# Q2g_ObstacleAvoidance

April 16, 2024

[ ]:
```
using JuMP
using Ipopt
using Plots
include("Q2c_VehicleDynamics.jl")

function circleShape(h,k,r)
     = LinRange(0, 2* , 500)
    h.+r*sin.( ), k.+r*cos.( )
end

x0 = [-10.0, 0.0, 0.0, 0.0, 0.0, 10.0, 0.0] #TODO Initial Condition
XL = [-40, -20, -3, -pi/5,  -pi/2, 5.0, -pi/12] # States Lower Bound
XU = [300, 20, 3, pi/5, pi/2, 15.0, pi/12] #TODO States Upper Bound
CL = [-2.6, -0.1] #TODO Control Lower Bound
CU = [2.6, 0.1] #TODO Control Upper Bound

model = Model(optimizer_with_attributes(Ipopt.Optimizer)) # Initialize JuMP model

numStates = 7 #TODO number of states
numControls = 2 #TODO number of control
PredictionHorizon = 8 #TODO Prediction Time
numColPoints = 81 #TODO
Δt = PredictionHorizon/(numColPoints - 1)# Time interval

@variables(model, begin
    # Set xst as a numColPoints x numStates matrix that is
    # between the upper and lower states bounds
    XL[i]   xst[j in 1:numColPoints, i in 1:numStates]    XU[i]
    #TODO Set u as a numColPoints x numControls matrix that is between the
    #  upper and lower control bounds
    CL[i]   u[j in 1:numColPoints, i in 1:numControls]   CU[i]
end)


fix(xst[1, 1], x0[1]; force = true) # set the initial condition for x-position value
```

```
#TODO Follow the same way, set the remaining initial conditions, set x0[2] to xst[1,2],... and so on.
fix(xst[1,2],x0[2];force = true)
fix(xst[1,3],x0[3];force = true)
fix(xst[1,4],x0[4];force= true)
fix(xst[1,5],x0[5];force= true)
fix(xst[1,6],x0[6];force= true)
fix(xst[1,7],x0[7];force= true)


x = xst[:, 1]; y = xst[:, 2]; v = xst[:, 3]; r = xst[:, 4];   = xst[:, 5];
ux = xst[:, 6]; sa = xst[:, 7];
ax = u[:, 1]; # retract variable
sr = u[:, 2];

#  xst = Matrix{Any}(undef, numColPoints, numStates)
# write the states derivative for all states & controls
 xst = Matrix{Any}(undef, numColPoints, numStates)
for i = 1:1:numColPoints
    xst[i, :] = @expression(model, VehicleDynamics(xst[i, :], u[i, :]))
end

# add constraint to each state using backward Euler method
for j = 2:numColPoints
    for i = 1:numStates
        @constraint(model, xst[j, i] == xst[j - 1, i] +Δt *  xst[j, i])
    end
end


# TODO write the cost function for each term
y_cost = @expression(model, sum((y[j])^2 * Δt for j= 1:1:numColPoints)) #global y position of C.G Cost
