Contents

q7

```
clc
clear
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Francesco Borrelli ME C231A 2015
% Kinematic Navigation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tic
N=50;
sampling=10;
%Var Defintions
z = sdpvar(2,N);

%Initial and terminal condition
z0 = [0;1];
zT = [850;1];
dzmin=-[20;2];
dzmax=[20;2];
zmin = [0;0];
zmax = [1000;7];

%Obstacle list
i=1;
obs{i}.center=[400;1];
obs{i}.LW=[200;2];
obs{i}.theta=0; %(in radiants)
i=i+1;
obs{i}.center=[800;5];
obs{i}.LW=[400;4];
obs{i}.theta=0; %(in radiants)


% some obtacle postprocessing
for j=1:length(obs)
    t=obs{j}.theta;
    % generate T matrix for each obstacle
    obs{j}.T=[cos(t), -sin(t);sin(t) cos(t)]*diag(obs{j}.LW/2);
    % polyehdral representaion
    obs{j}.poly=obs{j}.T*unitbox(2)+obs{j}.center;
end


%try to remove/add this one


%Constraints
%Setup Optimization Problem
```

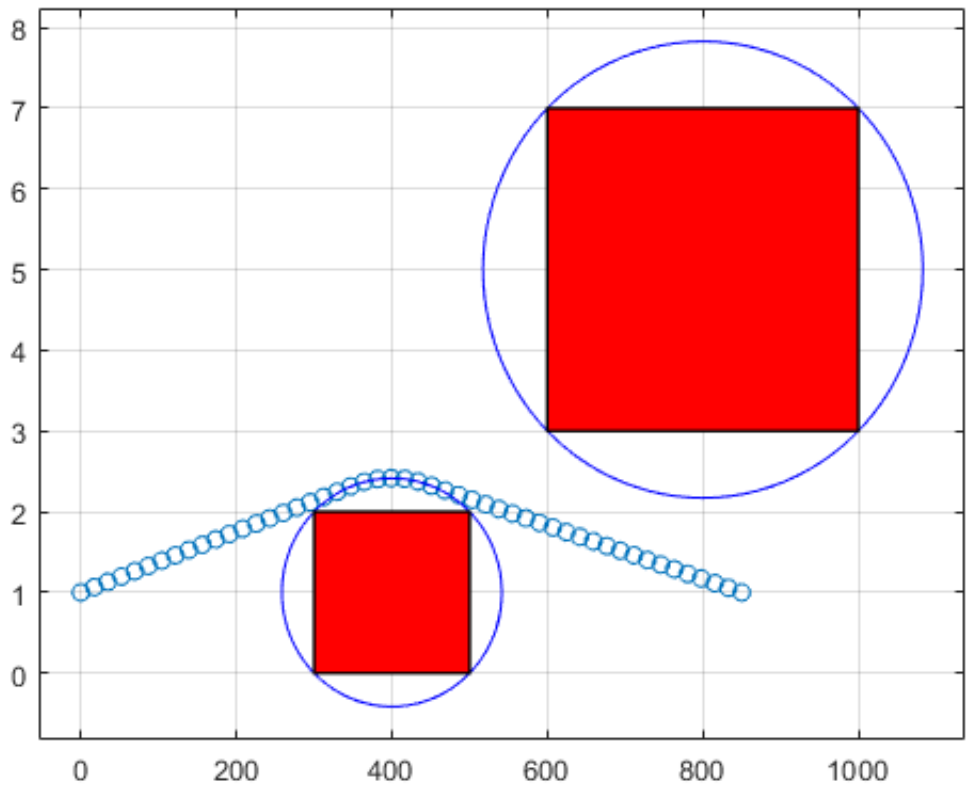
```
cost = 0;
Q=eye(2);
constr = [z(:,1)==z0,z(:,N)==zT];
for t = 2:N
      cost=cost+(z(:,t)-z(:,t-1))'*Q*(z(:,t)-z(:,t-1));
      constr = constr +[dzmin<= z(:,t)-z(:,t-1)<=dzmax];
      constr = constr + [zmin<=z(:,t)<= zmax];
      for k = 0:sampling-1
          for j=1:length(obs)
              xs=z(:,t-1)+k/sampling*(z(:,t)-z(:,t-1));
              constr = constr +[(xs-obs{j}.center)'*inv(obs{j}.T)'*inv(obs{j}.T)*(xs-obs{j
}.center)>=2];
          end
      end
end
options = sdpsettings('solver','ipopt');
%options.ipopt=ipoptset('linear_solver','MUMPS');
solvesdp(constr,cost,options);
z_vec = double(z);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Plotting Functions % to add title and labels
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
th = 0:pi/50:2*pi;
for j=1:length(obs)
    for l=1:length(th)
        z=[cos(th(l));sin(th(l))]*sqrt(2);
        y=obs{j}.T*z+obs{j}.center;
        xobs{j}(l) = y(1);
        yobs{j}(l) = y(2);
    end
end
```

```
Total number of variables.............................:       96
                     variables with only lower bounds:        0
                variables with lower and upper bounds:       96
                     variables with only upper bounds:        0
Total number of equality constraints.................:        0
Total number of inequality constraints...............:     1176
        inequality constraints with only lower bounds:        0
   inequality constraints with lower and upper bounds:        0
        inequality constraints with only upper bounds:     1176


Number of Iterations....: 244

                                   (scaled)                 (unscaled)
Objective...............:   1.4745068842705978e+03    1.4745068842705978e+04
Dual infeasibility......:   9.8279349396820243e-08    9.8279349396820243e-07
Constraint violation....:   0.0000000000000000e+00    0.0000000000000000e+00
Complementarity.........:   1.0000000000000004e-11    1.0000000000000004e-10
Overall NLP error.......:   9.8279349396820243e-08    9.8279349396820243e-07
```

```
Number of objective function evaluations             = 434
Number of objective gradient evaluations             = 245
Number of equality constraint evaluations            = 0
Number of inequality constraint evaluations          = 434
Number of equality constraint Jacobian evaluations   = 0
Number of inequality constraint Jacobian evaluations = 245
Number of Lagrangian Hessian evaluations             = 0
Total CPU secs in IPOPT (w/o function evaluations)   =      0.745
Total CPU secs in NLP function evaluations           =      0.301

EXIT: Optimal Solution Found.
```

