Ekonomická fakulta
Faculty of Economics

Jihočeská univerzita v Českých Budějovicích
University of South Bohemia in České Budějovice

Operations Research

Jana Klicnarová

České Budějovice | 2019

EUROPEAN UNION
European Structural and Investment Funds
Operational Programme Research, Development and Education

MINISTRY OF EDUCATION, YOUTH AND SPORTS

This textbook provides the basic introduction to Operations Research. It focuses on introduction to Linear Optimisation, Multi-criteria Decision Making, Data Envelopment Analysis and Project Analysis. The text is accompanied with prototype examples and Excel files with sample examples.

The textbook was created as part of the project Rozvoj JU - ESF, reg. number CZ.02.2.69/0.0/0.0/16_015/0002348, financed by the Operational Programme Research, Development and Education (European Structural and Investment Funds, 2014–2020).

Contents

Introduction

The aim of this text is to introduce some basic methods which can be useful in economic applications. More precisely, we are interested in following parts of Operations Research

- Linear optimisation – a construction of mathematical models, a graphical solution, a simplex method, SW for the solution of LO models, an economic interpretation of the solution.
- Multiple-criteria decision-making (MCDM) – construction of the weights, methods for MCDM.
- Data Envelopment Analysis (DEA) – identification of effective units, identification of problems for ineffective ones.
- Mathematical methods in Project Management – project network, Critical Path Method (CPM), PERT Method, Optimal cost and optimal duration of the project.

All methods of Operations Research follow the same scheme in fact. The steps of the procedure are presented in the picture .

The first point is the identification of the problem, the recognizing of Operations Research problem and exact definition of the problem (it includes for example solution requirements and so on).

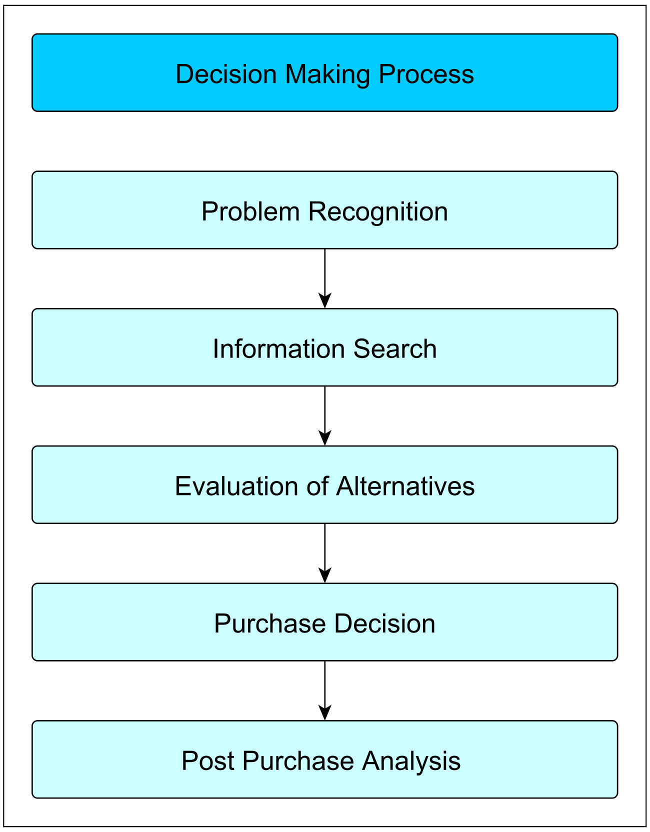
The second step is the issue of information research. We need to recognize important information which we need to involve in the solution procedure.

The third step is the definition of the problem in a mathematical way, formulation of the mathematical problem. (In this part we must identify the type of the problem.)

Then, we can start with the solution, we apply some of the suitable techniques of Operations Research furthermore we obtain a solution to our problem.

In the last part, we must explain a mathematical solution in an economical approach and we can also arrange the so-called post-optimization analysis. It means, for example, to examine how does the solution depend on initial values and so on.

Figure 1: Steps of a solution.



1 Linear Optimisation

Linear optimisation solves the problems of optimizing of some (linear) function (**objective function**) subject to some (linear) constraints. There are several types of problems which are included in this part of Operations Research. One of the typical types of these problems is so-called **production problem**.

In the production problem, we maximise the profit under some restrictions. Typically, the company wants to maximise its profit and the available supplies, resources and capacities are known. The question is what the optimal use of these items is?

To illustrate such problems better, let us formulate a prototype example. (It is **Standard Maximum Problem**, in which we are asked to find a solution maximising profit subject to given conditions.)

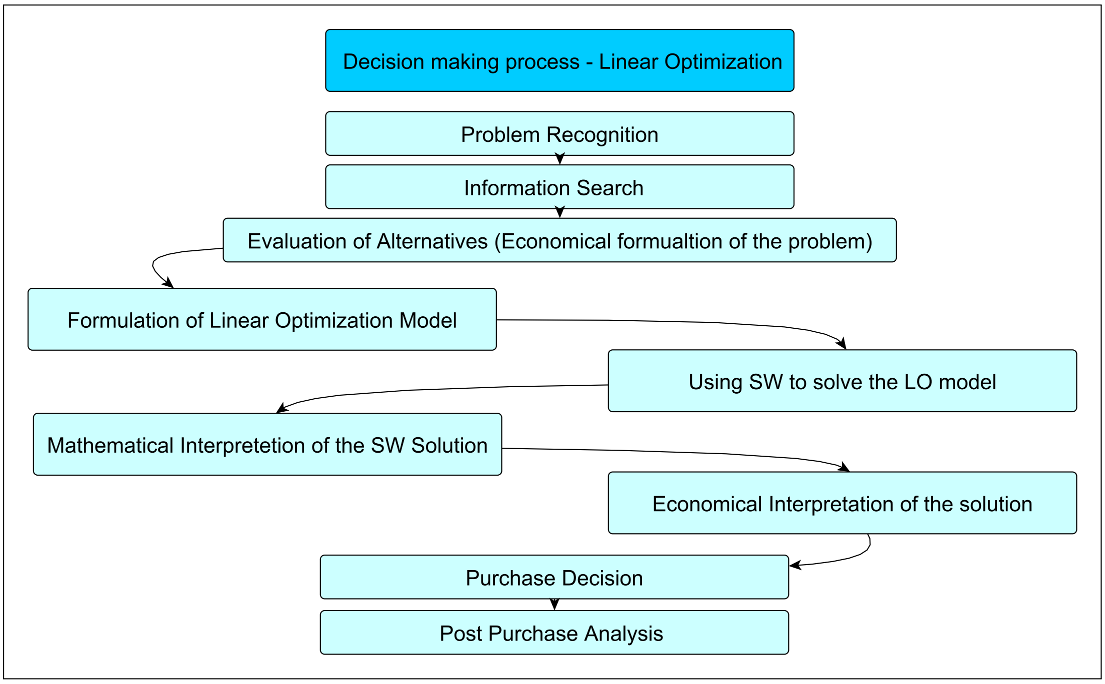
Example 1.1 (The Best Glass Co.) The Best Glass Co. produces high-quality glass windows. Now, they plan to use the remaining time of their production lines to start with the production of two new types of windows – let us call them Windows 1 and Window 2. All of these windows must go through three production lines, where the capacities of the lines are 60, 60, 85 hours. It is known that the unit of the first window type needs 2 hours at the first production line, 6 at the second one and 10 hours at the last production line. The unit of Windows 2 needs 10 hours at the first production line, 6 at the second one and 5 hours at the last production line.

The marketing division considers that the company could sell as many of either product as could be produced and it is supposed that the profit from each unit of Windows 1 would be 30 thousand dollars and from each unit of Windows 2 45.

It is not clear which mix of these two products would be most profitable.

Remark. The prototype example is a classic example of a problem for linear optimization – **The Standard Maximum Problem**. However, there exist many other types of problems which can be solved by linear optimisation methods, too. For example **The Standard Minimum Problem** – minimisation of cost under some conditions, **The Diet Problem** – optimal mixing of food to get daily required of nutrient with the lowest possible cost, **The Transportation Problem** – minimisation of transportation cost if we need to ship a given amount of commodity from producers to markets, **The Optimal Assignment Problem** – the optimal

Figure 1.1: Linear Optimisation Process.



assignment for example of workers to different jobs to get the best productivity; and so on. Some of these types of problems we will introduce later more precisely.

The solution to these problems can be based on the same method (simplex method), they differ only in the formulation of the issues.

Also, particular methods of solution for some problem exist (more effective) or methods for an issue where variables must be integers. However, it does not aim of this text to introduce these methods. Algorithms which allow us to solve integer linear optimisation are usually implemented in the software. Since the formulation of the problem and interpretation of the solution is the same, we would take care of it. On the other hand, in such case, the sensitivity analysis is not available (the sensitive analysis comes out from Simplex method).

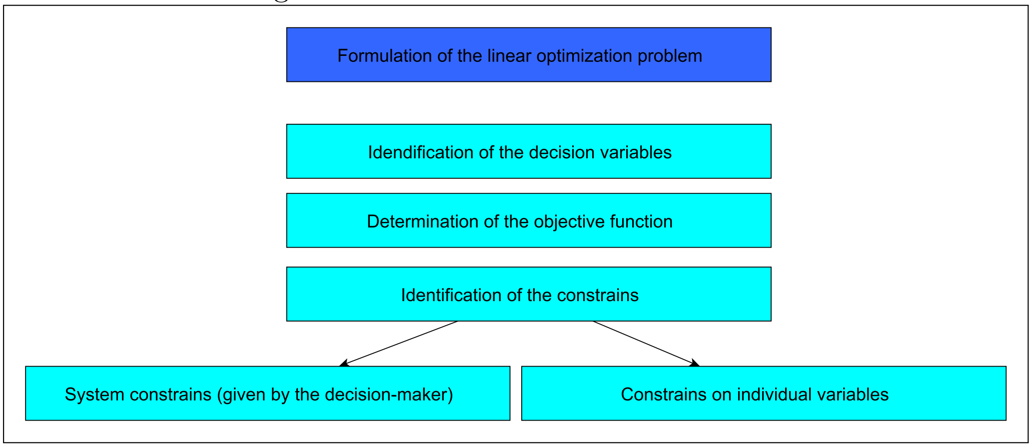
Now, let us introduce the process of the solution of LO problems. First, let us go back to the table given in the Introduction and show this table for Linear Optimisation process, see table 1.

In this text, we suppose that we know the economic problem and our aim is to continue with the solution. Prototype example is the formulation of the economic issue – the problem was identified, and also all the necessary information was given.

Hence, we should continue with the third step — formulation of the mathematical model – linear optimisation (programming) model. It is a mathematical representation of the economic (LP) model.

How to do it? The steps of this are described in the following table 1.

Figure 1.2: Construction of LO models.



As we can see from the picture, first we must identify the decision variables of the model. The decision variables are in contrast to problem data. The data are values that are either given or can be directly calculated from the given problem. Decision variables are some unknown values which the decision maker wants to know. In this example, the variables are the numbers of packages for each type of windows. Because in fact, the question in the example is how many packages of Windows 1 and how many packages of Windows 2 should the company produce to maximise its profit? So, we put x_1, resp. x_2 for the number of produced units of Windows 1, resp. 2. Then we search for the values of x_1 and x_2.

If we stated the decision variables, then we can start with the formulation of the linear programming (mathematical) model. Every linear optimisation model has three parts, see the picture 1.

The first part is the part of the objective function. There is the function which we want to, and there is also set what type of optimisation we want to do (typically to maximise or to minimise).

The second part is part of the main constraints. As was written above objective and all constraints functions must be linear; it means it must be in the form of the sum of constants multiplied by variables. The main constraints are all constraints which are explicitly given in the formulation of the problem (in our case, the restrictions on the capacity of the lines).

The last part is the part of constraints on variables – typically non-negativity of the variables, in special cases also integer type of variable, or binary type.

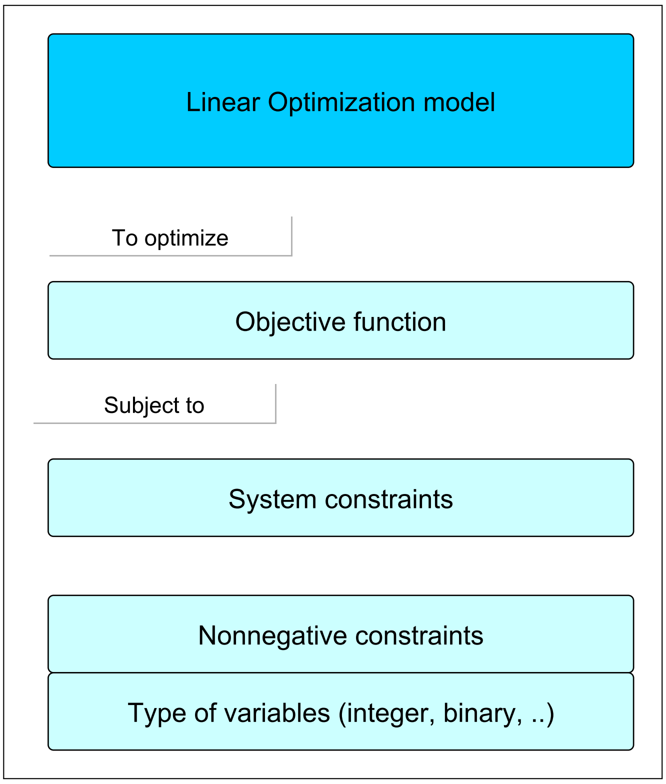
Let us formulate a prototype example to construct its linear optimisation model.

Hence the economic problem is given, and our aim is to construct the linear optimisation model. The first step of this construction is the identification of the decision variables.

Identification of the Decision Variables

In the prototype example, it can seem that there are two possibilities of decision variables – the number of hours at each production line or number of produced

Figure 1.3: Linear Optimisation Model.



units of each window type.

How to recognise which of these decision variables are the correct ones?

Let us suppose that we decide to set the number of hours to spend by the products at the production lines as the decision variables. Then, after the solution of the problem, the result would be the answer to the question of how many hours should spend both types of windows at each product line. Is it what the decision maker would like to know? Does it give him the answer to his question on how to achieve the best profit? No. He does not know how to reach it. So, it was not the correct choice.

Therefore, the decision variables will be:

x_1 the quantity of Windows 1 to produce;

x_2 the quantity of Windows 2 to produce.

Then, if we solve the problem, we answer the question of how many units of Windows 1 and how many units of Windows 2 the company should produce to optimise its profit. And this is what the manager would like to know.

If we have identified the decision variables, we can specify the objective function.

Determination of the objective function

In the prototype example, the company aims to maximise its profit. The profit depends on the number of produced units. It is known what the profit from each unit is. Hence, the whole profit can be written as

$$30x_1 + 45x_2,$$

where x_1, x_2 are (as was written above) decision variables which give us the number of units of Windows 1, resp. 2 to produce.

Hence, we can write the objective function as

$$\max\ 30x_1 + 45x_2.$$

Let us remark, that it is also essential to write the type of objective function (to write if we want to maximise or minimise it).

Now, we can continue with the construction of constraints.

Identification of Constraints The Best Windows Comp. wants to maximise its profit, but they could not produce as many units as they want, they are limited by some restrictions. In our prototype example, they are limited by the capacities of producing lines. We must all of these limits express as the constraints.

So, let us look more carefully at the capacity of the producing line 1. We know, that there are 60 hours available and that each unit of Windows 1 spends here 2 hours. Hence, if we produce x_1 units of Windows 1, we use $2x_1$ of the capacity fo the first production line. Each unit of Windows 2 needs 10 at this line, hence x_2 units needs $10x_2$ hours. Therefore, we can write:

$$2x_1 + 10x_2 \leq 60.$$

Similarly, we get for the second line:

$$6x_1 + 6x_1 \leq 60$$

and for the third line:

$$10x_1 + 5x_2 \leq 85.$$

Now, it seems that we are finished with constraints. It appears that there are no other ones. However, it is not true. We must add constraints which are not explicitly written in the problem but which must be fulfilled too. Typically, non-negativity of variables. In the prototype example, the decision variables are the number of produced units. Hence, it is clear, that we can not produce a negative number of units, so we must add constraints on non-negativity of variables:

$$x_1, x_2 \geq 0.$$

Now, we have the whole linear optimization model of our prototype example:

$$\begin{aligned} \max 30x_1 + 45x_2 & & \\ \text{subject to } 2x_1 + 10x_2 & \leq & 60, \\ 6x_1 + 6x_2 & \leq & 60, \\ 10x_1 + 5x_2 & \leq & 85, \\ x_1,\ x_2 & \geq & 0. \end{aligned}$$

The next step is the solution to this problem.

However, let us have some remarks on the construction of constraints in the linear optimisation model. Our prototype example is straightforward, there are no complicated constraints. Sometimes, we can have in the economic formulation some conditions which seem to be more complicated. Typically, we meet conditions as follows:

- it is needed to produce the same amount of Windows 1 as of Windows 2,
- more Windows 1 than Windows 2 needs to be produced,
- it has to be produced at least five more Windows 1 than it is produced Windows 2
- it has to be produced at least twice as many Windows 1 as it can to be produced Windows 2
- at least 30 per cent of the production is Windows 1.

These types of conditions are typically the source of problems for students. So, let us look more carefully on the construction of these constraints.