sr_cost = @expression(model, sum((sr[j])^2 * Δt for j= 1:1:numColPoints))
sa_cost = @expression(model, sum((sa[j])^2 * Δt for j= 1:1:numColPoints))
ux_cost = @expression(model, sum((ux[j] - 13)^2 * Δt for j= 1:1:numColPoints))
ax_cost = @expression( model, sum((ax[j])^2 * Δt for j=1:1:numColPoints)) # ax cost



#TODO define cost weight
w_y  = 0.05 # change later for 2f
w_sr = 2.0
w_ax = 0.2
w_ux = 0.2
w_sa = 1.0
```

```julia
block_list = [30.0 2 6] # block_list = [obstacle_x_center, obstacle_y_center,␣
↪radius]

# TODO add obstacle avoidance constraint
obs_constraint = @constraint(model, [i = 1:numColPoints],
1 <=  ((x[i]-block_list[1])^2+(y[i]-block_list[2])^2)/(block_list[3])^2
)


#Solve
@objective(model, Min, w_y * y_cost + w_sr * sr_cost + w_ax * ax_cost + w_ux *␣
↪ux_cost + w_sa * sa_cost) # objective value
# set_silent(model)  # Hide solver's verbose output
# @assert is_solved_and_feasible(model)
optimize!(model) # optimize model
StatesHis = value.(model[:xst]) # retrieve data


println("Your y cost is:  ", round(value(y_cost); digits = 3))

p = plot(size=(600, 350))
# plot!(p, StatesHis[:, 1], StatesHis[:, 2], tickfontsize = 10, xlabel = "X␣
↪(m)", ylabel = "Y (m)",guidefont=15) # path plot
# plot!(p, circleShape(block_list[1], block_list[2], block_list[3]), seriestype␣
↪= [:shape,], ;w = 0.5, c=:black, linecolor = :black, legend = false,␣
↪fillalpha = 0.2, aspect_ratio=:equal)
# plot(0:Δt:PredictionHorizon, StatesHis[:, 6], tickfontsize = 10, xlabel =␣
↪"time (s)", ylabel = "ux (m)",guidefont=15) # Speed plot

display(plot!(p, StatesHis[:, 1], StatesHis[:, 2], tickfontsize = 10, xlabel =␣
↪"X (m)", ylabel = "Y (m)",guidefont=15))# path plot
display(plot!(p, circleShape(block_list[1], block_list[2], block_list[3]),␣
↪seriestype = [:shape,], ;w = 0.5, c=:black, linecolor = :black, legend =␣
↪false, fillalpha = 0.2, aspect_ratio=:equal))
display(plot(0:Δt:PredictionHorizon, StatesHis[:, 6], tickfontsize = 10, xlabel␣
↪= "time (s)", ylabel = "ux (m)",guidefont=15)) # Speed plot
```

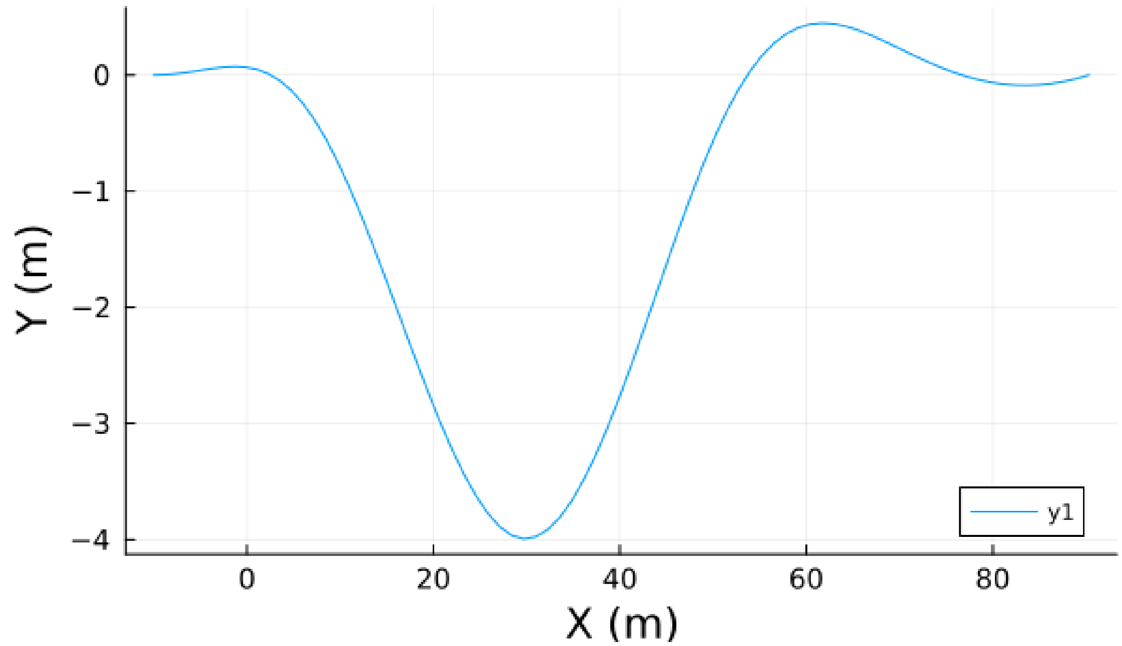
```
The states derivative is: [9.9 1.496 -1.004 2.587 0.1 1.0 0.1]

******************************************************************************
This program contains Ipopt, a library for large-scale nonlinear optimization.
