Linear and Quadratic Programming Assignment 2022
SC42056 Optimization for Systems and Control

Maria de Neves de Fonseca, 5611679
Qingyi Ren, 5684803

September 2022
Delft University of Technology

I. Configuring the mining setup

The university is contemplating to allow students to pay their tuition fee using a new cryptocurrency called DelftCoin. In this assignment we will explore some options for acquiring a setup of multiple mining units and how we can optimally control one such unit to provide enough DelftCoin for the tuition.

To mine for DelftCoin, the university offers two models of USB-powered mining devices: GreenMine which can mine 3.6 coins per day on average and a BlueMine model which mines 2.2 coins per day. They are priced at €250 and €120, respectively. The mining devices can be connected in series into a larger mining cluster. However, the computer is USB bandwidth limited, so only up to 10 mining devices can be installed. The budget is limited to €$2000 + 20E_1$.

A. *Linear Programming (LP) in standard form*

A general linear programming problem can be written as follows [1]:

Minimize the objective function

$$f(x_1, x_2, ..., x_n) = c_1x_1 + c_2x_2 + ... + c_nx_n$$

with respect to

$$a_{11}x_1 + ... + a_{1n}x_n = b_1$$

$$a_{21}x_1 + ... + a_{2n}x_n = b_2$$

$$\vdots$$

$$a_{m1}x_1 + ... + a_{mn}x_n = b_m$$

$$x_i \geq 0 \quad \text{with i=1,...,n}$$

Thus, the standard form for linear programming can be defined as

$$\boxed{\min c^T x \text{ s.t. } Ax = b, \quad x \geq 0}$$

Our mining setup is a maximization problem, since the objective is to get fastest mining rate of DelftCoin, expressed in coins per day. In other words, we want to know the maximum number of GreenMiners and BlueMiners we can have, when restricted to some constraints, to optimize for profit. This problem can be mathematically formulated as shown bellow, considering the number of GreenMiners (G) and number of BlueMiners (B) as continuous variables.

$$\max_{G,B} \quad 3.6G + 2.2B$$

$$250G + 120B \leq 2000 + 20E_1$$

$$G + B \leq 10$$

$$G, B \geq 0$$

Now, let's rewrite it into the standard form for linear programming problems. For this task, since we have a maximization problem and the standard form is written for minimization problems, we have to rewrite it as $\max c^T x = -\min(-c)^T x$. Similarly, the constraints need to be equality constraints instead of inequalities. For that we introduce dummy variables $s \geq 0$ with $Ax + Is = b$.

$$-\min_{G,B} \quad -3.6G - 2.2B$$

$$250G + 120B + s_1 = 2000 + 20E_1$$

$$G + B + s_2 = 10$$

$$G, B, s_1, s_2 \geq 0$$

A side note: E1, E2, and E3 are parameters ranging from 0 to 18 for each group according to a specific sum of the last three numbers of the student or employee IDs:

E1 = Da,1 + Db,1, E2 = Da,2 + Db,2, E3 = Da,3 + Db,3

with Da,3 the right-most digit of the ID of the first group member, Db,3 the right-most digit of the ID of the other group member, Da,2 the one but last digit of the ID of the first group member, etc. In our case:

E1 = 6+8 = 14, E2 = 7+0 = 7, E3 = 3+9 = 12

B. What is the mining rate of the resulting mining setup, and the optimal number of G and B to be installed?

To know the mining rate of the resulting mining setup, and the optimal number of G and B to be installed, the optimal solution of the LP problem was found by using the continuous formulation and `linprog` in MATLAB. Note that even though G and B should be integers, we are using `linprog` instead of `intlinprog` for now.

The function `linprog` finds the minimum of a problem specified by [2]

$$\min_{x} \quad c^T x \text{ such that} \begin{cases} Ax \leq b, \\ A_{eq}x = b_{eq}, \\ lb \leq x \leq ub \end{cases} \tag{1}$$

where the matrix $A = [250 \quad 120; 1 \quad 1]$ and vector $b = [2000{+}20E_1; \quad 10]$ are defined taking into account the problem inequality constraints. The vector c^T is defined as $c^T = [-3.6 \quad -2.2]$.

```
%% Question 1.1

clear all
close all

c=[-3.6 -2.2];
A=[250 120
   1 1];
b=[2000+20*14 10]';

lb=zeros(2,1);
ub=[];              % No upper bound
[x,fval,exitflag]=linprog(c,A,b,[],[],lb,ub)
rate=-c*x
```

The `exitflag` was equal to 1, therefore the problem converged to a solution x. The mining rate obtained is 33.6308 and the optimal number of G and B are 8.3077 and 1.6923, respectively. However, as mentioned before, G and B represent the number of each mining device, which should be integer. Knowing this, we can either round up or down the optimal solution of the LP problem obtained using Matlab, and check which one leads to the fastest mining rate while respecting the constraints.