Let us start with the first type. The condition: "It is needed to produce the same amount of Windows 1 as Windows 2." It is a straightforward condition, let us recall that x_1, resp. x_2 is an amount of produced Windows 1, resp. Windows 2. So, if the amounts should be the same, we can write it down as

$$x_1 = x_2.$$

The second type of conditions is very similar. We have to produce more Windows 1 than Windows 2, hence we can write

$$x_1 \geq x_2.$$

The third type of conditions is quite more complicated. We have to produce at least five more Windows 1 than we produce Windows 2. We know that we produce x_2 of Windows 2, hence we should add five more, it is $x_2 + 5$, and we need to produce at least such amount of Windows 1 (amount of Windows 1 is denoted by x_1. Therefore, we obtain:

$$x_1 \geq x_2 + 5.$$

If we are not sure if we are right, we check our solution in the following way. Let us suppose that we produce for example 4 Windows 2. In such a case if we want to fulfil the condition, we should produce at least $4 + 5$ Windows 1, so x_1 should be equal to or bigger than 9. Now, let us use the example number (4) in our constraint:

$$x_1 \geq x_2 + 5 = 4 + 5 = 9,$$

hence we have

$$x_1 \geq 9,$$

what is right. So our constraint is correct.

The fourth type of conditions is similar. We need to produce at least twice as many Windows 1 as Windows 2. It means if we produce x_2 of Windows 2 then twice as many Windows 1 as Windows 2 is equal to $2x_2$; hence the constraint can be written as

$$x_1 \geq 2x_2.$$

If we are not sure if we are right, we can again choose some value for x_2 and check our solution as was shown in the previous case.

The other type of condition which we want to present here is the type of condition where some percentages play a role. Our prototype condition says that at least 30 per cent of the production is Windows 1. Therefore, we can immediately write that $x_1 \geq stg$, where stg is 30 per cent of the production. (Suppose that percents are understood as a per cent of pieces.) What is the whole production? It is $x_1 + x_2$, than 30 percent of $(x_1 + x_2)$ is $0.3(x_1 + x_2)$. Hence, the constraint is in the form:

$$x_1 \geq 0.3(x_1 + x_2),$$

or

$$0.7x_1 - 0.3x_2 \geq 0.$$

Let us remark that many software for linear optimisation need to have constraints in such forms that all variables are on the left-hand side of the inequality, so then it is necessary to use the second expression.

1.1 Graphical Solution

In the case of the linear optimisation problem with only two variables, we can solve such problems graphically. To do it, we first need to display **the set of feasible solutions** and then if it is possible to add the objective function and identify the optimum solution.

First, let us explain how to display the set of feasible solutions. The solution is feasible if it satisfies all constraints.

To illustrate how to develop the solution, we show the graphical solution of Prototype example 1.1. Let us recall the mathematical model of this example.

$$\begin{array}{rcl} \max 30x_1 + 45x_2 & & \\ \text{subject to } 2x_1 + 10x_2 & \leq & 60, \\ 6x_1 + 6x_2 & \leq & 60, \\ 10x_1 + 5x_2 & \leq & 85, \\ x_1,\ x_2 & \geq & 0. \end{array}$$

Now, we need to display the set of points where all constraints are satisfied. Let us start with the displaying of the first constraint:

$$2x_1 + 10x_2 \leq 60.$$

This inequality can be displayed as a half-space which is bounded by the line given by the formula

$$2x_1 + 10x_2 = 60.$$

Therefore, first, we need to display the line with this formula. Each line is given by two points, hence, let us find two points of this line. First, let us suppose that $x_1 = 0$. If we fit it into the equation, we get

$$2 \cdot 0 + 10x_2 = 10x_2 = 60,$$

so we obtain $x_2 = 6$; therefore, the first point of the line is $[0, 6]$. Similarly, we get the second point of the line $[10, 4]$. So, we have the boundary of the half-space.

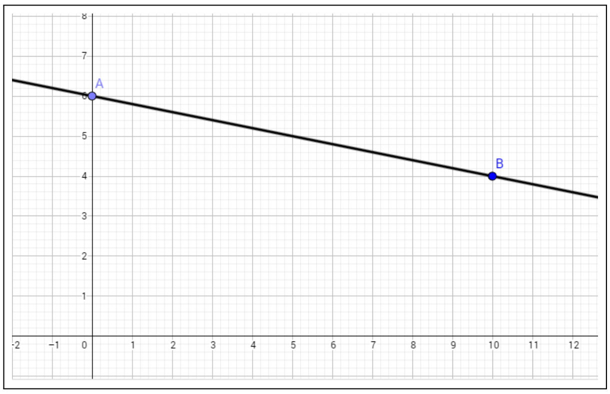
Now, we need to decide which part of the space is such that for all points at this set satisfy the inequality

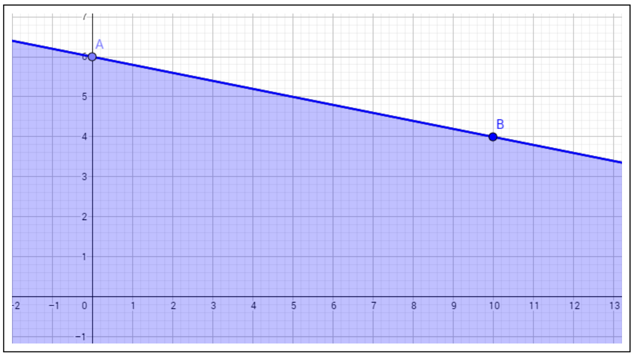
$$2x_1 + 10x_2 \leq 60.$$

To do it, let us choose some point which does not lie on the boundary, for example, the point $[0, 0]$ and let us check if the point satisfies the inequality:

$$2 \cdot 0 + 10 \cdot 0 = 0 \leq 60.$$

The inequality is fulfilled, so the searched half-space is such that which includes the point $[0, 0]$, see the following picture.





Similarly, we can construct the other conditions and the intersection of all these conditions gives us the set of feasible solutions.

This case is the typical one – there exists a non-empty bounded set of feasible solutions. In such a case there exists at least one optimum solution.

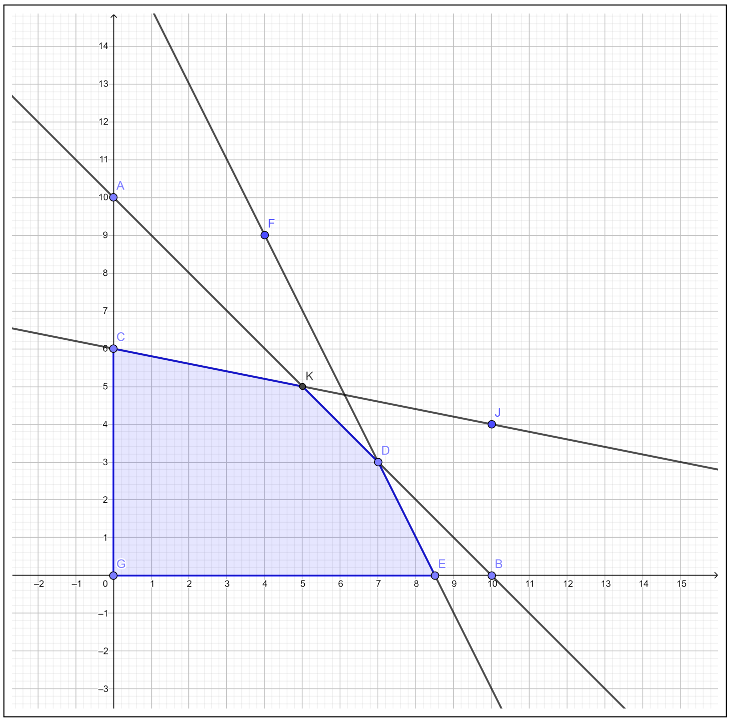
Now, there two possible ways how to identify the optimal solution. First, we can apply the **Fundamental Theorem of Linear Optimization**. Which states that if the set of feasible solutions is bounded and non-empty, then at least one of the basic solution (it has not been explained yet what it is) is the optimal one, in a weak formulation (applied for graphical solution), it says that the optimum solution of LO problems occurs at the region's corners. Therefore, we can identify all corners of the set of feasible solutions, to apply the objective function and the best value of the objective function identifies the optimum solution of the problem.

In Prototype example, we have following corners:

$$[0; 0], [8.5; 0], [7; 3], [0; 6].$$

Let us apply the objective function $30x_1 + 45x_2$:

Figure 1.4: The set of feasible solutions for Prototype example



corner	value of the objective function
$[0; 0]$	0
$[8.5; 0]$	250
$[7; 3]$	345
$[5; 5]$	375
$[0; 6]$	270

From the table, we can see, that the best solution is to produce 5 units of Windows 1 and 5 units of Windows 2, then the profit is expected to be 345.

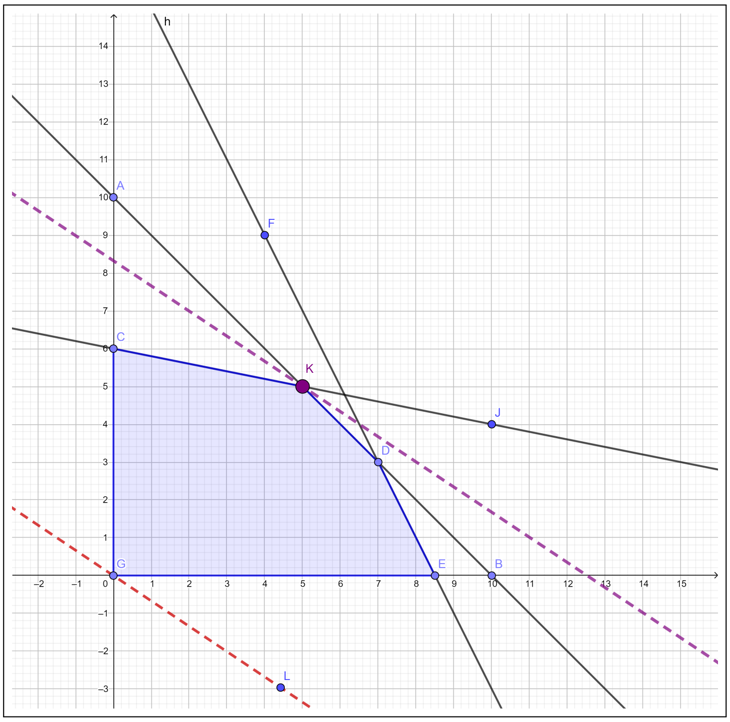
The other way, how to identify the optimal solution is to continue in the graphical way, to add the objective function into the graph and to find the optimal solution in the figure.

The objective function is:

$$p = 30x_1 + 45x_2,$$

where the p depends on the values of variables x_1 and x_2. However, for a fixed p, we get a line and for different values of p, we have parallel lines. So, we add to the graph the line $p = 30x_1 + 45x_2$, for one fixed p (typically for $p = 0$, but it is not important) and then we move (parallel) the line (in the case of maximisation)

Figure 1.5: Optimal solution of Prototype example



up and up where it still intersect the set of feasible solution – we stop at some the corner (or border) of the set of feasible solution – what is the optimal solution(s) of our problem, see the picture 1.1.

However, the intersection of half-spaces of conditions may be empty, in such a case there is no feasible solution and do not exist an optimum one. Or; typically when the setting of the example is not correct, the set of feasible solutions is unbounded and the objective can grow up to infinity, in such a case the optimum solution does not exist; the objective function is unbounded.

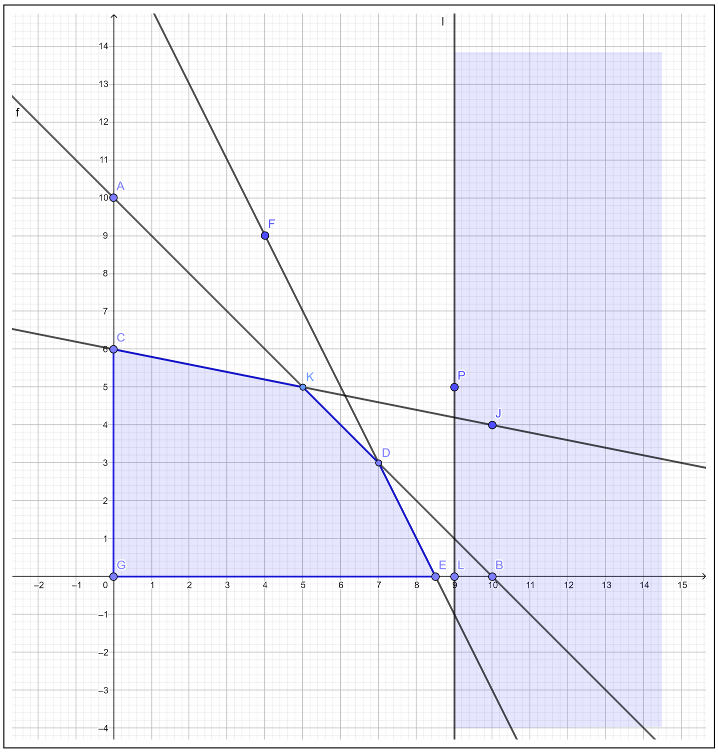
Let us introduce these cases more precisely.

Empty Set of Feasible Solutions To illustrate the case of the empty set of feasible solutions let us reconsider Prototype example 1.1 with only one extra condition – let us suppose that the company required to produce at least 9 units of Windows 1. It means one additional condition:

$$x_1 \leq 9.$$

Let us include this condition into the graph, and we can see that the intersection of the requirements is empty. It means no possible combination of outputs x_1, x_2 which satisfies all given conditions exists. So, no optimal solution exists, for more detail see 1.1.

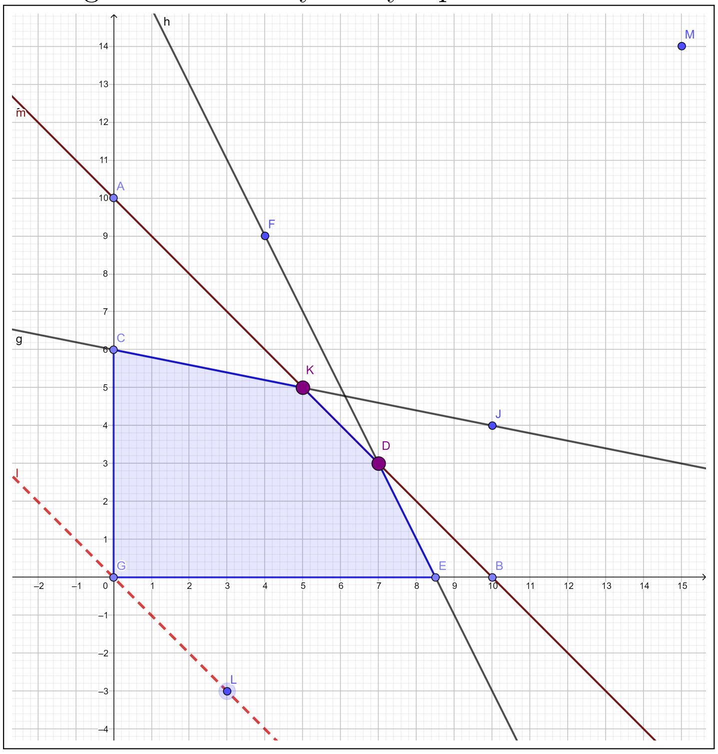
Figure 1.6: No feasible solution.



Unbounded Set of Feasible Solutions It also could happen that the set of feasible solution is unbounded and then no optimal solution exists, because it is unbounded. However, it typically means that we forgot some essential condition in the formulation of the problem because it is impossible (neither theoretically) to reach unlimited profit.

Infinity Many Optimum Solutions The last type of solution of Linear Optimization Problem (we had one optimal solution, no optimal solution (from two different reasons)) is the case of infinity many optimal solutions. This case takes place in special situations, when the profit line (generally half-space) is parallel with some constraint. To illustrate this case, let us reconsider our Prototype example 1.1 with a different supposed profit from each unit of Windows 2. In such a case we have a similar optimisation problem, where the set of feasible conditions is the same as in the Prototype example (all conditions are the same), only the objective function differs, more precisely, we want to solve the problem:

Figure 1.7: Infinity Many Optimal Solutions



$$
\begin{aligned}
\max 30x_1 + 30x_2 & \\
\text{subject to } 2x_1 + 10x_2 &\leq 60, \\
6x_1 + 6x_2 &\leq 60, \\
10x_1 + 5x_2 &\leq 85, \\
x_1,\ x_2 &\geq 0.
\end{aligned}
$$

If we include the profit function into the graph, we can see that it is parallel with one of the constraints. Hence, the possible optimal solutions will be all point on the abscissa DK or all convex combination of D and K, more precisely all solutions in the form

$$[5 + 2k; 5 - 2k],\ k \in [0, 1].$$

If we suppose that we solve the problem about units of Windows, we suppose the solution to be integers; then we have three possible solutions:

$$[5, 5],\ [6, 4],\ [7, 3].$$

For all these possible (and feasible) solutions we obtain the same profit of 300.

Figure 1.8: Data in Excel

	windows 1	windows 2				capacity
line 1	2	10				60
line 2	6	6				60
line 3	10	5				85
profit	30	45				

Figure 1.9: Variables Included into the Model

	windows 1	windows 2				capacity
line 1	2	10				60
line 2	6	6				60
line 3	10	5				85
profit	30	45				
variables	0	0				

1.2 Linear Optimisation with Solver

Methods for linear optimisation are implemented in many software. In this text, we will focus on Solver, add-in in Excel.

First, we need to prepare the data in the Excel sheet. In the first step we rewrite the data of our example in the following way, see the picture 1.8.

Now, all coefficients from the setting of the example are already set in the Excel, and we need to add variables. In our prototype example, we have two variables; hence we need to cells for them – we choose two cells and set them to be variables. We put the starting points into them – 0s, see the picture 1.9.

