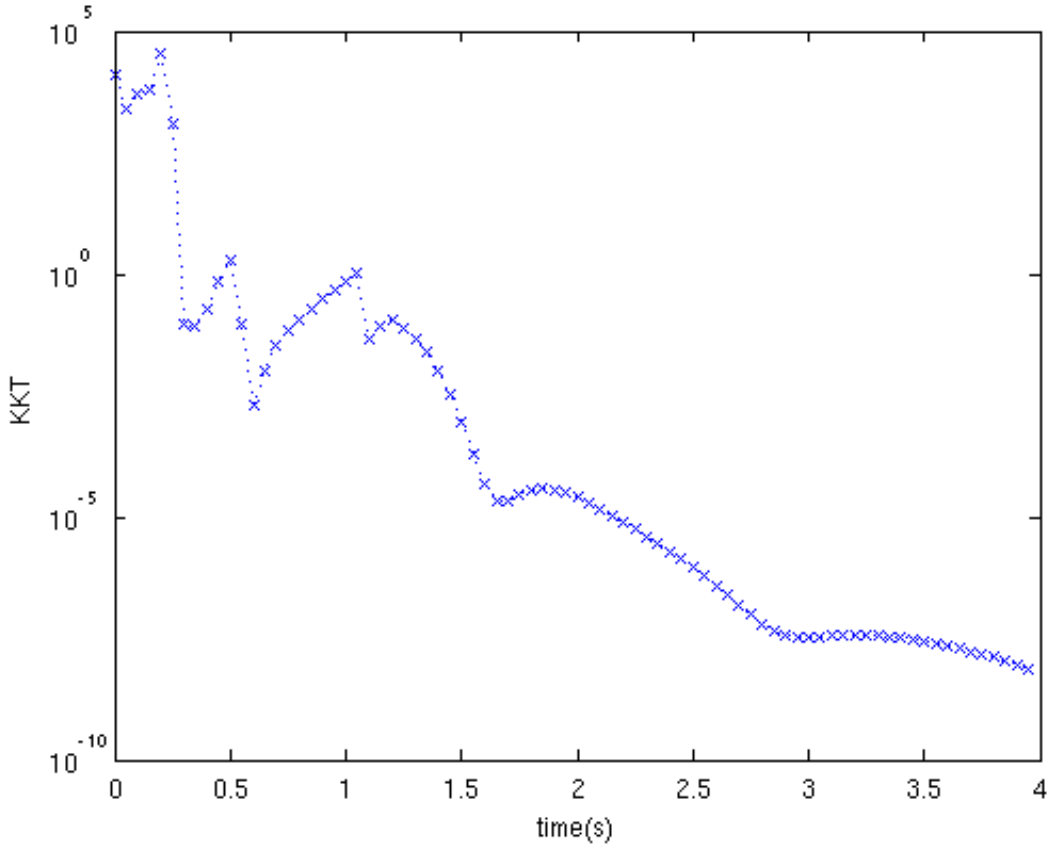
```
current time: 1.35       (RTI step -- QP error: 0,  KKT val: 4.79e-02,  CPU time: 208 µ
s)
current time: 1.4        (RTI step -- QP error: 0,  KKT val: 2.42e-02,  CPU time: 210 µ
s)
current time: 1.45       (RTI step -- QP error: 0,  KKT val: 9.87e-03,  CPU time: 200 µ
s)
current time: 1.5        (RTI step -- QP error: 0,  KKT val: 3.24e-03,  CPU time: 209 µ
s)
current time: 1.55       (RTI step -- QP error: 0,  KKT val: 8.63e-04,  CPU time: 200 µ
s)
current time: 1.6        (RTI step -- QP error: 0,  KKT val: 1.96e-04,  CPU time: 197 µ
s)
current time: 1.65       (RTI step -- QP error: 0,  KKT val: 4.83e-05,  CPU time: 197 µ
s)
current time: 1.7        (RTI step -- QP error: 0,  KKT val: 2.20e-05,  CPU time: 215 µ
s)
current time: 1.75       (RTI step -- QP error: 0,  KKT val: 2.17e-05,  CPU time: 219 µ
s)
current time: 1.8        (RTI step -- QP error: 0,  KKT val: 2.84e-05,  CPU time: 198 µ
s)
current time: 1.85       (RTI step -- QP error: 0,  KKT val: 3.51e-05,  CPU time: 205 µ
s)
current time: 1.9        (RTI step -- QP error: 0,  KKT val: 3.78e-05,  CPU time: 209 µ
s)
current time: 1.95       (RTI step -- QP error: 0,  KKT val: 3.60e-05,  CPU time: 205 µ
s)
current time: 2          (RTI step -- QP error: 0,  KKT val: 3.12e-05,  CPU time: 207 µ
s)
current time: 2.05       (RTI step -- QP error: 0,  KKT val: 2.53e-05,  CPU time: 206 µ
s)
current time: 2.1        (RTI step -- QP error: 0,  KKT val: 1.95e-05,  CPU time: 203 µ
s)
current time: 2.15       (RTI step -- QP error: 0,  KKT val: 1.46e-05,  CPU time: 213 µ
s)
current time: 2.2        (RTI step -- QP error: 0,  KKT val: 1.08e-05,  CPU time: 204 µ
s)
current time: 2.25       (RTI step -- QP error: 0,  KKT val: 7.80e-06,  CPU time: 205 µ
s)
current time: 2.3        (RTI step -- QP error: 0,  KKT val: 5.60e-06,  CPU time: 205 µ
s)
current time: 2.35       (RTI step -- QP error: 0,  KKT val: 3.99e-06,  CPU time: 205 µ
s)
current time: 2.4        (RTI step -- QP error: 0,  KKT val: 2.81e-06,  CPU time: 198 µ
s)
current time: 2.45       (RTI step -- QP error: 0,  KKT val: 1.95e-06,  CPU time: 263 µ
s)
current time: 2.5        (RTI step -- QP error: 0,  KKT val: 1.34e-06,  CPU time: 196 µ
s)
current time: 2.55       (RTI step -- QP error: 0,  KKT val: 9.04e-07,  CPU time: 260 µ
s)
current time: 2.6        (RTI step -- QP error: 0,  KKT val: 5.98e-07,  CPU time: 206 µ
s)
current time: 2.65       (RTI step -- QP error: 0,  KKT val: 3.88e-07,  CPU time: 273 µ
