



Solution for Project 5

Due date: Wednesday, December 20, 2023, 11:59 PM

Numerical Computing 2023 — Submission Instructions

(Please, notice that following instructions are mandatory:
submissions that don't comply with, won't be considered)

- Assignments must be submitted to iCorsi (i.e. in electronic format).
- Provide both executable package and sources (e.g. C/C++ files, MATLAB). If you are using libraries, please add them in the file. Sources must be organized in directories called:
Project_number_lastname_firstname
and the file must be called:
project_number_lastname_firstname.zip
project_number_lastname_firstname.pdf
- The TAs will grade your project by reviewing your project write-up, and looking at the implementation you attempted, and benchmarking your code's performance.
- You are allowed to discuss all questions with anyone you like; however: (i) your submission must list anyone you discussed problems with and (ii) you must write up your submission independently.

The purpose of this project is to implement the Simplex Method to find the solution of linear programs, involving both the minimisation and the maximisation of the objective function.

1. Graphical Solution of Linear Programming Problems [20 points]

Please consider the following two problems:

(1)

$$\begin{aligned} \min \quad & z = 4x + y \\ \text{s.t.} \quad & x + 2y \leq 40 \\ & x + y \geq 30 \\ & 2x + 3y \geq 72 \\ & x, y \geq 0 \end{aligned}$$

- (2) A tailor plans to sell two types of trousers, with production costs of 25 CHF and 40 CHF, respectively. The former type can be sold for 85 CHF, while the latter for 110 CHF. The tailor estimates a total monthly demand of 265 trousers. Find the number of units of each type of trousers that should be produced in order to maximise the net profit of the tailor, if we assume that the he cannot spend more than 7000 CHF in raw materials.

Start by writing problem (2) as a linear programming problem. Then complete the following tasks:

- Solve the system of inequalities.
- Plot the feasible region identified by the constraints.
- Find the optimal solution and the value of the objective function in that point.

(1) Solving the system of inequalities:

$$\begin{cases} x + 2y \leq 40 \\ x + y \geq 30 \\ 2x + 3y \geq 72 \\ x, y \geq 0 \end{cases} \Rightarrow \begin{cases} r_1 : y \leq 20 - \frac{1}{2}x \\ r_2 : y \geq 30 - x \\ r_3 : y \geq 24 - \frac{2}{3}x \\ x, y \geq 0 \end{cases}$$