Option	G	B	Mining rate	Total number of devices	Cost	Conclusions
1	8	2	33.2	10	2240	feasible and optimal
2	8	1	31.0	9	2120	feasible but not optimal
3	9	2	36.8	11	2490	unfeasible (violates constraints for #devices and cost)
4	9	1	34.6	10	2370	unfeasible (violates cost constraint)

TABLE I
POSSIBLE COMBINATIONS OF THE NUMBER OF DEVICES

After the analysis of table I, we can conclude that the feasible and optimal solution when considering G and B integer numbers is the one stated in option 1 where the number of G devices is 8 and the number of B devices is 2. For this case we are using the maximum amount of devices allowed, the mining rate is 33.2 and all the constraints are satisfied.

Figure 1 shows the graphical visualisation of the constraints for this problem and the respective feasible region being represented by the darkest blue area, since it is the set of solutions that satisfy all constraints. We can see that option 1 is indeed inside the feasible region.

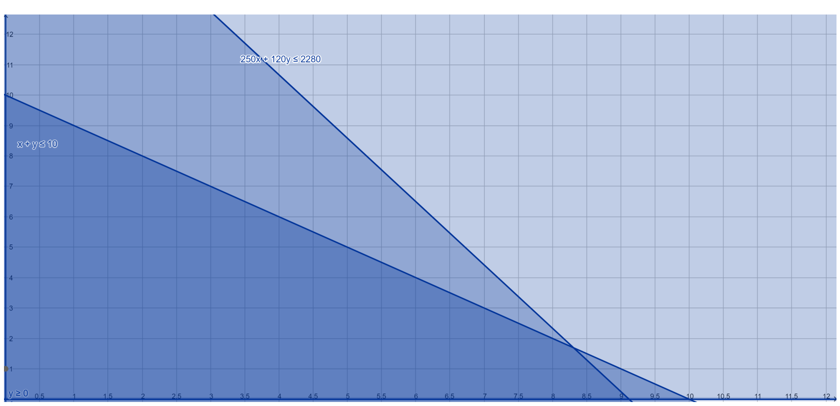

Fig. 1. Constraints and feasible region, x=G and y=B

An interesting observation is that even though we have three inequality constraints, only one of them is really limiting this setup, more precisely the limitation in the budget:

$$250G + 120B \leq 2000 + 20E_1$$

This can be seen in table I. If we want higher mining rates some constraints are not satisfied anymore, starting with option 4. For a mining rate of 34.6, the budget constraint is violated, being the first constraint to not be satisfied and, therefore, the one limiting the setup.

C. What is the cheapest setup to mine enough for your tuition fee in one year?

Assuming we want to get just enough DelftCoin to pay for a year's worth of tuition fee in 365 days, which is $8000 + 20E_2 = 8140$ DelftCoin, the optimization problem is now solved in order to get the cheapest setup possible for the described purpose. The new LP problem is then defined as

$$\min_{G,B} \quad 250G + 120B$$

$$250G + 120B \leq 2000 + 20E_1$$

$$G + B \leq 10$$

$$3.6 \times 365G + 2.2 \times 365B = 8000 + 20E_2$$

$$G, B \geq 0$$

In the standard form for linear programming problems we can formulate the problem as

$$\min_{G,B} \quad 250G + 120B$$

$$250G + 120B + s_1 = 2000 + 20E_1$$

$$G + B + s_2 = 10$$

$$3.6 \times 365G + 2.2 \times 365B = 8000 + 20E_2$$

$$G, B, s_1, s_2 \geq 0$$

For this case, the values of the matrices A, A_{eq}, and vectors b, b_{eq} and c from (1) were defined in the same way as in the previous question in Matlab, by taking into account the constraints and objective function.

```
%% Question 1.3

clear all
close all

c=[250 120];
A=[250 120
   1 1];
b=[2000+20*14 10]';

Aeq=[3.6*365 2.2*365];
```

```
beq=8000+20*7;

lb=zeros(2,1);
ub=[];              % No upper bound
[x,fval,exitflag]=linprog(c,A,b,Aeq,beq,lb,ub)
spent=c*x
```

Once again, the `exitflag` was equal to 1. Thus, the minimum amount of money that can be used to get the setup is €1228, and the optimal number of G and B are 0.2153 and 9.7847, respectively. However, G and B represent the number of each mining device, which should be integer, so we proceed to round up or down the optimal solution of the LP problem obtained using Matlab, and check which one leads to the cheapest setup while respecting the constraints.