s)
current time: 2.7        (RTI step -- QP error: 0,  KKT val: 2.46e-07,  CPU time: 195 µ
s)
current time: 2.75       (RTI step -- QP error: 0,  KKT val: 1.54e-07,  CPU time: 204 µ
s)
current time: 2.8        (RTI step -- QP error: 0,  KKT val: 9.67e-08,  CPU time: 197 µ
s)
current time: 2.85       (RTI step -- QP error: 0,  KKT val: 6.27e-08,  CPU time: 197 µ
```

```
s)
current time: 2.9        (RTI step -- QP error: 0,  KKT val: 4.42e-08,  CPU time: 198 µ
s)
current time: 2.95       (RTI step -- QP error: 0,  KKT val: 3.54e-08,  CPU time: 193 µ
s)
current time: 3          (RTI step -- QP error: 0,  KKT val: 3.24e-08,  CPU time: 202 µ
s)
current time: 3.05       (RTI step -- QP error: 0,  KKT val: 3.24e-08,  CPU time: 206 µ
s)
current time: 3.1        (RTI step -- QP error: 0,  KKT val: 3.38e-08,  CPU time: 200 µ
s)
current time: 3.15       (RTI step -- QP error: 0,  KKT val: 3.54e-08,  CPU time: 196 µ
s)
current time: 3.2        (RTI step -- QP error: 0,  KKT val: 3.67e-08,  CPU time: 201 µ
s)
current time: 3.25       (RTI step -- QP error: 0,  KKT val: 3.73e-08,  CPU time: 197 µ
s)
current time: 3.3        (RTI step -- QP error: 0,  KKT val: 3.71e-08,  CPU time: 208 µ
s)
current time: 3.35       (RTI step -- QP error: 0,  KKT val: 3.62e-08,  CPU time: 195 µ
s)
current time: 3.4        (RTI step -- QP error: 0,  KKT val: 3.47e-08,  CPU time: 205 µ
s)
current time: 3.45       (RTI step -- QP error: 0,  KKT val: 3.27e-08,  CPU time: 194 µ
s)
current time: 3.5        (RTI step -- QP error: 0,  KKT val: 3.03e-08,  CPU time: 193 µ
s)
current time: 3.55       (RTI step -- QP error: 0,  KKT val: 2.77e-08,  CPU time: 198 µ
s)
current time: 3.6        (RTI step -- QP error: 0,  KKT val: 2.50e-08,  CPU time: 193 µ
s)
current time: 3.65       (RTI step -- QP error: 0,  KKT val: 2.23e-08,  CPU time: 195 µ
s)
current time: 3.7        (RTI step -- QP error: 0,  KKT val: 1.97e-08,  CPU time: 205 µ
s)
current time: 3.75       (RTI step -- QP error: 0,  KKT val: 1.72e-08,  CPU time: 197 µ
s)
current time: 3.8        (RTI step -- QP error: 0,  KKT val: 1.49e-08,  CPU time: 204 µ
s)
current time: 3.85       (RTI step -- QP error: 0,  KKT val: 1.28e-08,  CPU time: 209 µ
s)
current time: 3.9        (RTI step -- QP error: 0,  KKT val: 1.09e-08,  CPU time: 209 µ
s)
current time: 3.95       (RTI step -- QP error: 0,  KKT val: 9.17e-09,  CPU time: 205 µ
s)
current time: 4          (RTI step -- QP error: 0,  KKT val: 7.67e-09,  CPU time: 198 µ
s)
```


