RL Homework 2 Report Question Number 2

Jainam Kothari

10-27-2024

Exercise 1

From the Previous homework Last question we can ask that we tried to solve a function in the format of the optimum control Problem Here are the equations formatted in LaTeX:

1. Optimal Control Problem:

$$\min_{\{x_n,u_n\}} \sum_{n=0}^{N-1} \left(\frac{1}{2} x_n^T Q x_n + \frac{1}{2} u_n^T R u_n - \bar{x}_n^T Q x_n \right) + \frac{1}{2} \bar{x}_N Q \bar{x}_N$$

subject to

$$x_{n+1} = \begin{bmatrix} 1 & 0 \\ 0 & \Delta t \end{bmatrix} x_n + \begin{bmatrix} 0 \\ A^t \end{bmatrix} u_n$$

$$x_0 = x_{init}$$

2. Rewritten in Matrix Form:

$$\min_y \frac{1}{2} y^T G y + g^T y$$

subject to

$$My = p$$

3. KKT Conditions:

The Lagrangian of the problem is

$$L(y, \lambda) = \frac{1}{2} y^T G y + g^T y + \lambda^T (My - p)$$

and the KKT conditions for optimality are

$$\begin{bmatrix} G & M^T \\ M & 0 \end{bmatrix} \begin{bmatrix} y \\ \lambda \end{bmatrix} = \begin{bmatrix} -g \\ p \end{bmatrix}$$

And here we found the y and λ
which will help us get the position, velocity, and control states of the drone.

Now we Need to also solve the same problem when the control is limited to 5 for both rotors and the horizontal and vertical velocities are bounded to 2m · s 1 .

And from this we have the limit in constraints now applied to the equation which is Here is the provided :

$$u_{\min} \leq u_n \leq u_{\max}$$

$$x_{\min} \leq x_n \leq x_{\max}$$

$$x_0 = x_{\text{init}}$$

now we can have new matrix H and h for this and use the Cvxopt colver from the QP library to solve it and this solution can be seen in the answer ipynb notebook .

Exercise 2

Function 1 of the Q2

$$f(x) = -e^{-(x-1)^2}$$

we get the Gradient of the function as ;

$$f'(x) = 2(x-1)e^{-(x-1)^2}$$

And we also get the Hessian of the function as :

$$f''(x) = 2e^{-(x-1)^2}\left(1 - 2(x-1)^2\right)$$

Now on getting the Hessian, we have two methods that is gradient and Newtons method that are general for all the questions and provided after finding the gradient and Hessian of all the question .

Function 2 of the Q2

$$f(x, y) = (1-x)^2 + 100(y - x^2)^2$$

we get the Gradient of the function as ;

$$\nabla f(x, y) = \begin{bmatrix} -2(1-x) - 400x(y - x^2) \\ 200(y - x^2) \end{bmatrix}$$

And we also get the Hessian of the function as :

$$\nabla^2 f(x, y) = \begin{bmatrix} 2 - 400(y - 3x^2) & -400x \\ -400x & 200 \end{bmatrix}$$

Function 3: Quadratic Function

$$f(x) = x^T \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} x + \begin{bmatrix} -1 \\ 1 \end{bmatrix}^T x$$

we get the Gradient of the function as ;

$$\nabla f(x) = \begin{bmatrix} 6 & 2 \\ 2 & 6 \end{bmatrix} x + \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

And we also get the Hessian of the function as :

$$\nabla^2 f(x) = \begin{bmatrix} 6 & 2 \\ 2 & 6 \end{bmatrix}$$

Function 4: 3D Quadratic Function

$$f(x) = \frac{1}{2}x^T \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix} x - \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T x$$

we get the Gradient of the function as ;

$$\nabla f(x) = \begin{bmatrix} 2 & 2 & 0 \\ 2 & 2 & 0 \\ 0 & 0 & 8 \end{bmatrix} x - \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},$$

And we also get the Hessian of the function as :

$$\nabla^2 f(x) = \begin{bmatrix} 2 & 2 & 0 \\ 2 & 2 & 0 \\ 0 & 0 & 8 \end{bmatrix}$$

Optimization Methods

Gradient Descent

Given

$$\min_x f(x)$$

In Gradient Descent, the update rule is:

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k)$$

where α_k is determined using a backtracking line search.

Newton's Method

Given

$$\min_x f(x)$$

In Newton's Method, the update rule is:

$$x_{k+1} = x_k - \alpha_k \left(\nabla^2 f(x_k)\right)^{-1} \nabla f(x_k)$$

Instead of directly inverting the Hessian, we solve $\nabla^2 f(x_k)p_k = -\nabla f(x_k)$ using np.linalg.solve to improve computational stability.

Backtracking Line Search

The backtracking line search adjusts α so that:

$$f(x_k + \alpha p_k) \leq f(x_k) + c\alpha \nabla f(x_k)^T p_k$$

where $c = 10^{-4}$ which is given in the question . This technique helps keep the step size small enough to avoid jumping past the minimum, which is especially useful for the Rosenbrock function.

