# Project Assignment in 42114 Integer Programming

Technical University of Denmark – Fall 2025

## Introduction

The project period formally starts on **Thursday, 2nd October** and the report must be handed in at the latest **Wednesday, 12th November at 23:59**. Questions to be answered are clearly marked in "question boxes". Please read the text and the questions carefully before answering them.

**Info:** Your final report must be uploaded to the appropriate group assignment on DTU Learn of the course. The report is expected to be not more than 10 pages long **excluding** tables, figures and appendices and must be written in Danish or in English. The project may be solved in groups and permissible group sizes are 1, 2, 3, or 4 persons.

## Modelling

A company produces a certain product and wants to plan its production for the next $n$ weeks so that it can meet its weekly forecast demand for the product at minimum cost. Weekly demand can be met by producing the product during the week, using inventory from the previous week, or backlogging unmet demand to the following week. No backlogging is possible beyond week $n$. Even though the company's customers tolerate backlogging, it is undesirable, and the company therefore places a limit on the total quantity backlogged across all weeks of $Q_b$ items (note that this is not per week, but *across* all $n$ weeks). Producing an item of the product in week $w = 1, 2 \ldots, n$, costs $p_w > 0$ (€), while holding an item of the product in inventory costs $h > 0$ (€) per week. Assume that the demand for the product in week $w$ is indicated by $d_w$. A single machine is available to produce the product, and this must be turned on to enable production.

**Question 1**

Formulate a Mixed Integer Linear Programming problem that can be used to optimize the company's production plan. If you use a big-M construction, state an appropriate value for each of these parameters.

**Question 2**

Restarting the machine needed for production can be a hassle, and the company would like to be able to limit the number of restarts to at most a pre-specified parameter $Q_s$. A restart is necessary when moving from a period with *no production* to a period *with production*. What changes are necessary to the model in Question 1 to be able to enforce this?

Question 3

Significant variation in weekly production volumes is undesirable. The company would like to investigate what happens to the production plan when the objective is to minimize the difference between the largest weekly production volume and the smallest (only considering weeks in which production occurs). What changes are necessary to the model in Question 1 to be able to optimize this objective?

Question 4

The company would now like to incorporate a maintenance requirement. Specifically, the machine must undergo one week of maintenance after it has been used in four production weeks. Idle weeks (with no production) delay this requirement; only production weeks count toward the total. Maintenance must occur before the fifth production week, lasts one week, stops production, and costs $\gamma > 0$ (€). What changes are necessary to the model in Question 1 to be able to enforce this?

## Formulations

Recall the container-packing problem from the exercises in Week 2. Given $n$ containers of various sizes $(v_1, v_2, \ldots, v_n)$, and assuming that each container $i = 1, 2, \ldots, n$ contains a known amount of liquid $a_i$ (>0) the aim is to combine the contents of the containers in such a way that the smallest number of **non-empty** containers is needed. When combining containers, it is not possible to split the liquid of a container across more than one container, and if a container is used, then the liquid that was originally in it must remain in it.

Consider an instance of this problem in which there are 15 containers with volumes and amounts specified in Table 1.

| | Container | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| $v_i$ | 400 | 450 | 520 | 330 | 400 | 350 | 250 | 300 | 280 | 310 | 340 | 290 | 275 | 180 | 310 |
| $a_i$ | 290 | 240 | 210 | 300 | 175 | 190 | 95 | 190 | 210 | 80 | 115 | 95 | 260 | 140 | 210 |

Table 1: Container

An optimal solution to this problem can be obtained using a Mixed Integer Programming model with two types of decision variables. The first, $y_j \in \{0, 1\}$, indicates whether container $j$ is selected ($y_j = 1$) or not ($y_j = 0$). The second, $x_{ij} \in \{0, 1\}$, specifies whether liquid $i$ is moved to container $j$ ($x_{ij} = 1$) or not ($x_{ij} = 0$).

Complete the following tasks.

Question 5

Explain the Mixed Integer Programming model of this problem given in Week 2. What dual bound does its LP-relaxation provide on the optimal objective value. Considering only the $x$ variables, which variable would you branch on? Perform the branching by fixing the chosen variable to zero on one branch and and to one on the other branch. Even if you still get fractional solutions do not branch further. Present solutions and solution values of the two "subproblems" generated in the branching. State the best dual bound obtained after your incomplete Branch-and-Bound search.