current time: 4s


x
time(s)
theta
time(s)

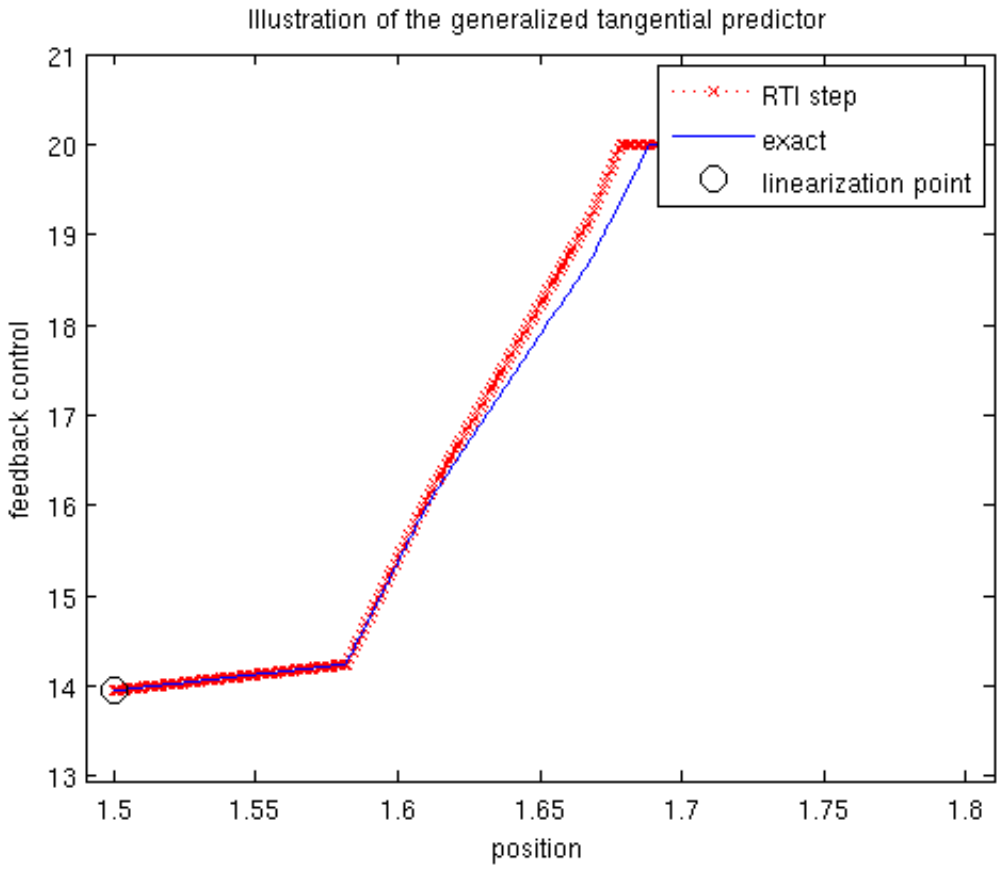

F
time(s)



Generalized tangential predictor of RTI for Nonlinear MPC

```
clc;
clear all;
close all;

% Define parameters:
N = 40;
Xref = [0 0 0 0];
X0 = [1.5 -0.2 0 0].';
input.x = repmat(X0.',N+1,1);
input.x0 = X0;

Uref = zeros(N,1);
input.u = Uref;

input.y = [repmat(Xref,N,1) Uref];
input.yN = Xref.';

input.W = diag([5e-1 1 5e-2 5e-2 2e-2]);
input.WN = diag([5e-1 1 5e-2 5e-2]);

% Converge the SQP method as the linearization point:
for k = 1:30
    output = acado_MPCstep(input);
    input.x = output.x;
    input.u = output.u;
end
X_RTI = output.x; U_RTI = output.u;

cost_RTI = []; u_RTI = [];
cost_SQP = []; u_SQP = [];
p_values = 1.5:0.001:1.8;
```

```
for p = p_values
    X0 = [p -0.2 0 0].';
    input.x0 = X0;

    % Perform one RTI step:
    input.x = X_RTI; input.u = U_RTI; % NOTE: we use the same, original linearization p
oint
    output = acado_MPCstep(input);
    cost_RTI = [cost_RTI; output.info.objValue];
    u_RTI = [u_RTI; output.u(1)];
    if output.info.status ~= 0
       error(['output status = ' num2str(output.info.status) ' for p = ' num2str(p)])
    end

    % Perform a fixed amount of 30 SQP steps:
    for k = 1:30
        output = acado_MPCstep(input);
        input.x = output.x;
        input.u = output.u;
    end
    cost_SQP = [cost_SQP; output.info.objValue];
    u_SQP = [u_SQP; output.u(1)];
    if output.info.status ~= 0
       error(['output status = ' num2str(output.info.status) ' for p = ' num2str(p)])
    end
end

figure;
plot(p_values, u_RTI, ':rx'); hold on;
plot(p_values, u_SQP, '-b');
plot(p_values(1), u_SQP(1), 'ko', 'MarkerSize', 10)
xlabel('position');
ylabel('feedback control')
legend('RTI step', 'exact', 'linearization point');
xlim([min(p_values)-0.01 max(p_values)+0.01])
ylim([min(u_SQP)-1 max(u_SQP)+1]);
title('Illustration of the generalized tangential predictor')
```



Published with MATLAB® R2014a