Figure 1: Exercise 1.1 - Feasible Region

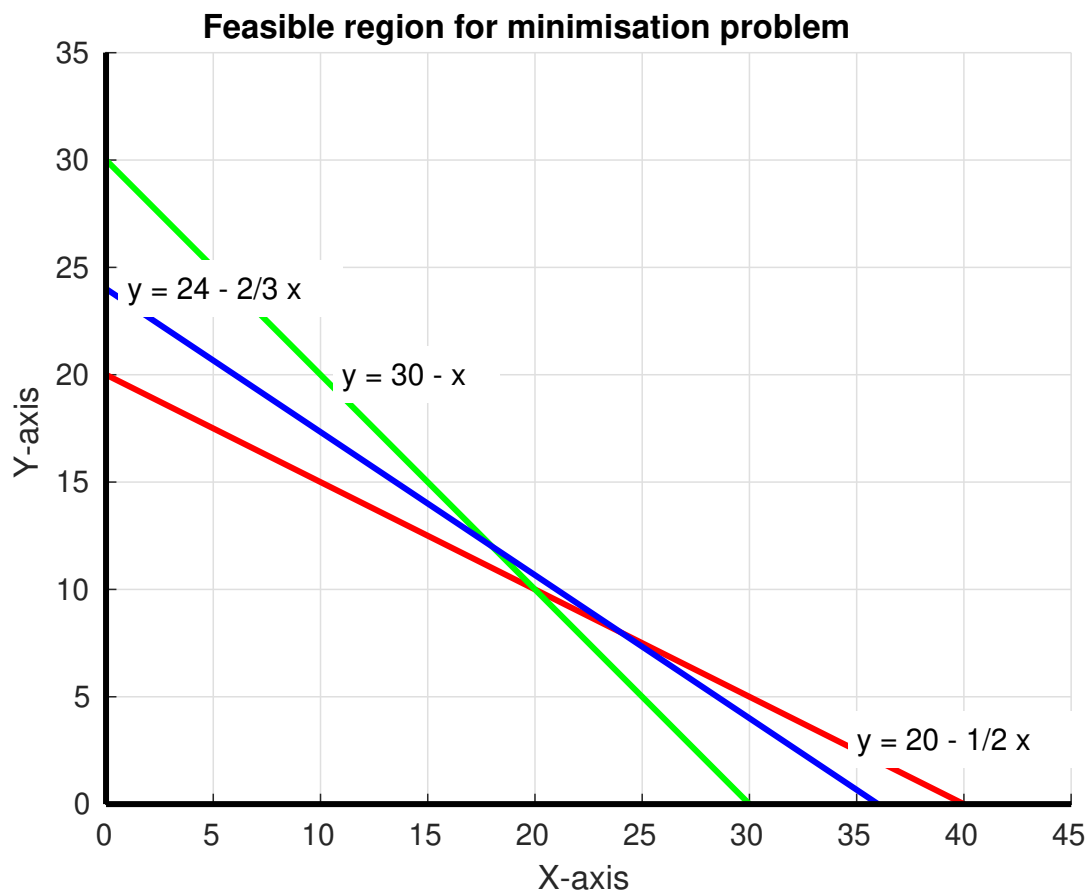


Figure 2: Exercise 1.1 - Feasible Region visualization



The figure 1 shows the feasible region of the problem (1): the 3 main lines and the non-negativity constraints in black corresponding to the x,y axis. The figure 2 shows the same feasible region, but marking their regions with their corresponding colors to clearly see the common area overlapping.

To get the points of the feasible region, we can solve the system of equations:

Intersections at $x = 0$:

$$\begin{cases} x \cap r_1 \\ x \cap r_2 \\ x \cap r_3 \end{cases} \Rightarrow \begin{cases} y \leq 20 - \frac{1}{2} \cdot 0 \\ y \geq 30 - 0 \\ y \geq 24 - \frac{2}{3} \cdot 0 \end{cases} \Rightarrow \begin{cases} P_1 = (0, 20) \\ P_2 = (0, 30) \\ P_3 = (0, 24) \end{cases}$$

Intersections at $y = 0$:

$$\begin{cases} y \cap r_1 \\ y \cap r_2 \\ y \cap r_3 \end{cases} \Rightarrow \begin{cases} 0 \leq 20 - \frac{1}{2}x \\ 0 \geq 30 - x \\ 0 \geq 24 - \frac{2}{3}x \end{cases} \Rightarrow \begin{cases} P_4 = (40, 0) \\ P_5 = (30, 0) \\ P_6 = (36, 0) \end{cases}$$

Intersections between lines:

$$\begin{cases} r_1 \cap r_2 \\ r_1 \cap r_3 \\ r_2 \cap r_3 \end{cases} \Rightarrow \begin{cases} 20 - \frac{1}{2}x = 30 - x \\ 20 - \frac{1}{2}x = 24 - \frac{2}{3}x \\ 30 - x = 24 - \frac{2}{3}x \end{cases} \Rightarrow \begin{cases} P_7 = (20, 10) \\ P_8 = (24, 8) \\ P_9 = (18, 12) \end{cases}$$

These 3 points constitute the feasible region because included in the regions of all constraints:

- | | | |
|-------------------|---------------------------------------|-----------------------------------|
| • $P_4 = (40, 0)$ | | • $z(P_4) = 4 \cdot 40 + 0 = 160$ |
| • $P_6 = (36, 0)$ | $\xrightarrow{\text{evaluate in } z}$ | • $z(P_6) = 4 \cdot 36 + 0 = 144$ |
| • $P_8 = (24, 8)$ | | • $z(P_8) = 4 \cdot 24 + 8 = 104$ |

The optimal solution is P_8 , with the value of the objective function z minimizing the problem.