In the next step, we need to prepare all the functions which we use in the model – objective function and all left-hand sides of the constraints. We remember, that all these functions are linear; hence we can write them as a scalar product of the vector of their coefficients and the vector of variables. We do it in the following way, see the picture 1.10.

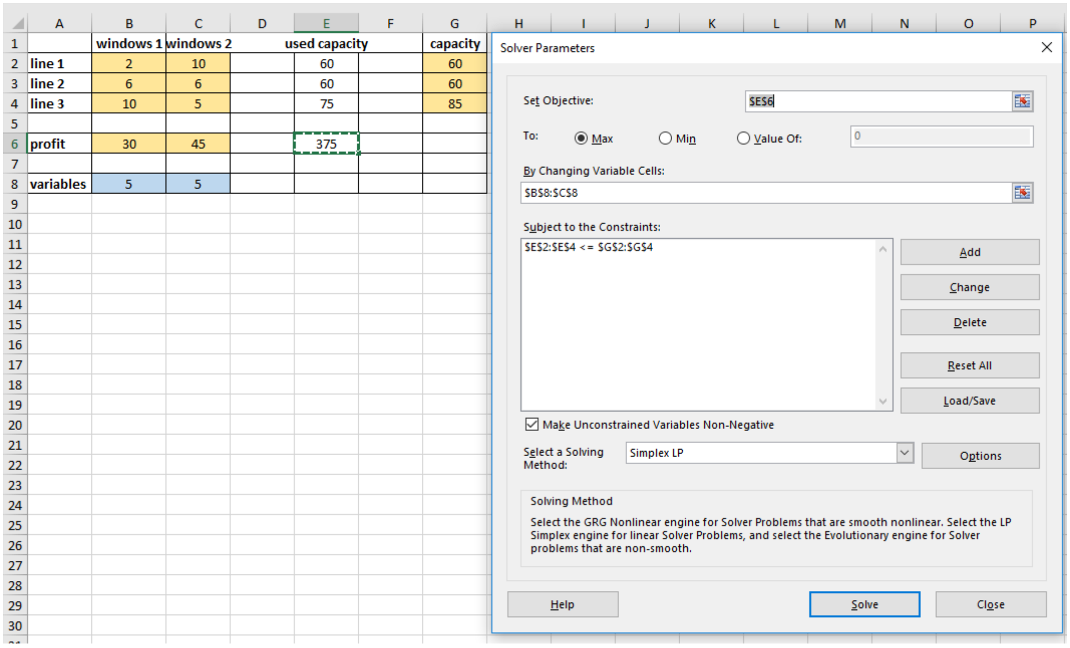
If we have everything prepared, we can open the Solver and set the problem in to it.

To the cell for the objective function, we set the cell, where we prepared the formula for the objective function. We set the type of objective function, denote

Figure 1.10: The Model for the Solver

	windows 1	windows 2	used capacity	capacity
line 1	2	10	`=SUMPRODUCT(B2:C2;$B$8:$C$8)	
line 2	6	6	`=SUMPRODUCTÍ(B3:C3;$B$8:$C$8)	
line 3	10	5	`=SUMPRODUCT(B4:C4;$B$8:$C$8)	
profit	30	45	`=SUMPRODUCT(B6:C6;$B$8:$C$8)	
variables	5	5		

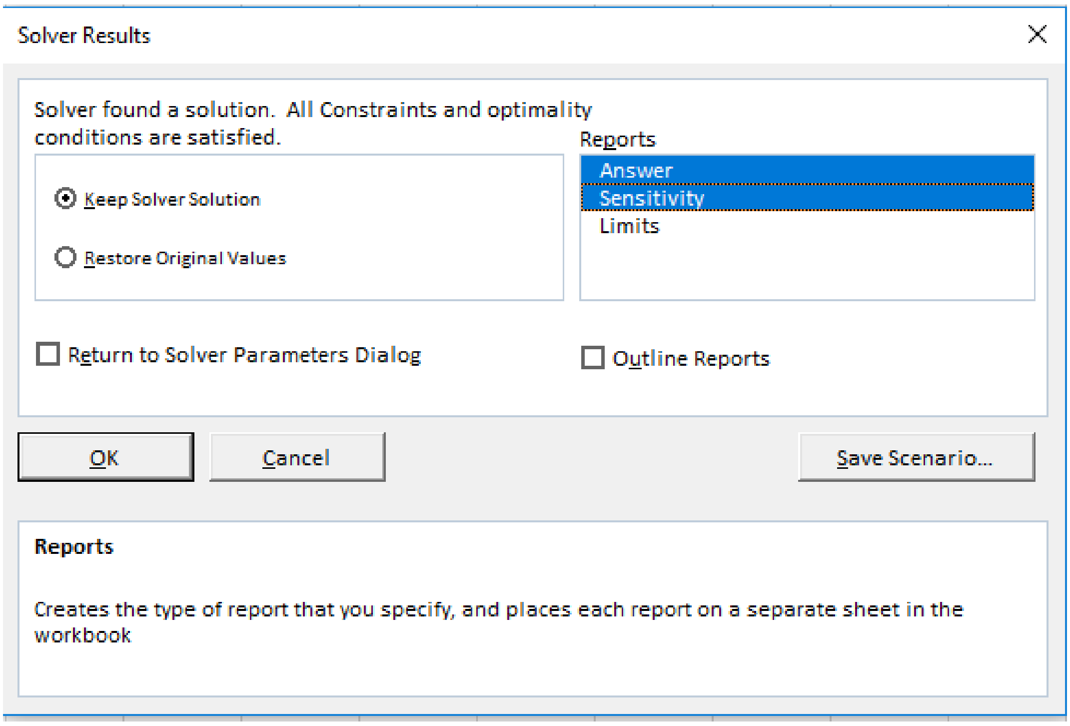
the variables (blue cells) and set all constraints.



Two possible ends of the Solver are:

- Solver found an optimal solution,
- Solver did not find an optimal solution.

In the first case, we tick to save the result and sensitive analysis for post-optimisation analysis, see the following figure.



In such a case we will get two more sheets – Sensitivity Report and Answer Report, where we find more detailed solution of our problem, for more details on these report see the following section.

In the other case, when the Solver could not find solution, the reports are not available. Usually, there three main reasons for the second case – no feasible solution, unbounded solution; or nonlinear problem.

1.2.1 Postoptimality Analysis

The post-optimality analysis also referred to as Sensitive analysis, carries the linear optimisation analysis beyond the determination of the optimal solution. After the determination of the optimal solution, we know the optimum values of variables to achieve the best value of the objective function. However, the optimisation model can answer many other questions. Questions as

- what does happen if the DM decides to produce a non-optimum product,
- in which price is advantageous to buy more supplies,
- in which price is advantageous to sell a part of supplies,
- if the profit of some product will be changed, will it change the result,
- and so on,

can be often answered without new optimisation.

More detailed results and answers to these question are in the Sensitivity Report sheet in Excel (generally, are given by values of so-called dual variables and stability ranges which are provided by the linear optimisation model, it is possible to get these answers from the final simplex tableau).

Microsoft Excel 16.0 Answer Report
Worksheet: [lp.xlsx]Example
Report Created: 3/6/2019 10:05:59 AM
Result: Solver found a solution. All Constraints and optimality conditions are satisfied.
Solver Engine
Engine: Simplex LP
Solution Time: 0.031 Seconds.
Iterations: 2 Subproblems: 0
Solver Options
Max Time Unlimited, Iterations Unlimited, Precision 0.000001, Use Automatic Scaling
Max Subproblems Unlimited, Max Integer Sols Unlimited, Integer Tolerance 1%, Assume NonNegative

Objective Cell (Max)

Cell	Name	Original Value	Final Value
$E$6	profit used capacity	375	375

Variable Cells

Cell	Name	Original Value	Final Value	Integer
$B$8	variables windows 1	5	5	Contin
$C$8	variables windows 2	5	5	Contin

Constraints

Cell	Name	Cell Value	Formula	Status	Slack
$E$2	line 1 used capacity	60	$E$2<=$G$2	Binding	0
$E$3	line 2 used capacity	60	$E$3<=$G$3	Binding	0
$E$4	line 3 used capacity	75	$E$4<=$G$4	Not Binding	10

Microsoft Excel 16.0 Sensitivity Report
Worksheet: [lp.xlsx]Example
Report Created: 3/6/2019 10:05:59 AM

Variable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$8	variables windows 1	5	0	30	15	21
$C$8	variables windows 2	5	0	45	105	15

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$E$2	line 1 used capacity	60	1.875	60	40	16
$E$3	line 2 used capacity	60	4.375	60	5.333333333	24
$E$4	line 3 used capacity	75	0	85	1E+30	10

A Change in an Objective Function Coefficient

It can be useful for the DM to know how much the contribution of a given variable to the objective function can be changed without an impact on the optimal solution. Such changes can occur because of new pricing policies, changed costs or some other factors.

If we keep the Sensitive analysis sheet in Excel, the first tableau provides us with

information about a change in the objective function coefficient. Each line is dedicated to the one variable in the objective function.
There to cases to consider (unfortunately, we have only one of them in Prototype example) – changes for the variable that is not currently included in the optimal solution and alterations for the variable that is now contained in the optimal solution.
The question was, how can we change the coefficient by the variables without an impact on the optimal solution. Now, let us focus on the variable not included in the optimal solution. In words of classical production problem, it is not optimal to produce such a product. Therefore, if we change the coefficient in one way (in case of profit type objective function down in the case of cost type objective function up) it can not change the optimal solution (the variable was not effective, and we only do it worse. However, the question is how can we change the coefficient to get the variable into the solution – it is given by so-called **range of insignificance**. As was already mentioned this range is unbounded from one side (the side depends on the type of the objective function). The second side gives us the value of the coefficient for which the variable could be already included in the optimal solution.
For the variables which are included in the optimal solution, we are interested in **range of optimality** that gives us for the variable coefficient a range where we can change it without an impact on the optimal solution. If we change the variable coefficient in this range (and all other conditions stay same), the optimal solution will be the same – we get the same values of all variables (therefore, the objective will be different).
In both cases we can also speak about so-called **reduced cost**, it is equal to zero if the variable is included in the optimal solution (except the case when we have a strict condition, for example, $x_3 \leq 4$ and we produce just 4). If it is equal to zero, then we can add this variable into an optimal solution without any impact on the value of the optimal value of the objective function. On the other hand, in case of a strictly positive value of reduced cost, we know that each added unit of the variable into the optimal solution will change the value of the objective function by the value of reduced cost.

Example 1.2 (Solution to The Best Glass Co. problem) In Prototype example we have the following values.
We can see that both variables are included in the optimum solution, so their reduced costs are equal to zero. The range of optimality for the variable x_1 is $[9, 45]$. It means that if the profit from product 1 will be changing between 9 and 45 it will be still optimal to produce 5 units of Windows 1 and 5 units of Windows 2. However, the objective function could change (it depends on the profits from units of Windows).
Similarly for the Window 2.

Let us suppose that we have in the example also variable x_3 – the number of units of Window 3 which is equal to zero in the optimal solution. Then the reduced cost could differ from zero. If the reduced cost is equal to zero, it means that there are more optimal solutions (we can include the x_3 into the optimal solution without any change of the objective function). If the reduced cost is some $C > 0$, it means that with each produced unit of Windows 3 the objective function goes down by C. In other words, if we want to provide such product without lost, we need to increase the profit at least by C (the range of insignificance also gives it).

A Change in the Right-hand Sides of a Constraint

To study the impact of the change in the RHS (right-hand side) of constraint on the optimal solution we need to understand to **shadow prices** and **range of feasibility**. Both of them are given in Sensitive analysis in Excel (and can be determinate from the final simplex tableau).
A shadow price is a marginal value; it indicates the impact that the one-unit change in the amount of a constraint would have on the value of the objective function. More precisely, a shadow price reveals the amount by which the value of the objective function would change if the level of the constraint was changed by one unit (if it increases or goes down depends on the type of the objective function and kind of change in the restrictions).
However, the level of the change must be in the range of feasibility. If it is out the range, we need to do the optimisation once again.

Example 1.3 (Solution to The Best Glass Co. Problem) In Prototype example we have in optimal solution first two lines fully use, there remains only the capacity of the line number 3. So, the shadow price for the line 3 is equal to 0. (If we change the capacity of the line 3 in the range $[75, \infty]$ it would not affect the optimal solution.
For lines 1 and 2 we are in a different situation, under optimal solution they are fully used. So if we change the capacity of one of them, it would change the optimal solution – we would be able to produce less or more (it depends on the type of change) of products, hence we will have higher or lower profit (also depends on the kind of change).
Let us focus on the variable x_1. The shadow price is 1.875 and the range of feasibility is $[44, 100]$. It means if we change the capacity of line 1 in the range of feasibility, the objective function will change by 1.875 per unit of the change of the capacity.
For example, if we know that we can run the line 1 in extra time in the cost of 1 per hour, then we see that we should do it because our profit will be 1.875 per hour. On the other hand, if we know that each saved hour on line 1 costs 2, then we di it because we lost only 1.875 per hour. (Everything still just in the range of feasibility.)

1.2.2 Classes of LO Problems

As was already mentioned above, there exist several classes of linear optimisation problems. The classical ones and the easiest to solve are **the standard maximum problems** and **the common minimum problems**. In these problems, all variables are constrained to be nonnegative, all constraints are inequalities, and there are no other special conditions (as variables must be integers or so on). For the solution of such problems, it is enough to apply the Simplex method. It is possible (in the case of a small number of variables) to solve these problems without using a computer; however, we will not do it here; for more details see other literature. In this text, we introduce only the solution by using some software, so it would not be too important for us if the problem is the classical one or not. For us, it will be essential to be able to formulate the mathematical model for the given economic problem. So, in this part, we will focus only on the differences among the different types of issues from the formulation of the mathematical model.

Diet Problem

Typically, in this type of problem, we consider several different types of the food each of them has a different proportion of given nutrients which are important for good health. We are interested in the optimal combination of the food to supply the required nutrients at a minimum cost.
Such example typically leads to the standard minimum problem.

Example 1.4 (Diet Problem) What is the optimal combination of the yoghurt and cereals for the health breakfast, if we need to get at $1200\,kJ$ of energy, $18/,g$ of proteins and $2\,g$ of calcium and not more than $1.2\,g$ of fat and $90\,g$ of carbohydrates at minimum cost? It is known that the $100\,g$ of the food contains the following amount of nutrients, the table also contains the price per $100\,g$ of food.

	En.(kJ)	Prot.(g)	Carb.(g)	Fat(g)	Ca(mg)	Price(CZK)
Yogurt	200	5	6	0,1	160	4
Cereals	1500	9	80	1,5	-	10

Example 1.5 (Solution to The Cutting Stock Problem) First, we need to identify the variables. It seems that there are two possibilities – x_1, x_2 – to be the amount of yoghurt and cereals or – $x_1, \ldots, x_5$ – to give the amount of energy, proteins and so on. Is it OK? To check if both possibilities are correct, let us consider that we already have a solution. In the first case, we say to the decision maker (DM) – use such amount of yoghurt and such amount of cereals – it is OK. He knows, how to mix the breakfast. In the second case, we say to the DM – use such amount of energy, such amount of proteins and so on. And he probably ask us — how??? So, we did not answer his question. So, the correct choice was only the first one. We have two variables:
x_1 – the amount of yogurt (in $100\,g$),
x_2 – the amount of cereals (in $100\,g$).
Then, we can formulate the objective function. If x_1, resp. x_2 gives us the amount of used yogurt, resp. cereals, it is clear that the price for such breakfast

is:

$$4x_1 + 10x_2.$$

The constrains are formulated in the similar way, therefore we get the following model:

$$c = 4x_1 + 10x_2 \;\rightarrow\; \text{min}.$$

Subject to conditions:

(a)	energy (kJ)	$200x_1 + 1500x_2 \geq 1200$
(b)	proteins (g)	$5x_1 + 9x_2 \geq 18$
(c)	carbohydrates (g)	$6x_1 + 80x_2 \leq 90$
(d)	fat (g)	$0,1x_1 + 1,5x_2 = 1,2$
(e)	calcium (mg)	$160x_1 \geq 2000$
		$x_1, x_2 \geq 0.$

The Cutting Stock Problem In this type of problems, we solve the problems of cutting standard-sized pieces of stock material, such as sheet metal, rolls of paper, fabric cloth and so on, into fragments of specified sizes while minimising material wasted or the number of cut pieces.
Therefore, the decision-maker wants to know how many pieces he has to divide in each way. Therefore, this problem is specific due to the nonstandard variables. In such issues, we first need to state all alternatives how the original pieces can be divided into required ones, and the number of these alternatives gives us the number of variables. Each variable identifies the number of original pieces cut in the given way. Let us give an example.

Example 1.6 (The Cutting Stock Problem) Let us suppose a company which needs 160 of 50cm-long bars, 200 of 70cm-long ones and 250 of 90cm-long bars. The machine produces all bars in the length of 2 meters. How should the company cut the bars if it wants to minimise the waste?