Results

For each function, we tested both optimization methods and compared their convergence behavior:

Function 1: Both methods found the best point quickly, but Newton's Method was faster because the function's exponential form is simple to work with.

Function 2: Gradient Descent took a lot of steps with the Rosenbrock function, which has a narrow valley, while Newton's Method handled it more efficiently.

Function 3: Both methods did well here, but Newton's Method was faster since the function was quadratic.

Function 4: For this 3D quadratic function, Newton's Method was effective in higher dimensions, as long as the Hessian stayed positive.

The plots illustrate how quickly each method converged and their step sizes. Generally, Newton's Method works better for functions with positive-definite Hessians, while Gradient Descent can struggle with non-quadratic functions.

Exercise 3

In this question, we will take the given equations as the comparison to the general equations and solve the question

We are given

$$\min_{\theta_n,\omega_n,u_n} \quad \sum_{n=0}^{300} 10(\theta_n-\pi)^2 + 0.1\omega_n^2 + 0.1u_n^2$$

subject to

$$\begin{aligned} \theta_{n+1} &= \theta_n + \Delta t\, \omega_n \\ \omega_{n+1} &= \omega_n + \Delta t\,(u_n - g\sin\theta_n) \\ \theta_0 &= \omega_0 = 0 \end{aligned}$$

This is same as the

$$\min_x \quad f(x) = 0$$

subject to $\quad g(x) = 0$

Here we are given the f(x) and constraints and we need to solve using SQP

First, we need to find the cost for the guess function which we can get using the method we used previously in the last homework

we can compare the given f(x) with

$f(x) = \min \quad \sum_{n=0}^{300} 1/2 * x^T Qx + g^T x$

based on this we can make the G matrix as we normally do, and since we do not have any R the G matrix will have only Q on the diagonal matrix;

also the x_n

$$\bar{x}_n = \begin{bmatrix} \bar{\theta}_n \\ \bar{\omega}_n \\ \bar{u}_n \end{bmatrix}$$

from this we can make the f(x) matrix for the later now we need to get the cost gradient matrix for getting the Lagrangian which is given by

$$G(\bar{x})\Delta x = g(\bar{x})$$
$$= \begin{bmatrix} \theta_n + \Delta t\, \omega_n - \theta_{n+1} \\ \omega_n + \Delta t\,(u_n - g\sin\theta_n) - \omega_{n+1} \end{bmatrix}$$

Now once we have calculated this we cam get the gradient of the cost function that we had before since we have the lagrangian given as

$$\mathcal{L}(x, \lambda, \mu) = f(x) + \lambda^T g(x)$$

where we need to find the gradient of this which is possible by getting the Lagrangian minimum for the convergence to the best step

thus we get the gradient of the cost function as

$$\nabla f(\bar{x}) = 2 \cdot \tfrac{1}{2} G\bar{x} + g$$

$$\Rightarrow \nabla f(\bar{x}) = G\bar{x} + g$$

and now we can write the minimum of the Lagrangian using it in the solving the KKT way and then we can use the simple line tracking that was provided in the pdf.

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x_k) & \nabla g(x_k)^T \\ \nabla g(x_k) & 0 \end{bmatrix} \begin{pmatrix} p_k \\ \lambda_{k+1} \end{pmatrix} = \begin{pmatrix} -\nabla f(x_k) \\ -g(x_k) \end{pmatrix}$$

now we have found the values of the pk and λ

Now we can apply the backtracking line search algorithm to get the alpha value we need and with that, we can conclude the part of exercise.

If $f(x_{\text{guess}}+\alpha p_x) < f_{\text{best}},$ then set $f_{\text{best}} \leftarrow f(x_{\text{guess}}+\alpha p_x)$ and accept the step.

Also, we need to get the constrained violation and that can be commute by to computing $|c(x)|$.

Now as we move to the second part of the exercise the new difference we have constrained on the velocity and now that adds a new constraint our question so we need to expand our KKT and solve for it the same as before

$$-4 \leq u_n \leq 4$$

make $C(\bar{x})\bar{x} \leq c(\bar{x})$

so our new KKT becomes

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x_k) & \nabla g(x_k)^T & \nabla c(x_k)^T \\ \nabla g(x_k) & 0 & 0 \\ \nabla c(x_k) & 0 & 0 \end{bmatrix} \begin{pmatrix} p_k \\ \lambda_{k+1} \\ \mu_{k+1} \end{pmatrix} = \begin{pmatrix} -\nabla f(x_k) \\ -g(x_k) \\ -c(x_k) \end{pmatrix}$$

now we have found the values of the pk and λ and μ

Now we can apply the backtracking line search algorithm to get the alpha value we need and with that, we can conclude the part of exercise.

If $f(x_{\text{guess}}+\alpha p_x) < f_{\text{best}},$ then set $f_{\text{best}} \leftarrow f(x_{\text{guess}}+\alpha p_x)$ and accept the step.

Also, we need to get the constrained violation and that can be commute by to computing $|c(x)|$.

Please see the notebook for the solution and code