(2) Rewrite the exercise as a linear programming problem:

- trousers type 1, 2
- production cost 25, 40
- selling price 85, 110
- total demand 265
- maximize profit
- spend less than 7000 in materials

$\xrightarrow{\text{max problem}}$

$$\begin{array}{ll} \max & z = 60x + 70y \\ \text{s.t.} & 25x + 40y \leq 7000 \\ & x + y \leq 265 \\ & x, y \geq 0 \end{array}$$

Solving the system of inequalities:

$$\begin{cases} 25x + 40y \leq 7000 \\ x + y \leq 265 \\ x, y \geq 0 \end{cases} \Rightarrow \begin{cases} r_1 : y \leq 175 - \frac{5}{8}x \\ r_2 : y \leq 265 - x \\ x, y \geq 0 \end{cases}$$

Figure 3: Exercise 1.2 - Feasible Region

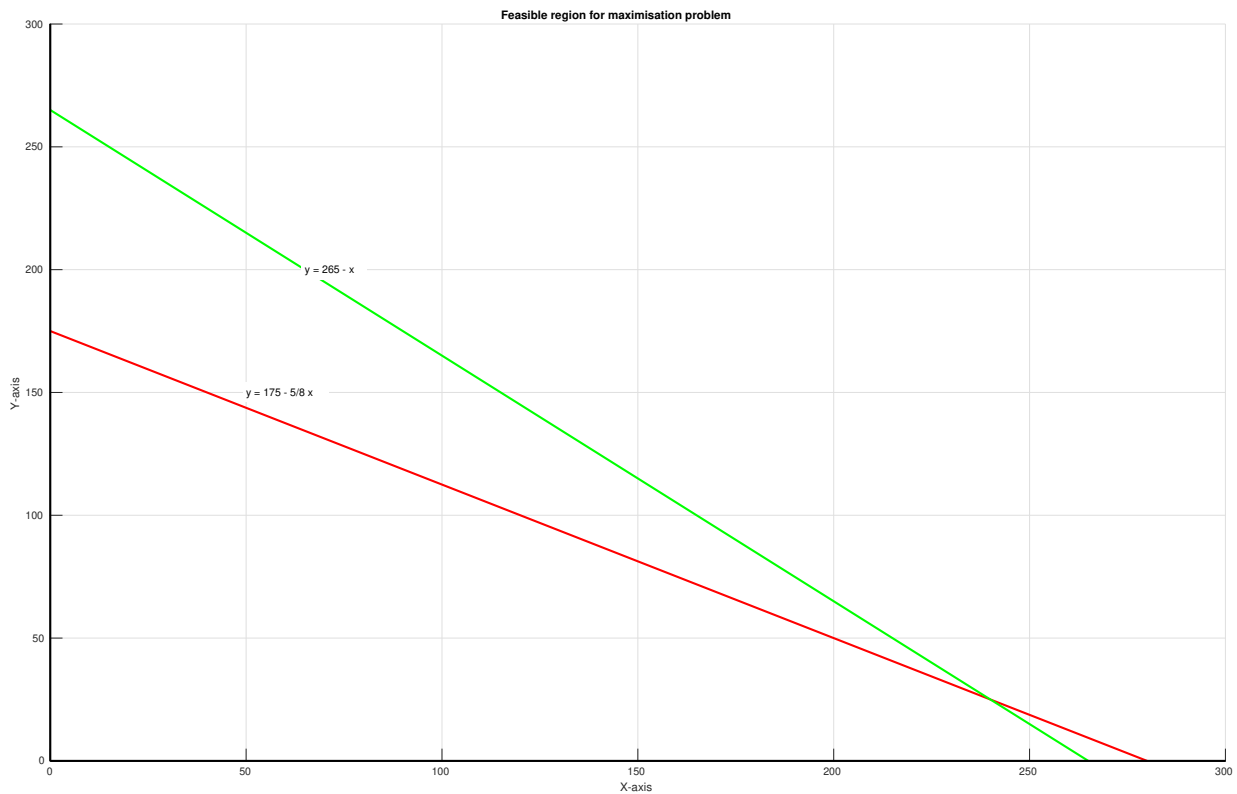
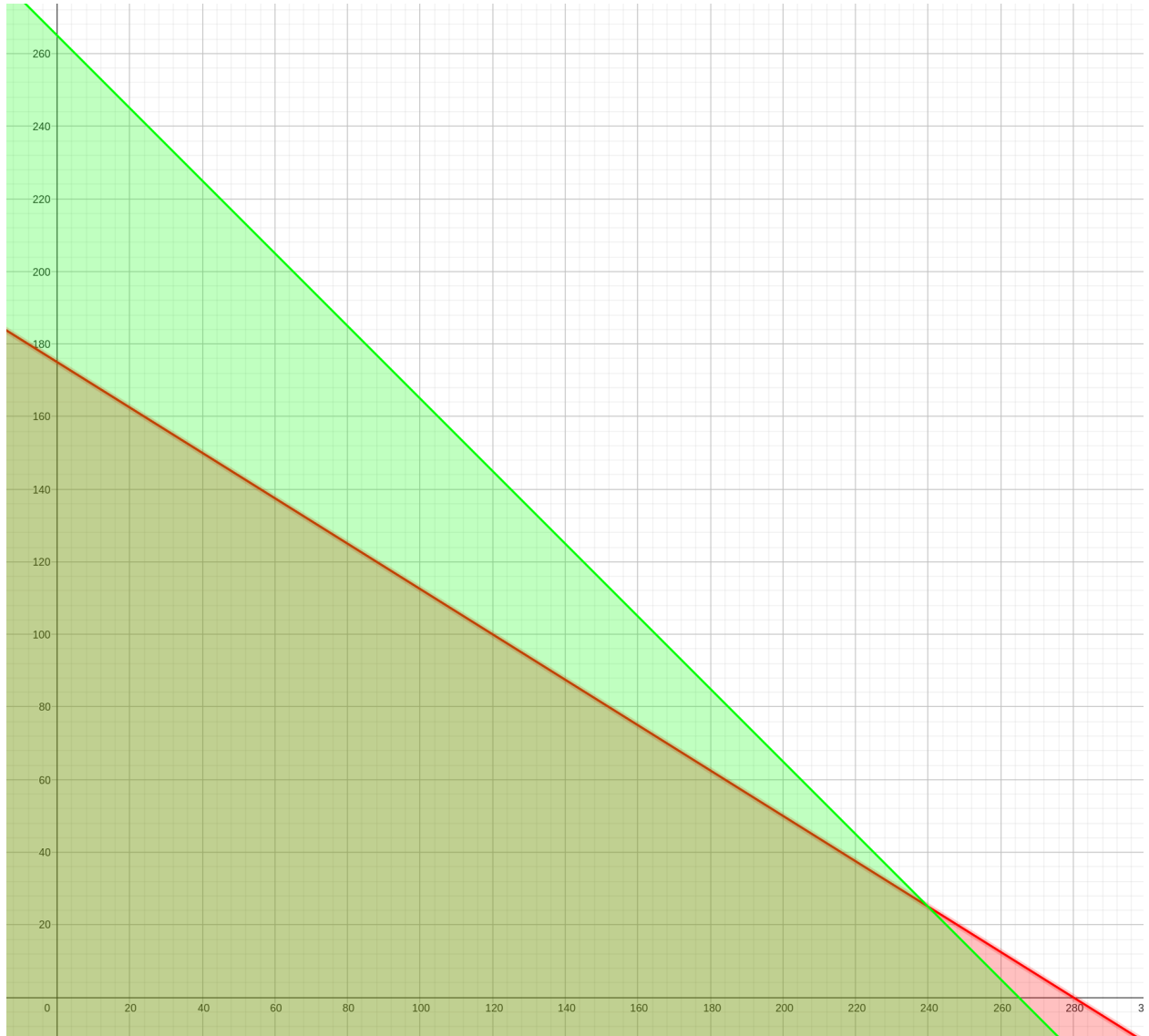