Example 1.7 (Solution to The Cutting Stock Problem) First, let us consider all possible alternatives:

length	alt. 1	alt. 2	alt. 3	alt. 4	alt. 5	alt. 6
50 cm	4	2	2	1	0	0
70 cm	0	0	1	2	1	0
90 cm	0	1	0	0	1	2
waste (cm)	0	10	30	10	40	20

Now, we can set x_1 to be the number of bars cut under the alternative 1 and so on.
If we know the variables, then it is easy to formulate the mathematical model.

$$10x_2 + 30x_3 + 10x_4 + 40x_5 + 20x_6 \;\rightarrow\; \text{min.}$$

Subject to conditions:

$$\begin{array}{lll} \text{(a)} & \text{50 cm:} & 4x_1 + 2x_2 + 2x_3 + x_4 \geq 160 \\ \text{(b)} & \text{70 cm:} & x_3 + 2x_4 + x_5 \geq 200 \\ \text{(c)} & \text{70 cm:} & x_2 + x_5 + 2x_6 \geq 250 \\ & & x_1, x_2, \ldots, x_6 \geq 0, \text{ integers.} \end{array}$$

The Transportation Problem In these problems, we consider several producers, production plants or ports which supply a particular commodity and several markets or companies where the merchandise must be shipped. The available commodity amount at each source is known, the required commodity

amount by each purchaser is known; and the cost for transportation of the commodity unit between each supplier and each purchaser is known, too. The question is how the commodity shipped at a minimum cost.
The main problem is again the correct setting of the variables. We are interested in the question of how much commodity should be shipped between each supplier and each purchaser. So, variables are x_{ij}'s which give us the amount of product shipped from the i-th supplier to the j-th purchaser. If we know variables, the formulation of the mathematical problem is natural.

Problems with Binary Variables In these problems, we typically need to choose something under some conditions. Usually, we need to select employees, projects or so on. Let demonstrate such a problem on the following example.

Example 1.8 (Projects) The company needs to decide which the possible projects will invest. The monthly costs may not exceed 200 thousand dollars. Further information – monthly costs per each project and its supposed annual yields are given in the following table. The company decided to choose at most three of the following projects.

Project	P_1	P_2	P_3	P_4	P_5
monthly cost (thousand dollars)	28	42	66	33	78
annual yield (thousand dollars)	56	82	104	63	120

Such projects should be chosen if the company wants to maximise annual yield and it is known that it is impossible to realise projects P_2 and P_4 at the same time, on the other hand, the company requires to realise at least one of the projects $P1$, $P3$, $P4$.

Example 1.9 (Solution to Projects) In such a case we have binary variables, variables $p_1, p_2, \ldots, p_5$ which take values 0 or 1 indicate if the project will be realized or not. (1 for realized, 0for not realized.)

Then the objective function is:

$$\max 56p_1 + 82p_2 + 104p_3 + 63p_4 + 120p_5.$$

The constraints are as follows.

$$
\begin{aligned}
p_1 + p_2 + p_3 + p_4 + p_5 &\leq 3 \\
28p_1 + 42p_2 + 66p_3 + 33p_4 + 78p_5 &\leq 200 \\
p_2 + p_4 &\leq 1 \\
p_1 + p_3 + p_4 &\geq 1, \\
p_1, p_2, \ldots, p_5 &- \quad \text{binary.}
\end{aligned}
$$

2 Multiple-criteria Decision-making

Multiple-criteria decision-making (MCDM), multiple-criteria decision analysis; or multiple-attribute decision-making is a part of operations research. In this part, the problems of alternatives evaluated under several criteria are solved. Typically, the criteria are in conflict – we want to buy the best thing cheap as is possible. MCDM problems are solved in daily life and also in settings such as economic problems, business, medicine and so on.

One of the typical MCDM problems in economics is portfolio optimisation – the aim is to find the portfolio with the biggest possible return with the smallest possible risk. However, it is well-known that a portfolio with a higher expected return usually has a higher measure of risk.

Generally, cost or price is usually one of the main criteria, and some measure of quality is typically another one. It is clear that such criteria are in conflict.

If we consider buying a new car, we take into account something like – cost, comfort, safety, and fuel economy -– it is unusual that the cheapest car is the most comfortable and the safest one.

It is clear that we do MCDM in our daily lives; usually, we weigh multiple criteria implicitly, we do not do any analysis, we do not think about any methods. On the other hand, if we should to decide some critical problem (for example as a manager in some company), we should be able to explain our decision. The MCDM methods are the way how to do it. It helps us to structure the problem and explicitly evaluate multiple criteria properly.

To show more precisely, what type of example we can solve by these methods, let us consider the following prototype example.

Example 2.1 (Council Tender) Town council announced tender for a building of a new sewer system. There are four announced criteria of optimisations - the price of the tender, the duration of the work, the economic results in the last year of the offering companies and the contribution of subsuppliers. Town council obtained three offers from three different firms, see the table.

company	**price (mil. CZK)**	**duration (months)**	**economic result (mil. CZK)**	**subsuppliers (%)**
A	20	26	2	10
B	24	30	6	30
C	18	28	4	25
cr. type	min	min	max	min
weights	0.4	0.3	0.1	0.2

The town manager discussed with the town council, and then he set the weights for individual criterions.

In this example, we have three alternatives — three offers and four criteria. In the first step, we need to set the weights; here they are given in this example. Therefore, we aim to identify the best alternative if we take into account all our criteria. Notably, the MCDM is a case, when we have a (finite) list of possible alternatives, and we know conditions under which we want to make a decision and also know objectives.

Typically, better evaluation under one of the criteria means worse evaluation under another one (s), hence in MCDM, there does not exist an optimum alternative. Usually, we have a list of possible "good" alternatives, and we aim to determine the "best" one, in fact, the **compromise** one. (Because the choice of the best one is subjective.)

As was mentioned above, in MCDM we have a list of all possible alternatives evaluated under several criteria. To start to solve such problems we first need to construct so-called **decision matrix** $\mathbb{R}$. The table 2.3 gives the decision matrix for Prototype example, wherein rows we have alternatives and in columns are considered criteria. Therefore, the element of the matrix r_{ij} gives us the evaluation of the i-th alternatives according to the j-th criterion.

Our decision matrix has three rows (= number of alternatives) and four columns (= the number of criteria). The element r_{ij} gives us the value of the alternative i under the criterion j.

Criteria Types and their Transformation As we can see in our Prototype example, we can have typically two criteria types – cost type (in our model the weight and the price) and profit type (in our case the waterproof rating and the expert evaluation).

Keeping the type of criteria during our analysis is essential. Some methods need to have all criteria of profit type; then it is necessary to transform cost type criteria into profit type (methods how to do it will be discussed later).

Existence of Feasible Solutions Similarly to linear optimisation, the first question is, if the feasible solution to this problem exists. The **feasible solution** is any solution which satisfies all conditions given by the decision maker.

Typically, we have in our first table only feasible solutions (usually, we do not include alternatives which do not match with our conditions). Sometimes, we have a list of several possible, and then we state some conditions and choose the feasible alternatives, then the set of feasible solutions can be narrower than the previous list of possible alternatives.

> **Example 2.2** The DM can look at the announcements and find 20 jobs offers; however, he is not interested in all of them. Hence, I can make the list of all of them. But all of them are not acceptable for me - for example for some of them I do not have a qualification, some of them I am not interested in it and so on. So I have to narrow the list of offers to only feasible ones. Typically, I do the list of only feasible offers for me as is done in Prototype example.
> There are also methods how to narrow the list – for example Conjunctive and Disjunctive methods, we will prescribe them later.

Compromise Solution If the feasible solution exists, we should be able to choose the compromise one, too. The choice of compromise solution depends on the choice of weights and choice of MCDM method.
Typically, in the MCDM, there does not exist a unique optimal solution. The unique optimal solution exists only in a case when one of the alternatives is so-called ideal or it dominates all others; see below what it means.
It is the main difference between MCDM and most other methods of operations research; typically there is not only one optimal solution. It is a reason, why we speak about
Before starting with the construction of weights and the introduction of MCDM methods, let us introduce so-called Pareto optimal alternatives and also the basic principle of MCDM.

Pareto Optimal Alternatives First, let us introduce **dominated alternatives**.

Definition 2.1. We say that the alternative A is **dominated** by the alternative B, if the alternative B is under all criteria better than (or the same as) the alternative A and at least under one criterion it is strongly better than the alternative B. We also say that the alternative B **dominates** the alternative A.

It is clear from the definition that the dominated alternatives cannot be the compromise ones (there exists at least one alternative which is better than dominated one).
So, in our decision-making process, we are interested in **nondominated alternatives, Pareto optimal ones**.

Definition 2.2. We say that the alternative is nondominated (or Pareto optimal) if any other alternative does not dominate it.

Example 2.3 (Solution to Council Tender Example) In Prototype example, we have no dominated alternative. To check it we need to compare every two alternatives if one of them is better than the second one. Let us compare Company A and Company B – Company A is better in price, duration; and number of subsuppliers, however, it is worse in economic result than Company B, hence it does not dominate Company B even it is dominated by it. Similar results we get if we compare Company A and C and Company B and C.

Utopia and Nadir alternatives In many methods MCDM, we use so called **Utopia alternative** and **Nadir alternative**. Both of these alternatives are hypothetic alternatives, first one has the best possible values under all criteria, the second one has the worst evaluation under all criteria.

Example 2.4 (Solution to Council Tender example) Utopia and Nadir alternatives in Prototype example

$$\text{utopia} = (18 \text{ CzK}; 26 \text{ months}; 6\text{mil. CzK}; 10\%),$$

$$\text{nadir} = (24 \text{ CzK}; 30 \text{ months}; 2\text{mil. CzK}; 30\%).$$

As was already mentioned there exist many methods of MCDM and the solution depends on the choice of the method. We can construct our own methods for decision-making. Therefore, how to recognize "good" method? Some basic properties of MCDM methods which should be satisfied exist. Let us explain them carefully.

2.0.1 Basic properties of MCDM methods

As we mentioned above, typically there is no unique solution of MCDM problem, the solution depends on the choice of weights, the choice of method, the option of data standardisation. However, how to recognise a "good" method of MCDM? The method should be such that the solution satisfies the following conditions:

Pareto Pptimal Solution The solution given by the method must be Pareto optimal (nondominated) alternative.

Determination The method must lead to a solution.

Uniqueness The method should give us a unique solution (after the setting of weights). It must identify one compromise solution.

Invariance to the Ordering of Criteria and Alternatives The choice of the compromise alternative should not depend on the order of criteria or alternatives.

Remark. It is a very trivial condition which says that for example the result of the tender cannot depend on the order of candidates - if we sort the candidates by their names or by the names of their companies or by the data of receiving their application or by their offer - the result should be still the same.

Invariance to Measure Units. The choice of compromise alternative should not be affected by the selection of measure units in which we evaluate the criteria. For example, if we want to choose the best offer according to the salary, the result should be the same if we set the salary in pounds, dollars, thousands of pounds and so on.

Invariance to the Addition of Non-optimal Alternatives. We should choose the same alternative does not matter if we have added some non-optimal (for example) dominated alternative to the list of feasible alternatives. The choice should also be the same whether we remove from the list of alternatives all dominated alternatives or not.

Fairness of the Method. The method should allow choosing any of nondominated alternatives by the setting of appropriate weights.

Comments on Properties of the MCDM Methods However, the properties as mentioned above of MCDM methods seem to be rational; we will see in the following lectures that some of them are not (unfortunately) satisfied by all methods. Also, some of the widely used methods do not meet some of these conditions. Hence, it is necessary to keep it in our mind when we make our final decision.

2.1 Construction of Weights

The first step in MCDM is the construction of the weights. The weights give us information about the importance of the criteria (from the point of the decision maker's view). Typically, there is one useful assumption for the weights – it is useful to suppose that the sum of weights is equal to one. Hence from now on, we will suppose that the sum of weights is equal to one.
Several types of methods on how to construct the weights exist. We can divide them according to **information about criteria preferences** which we need to assemble them. We can have following types of information about the preferences of criteria and in these cases to use for example following methods to construct weights. (A lot of methods of construction weights can be developed, so we mention only some of them.)

No Criteria Preferences
- Equal Weight Method

Ordinal Criteria Preferences
- Rank (Sum) Weight Method

- Fuller Triangle Method

Cardinal Criteria Preferences

- Reference Point Method
- Saaty Weight Method

No Criteria Preferences

In the case when we have no information about criteria preferences, there is only one possibility how to construct weights – **Equal Weights Method** – to evaluate each criterion by the same weight. In the case when we have n criteria; we use weights $1/n$ for each criterion, we can write:

$$v_i = 1/n, \text{ for all } i \in \{1, \dots, n\},$$

where n is the number of criteria.
Let us suppose that in our Prototype Example we have no information about criteria preferences. Let us set the weights.

Example 2.5 (Solution to Council Tender Example) Let us suppose that in our Prototype Example we have no information about criteria preferences. Let us construct the weights.
We have four criteria, hence $n = 4$, therefore we get

$$v_1 = v_2 = v_3 = v_4 = 1/4.$$

All weights are the same, and their sum is equal to one.

The following picture shows, how to develop this method in Excel.

	A	B	C	D
1			**Equal weights method**	
2			equal points	weights (standardization)
3	**Criterion 1**	**price (mil. CzK)**	1	=C3/$C$7
4	**Criterion 2**	**duration (months)**	1	=C4/$C$7
5	**Criterion 3**	**economic result (mil. CzK)**	1	=C5/$C$7
6	**Criterion 4**	**subsuppliers (%)**	1	=C6/$C$7
7		**sum**	4	
8				
9				

So, we get:

		Equal weights method	
		equal points	weights (standardization)
Criterion 1	**price (mil. CzK)**	1	**0,25**
Criterion 2	**duration (months)**	1	**0,25**
Criterion 3	**economic result (mil. CzK)**	1	**0,25**
Criterion 4	**subsuppliers (%)**	1	**0,25**
	sum	4	

Ordinal Criteria Preferences

In the case when we have only ordinal information about criteria, it means that we now only the rank of criteria but we are not able to measure the distance between criteria.

Rank (Sum) Weight Method Therefore, one of the possible ways how to construct weights is the following. First, we set the rank to each criterion. We have n criteria; hence the ranks are from 1 to n. For more important criterion we have a smaller number of rank, but in weights, the smaller number means less important criterion. Hence, we construct weights in the following way:

$$v_i = \frac{n+1-r_i}{n\frac{n+1}{2}}, \text{ for all } i \in \{1, \dots, n\},$$

where n is the number of criteria and r_i is the rank of ith criterion. This way of weight construction is called **Rank (Sum) Weight Method**.

Example 2.6 (Solution to Council Tender Example) Let us suppose that in our Prototype Example we know, that the most important criterion for the decision maker is the price, then the duration, then the number of subsuppliers and the less important one is the economic result. Let us construct the weights. We have four criteria, hence $n = 4$, therefore we get

$$v_i = \frac{4+1-r_i}{4\frac{4+1}{2}},$$

where r_i is the rank of criterion (according to the preference importance. So,

$$v_1 = 4/10,\ v_2 = 3/10,\ v_3 = 1/10,\ v_4 = 2/10;$$

or

criterion	rank	points	weight
price	1st	4	4/10
duration	2nd	3	3/10
n. of subsuppliers	3rd	2	2/10
economic result	4th	1	1/10
sum of points		10	

Fuller Triangle Method However, in case of a high number of criteria it could be difficult to order the criteria; hence we can use so-called Fuller's method. Under this method, we compare each pair of criteria and highlight the criterion which is more essential for us from the couple. Then we compute the number of preferences for each criterion, and by standardisation, we obtain the weights.

Example 2.7 (Solution to Council Tender Example) Let us order all possible couples of our criteria into the Fuller triangle and ask the DM to highlight the more important one from each couple, for the result see Figure ??.

FULLER TRIANGLE		
Criterion 1	**Criterion 1**	**Criterion 1**
Criterion 2	Criterion 3	Criterion 4
	Criterion 2	**Criterion 2**
	Criterion 3	Criterion 4
		Criterion 3
		Criterion 4

Then, we can construct weights in the following way. (Since the DM is consistent in his preferences, we add 1 to the number of preferences for all criteria (not to have zero weight), and we obtain the same weights as under Rank Weight Method. For the construction see the following figure.

	A	C	D	E
1		**Fuller Triangle Method**		
2				
3		Preference points (from FT)	Preference points + 1	weights (standardization)
4	**Criterion 1**	3	=C4+1	**=D4/$D$8**
5	**Criterion 2**	2	3	**0,3**
6	**Criterion 3**	0	1	**0,1**
7	**Criterion 4**	1	2	**0,2**
8			10	1

Cardinal Criteria Preferences

In the case when we have cardinal information about criteria preferences, it means that we can measure how much is the one criterion better than the second one. One of the typical how to construct the weights under cardinal information is the so-called Point method.

Reference Point Method This method is based on an evaluation of the criteria by points according to their importance. There are several ways how to do it. One of them is to set the point scale for the importance is for example from 0 to 10, and the DM is asked to assign some points to each criterion to express its importance. Then we sum all added points and set the weights as allocated points divided by the sum of all points.
The other way is to ask DM to allocate 100 points among the criteria in the way that it expresses its importance. Then to get weights, we only divide points by 100 (it is the sum of allocated points). Let us illustrate both of these ways on our Prototype Example.

Example 2.8 (Solution to Council Tender Example) To construct the weights, first, we need to ask the decision-maker to assign points to each criterion to give preference information, then we standardise the points, see the table below.

criterion	points(DM)	weight
price	10	10/25
duration	9	9/25
n. of subsuppliers	2	2/25
economic result	4	4/25
sum of points	25	

Saaty Method This method is not easy to use, on the other hand, it combines the profits of Fuller triangle method (we compare only pairs of criteria) and it allows us to use cardinal information. To construct weights by this method, we first need to construct so-called **Saaty matrix**, which has so many rows and columns as is the number of criteria. Each point s_ij gives us information how much more important is the ith criterion than the jth one for the decision maker. We ask DM to assign values into the table according to the following rules:

If the ith criterion is more important than the jth one for the decision maker assign:

9 - if the preference is extreme,

7 - if the preference is quite strong,

5 - if the preference is strong,

3 - if the preference is quite weak,

it is also possible to use the other integers bigger than one and less than or equal to nine to evaluate the measure of preference between criteria.

