Course › Week 7 › Proble... › ZMP

ZMP

Zero Moment Point

0.0/5.0 points (graded)

Consider the ZMP equation, which describes the center of mass dynamics of a walking robot.

$$x_{cm} - x_{zmp} = \frac{z_{cm}}{g}\ddot{x}_{cm}$$

where x_{cm} is the center of mass position, z_{cm} is the center of mass height, and x_{zmp} is the ZMP position.

To achieve this linear model, which assumptions/simplifications have been made? Select all that apply.

- [ ] The robot legs are massless
- [ ] $\dot{z}_{cm}$ is held constant. ✔
- [ ] The robot dynamics are linear
- [ ] Angular momentum is held constant. ✔

Submit You have used 0 of 1 attempt

Answers are displayed within the problem

ZMP Planning: Part A

0.0/8.0 points (graded)

For this problem, we will use a simple ZMP plan to construct a nominal trajectory for the center of mass that tracks the ZMP plan. Construct a linear system where the state $\mathbf{x} = \begin{bmatrix} x_{cm} \\ \dot{x}_{cm} \end{bmatrix}$ is the position of the center of mass $x_{cm} \in \mathbb{R}^2$ (for 3-D walking) and the velocity $\dot{x}_{cm}$. Take the output of to be the position of the ZMP, $x_{zmp} \in \mathbb{R}^2$. Let the input $u = \ddot{x}_{cm}$. Write your system in the form:

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu, \quad x_{zmp} = C\mathbf{x} + Du.$$

```
g = 9.81;   %gravity
z_cm = 1.1; %height of the center of mass
A =
B =
C =
D =

```

Unanswered

```
g = 9.81;   %gravity
z_cm = 1.1; %height of the center of mass

A = [zeros(2) eye(2); zeros(2,4)];
B = [zeros(2);eye(2)];
C = [eye(2) zeros(2)];
D = -eye(2)*z_cm/g;
```

Run Code

Submit You have used 0 of 3 attempts

Answers are displayed within the problem

ZMP Planning: Part B

0.0/8.0 points (graded)

With the system above, construct a linear quadratic optimal control problem to stablize the ZMP to the origin. Write the cost as $\int x_{zmp}^T Q_{zmp} x_{zmp}$

Convert this problem into standard LQR format, and call MATLAB's built in **lqr** function. See **help lqr** for an explanation of the term N. For $Q_{zmp} = I^{2\times 2}$, find the optimal cost-to-go S.

Rewrite this tracking problem in terms of the state and input, in the form

$$\int_0^\infty \mathbf{x}^T Q\mathbf{x} + q^T\mathbf{x} + u^T Ru + r^T u + 2\mathbf{x}^T Nudt$$

where we have discarded any constant terms in the cost. Note the "2", this is to fit your solution into MATLAB's **lqr** format.

```
g = 9.81;   %gravity
z_cm = 1.1; %height of the center of mass
A =
B =
Q =
R =
N =

[K,S] = lqr(A,B,Q,R,N);

```

Unanswered

```
g = 9.81;   %gravity
z_cm = 1.1; %height of the center of mass

% an example of Q_zmp
Q_zmp = eye(2);

% A, B, C, and D from last question:
A = [zeros(2) eye(2); zeros(2,4)];
B = [zeros(2);eye(2)];
C = [eye(2) zeros(2)];
D = -eye(2)*z_cm/g;

Q = C'*Q_zmp*C;
R = D'*Q_zmp*D;
N = C'*Q_zmp*D;

[K,S] = lqr(A,B,Q,R,N);
```

Run Code

Submit You have used 0 of 3 attempts

Answers are displayed within the problem

ZMP Planning: Part C

0.0/9.0 points (graded)

Repeat Part B, but instead track a desired reference trajectory $x^d_{zmp}(t)$, where the cost is $\int \left(x_{zmp} - x^d_{zmp}\right)^T Q_{zmp} \left(x_{zmp} - x^d_{zmp}\right) dt$

Rewrite this tracking problem in terms of the state and input, in the form

$$\mathbf{x}(T)^T Qf \mathbf{x}(T) + q_f^T \mathbf{x}(T) + \int_0^T \mathbf{x}^T Q(t)\mathbf{x} + q(t)^T \mathbf{x} + u^T R(t) u + r(t)^T u + 2\mathbf{x}^T N(t) u dt$$

where we have discarded any constant terms in the cost.

Download the stub code here, and implement the ZMP planner. For the final cost of the time-varying LQR problem, use the infinite horizon cost from Part B, but centered around the stationary end-point of the walking trajectory $\left(x_{cm} - x_{final}\right)^T S \left(x_{cm} - x_{final}\right)$.

A correct LQR controller should track the ZMP trajectory closely, and illustrate a smooth center of mass trajectory. For your working controller, what is $\ddot{y}_{cm}$ for $t = 1$ in the simulation? This should be the second element of u printed by the stub code.

Answer: .0376

Explanation

The time varying LQR costs are given by:

```
% The time varying costs for Q, R, N as in Part B.
% Now, however, the desired zmp position is zmp_traj, not a constant value (0),
% so we have linear terms as a result.
Q{1} = C'*Q_zmp*C;
Q{2} = -2*C'*Q_zmp*zmp_traj;
Q{3} = 0;


R{1} = D'*Q_zmp*D;
R{2} = -2*D'*Q_zmp*zmp_traj;
R{3} = 0;

options.Ny = C'*Q_zmp*D;

% Solve LQR at the final state to get infinite horizon cost
[~,S] = lqr(A,B,Q{1},R{1},options.Ny)
Qf{1} = S;
Qf{2} = -2*S*[zmp_traj.eval(T);0;0];
Qf{3} = 0;
```

Experiment by changing the walking speed and the amount of time at the beginning and end of the trajectory where the ZMP is motionless (t_beginning and t_end) in the code.

Submit You have used 0 of 3 attempts

Answers are displayed within the problem