Figure 4: Exercise 1.2 - Feasible Region visualization



The figure 3 shows the feasible region of the problem (2): the 2 main lines and the non-negativity constraints in black corresponding to the x,y axis as in the previous exercise. The figure 4 shows the same feasible region, but marking their regions with their corresponding colors red and green to clearly see the common area overlapping in orange.

To get the points of the feasible region, solve the system of equations:

Intersections at $x = 0$:

$$\begin{cases} x \cap r_1 \\ x \cap r_2 \end{cases} \Rightarrow \begin{cases} y \leq 175 - \frac{5}{8} \cdot 0 \\ y \geq 265 - x \end{cases} \Rightarrow \begin{cases} P_1 = (0, 175) \\ P_2 = (0, 265) \end{cases}$$

Intersections at $y = 0$:

$$\begin{cases} y \cap r_1 \\ y \cap r_2 \end{cases} \Rightarrow \begin{cases} 0 \leq 175 - \frac{5}{8}x \\ 0 \geq 265 - x \end{cases} \Rightarrow \begin{cases} P_3 = (280, 0) \\ P_4 = (265, 0) \end{cases}$$

Intersections between lines:

$$\begin{cases} r_1 \cap r_2 \end{cases} \Rightarrow \begin{cases} 175 - \frac{5}{8}x = 265 - x \end{cases} \Rightarrow \begin{cases} P_5 = (240, 25) \end{cases}$$

These 3 points constitute the feasible region because included in the regions of all constraints:

- | | | |
|---------------------|---------------------------------------|---|
| • $P_1 = (0, 175)$ | | • $z(P_1) = 60 \cdot 0 + 70 \cdot 175 = 12250$ |
| • $P_4 = (265, 0)$ | $\xrightarrow{\text{evaluate in } z}$ | • $z(P_4) = 60 \cdot 265 + 70 \cdot 0 = 15900$ |
| • $P_5 = (240, 25)$ | | • $z(P_5) = 60 \cdot 240 + 70 \cdot 25 = 16150$ |

The optimal solution is P_5 , with the value of the objective function z maximizing the net profit.

2. Implementation of the Simplex Method [30 points]

In this first part of the assignment, you are required to complete 2 functions which are part of a dummy implementation of the simplex method. Specifically you have to complete the TODOs in:

- *standardise.m*, which writes a maximisation or minimisation input problem in standard form;
- *simplexSolve.m*, which solves a maximisation or minimisation problem using the simplex method.

You are given also some already-implemented functions to help you in your task: *simplex.m* is a wrapper which calls all the functions necessary to find a solution to the linear program; *auxiliary.m* solves the auxiliary problem to find a feasible starting basic solution of the linear program; *printSol.m* is a function which prints the optimal solution found by the simplex algorithm. Finally, *testSimplex.m* presents a series of 6 problems to check if your implementation is correct, before moving to the next part of the assignment. Additional details to aid you in your implementation can be found in the comments inside the code.

The source code of the required files has been completed and extensively commented in Matlab.

3. Applications to Real-Life Example: Cargo Aircraft [25 points]

In this second part of the assignment, you are required to use the simplex method implementation to solve a real-life problem taken from economics (constrained profit maximisation).

A cargo aircraft has 4 compartments (indicated simply as S_1, \dots, S_4) used to store the goods to be transported. Details about the weight capacity and storage capacity of the different compartments can be inferred from the data reported in the following table:

Compartment	Weight Capacity (t)	Storage Capacity (m^3)
S_1	18	11930
S_2	32	22552
S_3	25	11209
S_4	17	5870

The following four cargos are available for shipment during the next flight:

Cargo	Weight (t)	Volume (m^3/t)	Profit (CHF/t)
C_1	16	320	135
C_2	32	510	200
C_3	40	630	410
C_4	28	125	520

Any proportion of the four cargos can be accepted, and the profit obtained for each cargo is increased by 10% if it is put in S_2 , by 20% if it is put in S_3 and by 30% if it is put in S_4 , due to the better storage conditions. The objective of this problem is to determine which amount of the different cargos will be transported and how to allocate it among the different compartments, while maximising the profit of the owner of the cargo plane. Specifically you have to:

1. Formulate the problem above as a linear program: what is the objective function? What are the constraints? Write down all equations, with comments explaining what you are doing.
2. Create a script *exercise2.m* which uses the simplex method implemented in the previous exercise to solve the problem. What is the optimal solution? Visualise it graphically and briefly comment the results obtained (are you surprised of this outcome on the basis of your data?).