plot routine

```
figure
axis([zmin(1) zmax(1) zmin(2) zmax(2)])
plot(z_vec(1,:),z_vec(2,:),'o')
hold on
for j=1:length(obs)
plot(xobs{j}, yobs{j},'b');
plot(obs{j}.T*unitbox(2)+obs{j}.center);
end
```



Comment

```
toc
% the computation time is faster than RRT in most cases.
```

```
Elapsed time is 0.501128 seconds.
```

Published with MATLAB® R2018a

Contents

q8

```
clc
clear
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Francesco Borrelli ME C231A 2015
% Kinematic Navigation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tic
N=50;
sampling=10;
%Var Defintions
z = sdpvar(2,N);

%Initial and terminal condition
z0 = [0;1];
zT = [850;1];
dzmin=-[20;2];
dzmax=[20;2];
zmin = [0;0];
zmax = [1000;7];

%Obstacle list
%Obstacle list
i=1;
obs{i}.center=[400;1];
obs{i}.LW=[200;2];
obs{i}.theta=0; %(in radiants)
i=i+1;
obs{i}.center=[800;5];
obs{i}.LW=[400;4];
obs{i}.theta=0; %(in radiants)


% integer variables
d = binvar(4*length(obs),(N-1)*sampling);
% bigM constant
bM=1000;


% some obstacle postprocessing
for j=1:length(obs)
    t=obs{j}.theta;
    % generate T matrix for each obstacle
    obs{j}.T=[cos(t), -sin(t);sin(t) cos(t)]*diag(obs{j}.LW/2);
    % polyehdral representaion
    obs{j}.poly=obs{j}.T*unitbox(2)+obs{j}.center;
end

%try to remove/add this one
```