Option	B	G	Cost	# of devices	Total amount mined	Conclusions
1	9	1	1330	10	8541	unfeasible (more than the amount needed to pay the fee)
2	9	0	1080	9	7227	unfeasible (less than the amount needed to pay the fee)
3	10	1	1450	11	9344	unfeasible (violates constraints for #devices and value of the fee)
4	10	0	1200	10	8030	unfeasible (less than the amount needed to pay the fee)

TABLE II
POSSIBLE COMBINATIONS OF THE NUMBER OF DEVICES

Looking at table II, it can be seen that neither of the options really satisfies all the constraints mainly because we just wanted mine exactly the amount needed to pay the fees, which is not possible when we want to write G and B as integers. However, we can check that if we would be willing to mine a bit more than the value of the fees, meaning that $3.6 \times 365G + 2.2 \times 365B \geq 8000 + 20E_2$, the integer solution for this optimization problem satisfying all the other constraints would the one presented in option 1 where the number of G devices is 1 and the number of B devices is 9. The cost of this setup would be €1330 and in one year (365 days) we could mine a total amount of 8541 DelftCoin which is more than enough to pay the fees.

Figure 2 shows the graphical visualisation of the constraints for this problem (if we assume we could mine a bit more than the value of the fees) and the respective feasible region being represented by the darkest blue area, since it is the set of solutions that satisfy all constraints. We can check that for this case, having 1 G device and 9 B devices would be indeed inside the feasible region. But if we assume we just want to mine the exact value of the fees, the solution would have to be on the line representing the function $3.6 \times 365G + 2.2 \times 365B = 8000 + 20E_2$, and none of the options from table II satisfies that.

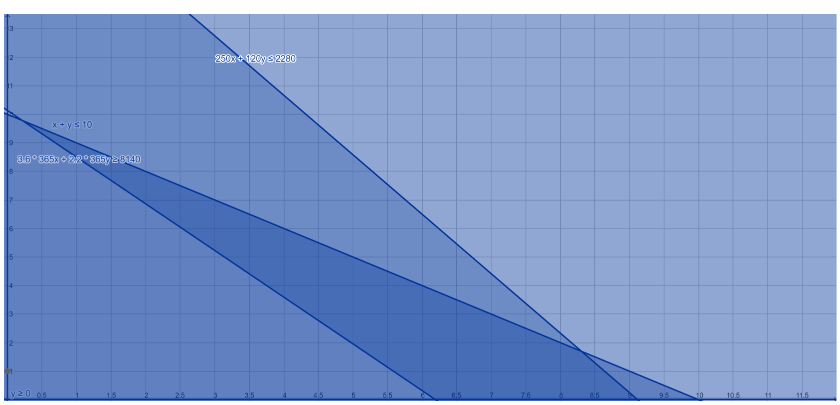

Fig. 2. Constraints and feasible region, x=G and y=B

II. Obtaining a discrete-time model for a single mining device

To try to reduce the costs of computation and cooling, and make the setup more efficient, a discrete-time model for a single mining device was obtained as a first step towards optimization. The temperature dynamics of one of these mining devices can be modeled by one state $T(t)$ representing the internal temperature at time t, an input for the ambient temperature $T^{amb}(t)$, and two inputs $\dot{q}^{in}(t)$ representing the power used by the mining device for computation and $\dot{q}^{out}(t)$representing the power used to cool the device. The dynamic model for the temperature inside the mining device, $T(t)$, is represented by the following differential equation:

$$\frac{dT(t)}{dt} = a_1[T^{amb}(t) - T(t)] + a_2[\dot{q}^{in}(t) - \dot{q}^{out}(t)]$$

where a_1 and a_2 are model parameters and $\dot{q}^{in}(t)$, $\dot{q}^{out}(t) \geq 0$.