 Ipopt is released as open source code under the Eclipse Public License (EPL).
         For more information visit https://github.com/coin-or/Ipopt
******************************************************************************
```

```
This is Ipopt version 3.14.14, running with linear solver MUMPS 5.6.2.

Number of nonzeros in equality constraint Jacobian...:     2473
Number of nonzeros in inequality constraint Jacobian.:      320
Number of nonzeros in Lagrangian Hessian.............:     3762

Total number of variables............................:      722
                     variables with only lower bounds:        0
                variables with lower and upper bounds:      722
                     variables with only upper bounds:        0
Total number of equality constraints.................:      560
Total number of inequality constraints...............:       81
        inequality constraints with only lower bounds:        0
   inequality constraints with lower and upper bounds:        0
        inequality constraints with only upper bounds:       81

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
   0  1.0130400e+02 9.50e+00 7.65e-01  -1.0 0.00e+00    -  0.00e+00 0.00e+00   0
   1  1.0204698e+02 8.39e+00 5.41e+00  -1.0 4.79e+01    -  1.76e-02 1.16e-01H  1
   2  1.0188145e+02 7.57e+00 4.85e+00  -1.0 4.08e+01    -  6.42e-02 9.88e-02f  1
   3  1.0195396e+02 6.97e+00 4.46e+00  -1.0 2.86e+01    -  6.25e-02 7.92e-02h  2
   4  1.0200340e+02 6.96e+00 4.46e+00  -1.0 5.17e+02    -  1.39e-02 4.00e-04h  5
   5  1.0281406e+02 6.57e+00 4.21e+00  -1.0 4.85e+01    -  4.75e-02 5.63e-02h  2
   6  1.0485210e+02 5.25e+00 3.29e+00  -1.0 2.14e+01    -  5.92e-02 2.01e-01H  1
   7  1.0680535e+02 3.89e+00 5.31e+00  -1.0 1.24e+01    -  7.04e-02 2.60e-01H  1
   8  1.0782040e+02 3.15e+00 5.31e+00  -1.0 1.10e+01    -  6.68e-02 1.90e-01h  1
   9  1.0878320e+02 2.37e+00 3.87e+00  -1.0 1.04e+01    -  8.95e-02 2.48e-01f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  10  1.0942251e+02 1.53e+00 2.31e+00  -1.0 9.65e+00    -  1.26e-01 3.54e-01f  1
  11  1.0885642e+02 6.66e-01 3.85e+00  -1.0 8.44e+00    -  2.02e-01 5.64e-01f  1
  12  1.0433388e+02 8.03e-02 4.65e+00  -1.0 5.84e+00    -  4.10e-01 8.79e-01f  1
  13  9.2102361e+01 2.81e-02 1.43e+00  -1.0 5.36e+00    -  5.11e-01 1.00e+00f  1