If the ith and jth criteria are indifferent for the decision maker assign:

1

If the ith criterion is less important than the jth one for the decision maker assign:

$$s_{ij} = 1/s_{ji}.$$

In the case when the decision maker is absolutely consistent in his preferences, than

$$s_{ij} = v_i/v_j.$$

(It could be also the key which can help to assign the values of the matrix.) In such a case, we have

$$s_{ij} = s_{ik} \cdot s_{kj}/\text{for all } k = \{1, \ldots, n\}.$$

If the previous equation is satisfied for all possible combinations of i, j, k then we speak about the fully consistent matrix and we can easily construct weights by geometric mean. More precisely, for each criterion we compute the geometric mean (in Excel the function GEOMEAN) from the values which were assigned to this criterion (from each matrix row); to get weights we standardise these values. In case, when the matrix is not fully consistent; this procedure need not lead to the correct weights. Therefore, we use this process and then ask DM to verify the solution, i.e. we show him weights and ask him if he agrees with given results. If not, we show some inconsistency in the matrix and ask him to reconsider the matrix. Then we apply the procedure once again.

Example 2.9 (Solution to Council Tender Example) First, we ask the DM to fulfil the Saaty matrix according to rules introduced above. Then we apply the geometric mean to evaluate each criterion and at the end, we standardise the weights, see the following table.

	A	B	C	D	E	F	G
1		**Saaty Weight Method**					
2	**Saaty matrix**						
3		**Criterion 1**	**Criterion 2**	**Criterion 3**	**Criterion 4**	Geometric mean	weights (standardization)
4	**Criterion 1**	1	5	8	7	=GEOMEAN(B4:E4)	**=F4/$F$8**
5	**Criterion 2**	0,2	1	6	5	1,56508458	**0,242119389**
6	**Criterion 3**	0,125	0,16666667	1	0,5	0,319471552	**0,049422413**
7	**Criterion 4**	0,14285714	0,2	2	1	0,488923022	**0,075636643**
8					sum	6,464102644	

2.2 Methods of MCDM

Many methods of MCDM exist, it is not the aim of this text to introduce all of them. It this text, we choose some typical methods and introduce their ideas and information which they can give us.
Every good method of MCDM fulfils the following conditions:

- Dominated alternatives cannot be chosen as the best ones.
- The choice of the best alternative does not depend on the rank of the criteria.
- The choice of the best alternative does not depend on the rank of the alternatives.
- The choice of the best alternative does not depend on the scale.

Methods of MCDM can be divided into several groups for example according to the type of information which must be given about evaluation of alternatives under criteria.

Ordinal Information
- Lexicographical method
- Rank method

Cardinal Information
- Point method
- Weighted Sum Method
- TOPSIS
- ELECTRE

2.2.1 Data

Before each analysis, we need to have the data in the form which we can use for the analysis. There are several problems we often need to handle it.

Missing values We may need to compare given alternatives and unfortunately, some of them are not evaluated under some criterion. How to handle such a problem?
One of the possible ways is to remove the alternative from the analysis – we do not have enough information to do the study. However, the DM can ask us to include it. So, we can omit the criterion from the analysis (we are not able to compare all alternatives under it), but if the criterion is essential for the DM, it is not a good way, too.
So, if we need to include all alternatives and all criteria into the decision-making process, we need to assign some value to the missing evaluation of the alternative. The question is how to estimate the assessment. Usually, there are two main reasons why the assessment is missing. One of them is that it is not essential for

other decision-makers, so it is not given. The second is that the evaluation of this alternative is under this criterion so bad, that it is not given. So, we should be able to decide which of the possibilities is the case. In the case of the first possibilities, we can assign mean (or a value less bad than the mean (from the evaluation of the criterion under other alternatives)); in the second case, we should assign the worst evaluation under the criterion or something worse.

Transformation of the type of objective function In many real-life problems, we have both kinds of type of objective function – cost and benefit types typically. However, some methods need to have all criteria in benefit type. How to transform cost-type objective function into benefit-type?
Several ways how to transform the type of objective function exists. One of them is the change of sign, i.e. instead of $\min f(x)$ we use $\max(-f(x))$. This way is straightforward to do, but the economic interpretation of such values is often impossible (typically, we get negative values at places where no negative values could exist). The other way (which omit negative values) is the transformation of $\min f(x)$ into $\max(1/f(x))$ (it is possible only if any value of $f(x)$ is not equal to zero). The problem of negative values is solved, but the interpretation of such values still does not exist. The mainly use transformation is the transformation of $\min f(x)$ into $\max(M - f(x))$, where M is the biggest possible value of $f(x)$ or the biggest acceptable value. In such a case we usually have a good interpretation (it shows us how much we save up) but the choice of M is individual and the different choice of M can lead to different decision-making result.

Example 2.10 (Solution to Council Tender example) In our Prototype example, we have three cost-type criteria. Therefore, the question is how to transform them into benefit-type criteria. Let us focus on the criterion price for example. The companies offer to do the project at the cost of 20, 24, 18 mill. CzK.
In case, when we decide to apply the easiest possible transformations – $\min f(x)$ transform into $\max(-f(x))$; resp. $\min f(x)$ into $\max(1/f(x))$, we get benefit-type criterion with values -20, -24, -18; resp. $1/20$, $1/24$, $1/18$. In both cases, it is not clear, how to interpret the values, in the first case we get negative values, what can bring problems during optimization process. Therefore, let us focus on the transformation of $\min f(x)$ into $\max(M - f(x))$. The question is how to choose the constant M. There are two main ways. First, set

$$M := \max_x f(x),$$

in our case $M = \max\{20, 24, 18\} = 24$. And transformed values are 4, 0, 6. The interpretation is a possible saving against the worst offer. However, this transformation depends on the worst value under this criterion, so, it may be

affected by some added dominated alternative.
To avoid this problem, we can set M at the beginning of the evaluation process. We ask the DM, what is the biggest price he can pay and it will be M. All offers, where the price is bigger than such value we exclude from the decision-making process (it is not acceptable for the DM). In Prototype example, the DM can say that the maximum possible price for such a project is 30 mill. CzK, then we can transform our criterion into benefit-type criterion with values 10, 6, 12. These values can be interpreted as possible saving against the highest acceptable price. However, it is clear that the choice of M is the individual.

In case, when we have only a few alternatives and few criteria, it is possible to solve the problems "on the paper". However, in a case when we have a higher number of alternatives or criteria, it is more comfortable to solve these problems in some software. In this text, we present the solution in Excel. In the following, we explain each method of how to apply it on the paper and also demonstrate how to use it in Excel. Admittedly, there is also software for these kinds of decision-making problems; however, it is not the aim of this text to introduce it.
To solve these problems in Excel, first, we set the problem into the Excel sheet, for example in the following way (for Prototype example). (We also set the weights and compute utopia and nadir alternatives.)
If many alternatives exist, we can first apply the following methods to narrow the number of feasible alternatives. We also can repeat these methods – to change the bounds to get less or more alternatives.

Conjunctive methods We choose only such alternatives which fulfil conditions under all criteria.

Example 2.11 (Solution to Council Tender example) In the tender could be written that the town council requires only projects in duration less than or equal to 30 months and in price up to 30 mill. CzK. Hence, no offer which exceeds one of these numbers is feasible. The project is unacceptable if it takes more than 30 months or the price is higher than 30 mill. CzK.

Disjunctive method We accept all alternatives which fulfil the given conditions at least under one criterion.

Example 2.12 (Solution to Council Tender example) In the tender could be written that the town council requires only projects in duration less than or equal to 30 months or in price up to 20 mill. CzK. Hence, the acceptable offers must have a duration less than 30 months, or if it exceeds 30 months, then the price must be less than 20 mill. CzK.

Ordinal information

Lexicographical method For the application of this method, it is enough to have only ordinal information about alternatives preferences under all criteria. We compare the alternatives under the most important criterion, in the case of two best alternatives, we examine them under the second criterion and so on. The main advantage of this approach is that it is straightforward to use, on the other hand, it takes into account evaluation under only one criterion, which is a significant disadvantage.

Example 2.13 (Solution to Council Tender example) Let us solve Prototype example with given weights be the Lexicographical method. According to the given weights, we can see that the most important criterion is the price. So, we order the alternative according to the price:
1st company C, 2nd Company A and the last one Company A.

Rank method This method also works only with ordinal information about alternatives preferences under criteria, nevertheless, it takes into account evaluation under all criteria. It is based on the weighted rank of alternatives.

$$\mathbf{Z} = (z_{ij} = v_i r_{ij}).$$

Then we put

$$p_i = \sum_j z_{ij}.$$

The best alternative has the smallest value of p.
Let us show the solution of Prototype example by this method.

Example 2.14 (Solution to Council Tender example) First, let us recall the decision matrix:

Table 2.1: Council Tender Example – Rank Method.

comp.	**price (mil.CzK)**	**duration (months)**	**ec. result (mil. CzK)**	**subsuppliers (%)**
company A	20	26	2	10
company B	24	30	6	30
company C	18	28	4	25
criterion type	min	min	max	min
weights	0.4	0.3	0.1	0.2

Now, we can set the ranks of each alternative under each criterion:

Table 2.2: Council Tender Example – Orders.

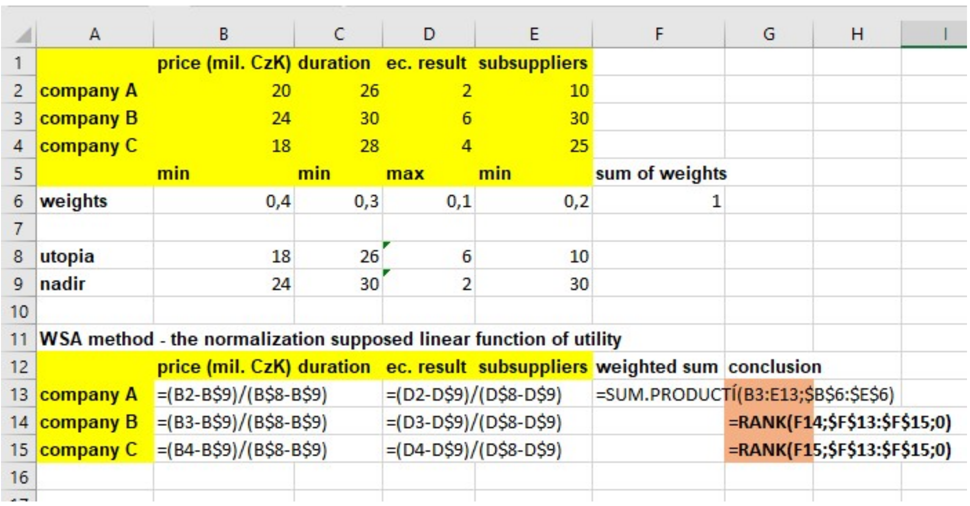
comp.	**price (mil.CzK)**	**duration (months)**	**ec. result (mil. CzK)**	**subsuppliers (%)**
company A	2nd	1st	3rd	1st
company B	3rd	3rd	1st	3rd
company C	1st	2nd	2nd	2nd

Therefore, we get following weighted ranks:

Table 2.3: Council Tender Example – Solution by the Rank Method.

company	**computation**	**result**	**rank**
company A	$2 \cdot 0.4 + 1 \cdot 0.3 + 3 \cdot 0.1 + 1 \cdot 0.2$	1.6	1st - 2nd
company B	$3 \cdot 0.4 + 3 \cdot 0.3 + 1 \cdot 0.1 + 3 \cdot 0.2$	3.6	3rd
company C	$1 \cdot 0.4 + 2 \cdot 0.3 + 2 \cdot 0.1 + 2 \cdot 0.2$	1.6	1st- 2nd

Using this approach, we cannot decide (it is not obvious) which of the alternative is the best one — Companies A and C have the same evaluation.
As was mentioned above, it is possible to use Excel to solve this issue. In Excel, we can use a function RANK – which gives us the rank of the value from the given list. The last argument of the function is the parameter of ranking – 0 for ranking from the largest to the smallest and 1 for from the lowest to the largest. For more details see the following picture or the attached Excel file.



	A	B	C	D	E	F	G	H	I
1		price (mil. CzK)	duration	ec. result	subsuppliers				
2	company A	20	26	2	10				
3	company B	24	30	6	30				
4	company C	18	28	4	25				
5		min	min	max	min	sum of weights			
6	weights	0,4	0,3	0,1	0,2	1			
7									
8	utopia	18	26	6	10				
9	nadir	24	30	2	30				
10									
11	WSA method - the normalization supposed linear function of utility								
12		price (mil. CzK)	duration	ec. result	subsuppliers	weighted sum	conclusion		
13	company A	=(B2-B$9)/(B$8-B$9)		=(D2-D$9)/(D$8-D$9)		=SUM.PRODUCTÍ(B3:E13;$B$6:$E$6)			
14	company B	=(B3-B$9)/(B$8-B$9)		=(D3-D$9)/(D$8-D$9)			=RANK(F14;$F$13:$F$15;0)		
15	company C	=(B4-B$9)/(B$8-B$9)		=(D4-D$9)/(D$8-D$9)			=RANK(F15;$F$13:$F$15;0)		
16									

Point method The principle of this method is similar to the Preference point method for the criteria of weight construction. We need to have cardinal information about all alternatives under all criteria, then we assign points (utility) to each alternative under each criterion and apply weighted sums. (In Excel by applying SUM.PRODUCT function.) The higher weighted sum is better. So, the best alternative has the highest weighted sum.

Weighted Sum Method (WSM) In fact, this method can be also called as a method of linear utility function. (In case, when we suppose linear utility function, then this method is equivalent to maximization of weighted utility. In the first step of this method, we use following standardization:

$$s_{ij} = \frac{r_{ij} - n_j}{u_j - n_j},$$

where n_j; resp. u_j stand for nadir; resp. utopia value under jth criterion, i.e. in case of benefit-type criterion:

$$n_j = \min_i r_{ij} \quad \text{and} \quad u_j = \max_i r_{ij},$$

in case of cost-type criterion:

$$n_j = \max_i r_{ij} \quad \text{and} \quad u_j = \min_i r_{ij}.$$

The main advantage of this method is its simplicity. On the other hand, the main disadvantage of this method is a possible dependence on an added dominated alternative. The way how to handle this problem is the usage of conjunctive and disjunctive methods before the optimisation.

Let us demonstrate how to use this method for our Prototype example.

Example 2.15 (Solution to Council Tender example) First, we need to standardise the values. We apply the above-described method.

Table 2.4: Council Tender Example – Data.

comp.	**price (mil.CzK)**	**duration (months)**	**ec. result (mil. CzK)**	**subsuppliers (%)**
A	20	26	2	10
B	24	30	6	30
C	18	28	4	25
cr. type	min	min	max	min
weights	0.4	0.3	0.1	0.2
utopia alt.	18	26	6	10
nadir alt.	24	30	2	30
STAND.				
A	$\frac{20-24}{18-24}$	$\frac{26-30}{26-30}$	$\frac{2-2}{6-2}$	$\frac{10-30}{10-30}$
B	$\frac{24-24}{18-24}$	$\frac{30-30}{26-30}$	$\frac{6-2}{6-2}$	$\frac{30-30}{10-30}$
C	$\frac{18-24}{18-24}$	$\frac{28-30}{26-30}$	$\frac{4-2}{6-2}$	$\frac{25-30}{10-30}$

Therefore, we can apply weights and get the following weighted sums:

Table 2.5: Council Tender Example – Solution by the WSM.

company	**computation**	**result**	**rank**
company A	$\frac{2}{3} \cdot 0.4 + 1 \cdot 0.3 + 0 \cdot 0.1 + 1 \cdot 0.2$	0.77	1st
company B	$0 \cdot 0.4 + 0 \cdot 0.3 + 1 \cdot 0.1 + 0 \cdot 0.2$	0.1	3rd
company C	$1 \cdot 0.4 + \frac{1}{2} \cdot 0.3 + \frac{1}{2} \cdot 0.1 + \frac{1}{4} \cdot 0.2$	0.65	2nd

It is possible to apply this method in Excel, see the following picture or the attached Excel file.

	A	B	C	D	E	F	G	H	I
1		price (mil. CzK)	duration	ec. result	subsuppliers				
2	company A	20	26	2	10				
3	company B	24	30	6	30				
4	company C	18	28	4	25				
5		min	min	max	min	sum of weights			
6	weights	0,4	0,3	0,1	0,2	1			
7									
8	utopia	18	26	6	10				
9	nadir	24	30	2	30				
10									
11	WSA method - the normalization supposed linear function of utility								
12		price (mil. CzK)	duration	ec. result	subsuppliers	weighted sum	conclusion		
13	company A	=(B2-B$9)/(B$8-B$9)		=(D2-D$9)/(D$8-D$9)		=SUM.PRODUCTÍ(B3:E13;$B$6:$E$6)			
14	company B	=(B3-B$9)/(B$8-B$9)		=(D3-D$9)/(D$8-D$9)			=RANK(F14;$F$13:$F$15;0)		
15	company C	=(B4-B$9)/(B$8-B$9)		=(D4-D$9)/(D$8-D$9)			=RANK(F15;$F$13:$F$15;0)		
16									

TOPSIS This method is a bit more complicate to compute (so, we will run it in Excel), on the other hand, it allows us to use euclidian metric what is more convenient than the assumption of a linear utility function, for example.
For applying this method, we first need to have all the criteria in benefit-type. Then we do the first standardisation – we divide each number in the table by the (euclidian) norm if the column vector. Then we apply weights and find out utopia and nadir alternative. Then it remains to compute for each alternative its distance (euclidian) from utopia and nadir alternative and compare these values under given criterion. For more details see attached Excel file.