To identify the model parameters that govern the mining device, the continuous-time model is transformed into a discrete-time version using the following approximation:

$$\frac{dT_k}{dt} \approx \frac{T_{k+1} - T_k}{\Delta t}$$

where T_k represents the internal temperature at time step k. The resulting model can be obtained by doing the following operations:

$$\frac{dT_k}{dt} \approx \frac{T_{k+1} - T_k}{\Delta t} \Rightarrow \frac{T_{k+1} - T_k}{\Delta t} = a_1[T_k^{amb} - T_k] + a_2[\dot{q}_k^{in} - \dot{q}_k^{out}] \Leftrightarrow$$

$$\Leftrightarrow T_{k+1} = [1 - a_1\Delta t]T_k + [a_2[\dot{q}_k^{in} - \dot{q}_k^{out}] + a_1 T_k^{amb}]\Delta t \Leftrightarrow T_{k+1} = AT_k + B[\dot{q}_k^{in}, \dot{q}_k^{out}, T_k^{amb}]^T$$

where $A = [1 - a_1\Delta t]$ and $B = [a_2\Delta t \quad -a_2\Delta t \quad a_1\Delta t]$.

III. Identifying the model parameters of a single mining devices

The power consumption for computation and fans respectively, as well as the internal and ambient temperature were recorded. These were sampled at intervals of 1 minute (i.e., $\Delta t = 60s$) and were be obtained by downloading the file `measurements22.csv` from Brightspace. The Matlab function `readtable` was used to read in the available data as shown bellow. The data was saved in the respective variables in the following order T_k, T_k^{amb}, $\dot{q}_k^{in}$ and $\dot{q}_k^{out}$.

```
T = readtable("measurements22.csv");

delta_t = 60;
T_k = T.Var1;
q_in = T.Var3;
q_out = T.Var4;
T_amb = T.Var2;
T_k_1 = T_k(2:end);
```

The goal was to solve the following optimization problem:

$$\min_{a_1,a_2} \sum_{k=1}^{N} (T_{k+1} - (AT_k + B[\dot{q}_k^{in}, \dot{q}_k^{out}, T_k^{amb}]^T))^2$$

The problem can be re-written into an unconstrained quadratic programming problem in the form of:

$$\boxed{\min_x \frac{1}{2} x^T H x + c^T x}$$

Thus, our problem could be written in order to find the matrices H and c as follows

$$\min_{a_1,a_2} \sum_{k=1}^{N} (T_{k+1} - (AT_k + B[\dot{q}_k^{in}, \dot{q}_k^{out}, T_k^{amb}]^T))^2$$

$$= \min_{a_1,a_2} \sum_{k=1}^{N} (T_{k+1} - T_k + a_1 T_k \Delta t - a_2 \dot{q}_k^{in} \Delta t + a_2 \dot{q}_k^{out} \Delta t - a_1 T_k^{amb} \Delta t)^2$$

$$= \min_{a_1,a_2} \sum_{k=1}^{N} (a_1(T_k \Delta t - T_k^{amb} \Delta t) + a_2(-\dot{q}_k^{in} \Delta t + \dot{q}_k^{out} \Delta t) + T_{k+1} - T_k)^2$$

$$= \min_{a_1,a_2} \sum_{k=1}^{N} ([T_k \Delta t - T_k^{amb} \Delta t \qquad - \dot{q}_k^{in} \Delta t + \dot{q}_k^{out} \Delta t] \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + T_{k+1} - T_k)^2$$

$$= \min_{a_1,a_2} \sum_{k=1}^{N} (\underbrace{[T_k \Delta t - T_k^{amb} \Delta t \qquad - \dot{q}_k^{in} \Delta t + \dot{q}_k^{out} \Delta t]}_{\phi} \underbrace{\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}}_{x} - \underbrace{(-T_{k+1} + T_k)}_{Y})^2$$

$$= \min_x \sum_{k=1}^{N} (\phi x - Y)_k^2 = \min_x (\phi x - Y)^T (\phi x - Y) = \min_x (x^T \phi^T - Y^T)(\phi x - Y)$$

$$= \min_x \quad x^T \phi^T \phi x - x^T \phi^T Y - Y^T \phi x + Y^T Y = \min_x \quad \frac{1}{2} x^T \underbrace{2\phi^T \phi}_{H} x + \underbrace{(-2\phi^T Y)^T}_{c^T} x + \underbrace{Y^T Y}_{d}$$

$$= \min_x \quad \underbrace{\frac{1}{2} x^T H x + c^T x}_{f(x)} + d = \min_x f(x) + d, \text{ with } d \text{ being a scalar.}$$