  14  6.3448871e+01 2.10e-01 1.03e+00  -1.0 1.50e+01    -  3.62e-01 1.00e+00f  1
  15  4.5590440e+01 4.65e-02 5.68e-01  -1.0 8.58e+00    -  5.22e-01 7.86e-01f  1
  16  3.9066811e+01 2.99e-02 1.09e+00  -1.0 1.12e+01    -  1.09e-01 3.62e-01f  1
  17  3.3439600e+01 1.79e-02 3.76e+00  -1.0 1.12e+01    -  6.46e-02 4.01e-01f  1
  18  2.5405825e+01 8.49e-03 1.47e+01  -1.0 9.70e+00    -  5.07e-02 9.06e-01f  1
  19  2.0638058e+01 2.57e-03 1.20e+01  -1.0 7.42e+00    -  6.55e-02 1.00e+00f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  20  1.7321446e+01 2.09e-03 7.31e+00  -1.0 1.13e+01    -  4.59e-01 4.00e-01f  1
  21  1.2523872e+01 4.57e-03 1.37e+01  -1.0 1.07e+01    -  4.95e-02 1.00e+00f  1
  22  1.0603519e+01 3.13e-03 8.06e+00  -1.0 1.15e+01    -  2.80e-01 3.79e-01f  1
  23  8.4492756e+00 2.14e-03 3.59e+00  -1.0 1.13e+01    -  1.44e-01 5.70e-01f  1
  24  8.2619953e+00 9.69e-04 1.84e+01  -1.0 3.64e+00    -  5.88e-02 1.00e+00f  1
  25  6.1076288e+00 1.28e-03 1.02e+01  -1.0 1.57e+01    -  4.31e-01 4.45e-01f  1
  26  5.7933335e+00 5.09e-04 6.29e+00  -1.0 3.70e+00    -  2.12e-01 1.00e+00f  1
  27  4.6635204e+00 9.67e-04 2.89e+00  -1.0 1.35e+01    -  5.41e-01 5.41e-01f  1
  28  4.9409531e+00 9.72e-05 1.90e+00  -1.0 3.78e-01    -  5.15e-01 1.00e+00f  1
```

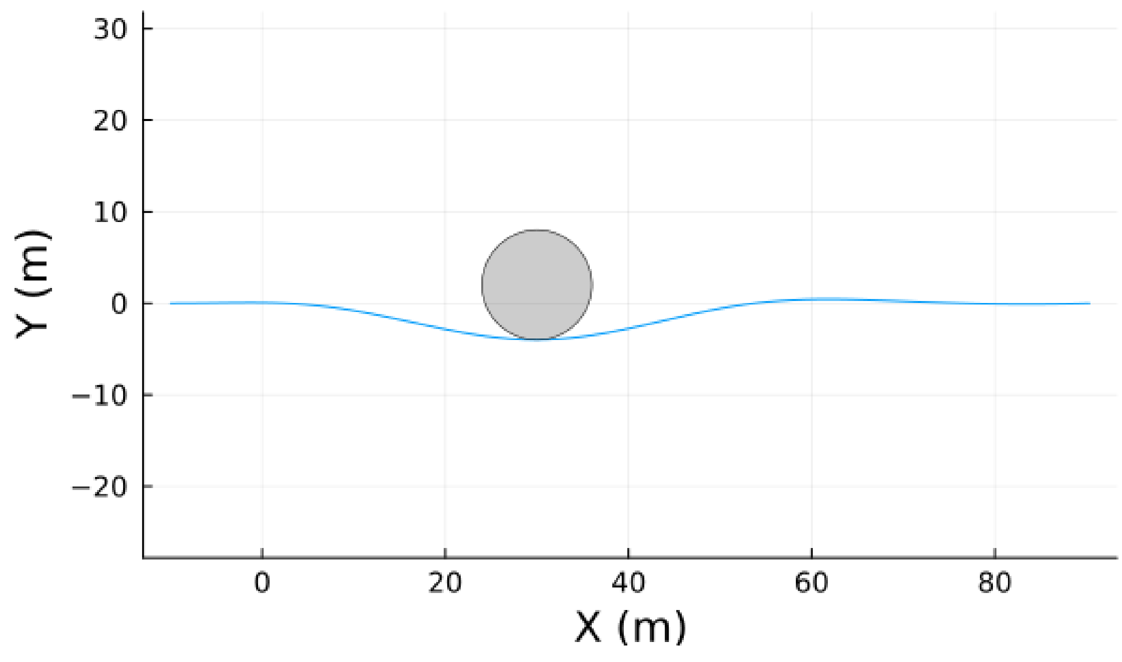
```
  29  4.5356384e+00 4.20e-04 1.91e-01  -1.0 5.06e+00    -  1.00e+00 1.00e+00f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  30  3.7970469e+00 4.89e-03 1.56e-01  -2.5 5.56e+00    -  8.97e-01 8.23e-01h  1
  31  3.4781707e+00 3.01e-03 1.33e-01  -2.5 1.96e+00    -  9.01e-01 1.00e+00h  1
  32  3.3747261e+00 1.93e-03 1.02e-03  -2.5 3.13e-01    -  1.00e+00 1.00e+00h  1
  33  3.3124500e+00 1.28e-03 3.04e-02  -3.8 2.51e-01    -  7.90e-01 1.00e+00h  1
  34  3.2934684e+00 3.17e-04 1.73e-04  -3.8 1.28e-01    -  1.00e+00 1.00e+00h  1
  35  3.2856980e+00 1.05e-04 7.06e-04  -5.7 6.01e-02    -  9.43e-01 1.00e+00h  1
  36  3.2848904e+00 7.96e-06 6.69e-06  -5.7 1.71e-02    -  1.00e+00 1.00e+00h  1
  37  3.2848271e+00 7.67e-07 5.38e-07  -5.7 4.61e-03    -  1.00e+00 1.00e+00h  1
  38  3.2847616e+00 1.05e-07 4.82e-06  -8.6 2.44e-03    -  9.93e-01 1.00e+00h  1
  39  3.2847602e+00 4.75e-09 1.13e-09  -8.6 3.99e-04    -  1.00e+00 1.00e+00h  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  40  3.2847602e+00 1.39e-11 3.86e-12  -8.6 2.36e-05    -  1.00e+00 1.00e+00h  1