ELECTRE and others Many methods od MCDM exist, some of them are computationally simple, some of them quite complicated. Since MCDM methods are widely used, many SW for MCDM exist, in this course, we use an Excel Macro SANNA. SANNA was developed at The University of Economics in Prague and it involves several basic MCDM methods. It can be downloaded from nb.vse.cz/ $\sim$ jablon.

3 Data Envelopment Analysis (DEA)

In the previous chapter, we introduced the basic methods of MCDM. As was written there, the solution to such problems depends on the choice of the weights and also on the selection of the method. The aim of this chapter is different, in this chapter, the objective of the analysis is the identification of so-called effective units (alternatives in terms of the previous chapter), where the units (alternatives) are supposed to have some know inputs (cost-type criteria in terms of the last chapter) and outputs (profit-type criteria).

More precisely, Data Envelopment Analysis (DEA) is a technique used to evaluate the technical efficiency of examined units. What do we mean by **technical efficiency**? Vaguely speaking, we search for units which achieve the best outputs at the smallest inputs.

Data envelopment analysis (DEA) also called frontier analysis, was first put forward by Charnes, Cooper, and Rhodes in 1978. Since the technique was first proposed much theoretical and empirical work has been done. Many studies have been published dealing with applying DEA in real-world situations.

It is a performance measurement technique which can be used for evaluating the relative efficiency of decision-making units (DMU's) in organisations. A DMU is a distinct unit within an organisation that has flexibility with respect to some of the decisions it makes, but not necessarily complete freedom concerning these decisions.

By this technic, we can typically evaluate activities of bank branches, hospitals, tax offices, departments of some company, schools and university departments, government institutions and so on. To use this technic, we need to have several equivalent units evaluated in several inputs (at least in one) and several outputs (at least in one). Then we can run DEA and answer the question, which of the units are effective and what should improve the non-effective ones to become effective.

To understand DEA methodology, let us first consider the most straightforward possible case – single input and single output case. Let us formulate the following prototype example.

Example 3.1 (Prototype Example – Business Chain) The business chain has eight branches. For each branch, we know the number of employees and daily sales (in 10 thousand CzK), see the following table.

Branch	A	B	C	D	E	F	G	H
N. of employees (x)	2	2	3	4	5	5	6	8
Sales (y)	1	4	2	3	4	2	3	5

Which branch is effective and which need some improvements?

3.1 1 Input - 1 Output Problems

As was already mentioned, this case is the simplest one – there are only a single input and single output, as is presented in Prototype example. In such a case it is the easiest way how to measure the efficiency of the unit to use ratios.
Rations are commonly used methods, however, in 1-1 case, it is the easiest to use it. Because there is no doubt how to measure input and how to measure output – since we have only one input and one output, we compare the ratios of outputs and inputs (it does not matter which scale of inputs and outputs we use, we only have to use the same for all units). Such a ratio can be viewed as the number of outputs which give us a unit input for each decision-making unit. It is clear that the higher number stands for more efficient DMU.
Hence in our prototype example, we have:

Branch	A	B	C	D	E	F	G	H
efficiency y/x	0.5	2	0.67	0.75	0.8	0.4	0.5	0.63

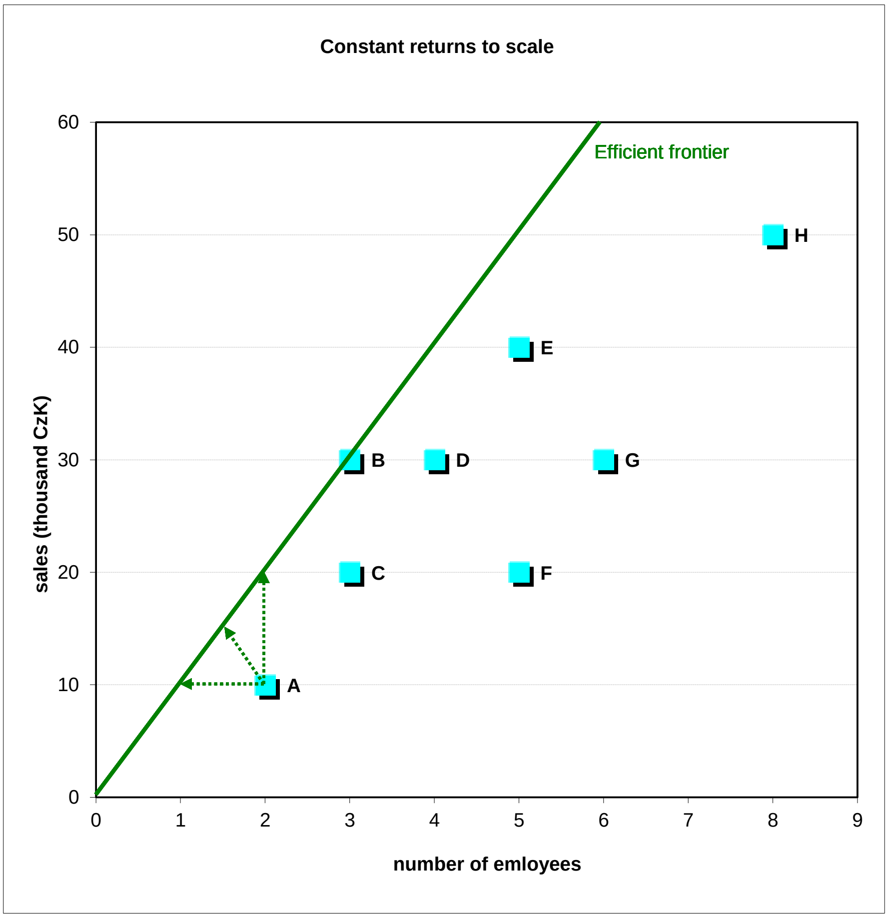
Therefore, we can see that the branch B has the highest ratio of sales per employee – we can say that the branch B is the most effective branch. On the other hand, the branch F has the lowest ratio of sales per employee, so we can say that it is the less effective branch.
Now, the natural question raises – how much effective are the individual branches – in comparison with the most effective branch B? It is clear that it is possible to have the output 2 per unit input, let us say, that it is 100%. Then, we can compare all other branches to the most effective one and calculate their so-called **relative efficiency with respect to the effective branch** B. To do it, we divide the ratio for any branch by 2 (the value for the branch B) and multiply it by 100 to convert to a percentage.
So, in our example we get:

Branch	A	B	C	D	E	F	G	H
efficiency y/x	0.25	1	0.335	0.375	0.4	0.2	0.25	0.315

Now, it is easily seen that all other branches are much less efficient than the branch B. If we suppose that all branches are comparable (from the point of view of non-included measures), then we can say that for example the branch A is effective from only 25% and if we want A to be effective it must change inputs or outputs or both of them such that their ratio will be the same as the ratio of output and input of the most effective branch.

such easy examples we can also solve graphically, see the following picture 3.1.

Figure 3.1: DEA - 1 input - 1 output problem



The line that goes through the beginning and includes the most significant angle with the axel x is called **efficient frontier**. All branches which lie at the frontier are effective (have the same efficiency) and all ones under the frontier are ineffective. The ineffective branch can become effective if it changes the inputs or output; or both of it in the way to move itself to the efficient frontier.

3.2 1 Input - 2 Outputs Problems

In this section, we introduce the last set of DEA problems which can be solved in the graphical way – 2I-1O problems.

Let us consider the following prototype example.

Example 3.2 (Prototype Example – Business Chain) The business chain has seven branches. For each branch, it is known the number of employees, number of customers per hour and daily sales (in 10 thousand CzK), see the following table.

Branch	A	B	C	D	E	F	G
Inputs							
n. of employees (x)	2	3	1	1	2	2	4
Outputs							
n. of customers per hour (y_1)	2	6	2	4	8	10	24
sales in 10 thousand CzK (y_2)	10	21	3	3	12	10	8

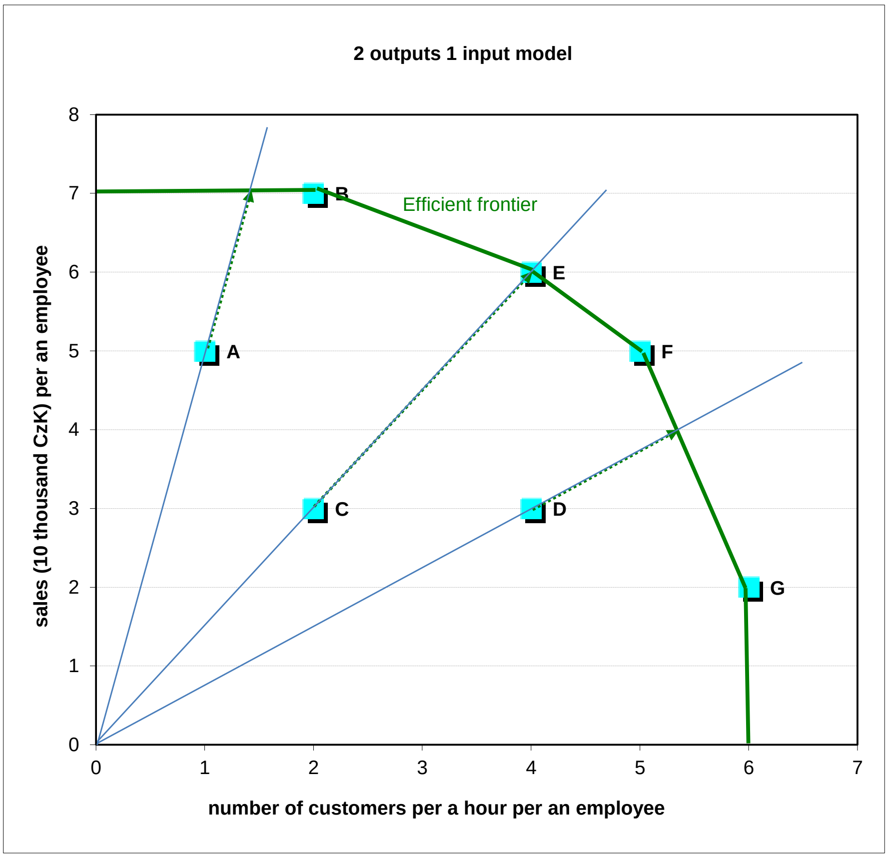
Is it possible to compare the branches and to decide which of them are effective?

It is clear that we are not able to display all these data into one 2D-graph, however, we have only one input, so, we can normalise the date into the number of outputs per unit input and then we get only two variables and will be able to draw a graph.

In the following table let us have the normalised data.

Branch	A	B	C	D	E	F	G
Outputs per unit input							
n. of customers per hour and one employee (y_1/x)	1	1	2	4	4	5	6
sales in 10 thousand CzK per one employee (y_2/x)	5	7	3	3	6	5	2

Now, let us display data in the graph **??**.



Which branches are effective? Since we are not able to compare the outputs, we denote a unit as an effective if there is no convex combination of other units better in both outputs, in a graphical way, we draw an efficient frontier as a convex envelopment of branches.

In Prototype example, we denote four branches as effective ones – B, E, F, G. All others are ineffective and can become effective if they change their outputs or input in the way to move itself to the efficient frontier. To measure their inefficiency; we can use the ratio between the distance of the beginning and the branch and between the beginning and its peer branch (image of the branch at the efficient frontier), i.e. for the branch C:

$$\frac{|OC|}{|OE|}.$$

In the case of the branch, C its peer branch exists, and it is the branch E. Generally, the peer unit does not exist, and it is given by a cross over point of the efficient frontier and the line provided by the beginning and the branch.

3.3 2 Inputs - 1 Output problems

There two other cases, when we can use a graphical way of the solution – 1 input 2 outputs problems and 2 outputs 1 input problems. In this section, let us introduce the graphical solution of the 1I-2O problems.

Let us consider the following prototype example.

Example 3.3 (Prototype Example – Business Chain) The business chain has nine branches. For each branch, we know the number of employees, overhead cost (in thousand CzK) and daily sales (in 10 thousand CzK), see the following table.

Branch	A	B	C	D	E	F	G	H	I
Inputs									
n. of employees (x_1)	12	7	16	8	4	5	18	20	12
over head cost (thousand CzK) (x_2)	9	3	2	4	8	2	12	10	5
Outputs									
sales (10 thousand CzK) (y)	3	1	2	2	2	1	3	4	2

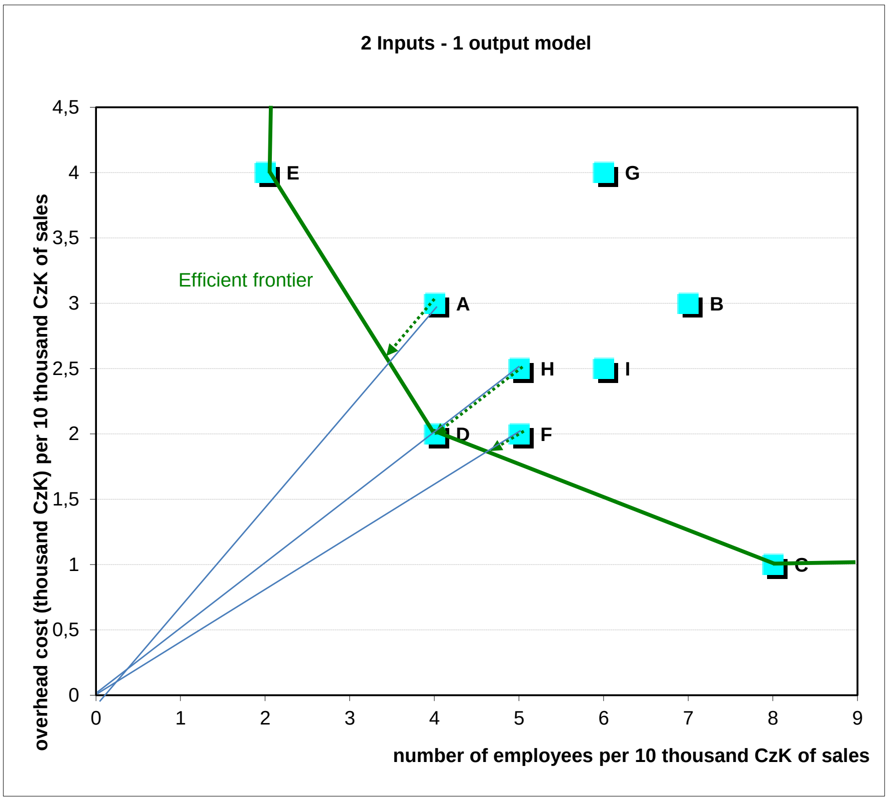
Is it possible to compare the branches?

Again, it is clear that we are not able to display all these data into one 2D-graph, however, we have only one output, so, we can normalise the date into the number of inputs needed per unit output and then we get only two variables and will be able to draw a graph.

In the following table let us have the normalised data.

Branch	A	B	C	D	E	F	G	H	I
Inputs per unit output									
n. of employees per unit sales (x_1/y)	4	7	8	4	2	5	6	5	6
over head cost per unit sales (x_2/y)	3	3	1	2	4	2	4	2.5	2.5

Now, let us display data in the following graph.



3.4 Linear Optimisation in DEA

As was shown above some small DEA problems can be solved graphically. However, when we have more than two inputs or outputs, or two inputs and two outputs, then we are not able to apply a graphical solution.

A general solution of DEA models uses linear optimisation. The basic idea is straightforward, for each unit we search for weights for inputs and outputs such that the efficiency of the unit is as height as is possible.

In mathematical formulation, we search for weights under which the efficiency of given unit is maximally and efficiencies of all units are less than or equal to one.

However, if write down the model directly, it is not the linear optimisation problem, the objective function and constraints are fractions with variables in both numerators and denominators. Fortunately, there is an easy way how to turn this model into a linear one. For the objective function, we fix (without loss of generality) either the numerator to be equal to one, or the denominator to be equal to one. Constraints are easily converted to linear conditions by multiplying

both sides of each constraint by the denominator.
Then we get two possible alternative linear models (dual models). When there exists a solution with objective function equal to one, the tested unit is effective if not it is not.
If we solve the problem in software, the sensitive analysis could help us to identify peer units for each ineffective unit and also give the answer how should the inefficient unit change its inputs, resp. outputs, to be effective.

3.4.1 Basic Notation

First, let us introduce the basic notation.
Let us suppose to have p units to analyse, the number of unit to be denoted by k, therefore $k = 1, 2, \ldots, p$; Each unit to have m inputs and n outputs.
We denote

$$\begin{aligned} x_{ik} \quad & – \text{i-th input of k-th unit}, i = 1, 2, \ldots, m,\ k = 1, 2, \ldots, p, \\ y_{jk} \quad & – \text{j-th output of k-th unit}, j = 1, 2, \ldots, n,\ k = 1, 2, \ldots, p. \end{aligned}$$

Weights which are assigned to each input, resp. output are denote by u_i, resp. v_j.
In the general case, we measure the efficiency as

$$\text{efficiency} = \frac{\text{weighted sum of outputs}}{\text{weighted sum of inputs}}, \tag{3.1}$$

in mathematical formulation:

$$e_k = \frac{\sum_{j=1}^{n} v_j y_{jk}}{\sum_{i=1}^{m} u_i x_{ik}},\ k = 1, 2, \ldots, p, \tag{3.2}$$

where u_i, v_j are weights for inputs and outputs and x_{ik} is the value of i-th iput of k-th unit and y_{jk} gives the value of j-th output of k-th unit.