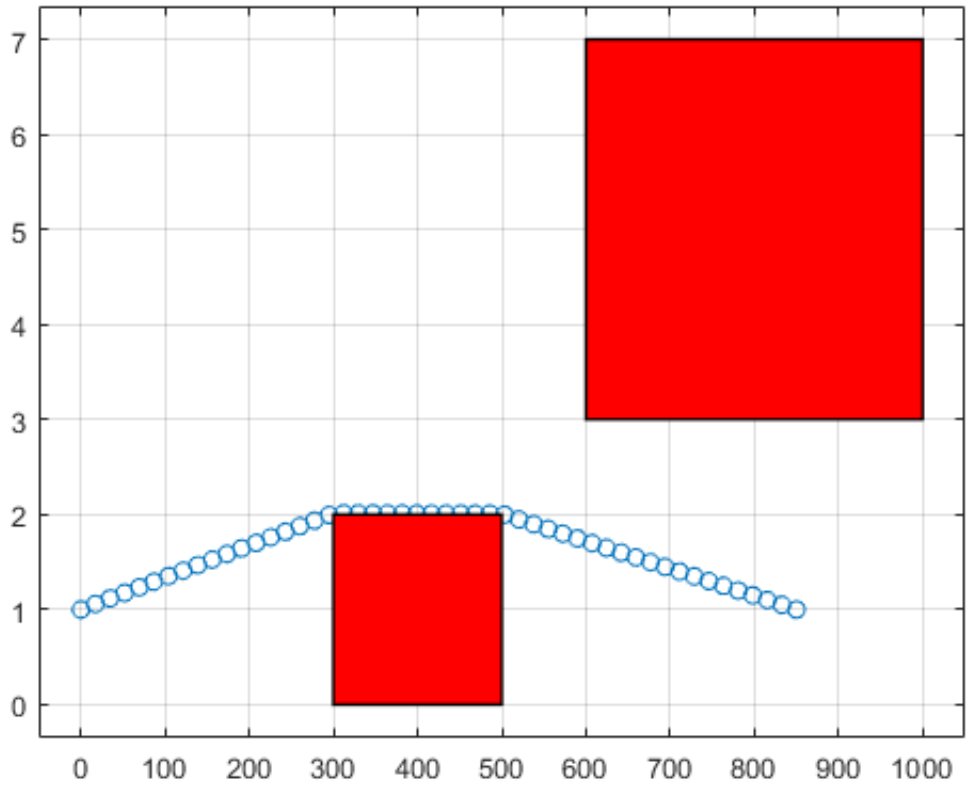
```
%z_obs{4}=[3;7];
%d_obs{4}=8;
%Qobs{4}=diag([1,10]);

%Constraints
%Setup Optimization Problem
cost = 0;
constr = [z(:,1)==z0;z(:,N)==zT];
Q=eye(2);
%constr = [zmin<=z(:,N)<= zmax, z(:,1)==z0,z(:,N)==zT];
for t = 2:N
      cost=cost+(z(:,t)-z(:,t-1))'*Q*(z(:,t)-z(:,t-1));
      constr = constr +[dzmin<= z(:,t)-z(:,t-1)<=dzmax];
      constr = constr +[zmin<= z(:,t)<=zmax];
      for k = 0:sampling-1
          for j=1:length(obs)
              zs=z(:,t-1)+k/sampling*(z(:,t)-z(:,t-1));
              [H,K]=double(obs{j}.poly);
              constr = constr +[H(1,:)*(zs)>=K(1)-(1-d((j-1)*4+1,(t-2)*sampling+k+1))*bM .
..
                                H(2,:)*(zs)>=K(2)-(1-d((j-1)*4+2,(t-2)*sampling+k+1))*bM .
..
                                H(3,:)*(zs)>=K(3)-(1-d((j-1)*4+3,(t-2)*sampling+k+1))*bM .
..
                                H(4,:)*(zs)>=K(4)-(1-d((j-1)*4+4,(t-2)*sampling+k+1))*bM .
..
                                d((j-1)*4+1,(t-2)*sampling+k+1)+d((j-1)*4+2,(t-2)*sampling
+k+1)+d((j-1)*4+3,(t-2)*sampling+k+1)+d((j-1)*4+4,(t-2)*sampling+k+1)>=1];
          end
      end
end
options = sdpsettings('solver','gurobi');
%options.ipopt=ipoptset('linear_solver','MUMPS');
solvesdp(constr,cost,options);
z_vec = double(z);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Plotting Functions % to add title and labels
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
Academic license - for non-commercial use only
Optimize a model with 5296 rows, 4020 columns and 15880 nonzeros
Model has 198 quadratic objective terms
Variable types: 100 continuous, 3920 integer (3920 binary)
Coefficient statistics:
  Matrix range     [5e-04, 1e+03]
  Objective range  [0e+00, 0e+00]
  QObjective range [2e+00, 4e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 1e+03]
Presolve removed 4328 rows and 3388 columns
Presolve time: 0.03s
Presolved: 968 rows, 632 columns, 2893 nonzeros
Presolved model has 190 quadratic objective terms
```

```
Variable types: 96 continuous, 536 integer (536 binary)
Found heuristic solution: objective 15613.676716
Found heuristic solution: objective 14853.329930

Root relaxation: objective 1.474490e+04, 810 iterations, 0.01 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0 14744.8980    0  328 14853.3299 14744.8980  0.73%     -    0s
     0     0 14744.8980    0  374 14853.3299 14744.8980  0.73%     -    0s
     0     0 14744.8980    0  306 14853.3299 14744.8980  0.73%     -    0s
     0     0 14744.8980    0  306 14853.3299 14744.8980  0.73%     -    0s
     0     0 14744.8980    0  255 14853.3299 14744.8980  0.73%     -    0s
     0     0 14744.8980    0  248 14853.3299 14744.8980  0.73%     -    0s
     0     2 14744.8980    0  248 14853.3299 14744.8980  0.73%     -    0s
*  204    76             106    14853.313802 14744.9943  0.73%  11.7    0s
H  263    23                    14745.018326 14744.9943  0.00%  11.0    0s
H  274    18                    14745.006375 14744.9943  0.00%  10.6    0s

Cutting planes:
  Clique: 311
  MIR: 12

Explored 282 nodes (4730 simplex iterations) in 0.37 seconds
Thread count was 4 (of 4 available processors)

Solution count 5: 14745 14745 14853.3 ... 15613.7

Optimal solution found (tolerance 1.00e-04)
Best objective 1.474500637480e+04, best bound 1.474499433966e+04, gap 0.0001%
```

plot routine

```
figure
plot(z_vec(1,:),z_vec(2,:),'o')
hold on
for j=1:length(obs)
plot(obs{j}.T*unitbox(2)+obs{j}.center);
end
```



The time is bit slower than the NLP.

```
toc
```

```
Elapsed time is 0.741042 seconds.
```

Published with MATLAB® R2018a