Number of Iterations…: 40

                                   (scaled)                 (unscaled)
Objective…:   3.2847602184500948e+00    3.2847602184500948e+00
Dual infeasibility…:   3.8614831090414654e-12    3.8614831090414654e-12
Constraint violation…:   1.3932396902838207e-11    1.3932396902838207e-11
Variable bound violation:   0.0000000000000000e+00    0.0000000000000000e+00
Complementarity…:   2.7585141641059175e-09    2.7585141641059175e-09
Overall NLP error…:   2.7585141641059175e-09    2.7585141641059175e-09


Number of objective function evaluations             = 55
Number of objective gradient evaluations             = 41
Number of equality constraint evaluations            = 55
Number of inequality constraint evaluations          = 55
Number of equality constraint Jacobian evaluations   = 41
Number of inequality constraint Jacobian evaluations = 41
Number of Lagrangian Hessian evaluations             = 40
Total seconds in IPOPT                               = 0.627

EXIT: Optimal Solution Found.
Your y cost is:  25.81

******************************************************************************
This program contains Ipopt, a library for large-scale nonlinear optimization.
 Ipopt is released as open source code under the Eclipse Public License (EPL).
         For more information visit https://github.com/coin-or/Ipopt
******************************************************************************

This is Ipopt version 3.14.14, running with linear solver MUMPS 5.6.2.

Number of nonzeros in equality constraint Jacobian…:     2473
Number of nonzeros in inequality constraint Jacobian.:      320
```

```
Number of nonzeros in Lagrangian Hessian…:     3762

Total number of variables…:      722
                     variables with only lower bounds:        0
                variables with lower and upper bounds:      722
                     variables with only upper bounds:        0
Total number of equality constraints…:      560
Total number of inequality constraints…:       81
        inequality constraints with only lower bounds:        0
   inequality constraints with lower and upper bounds:        0
        inequality constraints with only upper bounds:       81

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
   0  1.0130400e+02 9.50e+00 7.65e-01  -1.0 0.00e+00    -  0.00e+00 0.00e+00   0