3.4.2 CCR input oriented model

These models suppose constant returns. The results show us the way for ineffective units how to change their inputs to become efective. To decide if the unit is effective, we need to construct the model for the unit. Therefore, to model for the unit H (one of the given p units) is:

$$e_H = \frac{\sum_{j=1}^{n} v_{jH} y_{jH}}{\sum_{i=1}^{m} u_{iH} x_{iH}} \quad \to \quad \max,$$

which maximise the ration of weighted outputs to weighted inputs subject to the following constraints:

$$\frac{\sum_{j=1}^{n} v_{jH} y_{jk}}{\sum_{i=1}^{m} u_{iH} x_{ik}} \leq 1, \ \forall k = 1, 2, \ldots, p,$$

which give us the conditions that the efficiency of each unit is at most equal to one and constraints:

$$v_{jH} \geq 0, \ \forall j = 1, 2, \ldots, n,$$
$$u_{iH} \geq 0, \ \forall i = 1, 2, \ldots, m,$$

where the non-negativity of weights is given.

However, such model does not satisfy conditions for linear model, so it is quite difficult to solve it. Fortunately, this model is equivalent with the following one, which is already linear.

$$labeleq : CCRline\acute{a}rn\acute{i}modelvstupprim\acute{a}re_H = \sum_{j=1}^{n} v_{jH} y_{jH} \rightarrow \max$$
$$\sum_{i=1}^{m} u_{iH} x_{iH} = 1$$
$$-\sum_{i=1}^{m} u_{iH} x_{ik} + \sum_{j=1}^{n} v_{jH} y_{jk} \leq 0, \ \forall k = 1, 2, \ldots, p,$$
$$u_{jH} \geq 0, \ \forall j = 1, 2, \ldots, n,$$
$$u_{iH} \geq 0, \ \forall i = 1, 2, \ldots, m.$$

Example 3.4 (Town Offices) The district includes seven towns. The town offices are evaluated by wage costs (mill. CzK per year), operating costs (mill. CzK per year), value of municipality property; and by the number of inhabitants as is shown in the following table

Office	A	B	C	D	E	F	G
Wage Costs	3,5	3	2,8	4	3,8	3,6	3,9
Operating Costs	1,5	1,4	1,6	1,7	1,3	1,25	1,8
No. of Inhabitants	1020	900	1200	1300	1100	800	1150
Property	50	52	48	55	53	46	54

Example 3.5 (Solution to Town Offices) Since, we need to evaluate seven units, we must construct seven models. The primary model for the first town, unit A is as follows.

$$\begin{aligned}
e_1 = 1020u_{11} + 50u_{21} &\rightarrow \max \\
3,5v_{11} + 1,5v_{21} &= 1 \\
-3,5v_{11} - 1,5v_{21} + 1020u_{11} + 50u_{21} &\leq 0 \\
-3v_{11} - 1,4v_{21} + 900u_{11} + 52u_{21} &\leq 0 \\
-2,8v_{11} - 1,6v_{21} + 1200u_{11} + 48u_{21} &\leq 0 \\
-4v_{11} - 1,7v_{21} + 1300u_{11} + 55u_{21} &\leq 0 \\
-3,8v_{11} - 1,3v_{21} + 1100u_{11} + 53u_{21} &\leq 0 \\
-3,6v_{11} - 1,25v_{21} + 800u_{11} + 46u_{21} &\leq 0 \\
-3,9v_{11} - 1,8v_{21} + 1150u_{11} + 54u_{21} &\leq 0 \\
u_{j1} &\geq 0,\ j = 1,2, \\
v_{i1} &\geq 0,\ i = 1,2.
\end{aligned}$$

Results:

Variable	Value
e_1	0,909864
v_{11}	0,135
v_{21}	0,3516
u_{11}	0,0003
u_{21}	0,012

Therefore, we observe that the first town, unit A, in ineffective, since the value of e_1 is less than 1. Resulting weights state the benefit of each criterion to the resulting efficiency. For unit H, we can compute a benefit of i-th input as 3.3

$$\frac{x_{iH}v_{iH}}{\sum_{i=1}^{m} x_{iH}v_{iH}}, \tag{3.3}$$

similarly, for j-th output we obtain 3.4

$$\frac{y_{jH}u_{jH}}{\sum_{j=1}^{n} y_{jH}u_{jH}}. \tag{3.4}$$

For the unit A and its two output we have the following benefits:

$$\frac{3{,}5\cdot 0{,}135}{3{,}5\cdot 0{,}135+1{,}5\cdot 0{,}3516} = 47\%,$$

$$\frac{1{,}5\cdot 0{,}3516}{3{,}5\cdot 0{,}135+1{,}5\cdot 0{,}3516} = 53\%,$$

The same for its outputs:

$$\frac{1020\cdot 0{,}0003}{1020\cdot 0{,}0003+50\cdot 0{,}012} = 34\%,$$

$$\frac{50\cdot 0{,}012}{1020\cdot 0{,}0003+50\cdot 0{,}012} = 66\%.$$

to propose to unit A how to improve its input to become effective, we need to solve the dual model for this unit. (Or, we can get the solution from the sensitive analysis report from SW.)

$$\begin{aligned}
z_1 &\rightarrow \min\\
3,5z_1 - 3,5\lambda_{11} - 3\lambda_{21} - 2,8\lambda_{31} - 4\lambda_{41} - 3,8\lambda_{51} - 3,6\lambda_{61} - 3,9\lambda_{71} &\geq 0\\
1,5z_1 - 1,5\lambda_{11} - 1,4\lambda_{21} - 1,6\lambda_{31} - 1,7\lambda_{41} - 1,3\lambda_{51} - 1,25\lambda_{61} - 1,8\lambda_{71} &\geq 0\\
1020\lambda_{11} + 900\lambda_{21} + 1200\lambda_{31} + 1300\lambda_{41} + 1100\lambda_{51} + 800\lambda_{61} + 1150\lambda_{71} &\geq 1\\
50\lambda_{11} + 52\lambda_{21} + 48\lambda_{31} + 55\lambda_{41} + 53\lambda_{51} + 46\lambda_{61} + 54\lambda_{71} &\geq 5\\
\lambda_{k1} &\geq 0,\\
k = 1, 2, \ldots, 7.
\end{aligned}$$

Variable	Value
z_1	0,909864
z_1	0,909864
λ_{11}	0
λ_{21}	0,3529
λ_{31}	0,2234
λ_{41}	0
λ_{51}	0,3948
λ_{61}	0
λ_{71}	0

From the results, we can see that the peer units for the unit A are units B, C and E (second, third and fifth town in the district), because their variables λ_{21}, λ_{31}; and λ_{51} differ from zero. The variable λ_{21} corresponds to the second unit (according to the first index) and so on. To get the optimum inputs, we use these values and inputs of peer units in the following way, for the first input of unit A:

$$x^{'}_{11} = \lambda_{21}x_{12} + \lambda_{31}x_{13} + \lambda_{51}x_{15} = 0{,}3529\cdot 3 + 0{,}2234\cdot 2,8 + 0{,}3948\cdot 3,8 = 3{,}1845.$$

For the second input of unit A:

$$x'_{21} = \lambda_{22}x_{22} + \lambda_{32}x_{23} + +\lambda_{52}x_{25} = 0{,}3529 \cdot 1,4 + 0{,}2234 \cdot 1,6 + +0{,}3948 \cdot 1,3 = 1{,}3648$$

So, the unit A should decrease its wage costs from 3.5 to 3.1845 and its operating costs from 1.5 mill. CzK to 1.3648 mill. CzK to become effective.

4 Project Management with CPM/PERT Methods

Many people, managers require to plan projects. It means, they must coordinate a lot of activities, a lot of workers. They need to optimize the time schedule and cost of the project. Some of the activities depend on each other (in some way), some activities are independent on other ones. The typical dependence is in the way, that it is given, that some activity cannot start before the other activity is finished. (For example, if we want to paint our house, first, we must remove or cover furniture.) Or, we have blue-collars, who work on one activity, so they cannot at the same time do on the other one. The next example of the possible dependence is a condition that two activities must run parallel (for example, we should serve meal and drinks for dinner at one time).

If we propose a project, we are interested in the following questions:

- What is the shortest possible time in which we can finish the project and how to manage it?
- How many people do we need to finish the project in the time?
- What is the cost of the project? Is it possible to optimize the cost of the project?

In the first part of this chapter, we answer the first question. The other questions we respond in other parts.

First, if we want to analyze any project, we must define the economic problem. In the case of projects, we need to distinguish, what we want to succeed, what are the activities, which we require to finish. The manager of the projects must settle the time necessary for each activity and the relationships among activities.

4.1 Project Network

If the economic problem is well-defined, we can start with the construction of the mathematical problem. There are exist several approaches on how to formalize the economic problem in a mathematical way.

Here, we will display the problem in the graph. If we want to present the project as a graph, in fact, we construct an **oriented network**, it is called a **project network**.

In fact, there two kinds of project networks. The first one is the **activity-on-arc (AOA) project network**. In this case, each activity is represented by an arc; nodes are used to separate activities from each of its immediate predecessors.

The second type of project networks are the so-called **activity-on-node (AON) project networks**. Where each activity is represented by a node; arcs are used just to show the precedence relationships among the activities.

From now on, we will use only the activity-on-node projects network. One of the reason, why we chose this way is that in such a case the network presentation is unique.

Hence, let us describe the construction of this type of project network. The network is such type of a graph, which starts by one node and ends by one node, too. The starting node is a node, where the beginning of the project is, it displays a situation when no activity has started yet The last node is the node, where all activities are already finished. Between these two nodes, there are all other nodes which present all activities, one node presents one activity. Therefore, in case of a project which consists of 5 activities, it is presented by a project network which contains 7 nodes – first one, last one and five nodes between them (= 5 activities must be done).

4.2 Critical Path Method (CPM)

Now, let us solve the problem of the shortest possible duration of the project. This problem we solve in the following steps.

- We display the project network.
- We identify the **earliest start time (EST)** and **earliest finish time (EFT)** for each activity, each node.
- We identify the **latest finish time (LFT)** and the **latest start time (LST)** for each activity.
- We identify the **critical path** of the project and critical activities.
- For non-critical activities, we identify

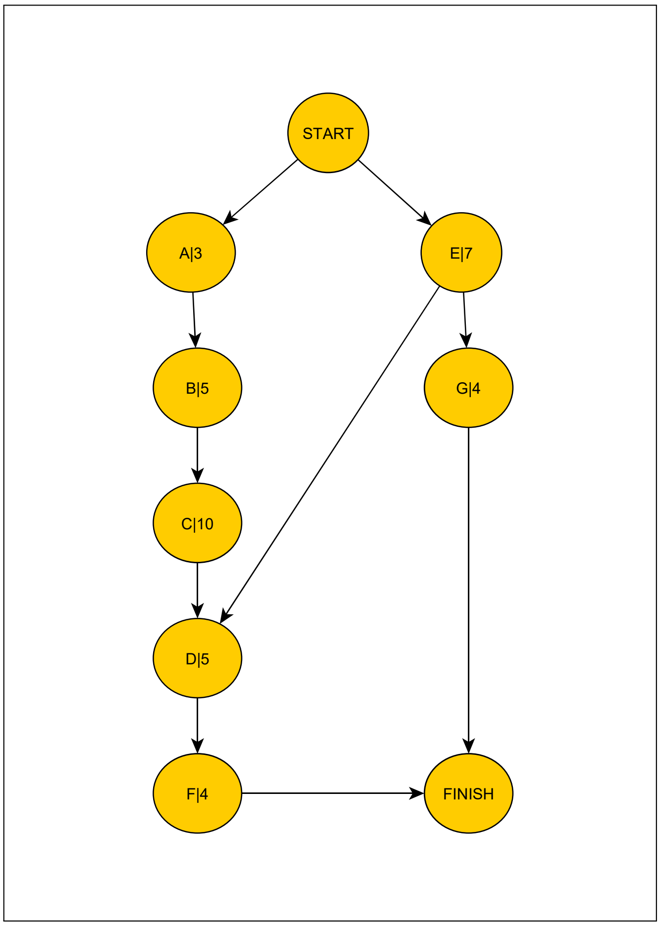
Let us consider the following example to explain how to solve these problems.

Example 4.1 REP company was asked to reconstruct the house of NH company. NH company identifies works, which is necessary to do and REP company estimated the duration of each activity. To plan the project it was also necessary to identify predecessors for each activity, the boss of REP company decided, which activities had to be finished before the beginning of others. All information is provided in the following table.

activity	**description**	**duration (days)**	predecessors
A	eviction of technic equipment	3	–
B	eviction of furniture	5	A
C	floor repairs	10	B
D	interior plumbing	5	E
E	exterior plumbing	7	–
F	interior painting	4	D
G	exterior painting and fixtures	4	E
H	Install the flooring	4	F

The first step to solve this problem is the displaying of the project network. As we wrote above, the network of this project contains 10 nodes (the project has 8 activities and start and finish nodes). Oriented arcs show the relations between activities – for each node (activity) the incoming arcs go from all immediate predecessors and outgoing go to immediately followers.

The project network for our example problem is as follows.



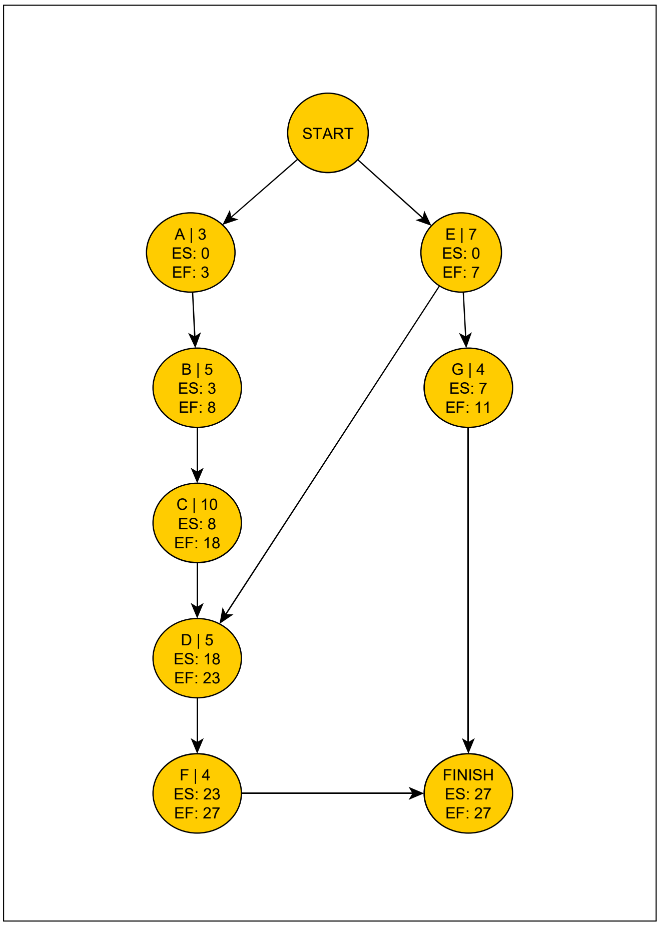
The second step is to identify the earliest start time and the earliest finish time for each node – each activity. Let us explain **Earliest Start Time Rule**. **The earliest start time of activity (ES)** is the earliest time, when the activity can start. Every activity can start when all its predecessors are finished, so the earliest start time of the activity is equal to the largest of **the earliest finish times** of its immediate predecessors. In symbols, ES is the largest EF of the immediate predecessors.

The earliest finish time of the activity is equal to the earliest start time plus the estimated duration of the activity (it is the earliest time when the activity can finish, hence we suppose that if it starts at the earliest start and it takes the estimated duration).

To set up the earliest start times and the earliest finish time we go from the beginning of the project. First, we set the earliest start time and the earliest finish time for the node START – we suppose to start at time zero (and no activity is at this node, so the duration of the activity on the node is equal to zero), so $ES = EF = 0$. Then we continue with activities which have no predecessor and set their $ES = 0$. If we know the earliest start time of a node we can set up the earliest finish time of the node:

$$EF := ES + d_a,$$

where d_a is the estimated duration of the activity.
We continue from the beginning to the end to finish the setting of. We get the following network.



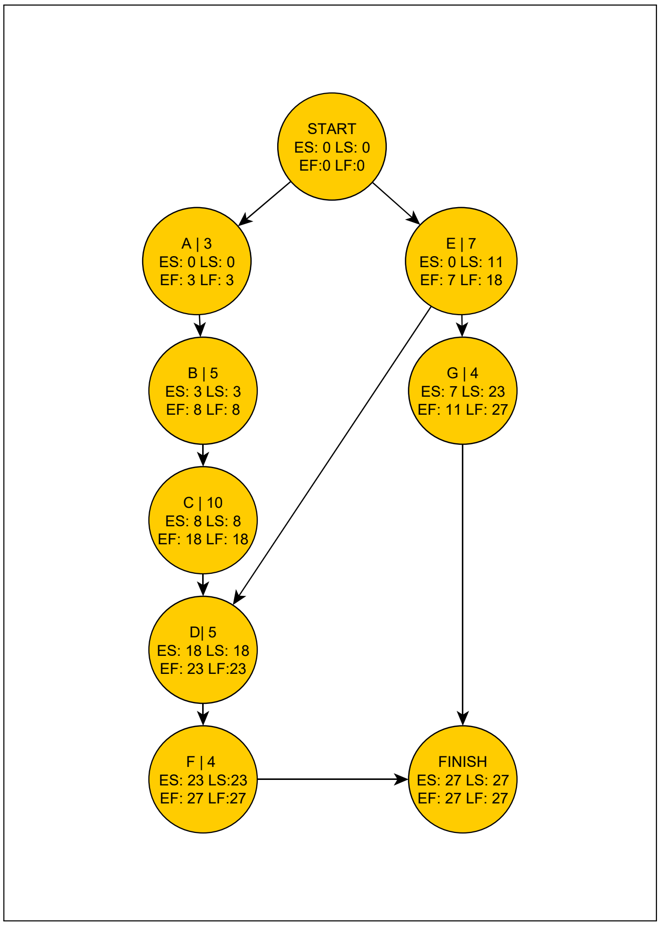
If we know ES and EF for all activities, we know the shortest possible duration of the project (it is equal to the earliest start of FINISH node). The following step is the identification of **critical activities**. Critical activities are all activities which are on the **critical path**. Activities which must start at their earliest possible starts to finish the project in the shortest possible time. To identify critical path we must compute for all activities their **latest finishes (LF)** and **latest starts**. The activities which latest starts coincide with their earliest starts are the critical ones.

