

	Notations	Definitions
Sets	$I$	set of users
	$V_1$	set of potential facility on level 1
	$V_2$	set of potential facility on level 2
Parameters	$f_{j_1}$	fixed cost for opening facility $j_1 \in V_1$
	$f_{j_2}$	fixed cost for opening facility $j_2 \in V_2$
	$c_{ij_1j_2}$ $i \in V_1, j_2 \in V_2$	price of supplying user $i$ through the network $j_1j_2$ , $j_1 \in V_1, j_2 \in V_2$
Variables	$y_{j_1}$	binary variable, decision for opening facility $j_1 \in V_1$
	$y_{j_2}$	binary variable, decision for opening facility $j_2 \in V_2$
	$x_{ij_1j_2}$ $(j_1, j_2)$	binary variable, decision for assignment user $i$ to pair $(j_1, j_2)$

TUFLP(Two-level uncapacitated facility location problem) is a problem consisting of a set of users  $I$  and two levels of service facilities  $V_1$  and  $V_2$ . It is assumed that all service facilities are uncapacitated with fixed costs  $f_{j_r}$  for establishing service facilities  $j_r$ , for  $r = 1, 2$ , as well as transportation costs  $c_{ij_1j_2}$  to user  $i$  through a pair of service facilities  $j_1, j_2$ . The problem consists of determining which service facility to open at which level so that each user is served by a pair of open service facilities  $(j_1, j_2)$  minimizing the total cost. Binary variables  $y_{j_r}$  are introduced which are equal to 1 if and only if the service object  $j_r \in V_r$  is open, otherwise they are equal to 0,  $x_{ij_1j_2}$  which are equal 1 if the user  $i$  is assigned to pair  $(j_1, j_2)$ , otherwise they are equal to 0. Then the TUFLP problem can be formulated as follows:

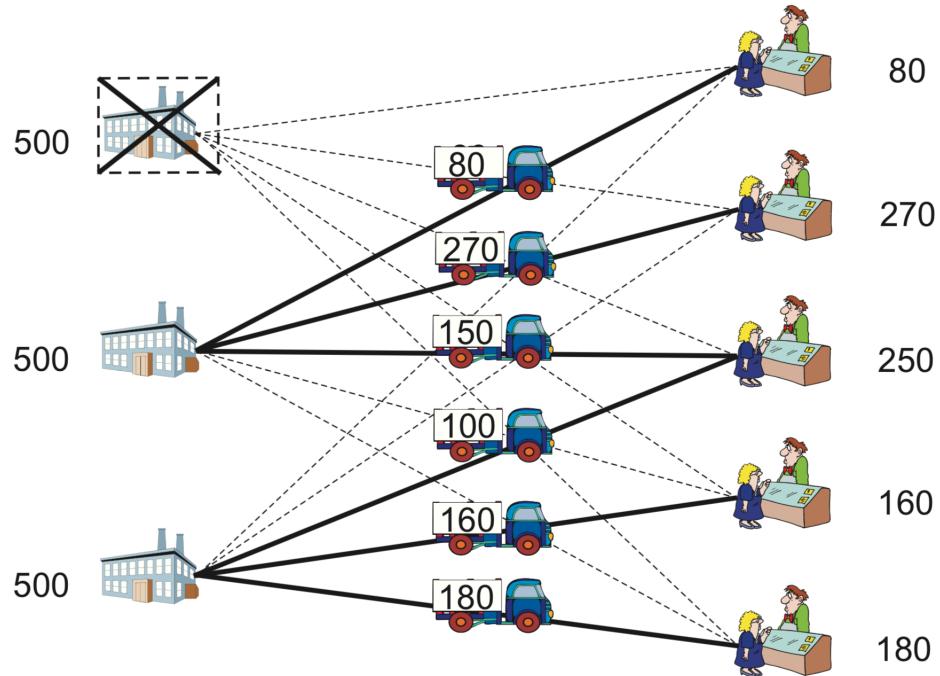
$$\begin{aligned}
& \min \sum_{i \in I} \sum_{j_1 \in V_1} \sum_{j_2 \in V_2} c_{ij_1j_2} x_{ij_1j_2} + \sum_{r=1}^2 \sum_{j_r \in V_r} f_{j_r} y_{j_r} \\
& s.t. \sum_{j_2 \in V_2} \sum_{j_1 \in V_1} x_{ij_1j_2} = 1 \quad i \in I
\end{aligned} \tag{1}$$

$$\sum_{j_1 \in V_1} x_{ij_1 j_2} \leq y_{j_2} \quad i \in I, \quad j_2 \in V_2 \quad (2)$$

$$\sum_{j_2 \in V_2} x_{ij_1 j_2} \leq y_{j_1} \quad i \in I, \quad j_1 \in V_1 \quad (3)$$

$$x_{ij_1 j_2} \in \{0, 1\} \quad i \in I, \quad j_1 \in V_1, \quad j_2 \in V_2 \quad (4)$$

$$y_{j_r} \in \{0, 1\} \quad j_r \in V_r, \quad r = 1, 2 \quad (5)$$



Solution representation:

The solution to the problem is presented as a two-segment binary sequence of length  $|V1| + |V2|$ , the first part is of length  $|V1|$ , and the second part is part of the length  $|V2|$ . The solution is generated by  $y_{jr} = 1, r = 1, 2$  if resource  $j_r, r = 1, 2$  is established and  $y_{jr} = 0, r = 1, 2$  if resource  $j_r, r = 1, 2$  is not established. The probability that a resource is established at the first level is 0.35 , and at the second level 0.25. The neighborhood  $N(s)$  of the solution  $s$  defined in this way consists of all the solutions  $s_0$  which are obtained as follows: If it is random the selected resource at the first level is established, it is turned off, and vice versa, if it is turned off, it is established, in the same way handles a randomly selected resource at the second level.

Matrix sort:

A matrix of dimension  $|V1| \times |V2|$  is constructed for each user  $i \in I$  with user supply prices and through par of service objects  $(j_1, j_2)$  so that each represents the price for the pair  $(j_1, j_2)$ . When calculating the value of objective function, especially the value of the sum  $\sum_{i \in I} \sum_{j_1 \in V_1} \sum_{j_2 \in V_2} c_{ij_1j_2} x_{ij_1j_2}$  the minimum price among the established resources by searching for the minimum element of the matrix by those types and columns that indicate they are resources established. Due to the intensive calculation of the value of the objective function before the actual application of the algorithm (immediately after loading data) the price matrix for each user  $i \in I$  is sorted ascending into a single array, with the original matrix indexes to know which resources are established. In this way, the total calculation time of the objective function is reduced, as well as the total execution time of the entire algorithm, because the first element of the sequence is taken for which resources on both the first and second levels are established, and further search of the array stops because all elements are after of the selected element where both resources are eventually established, i.e. with a higher cost of supplying user  $i$ .