Question 6

Repeat the branching component of Question 5, but now consider branching on the $y$ variables. Discuss any differences you observe.

Question 7

Design a greedy heuristic that can be used to provide a feasible solution to the problem. Clearly indicate the greedy principle you follow. Based on your answers to Questions 5 and 6, what can you say about the value of the optimal objective value to the Mixed Integer Program?

Question 8

Can any of the $x_{ij}$ variables be immediately removed from the Mixed Integer Programming model (since they will be at value zero in any optimal solution)? Explain your choice.

Question 9

Consider the optimal solution to the LP-relaxation. Identify a constraint in terms of the *$y$-variables only* that if added to the formulation would result in a better formulation of the problem. Explain your choice.

## Lagrangian Relaxation

Continuing with the container packing problem given in Table 1, you would now like to investigate the quality of the dual bounds obtained by applying different Lagrangian Relaxations of the problem.

Complete the following tasks.

Question 10

Identify *two* different Lagrangian Relaxations of the Mixed Integer Programming formulation of the problem considered in Question 5. Discuss their respective potential to provide better dual bounds than the LP-relaxation of the Mixed Integer Programming formulation.

Question 11

For each of the Lagrangian Relaxations given in Question 10, state the respective Lagrangian Dual problems.

Question 12

Explore and compare different step size rules when applying the subgradient method to optimize the Lagrangian Dual problems. What is the best dual bound provided by each of the Lagrangian Dual problems?

Question 13

Reflect on your answers to Question 12. Do the results support the intuition you gave in Question 10? Do the best solutions to your Lagrangian Dual problem provide feasible solutions to the original problem, and if not, can they be repaired easily?

## Dynamic Programming

In disaster scenarios, drones provide rapid delivery of supplies to areas that may be blocked or unsafe. The disaster zone can be represented as a 2D, $(n \times n)$ grid, where each cell has an associated traversal time and some cells may be obstacles. Drones can move horizontally (east), vertically (south), or diagonally (south east), with diagonal moves taking more time to reflect the longer distance covered. Given an $n \times n$ cost matrix, $C$, where each element reflects the horizontal and vertical travel times for the corresponding grid cell, and assuming that the drone is currently located in the top left corner of the grid cell (1,1), the aim is to find the shortest path to the bottom right corner of grid cell (n,n), i.e., bottom right. Diagonally traversing a grid cell $(i, j)$ is assumed to take approximately 40% more time than $c_{ij}$. Figure 1 illustrates the costs associated with grid cell $(i, j)$.

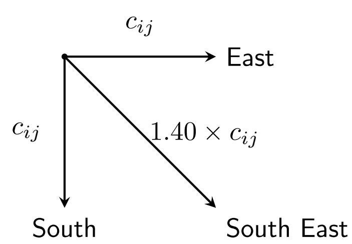


Figure 1: Grid cell cost example

You suspect that the optimal solution to this problem can be determined via dynamic programming.

Complete the following tasks.

Question 14

Identify the stages and states of your dynamic program.

Question 15

Give the dynamic programming recursion that will compute the shortest path for the drone through the grid.

Question 16

Use the suggested dynamic programming method to find the optimal objective value to a drone routing problem over a 7×7 with the following cost matrix (an x indicates the grid cell is not traversable). **Note that no Julia programming is necessary to answer this question.**

$$\begin{bmatrix} 6 & 9 & 14 & 8 & 13 & 13 & 11 \\ 5 & 16 & 12 & 12 & 13 & 6 & 16 \\ 8 & 15 & \mathrm{x} & 13 & 8 & 12 & 14 \\ 10 & 16 & 14 & 6 & 14 & 8 & 12 \\ 15 & 13 & 15 & 8 & 16 & 14 & 6 \\ 8 & 9 & 15 & 14 & \mathrm{x} & 15 & 14 \\ 6 & 12 & 13 & 8 & 12 & 10 & 7 \end{bmatrix}$$

Question 17

From your working to Question 16, identify the shortest path.