   1  1.0204698e+02 8.39e+00 5.41e+00  -1.0 4.79e+01    -  1.76e-02 1.16e-01H  1
   2  1.0188145e+02 7.57e+00 4.85e+00  -1.0 4.08e+01    -  6.42e-02 9.88e-02f  1
   3  1.0195396e+02 6.97e+00 4.46e+00  -1.0 2.86e+01    -  6.25e-02 7.92e-02h  2
   4  1.0200340e+02 6.96e+00 4.46e+00  -1.0 5.17e+02    -  1.39e-02 4.00e-04h  5
   5  1.0281406e+02 6.57e+00 4.21e+00  -1.0 4.85e+01    -  4.75e-02 5.63e-02h  2
   6  1.0485210e+02 5.25e+00 3.29e+00  -1.0 2.14e+01    -  5.92e-02 2.01e-01H  1
   7  1.0680535e+02 3.89e+00 5.31e+00  -1.0 1.24e+01    -  7.04e-02 2.60e-01H  1
   8  1.0782040e+02 3.15e+00 5.31e+00  -1.0 1.10e+01    -  6.68e-02 1.90e-01h  1
   9  1.0878320e+02 2.37e+00 3.87e+00  -1.0 1.04e+01    -  8.95e-02 2.48e-01f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  10  1.0942251e+02 1.53e+00 2.31e+00  -1.0 9.65e+00    -  1.26e-01 3.54e-01f  1
  11  1.0885642e+02 6.66e-01 3.85e+00  -1.0 8.44e+00    -  2.02e-01 5.64e-01f  1
  12  1.0433388e+02 8.03e-02 4.65e+00  -1.0 5.84e+00    -  4.10e-01 8.79e-01f  1
  13  9.2102361e+01 2.81e-02 1.43e+00  -1.0 5.36e+00    -  5.11e-01 1.00e+00f  1
  14  6.3448871e+01 2.10e-01 1.03e+00  -1.0 1.50e+01    -  3.62e-01 1.00e+00f  1
  15  4.5590440e+01 4.65e-02 5.68e-01  -1.0 8.58e+00    -  5.22e-01 7.86e-01f  1
  16  3.9066811e+01 2.99e-02 1.09e+00  -1.0 1.12e+01    -  1.09e-01 3.62e-01f  1
  17  3.3439600e+01 1.79e-02 3.76e+00  -1.0 1.12e+01    -  6.46e-02 4.01e-01f  1
  18  2.5405825e+01 8.49e-03 1.47e+01  -1.0 9.70e+00    -  5.07e-02 9.06e-01f  1
  19  2.0638058e+01 2.57e-03 1.20e+01  -1.0 7.42e+00    -  6.55e-02 1.00e+00f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  20  1.7321446e+01 2.09e-03 7.31e+00  -1.0 1.13e+01    -  4.59e-01 4.00e-01f  1
  21  1.2523872e+01 4.57e-03 1.37e+01  -1.0 1.07e+01    -  4.95e-02 1.00e+00f  1
  22  1.0603519e+01 3.13e-03 8.06e+00  -1.0 1.15e+01    -  2.80e-01 3.79e-01f  1
  23  8.4492756e+00 2.14e-03 3.59e+00  -1.0 1.13e+01    -  1.44e-01 5.70e-01f  1
  24  8.2619953e+00 9.69e-04 1.84e+01  -1.0 3.64e+00    -  5.88e-02 1.00e+00f  1
  25  6.1076288e+00 1.28e-03 1.02e+01  -1.0 1.57e+01    -  4.31e-01 4.45e-01f  1
  26  5.7933335e+00 5.09e-04 6.29e+00  -1.0 3.70e+00    -  2.12e-01 1.00e+00f  1
  27  4.6635204e+00 9.67e-04 2.89e+00  -1.0 1.35e+01    -  5.41e-01 5.41e-01f  1
  28  4.9409531e+00 9.72e-05 1.90e+00  -1.0 3.78e-01    -  5.15e-01 1.00e+00f  1