Note that the matrix H is symmetric positive definite and, therefore, $\frac{1}{2} x^T H x$ is a convex function. Also, since d is a scalar and the goal is to get the optimal solution x^*, one can assume that $\min_x f(x) + d$ leads to the same solution as $\min_x f(x)$. Moreover, it is worth noticing that since we are in the presence of an unconstrained quadratic programming problem, the optimal solution x^* can be drawn analytically:

$$\boxed{\bigtriangledown f(x^*) = Hx^* + c = 0 \Leftrightarrow x^* = -H^{-1} c}$$

```
Y = -T_k_1+T_k(1:end-1);
phi = [(T_k(1:end-1)-T_amb(1:end-1))*delta_t (-q_in(1:end-1)+q_out(1:end-1))*delta_t
    ];
H = 2*phi'*phi;
c = -2*phi'*Y;

%-------check if H is symmetric positive definite------%
% try chol(H)
%     disp('Matrix is symmetric positive definite.')
```

```
% catch ME
%     disp('Matrix is not symmetric positive definite')
% end

x = -inv(H)*c
```

The estimated values for a_1 and a_2 are then 0.0005 and 0.0020, respectively.

IV. Optimizing the cost for mining of a single mining device

The objective now is to maximize the mining rate while minimizing the cost for operating the miner over some horizon of N steps, starting at k = 1 for some standardized input signals, while making sure that the dynamic model is followed throughout the time horizon.

The rate of mining is directly proportional to $\dot{q}^{in}$, which is capped at $\dot{q}^{in}_{max} = 125$ W. To relate the yield of DelftCoin to the energy costs, we avoid speculation and set the price at €$0.38 + 0.01E_3$ per coin. It also takes 8 hours at full power to mine one coin, which results in a yield of $\psi = 0.38 + 0.01E_3$ €/kWh for power spent on $\dot{q}^{in}$. The cost of energy ϕ_k (€/kWh) is provided in the `measurements22.csv` file. Finally, the internal temperature T is also constrained to satisfy the operational range of the mining device, $T^{max} = 95$ °C.

Consider a horizon of 1 day cycle, i.e. $N = 1440$, assume that $T_1 = 23.3$ °C, and the model parameters, a_1 and a_2, are the ones found in the previous task. We can define the optimization problem as:

$$\min_{\substack{T_2,\ldots,T_{N+1} \\ \dot{q}^{in}_1,\ldots,\dot{q}^{in}_N}} \sum_{k=1}^{N} \phi_k(\dot{q}^{in}_k + \dot{q}^{out}_k)\Delta t - \psi \dot{q}^{in}_k \Delta t$$

$$\begin{aligned} \text{s.t. } T_{k+1} &= AT_k + B[\dot{q}^{in}_k, \dot{q}^{out}_k, T^{amb}_k]^T, \quad \text{k=1,...,N}, \\ 0 &\le \dot{q}^{in}_k \le \dot{q}^{in}_{\max}, \quad \text{k=1,...,N}, \\ T_k &\le T^{\max}, \quad \text{k=2,...,N+1}. \end{aligned}$$

First we read the data into the respective variables the same way as in section III. All the variables are also defined and converted to the right units. For instance, since we are dealing with time in seconds and power in Watt, ϕ_k and ψ are converted to €/W-second by dividing them by $60 \times 60 \times 1000 = 3600000$.

```
T = readtable("measurements22.csv");

%variables
delta_t = 60;
T_k = T.Var1;
q_in = T.Var3;
q_out = T.Var4;
T_amb = T.Var2;
T_k_1 = T_k(2:end);
phi = T.Var5/3600000;
E3 = 12;
psi = (0.38+0.01*E3)/3600000;
N = 1440;
T_max = 95;
q_in_max = 125;
T_1 = 23.3;
a_1 = 0.0005;
a_2 =  0.0020;
```