This is a maximisation problem, where the objective function is to maximize the profit. For the problem to be formulated as a linear program, we define:

- incognitas $x_{c,s}$
- cargo $c \in C = \{C_1, C_2, C_3, C_4\}$
- compartment $s \in S = \{S_1, S_2, S_3, S_4\}$

$$\begin{aligned} \max z &= 135 \cdot (x_{1,1} + 1.1x_{1,2} + 1.2x_{1,3} + 1.3x_{1,4}) \\ &+ 200 \cdot (x_{2,1} + 1.1x_{2,2} + 1.2x_{2,3} + 1.3x_{2,4}) \\ &+ 410 \cdot (x_{3,1} + 1.1x_{3,2} + 1.2x_{3,3} + 1.3x_{3,4}) \\ &+ 520 \cdot (x_{4,1} + 1.1x_{4,2} + 1.2x_{4,3} + 1.3x_{4,4}) \end{aligned}$$

With weight capacity constraints in t unit measure:

$$\sum_{c \in C} x_{c,1} \leq 18, \quad \sum_{c \in C} x_{c,2} \leq 32, \quad \sum_{c \in C} x_{c,3} \leq 25, \quad \sum_{c \in C} x_{c,4} \leq 17$$

With storage capacity constraints in m^3 unit measure: each cargo volume has been transformed to m^3 unit measure by multiplying weight and volume e.g. $16 t \cdot 320 m^3/t = 5120 m^3$

$$\begin{aligned} 5120 m^3 \cdot x_{1,1} + 16320 m^3 \cdot x_{2,1} + 25200 m^3 \cdot x_{3,1} + 3500 m^3 \cdot x_{4,1} &\leq 11930 m^3 \\ 5120 m^3 \cdot x_{1,2} + 16320 m^3 \cdot x_{2,2} + 25200 m^3 \cdot x_{3,2} + 3500 m^3 \cdot x_{4,2} &\leq 22552 m^3 \\ 5120 m^3 \cdot x_{1,3} + 16320 m^3 \cdot x_{2,3} + 25200 m^3 \cdot x_{3,3} + 3500 m^3 \cdot x_{4,3} &\leq 11209 m^3 \\ 5120 m^3 \cdot x_{1,4} + 16320 m^3 \cdot x_{2,4} + 25200 m^3 \cdot x_{3,4} + 3500 m^3 \cdot x_{4,4} &\leq 5870 m^3 \end{aligned}$$

With non-negativity constraints, meaning the quantities in each compartment must be positive:

$$\begin{aligned} x_{1,1} + x_{2,1} + x_{3,1} + x_{4,1} &\geq 0 \\ x_{1,2} + x_{2,2} + x_{3,2} + x_{4,2} &\geq 0 \\ x_{1,3} + x_{2,3} + x_{3,3} + x_{4,3} &\geq 0 \\ x_{1,4} + x_{2,4} + x_{3,4} + x_{4,4} &\geq 0 \end{aligned}$$

4. Cycling and Degeneracy [10 points]

Consider now the following simple problem:

$$\begin{aligned} \max z &= 3x_1 + 4x_2, \\ \text{s.t. } 4x_1 + 3x_2 &\leq 12 \\ 4x_1 + x_2 &\leq 8 \\ 4x_1 + 2x_2 &\leq 8 \\ x_1, x_2 &\geq 0. \end{aligned}$$

1. Create a script *exercise3.m* which uses the simplex method implemented above to solve this problem. Do you achieve convergence within the maximum number of iterations (given by the maximum number of possible basic solutions)? Do you notice any strange behaviour? (*hint*: check, e.g., the indices of the entering and departing variables)
2. Look at the number of constraints and at the number of unknowns: what can you notice about the underlying system of equations? Represent them graphically and try to use this information to explain the behaviour of your solver in the previous point.