  29  4.5356384e+00 4.20e-04 1.91e-01  -1.0 5.06e+00    -  1.00e+00 1.00e+00f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  30  3.7970469e+00 4.89e-03 1.56e-01  -2.5 5.56e+00    -  8.97e-01 8.23e-01h  1
  31  3.4781707e+00 3.01e-03 1.33e-01  -2.5 1.96e+00    -  9.01e-01 1.00e+00h  1
```

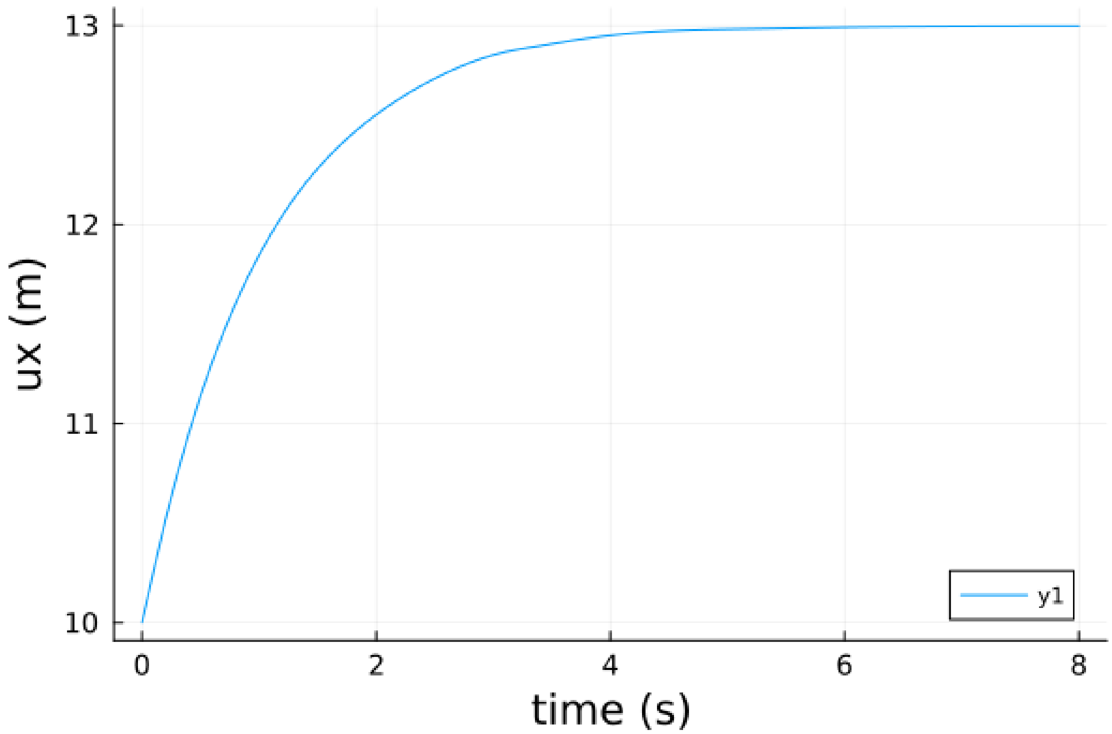
```
  32  3.3747261e+00 1.93e-03 1.02e-03  -2.5 3.13e-01    -  1.00e+00 1.00e+00h  1
  33  3.3124500e+00 1.28e-03 3.04e-02  -3.8 2.51e-01    -  7.90e-01 1.00e+00h  1
  34  3.2934684e+00 3.17e-04 1.73e-04  -3.8 1.28e-01    -  1.00e+00 1.00e+00h  1
  35  3.2856980e+00 1.05e-04 7.06e-04  -5.7 6.01e-02    -  9.43e-01 1.00e+00h  1
  36  3.2848904e+00 7.96e-06 6.69e-06  -5.7 1.71e-02    -  1.00e+00 1.00e+00h  1
  37  3.2848271e+00 7.67e-07 5.38e-07  -5.7 4.61e-03    -  1.00e+00 1.00e+00h  1
  38  3.2847616e+00 1.05e-07 4.82e-06  -8.6 2.44e-03    -  9.93e-01 1.00e+00h  1
  39  3.2847602e+00 4.75e-09 1.13e-09  -8.6 3.99e-04    -  1.00e+00 1.00e+00h  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  40  3.2847602e+00 1.39e-11 3.86e-12  -8.6 2.36e-05    -  1.00e+00 1.00e+00h  1

Number of Iterations…: 40

                                   (scaled)                 (unscaled)
Objective…:   3.2847602184500948e+00    3.2847602184500948e+00
Dual infeasibility…:   3.8614831090414654e-12    3.8614831090414654e-12
Constraint violation…:   1.3932396902838207e-11    1.3932396902838207e-11
Variable bound violation:   0.0000000000000000e+00    0.0000000000000000e+00
Complementarity…:   2.7585141641059175e-09    2.7585141641059175e-09
Overall NLP error…:   2.7585141641059175e-09    2.7585141641059175e-09


Number of objective function evaluations             = 55
Number of objective gradient evaluations             = 41
Number of equality constraint evaluations            = 55
Number of inequality constraint evaluations          = 55
Number of equality constraint Jacobian evaluations   = 41
Number of inequality constraint Jacobian evaluations = 41
Number of Lagrangian Hessian evaluations             = 40
Total seconds in IPOPT                               = 0.627

EXIT: Optimal Solution Found.
Your y cost is:  25.81
```



0
−1
−2
−3
−4
Y (m)
0
20
40
60
80
X (m)
y1

30
20
10
0
−10
−20
Y (m)
0
20
40
60
80
X (m)


13
12
11
10
ux (m)
0
2
4
6
8
time (s)
y1