Since we are minimising in function of T_{k+1} and $\dot{q}_k^{in}$ for $k = 1, ..., N$, which compose the vector $x \in R^{2N\times 1}$, the problem can be re-written in matrix form as follows

$$x = \begin{bmatrix} T_2 \\ T_3 \\ \vdots \\ T_{N+1} \\ \dot{q}_1^{in} \\ \dot{q}_2^{in} \\ \vdots \\ \dot{q}_N^{in} \end{bmatrix}$$

$$\min_{\substack{T_2,...,T_{N+1} \\ \dot{q}_1^{in},...,\dot{q}_N^{in}}} \sum_{k=1}^{N} \phi_k(\dot{q}_k^{in} + \dot{q}_k^{out})\Delta t - \psi\dot{q}_k^{in}\Delta t =$$

$$= \min_x \underbrace{\begin{bmatrix} \overbrace{0 \quad \cdots \quad 0}^{N} & \overbrace{(\phi_1 - \psi)\Delta t \quad \cdots \quad (\phi_N - \psi)\Delta t}^{N} \end{bmatrix}}_{c^T} x + \underbrace{\sum_{k+1}^{N} \phi_k \dot{q}_k^{out}\Delta t}_{d}$$

where d is a scalar and, therefore, doesn't affect the values of x, but only the value of the objective function. Thus, to find the values of x we can just solve the following

$$\min_x \underbrace{\begin{bmatrix} \overbrace{0 \quad \cdots \quad 0}^{N} & \overbrace{(\phi_1 - \psi)\Delta t \quad \cdots \quad (\phi_N - \psi)\Delta t}^{N} \end{bmatrix}}_{c^T} x$$

Our constraints can be re-written as follows

$$T_{k+1} = AT_k + B[\dot{q}_k^{in}, \dot{q}_k^{out}, T_k^{amb}]^T \Leftrightarrow T_{k+1} = AT_k + a_2\dot{q}_k^{in}\Delta t - a_2\dot{q}_k^{out}\Delta t + a_1 T_k^{amb}\Delta t$$

$$\Leftrightarrow T_{k+1} - AT_k - a_2\dot{q}_k^{in}\Delta t = -a_2\dot{q}_k^{out}\Delta t + a_1 T_k^{amb}\Delta t$$

$$\Leftrightarrow T_{k+1} - AT_k - a_2\dot{q}_k^{in}\Delta t = -a_2\dot{q}_k^{out}\Delta t + a_1 T_k^{amb}\Delta t$$

$$\Leftrightarrow \underbrace{\begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & -a_2\Delta t & 0 & 0 & \cdots & 0 \\ -A & 1 & \ddots & \vdots & 0 & 0 & -a_2\Delta t & 0 & \ddots & \vdots \\ 0 & -A & \ddots & 0 & \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 & 0 & \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -A & 1 & 0 & \cdots & \cdots & 0 & -a_2\Delta t \end{bmatrix}}_{\substack{N\times 2N \\ A_{eq}}} \underbrace{\begin{bmatrix} T_2 \\ T_3 \\ \vdots \\ T_{N+1} \\ \dot{q}_1^{in} \\ \vdots \\ \dot{q}_N^{in} \end{bmatrix}}_{\substack{2N\times 1 \\ x}} = \underbrace{\begin{bmatrix} -a_2\dot{q}_1^{out}\Delta t + a_1 T_1^{amb}\Delta t + AT_1 \\ -a_2\dot{q}_2^{out}\Delta t + a_1 T_2^{amb}\Delta t \\ \vdots \\ -a_2\dot{q}_N^{out}\Delta t + a_1 T_N^{amb}\Delta t \end{bmatrix}}_{\substack{N\times 1 \\ b_{eq}}}$$

$$\begin{cases} 0 \le \dot{q}_k^{in} \le \dot{q}_{max}^{in} \\ T_{k+1} \le T^{max} \end{cases}, \text{ for } k = 1, ..., N \Leftrightarrow \underbrace{\begin{bmatrix} I_{N\times N} & 0_{N\times N} \\ 0_{N\times N} & I_{N\times N} \\ 0_{N\times N} & -I_{N\times N} \end{bmatrix}}_{A_{ineq}} \underbrace{\begin{bmatrix} T_2 \\ T_3 \\ \vdots \\ T_{N+1} \\ \dot{q}_1^{in} \\ \vdots \\ \dot{q}_N^{in} \end{bmatrix}}_{x} \le \underbrace{\begin{bmatrix} T^{\max}_{N\times 1} \\ \dot{q}^{in}_{\max\ N\times 1} \\ 0_{N\times 1} \end{bmatrix}}_{b_{ineq}}$$