To get the latest finishes we go from the end of the project to the beginning. First, we put the latest finish of the node FINISH to be equal to its earliest possible finish (it means we want to finish the project as early as is possible).

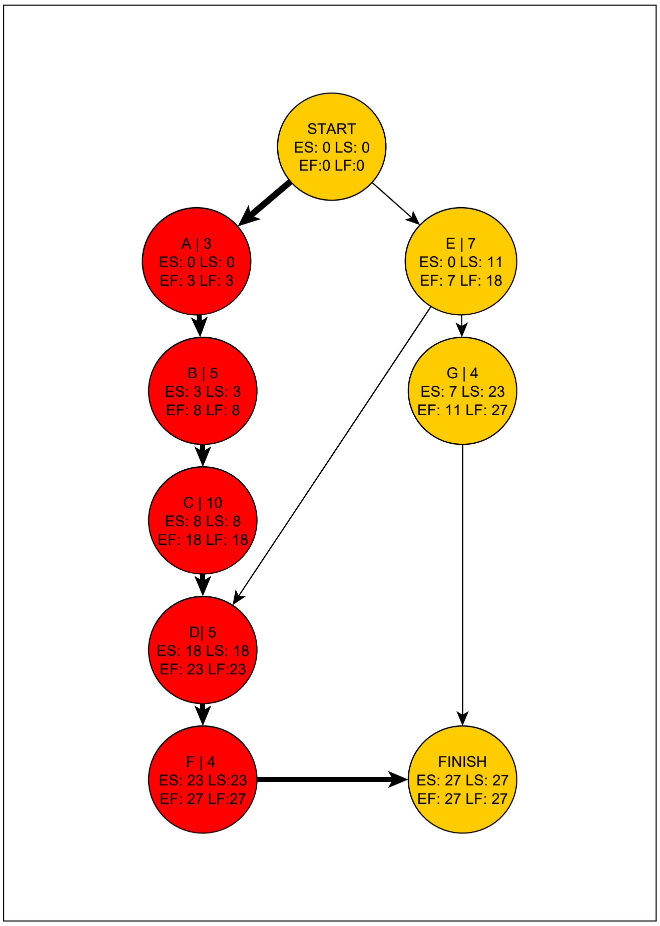
When we know the latest finish of the node – activity, then we can compute the latest start time for the activity – the activity must start in such a time that it finishes at the latest finish time, hence

$$LS := LF - d_a,$$

where d_a is the duration of the activity.
We can set the latest finish of the activity if we know the latest start times of all the following activities. The LF of the activity is the smallest of all followers LSs (the activity must be finished when any following activity starts).
For our example, we get the following graph.



Now, we can identify critical activities and the critical path. Critical activities are activities which must start at the earliest start time to finish the project in time. So, critical activities are such that their earliest start time is equal to their latest possible start time. In our example we have five critical activities:
$A,\ B,\ C,\ D,\ F$ – are on the so-called critical path.
In the following graph, we display it in red.



This method we now applied to find the shortest possible duration of the project is called the **Critical Path Method (CPM)**.

As it was mentioned above the activities on the critical path must begin as soon as possible. However, what about the activities which are not on the critical path – in our example activities E and G. For the activity E we can see that the earliest start is at 0 and the latest start is at 7, hence there is a slack of 5 days for this activity. Generally, we can get the slack for activity as

$$slack = LS - ES$$

for each activity (so the slack for activities on the critical path is equal to zero).

4.3 CPM with Excel

To identify the critical path, we can also use Excel. First, we set the activities, their durations and their predecessors in the Excel sheet, see picture 4.1.

Figure 4.1: CPM with Excel

Activity	duration	predecessors
Start	0	
A	3	
B	5	A
C	10	B
D	5	C,E
E	7	
F	4	D
G	4	E
Finish		

In the second step, we start with the computation of the earliest start for each activity and the earliest finish of each activity. Generally, the earliest start of every activity is the maximum from the latest starts of each immediate predecessors. The earliest finish of each activity is equal to the earliest start of this activity plus the duration of the activity, for more detail see the following picture 4.2. So, we set the earliest possible starts and the earliest possible finishes for all activities. The duration of the whole project is the highest value of the latest possible finish. In our example, we can see that the duration of the project is equal to 27.

Figure 4.2: CPM

Activity	duration	predecessors	ES	EF	LS
Start	0		0	0	
A	3		0	=D3+B3	
B	5	A	=E3	8	
C	10	B	8	18	
D	5	C,E	=MAX(E5;E7)		
E	7		0	7	
F	4	D	23	27	
G	4	E	7	11	
Finish			=MAX(E8;E9)		

Now, we need to state the latest possible finishes and the latest possible starts for each activity, we begin at the end and go to the start. If we want to finish the project in the shortest possible time, we have to finish the latest activities at the time of the duration of the project. The latest possible start of each activity is equal to the latest possible finish abstracted the duration of the activity. The latest possible finish of the activity is equal to the smallest value of the latest possible starts of each immediate predecessors, for more details see the following picture **??**.

4.4 PERT Method

In the previous part, we used fixed durations of activities in our project. We supposed that we were able to estimate the real duration of all activities. However in real-life problems, we usually only estimate the real duration. Hence, we would consider durations of activities to be stochastic, more precisely to be random variables with some characterization.
A random variable is characterized by its distribution law. The times necessary for activities are usually supposed to follow a beta distribution. Beta distribution law has three parameters (m, o, p) as follows:

m – most likely estimate (m), it is the estimate of the most likely value of the duration, o – optimistic estimate (o), the estimate of the duration under the most favorable conditions,

p – pessimistic estimate (p), the estimate of the duration under the most unfavorable conditions.

Usually, the companies are able to estimate these three values for all activities. Why do we consider the beta distribution instead of usually used normal distribution. In the case of the activity duration, the beta distribution is more suitable. There are the following reasons:

- beta distribution is bounded – the values of the random variable are between p and o,
- beta distribution can be asymmetric – the asymmetry is given by the position of m between p and o (if m in half between p and o then the distribution is symmetric).

It is well-known from courses on Probability Theory that the sum of independent normally distributed random variables is normally distributed random variables. Unfortunately, it is not known the distribution of the sum of independent beta distributed random variables. So, if we suppose beta distributed duration of activities in the project, we identify a critical path then we do not know the distribution of the duration of the whole project. So, there are two main ways how to handle this problem. One of them (which is not included in this subject) is using simulation technic to estimate the distribution of the project duration. The second one is to suppose that the Central Limit Theorem takes a place. Let us recall in short some basic condition for application of Central Limit Theorem:

- the durations of activities are independent random variables,
- there is enough (more than 40) activities on the critical path,
- activities which are not on the critical path are not important.

The first two conditions are important conditions for CLT taking place. In the case of dependence among activity durations, CLT would fail. Also in the case of a small number of activities, the CLT cannot be applied.
What is the meaning of the third condition? In fact, if we apply CLT in this way, we do not take into account the non-critical activities. It does not matter only in the case when these activities do not play an important role. When is this condition violated? Let us suppose for example the following example. Let us have two parallel ways in the graph, one of them is critical and the second is not (in the sense of comparison of expected duration). However, especially in the case, when there is no important difference between the expected times of these

two ways and if the variance of the time of non-critical way is bigger than the variance of the critical one, then in fact, in realizations, our "non-critical way" can be run as a critical. So, the usage of CLT fails (this way is not included). The above-mentioned problems are the main disadvantages of using CLT (PERT) methods. On the other hand, there is no other easy method to estimate the duration of the stochastic project (except simulation technics).

To do (??)

So, let us suppose that it is possible to apply CLT, hence we will apply the PERT method in our case. What are the steps of the PERT method?

PERT method:

- For each activity, computation of expected duration and standard deviation of duration from entered values.
- CPM methods applied for expected durations.
- Identification of critical path, estimation of the expected duration of the project and its standard deviation.
- Probability computations.

To compute the expected duration and standard deviation for each activity, we apply a well-known (from Probability theory) formula, which set that for β-distribution:

$$\mu = \frac{p + 4m + o}{6},$$

and

$$\sigma^2 = \left(\frac{o - p}{6}\right)^2.$$

Surely, all of these computations, we can do with Excel, see the picture **??**.

Figure 4.3: PERT with Excel

Activities	predecessors	pesimistic	most likely	optimistic	expected	variance
Start			0			
A		2	3	4	=(C3+4*D3+E3)/6	
B	A	3	5	7	5	=((E4-C4)/6)^2
C	B	7	10	11	9,666667	0,444444
D	C,E	4	5	9	5,5	0,694444
E		5	7	9	7	0,444444
F	D	4	4	4	4	0
G	E	1	4	6	3,833333	0,694444
Finish						

As a next step, we apply expected activity durations to identify the critical path and critical activities.
To compute the expected duration of the project, we sum the expected durations of all critical activities, it gives us the expected duration of the project.
However, in this case, we know that the activity durations are random variables, hence the project duration must be a random variable, too. We apply the CLT and suppose that it follows a normal distribution with expected value the expected duration of the project and with variance equal to the sum of critical activities variances.
If we know the distribution of the random duration, we can do some computations.
We can be interested in the following questions.

- What is the probability that the project finish early than in time XY?
- What is the duration of the project which will be exceeded with probability 5%?

For a better idea about the answers to these questions, we can draw a graph of the density of normal distribution. From this graph, we can easily see the idea of the answers, see the graph **??**.
We suppose normal limit distribution. The normal distribution is used very often in many practical applications and it has a lot of useful properties. One of them is the so-called Law of 1-2-3 standard deviations. What is it? This law says that the probability of random result to be mean plus or minus one standard deviation is equal to 0.67 (does not matter, what the mean is or what the standard deviation is). In case of the result between mean minus two standard deviations and mean plus two deviations, the probability is 0.95 and for three deviations, we get 0.995.
In other words, we know, that 2/3 of observations is equal to mean +- standard deviation. With probability 0.95 the distance of the observation from the mean is less than two standard deviations. Look at the graph for more details.
To answer the questions about probabilities exactly, we can use for example Excel (probability calculator). More precisely, the function NORMDIST and NORMINV. In the function NORMDIST, we set the mean, standard deviation, and x and it returns the probability of the value less than or equal to x of a random variable following a normal distribution with entered parameters.
In the function NORMINV, we set mean, standard deviation and probability and it returns the x such that the random variable following a normal distribution with entered parameters is less than or equal to x with entered probability.

4.5 Crashing

Let us suppose the following problem. We know the duration of each activity and the cost of the activity. We know also the cost of each day of the duration of the project (office rental, secretary, and so on). It is also known that it is possible to crash some of the activities – to finish them in a shorter time, however, if we want to do it, we have to pay more (extra bonus for overtime, overwork, extra workers, special technology and so on) for their realization. On the other hand, it is possible that the whole project will be cheaper – we save the cost for the project. If we crash some activity on the critical path, then we save some money because we save some days of the project – more precisely, we speak about **direct and indirect costs**.

Let us recall what do we mean by direct and indirect costs. Direct costs are that can be directly attributed to a specific activity, e.g. labor, raw materials, and equipment rental costs. Indirect costs are costs that cannot be directly attributed to a specific activity, they are connected with the whole project, e.g. management, general administration, rental and utility costs.

Direct cost grows up if we crash the activity (we need more people, special material, bonuses and so on), on the other hand, indirect cost grow up with the length of the project, so if the whole project is crashed then the indirect cost is reduced.

The question is, what is the cost-optimal duration of the project.

First, let us introduce a prototype example.

Example 4.2 (ALEA comp.) ALEA comp. needs to finish a planned project within 12 months. The project has 4 activities – A, B, C, D – with following properties. The project manager found out that it is not possible to finish the project in time with given durations of activities. So, it would be necessary to crash any activities to be able to finish the project on time. Therefore, the project manager wrote down also the shortest possible duration of each activity (in months) and estimate the cost for each activity in such a case. Which activities should be crashed to finish the project in time with minimal cost? It is known that indirect cost is $6,000 per month.
Is it better to finish the project in time or to pay a penalty $8000?

Act.	Pred.	Normal time	Crash time	Normal cost.	Crash cost
A		8	5	$ 25,000	$ 40,000
B		9	7	$ 20,000	$ 30,000
C	A	6	4	$ 16,000	$ 24,000
D	B	7	4	$ 27,000	$ 45,000

Example 4.3 (Solution to ALEA comp.) When all activities take its normal durations the project will take 16 months. The critical activities are B and D.

Total cost is $184,000 (indirect cost 16·6 = 96 and direct cost 25+20+16+27 = 88). If we want to finish the project in shorter time, we need to decide which activities we will crash and what will be the price for it (on the other hand we save on indirect cost).

When we want to solve such a problem, we first need to know which activities is possible to crash and with what is the cost per week saved. Hence, typically, we prepare a table, where we denote, how can be the activity crashed and we estimate for each activity the cost per one week saved using the following formula:

$$\frac{\text{crash cost} - \text{normal cost}}{\text{normal duration} - \text{crach duration}}.$$

Example 4.4 (Solution to ALEA comp.) Let us do it for Prototype example, we put it into the following table.

Activity	Maximal reduction time (months)	cost per 1 month saved
A	3	$ 5,000
B	2	$ 5,000
C	2	$ 4,000
D	3	$ 6,000

To optimize the cost we can use the way which is introduced in the following or we apply methods of LP.

The first method shows us the principle of optimization, on the other hand, it is suitable just for small problems. With a higher number of activities, it is more complicated.

The algorithm of the method is very easy.

1. Use CPM with the normal duration of the activities to identify the critical path. Compute direct, indirect, and total costs in this case.
2. Choose the activity on the critical way (one activity at each critical paths in case of more than one critical paths) with the smallest cost per one week saved which is still possible to crash.
3. Crash the chosen activity(ies) and compute direct, indirect, and total costs in this case.
4. Continue with repeating steps 2 and 3 till it is possible to crash or till the total cost goes down or till we catch the duration of the project we need.

This solution can be written in the table, where we write down all necessary information.

Example 4.5 (Solution to ALEA comp.) The table for Prototype example follows.

Total time	cr. act.	Δt	Δc	indirect c.	direct c.	total cost
16	-	-	-	\$ 96,000	\$ 88,000	\$ 184,000
15	B	1	\$ 5,000	\$ 90,000	\$ 93,000	\$ 183,000
14	B	1	\$ 5,000	\$ 84,000	\$ 98,000	\$ 182,000
13	D	1	\$ 6,000			
	C	1	\$ 4,000	\$ 78,000	\$ 108,000	\$ 186,000
12	D	1	\$ 6,000			
	C	1	\$ 4,000	\$ 72,000	\$ 118,000	\$ 190,000

The cheapest way is to realize the project in 14 month in the cost of \$ 182,000. If we need to finish the project within 12 months, the cost will be \$ 190,000.

Remark. We can see from the table, that to get the duration of the project from 13 to 12, it is necessary to crash two activities. If the project takes 13 months, there are two critical paths, so we need to crash both of them and it is not possible (or too expensive) to do it by crashing only one activity.

Linear optimization in project crashing

The aim of the optimization is typically to find the least expensive way of crashing activities to finish the project in a deadline or to find the cheapest way of the realization of the project.
In the words of linear optimization, we require to minimize a cost under some restrictions; i.e. subject to some constraints. Let us focus on Prototype example and show the construction of the LO-model.

Example 4.6 (Solution to ALEA comp.) First, let us choose the variables. Put y_i for the time when the activity i will be finish and y for the time when the whole project will finish. The other variables are the crashing ones – set x_i for the reduction in the time of the activity i due to the crashing of this activity.
Let us assume that we are interested in the cheapest possible way how to complete the project. Then we can formulate the problem:
The objective function presents the total cost. The total cost is the sum of direct costs in normal time for all activities, crashing costs, and indirect costs per project (depending on the duration of the project):

$$\min 5000x_A + 5000x_B + 4000x_c + 6000x_D + 6000y + 88000,$$

subject to

$$
\begin{aligned}
y_A &\geq 8 - x_A \\
y_B &\geq 9 - x_B \\
y_C &\geq y_A + 6 - x_C \\
y_D &\geq y_B + 7 - x_D \\
y &\geq y_C \\
y &\geq y_D \\
x_A &\leq 3 \\
x_B &\leq 2 \\
x_C &\leq 2 \\
x_D &\leq 3 \\
x_A, x_B, x_C, x_D &\geq 0.
\end{aligned}
$$

The constraints ensure that predecessors finish in time, respectively the following activities start after finishing predecessors.
In the situation when we need to finish the project within some deadline, we can add a constraint, for example

$$y \leq 12.$$

This model can be solved by Excel, for more detail see the attached file.