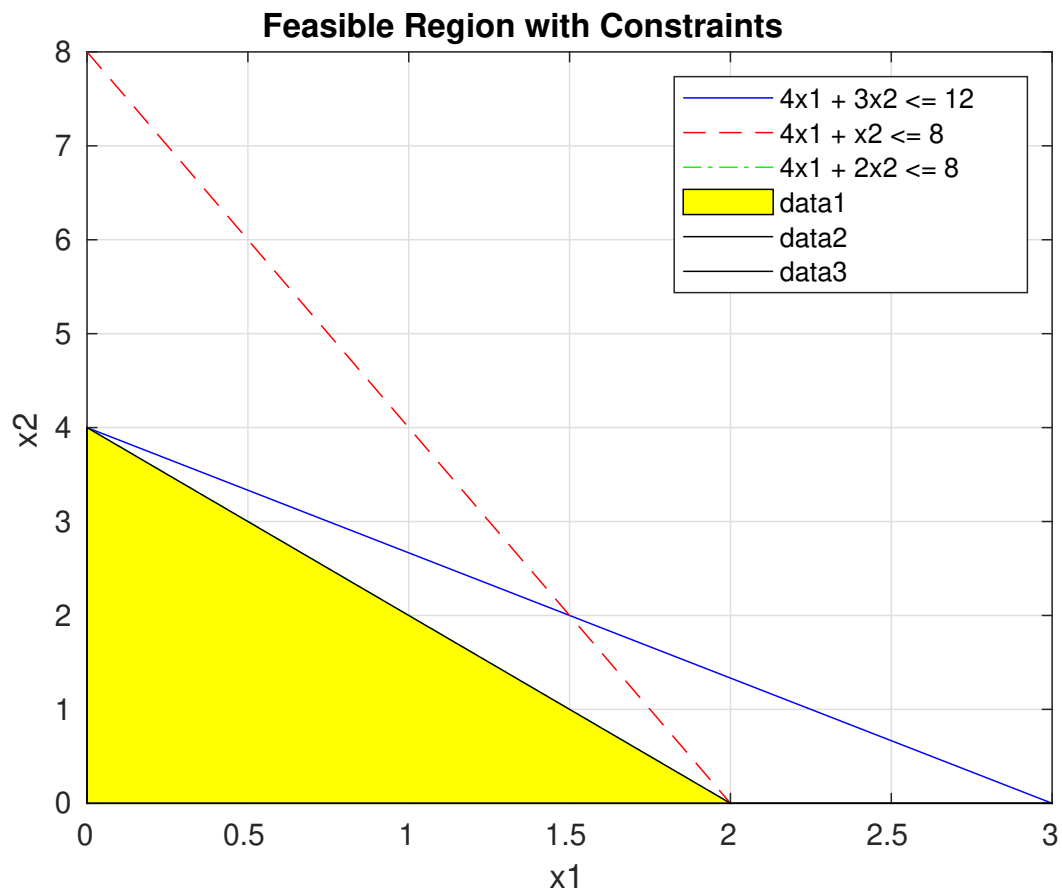
(1) The script *exercise3.m* tries to solve the problem with the simplex method implemented in the previous exercise. The problem is not solved because the maximum number of iterations is reached. Matlab prints the following message:

```
Error using simplexSolve
Incorrect loop, more iterations than the number of basic solutions
```

To recreate it I used the simplex method implemented in the previous exercise that passed the tests. The linearization problem was given in the same format the tests are given. Hardcoding a greater number of iterations than the maximum number of basic solutions up to $1e + 9$ either gives back the same error or takes too long. Hence, the system is not able to converge. This phenomenon could be related to cycling: multiple variables end up swapping the same pair(s) over and over again, hindering the capacity of the simplex method to achieve convergence. The next questions enlighten this behavior.

(2) The number of unknowns is 2: x_1, x_2 . The number of constraints is 3. Hence, the system of equations may be underdetermined or have multiple solutions. In Figure 5 we can better understand the behavior and performance of the simplex solver by noticing that there is a degeneracy, where more than one optimal solution exists: the feasible region in yellow is where all constraints are satisfied simultaneously and since more than one basic feasible solution corresponds to the same corner point of the feasible region, the overlapping of the lines means that different constraints define the same edge of the feasible region. The simplex method is then not able to move to a new vertex because multiple constraints are tight at the current vertex, potentially leading to cycling. The source code to generate the plot is in the file *exercise4_plot.m*.

Figure 5: Exercise 4.2 - Feasible Region with constraints



5. Quality of the Code & Report [15 points]

Each project sums up to 100 points, out of which 15 points are dedicated to the general quality of your written report and of the code submitted. Your report should be a coherent document. If there are theoretical questions, explain and justify your answers. If you made a particular choice in your implementation that might be out of the ordinary, please explain it in the report. The code you submit should be self-contained and executable. It should include the set-up for all the results that you obtained and listed in your report. Furthermore, it should be readable and include comments explaining the more complicated steps.