As it can be seen, this problem can be written in the general form of a linear programming problem as showed in question 1. Therefore, the function `linprog` can be used for this problem. The matrices A, A_{eq}, and vectors b, b_{eq} and c^T from (1) are now A_{ineq}, A_{eq}, b_{ineq}, b_{eq} and c^T, respectively, to differentiate from A referring to $[1-a_1\Delta t]$. These are defined taking into account the constraints and objective function as defined above.

```
%optimization problem
c=[zeros(1,N), delta_t*(phi(1:N)'-psi)];

D=zeros(1,N-1);
for i=1:numel(D)
    D(i)=-(1-a_1*delta_t);
end

J=zeros(1,N);
for i=1:numel(J)
    J(i)=-a_2*delta_t;
end

A_eq_1 = diag(ones(1,N))+diag(D,-1);
A_eq_2 = diag(J);

A_eq = [A_eq_1 A_eq_2];

b_eq_1=zeros(1,N);
for i=1:numel(b_eq_1)
    b_eq_1(i)=-a_2*delta_t*q_out(i)+a_1*delta_t*T_amb(i);
end

b_eq_2 = [(1-a_1*delta_t)*T_1, zeros(1,N-1)];

b_eq = (b_eq_1+b_eq_2)';

A = [eye(N) zeros(N,N);
     zeros(N,N) eye(N);
     zeros(N,N) -eye(N)];

b = [T_max*ones(N,1);q_in_max*ones(N,1); zeros(N,1)];

[x,fval,exitflag]=linprog(c,A,b,A_eq,b_eq,[],[]);
total_cost = c*x + sum(phi(1:N).*q_out(1:N)*delta_t)
```

The optimized cost of mining with this specific mining device is -0.1746€, which means we gained 0.1746€. For this optimal solution, the behaviour of the internal temperature T and the power used by the mining device for computation $\dot{q}^{in}$ are presented in figure 3. It can be seen that to minimize the cost and maximize the yield, the internal temperature of the device reaches its maximum limit and the power used by the mining device for computation firstly reaches its maximum and then is kept relatively constant around 90 Watt. Of course there are some oscillations for both results due to the recorded values of the other variables, $\dot{q}^{out}$ and T^{amb}, from `measurements22.csv`.

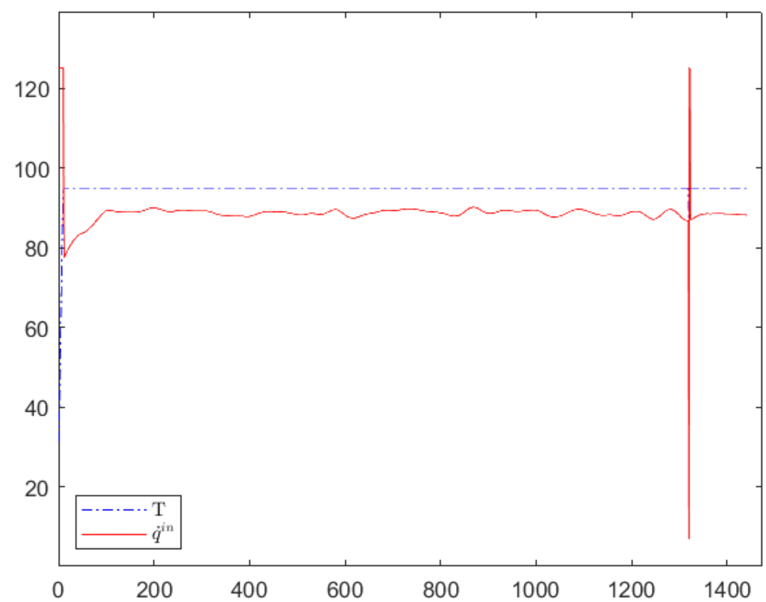

Fig. 3. Behaviour of the internal temperature T and the power used by the mining device for computation $\dot{q}^{in}$

References

[1] Optimization for Systems and Control (SC42056). (2020, October 17). Retrieved from https://www.dcsc.tudelft.nl/ bdeschutter/osc

[2] Linear Program Linear Inequality Constraints Example. (2022, September 30). Retrieved from https://nl.mathworks.com/help/optim/ug/linprog.html