EE 711: Optimization Techniques for Cyber Physical Systems
Department of Electrical and Computer Engineering Wichita State University
Assignment 12
Due: May 13, 2022; 11:59 pm

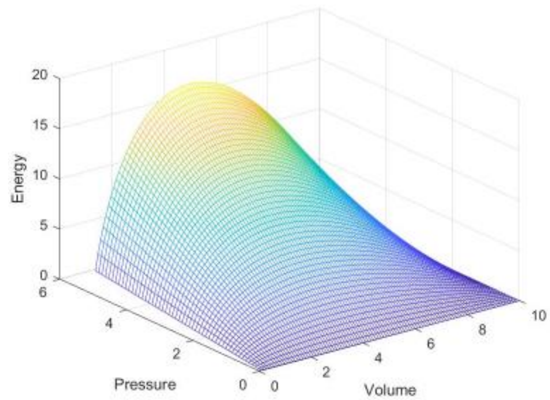
Problem 1 (120%):
For the original problem you have chosen towards your Project 1, use a metaheuristic method to find the optimal solution. You must write your own script using MATLAB or Python.

The objective function to be solved is:

$$\max_{P_t, V_w}\{Energy\} = P_t V_w ln\left(\frac{K}{V_w}\right)$$
$$P_t \leq 5$$
$$V_w \leq 10$$

Exploring nature of the objective function:

The constrained objective function is shown in below figure



Checking for convexity:

Taking hessian matrix

$$H = \begin{bmatrix} 0 & -0.307 \\ -0.307 & -0.5 \end{bmatrix}$$

$$|\lambda I - H| = \left|\begin{bmatrix} \lambda - 0 & 0.307 \\ 0.307 & \lambda + 0.5 \end{bmatrix}\right| = 0$$

$$\lambda(\lambda + 0.5) - 0.094 = 0$$
$$\lambda = 0.146, -0.646$$

Therefore, there is only saddle point.

In this work, I'm trying to develop a solution approach for the original problem with modified gradient search. In original gradient search only, single point is used as started in this work multiple random initial points are selected as starting points then gradient search with big step size is used to reach approximate min/max point. Once a feasible zone for extremum is found, step size is reduced and then again, the gradient search used to find the extremum with high accuracy. This part is developed by me based on class discussion other inferences and standard gradient search algorithms were referred.

Steps:

Inputs: Number of iterations. Number of initial points

Start from some random points x_i, y_i .

Find the gradient of each starting points.

Set some destination points for each initial point based on the estimated gradient and step size.

$X_j = X_i + t^* \ \nabla f$

find the function value at X_j ($f(X_j)$)

if $f(X_j) > f(X_i)$ % assume this is a maximization function

find new gradient and move

else don't move to new point. Come back to original point.

In this way all the points will search for new points and find the possible highest value until the number of iterations finish.

Then all the final points will be compared and a pint with maximum function value will be selected as argmax.

Pseudo code:

Input:
$f(x), t(step\ size), \tau(tolerance), n(number\ of\ initial\ points)$

Set , τ ,n,X_i

Pick multiple initial points $x_i = [3,3], [4,2], [1,3],[10,1],[5,6]$

t=0.5;

Do the below Steps until $f(x_{new}) - f(x_{old}) < \tau$ and all the other points are stationary.

1. Do until $f(x) - f(x_k + tv) < f(x) - t * \nabla f(x_{old})^T$
2. $x_{new} = x_{old} + tv$
3. Once for all initial points $f(x_{new}) - f(x_{old}) < \tau$ Stop.

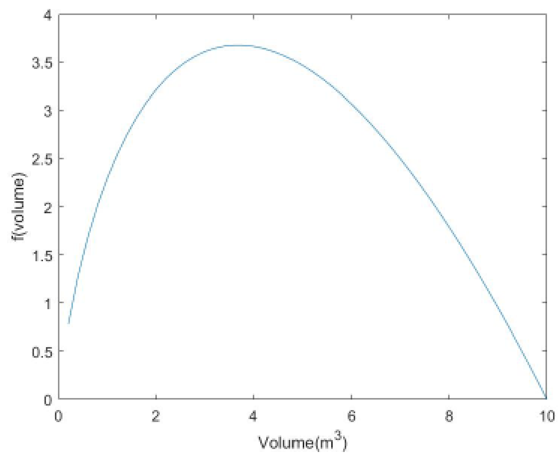
The code for this work is not fully developed. Therefore an alternative solution method is discussed in next section.

The original objective function can be efficiently solved by below discussed method. Since the objective function of this work has a special characteristic.

It can be seen from the 3d plot of the original function that the increase in pressure (P_t) will always increase the output energy. In this work maximum possible pressure is taken as 5 Nm^{-2}. Therefore, the objective function can be simplified as

$$\max_{P_t, V_1} V_1 \, ln\left(\frac{10}{V_1}\right)$$

Then the plot for this function becomes as:



Therefore, the solution can be found by doing gradient search only in volume direction. And optimum values are obtained when V_1 =3.678 m^3 and P_t=5 Nm^{-2}. Maximum energy for this storage value is 1.839 MW.