# Remarks on 0-1 Optimization Problems with Superincreasing and Superdecreasing Objective Functions

Marian Chudy

**Abstract** The set of particular 0-1 optimization problems solvable in polynomial time has been extended. This becomes when the coefficients of the objective function belong to the set of superincreasing or superdecreasing types of sequence. We have defined special superincreasing sequences which we call the nearest up and nearest down to the sequence $(c_j)$ of objective function coefficients. They are applied to calculate the upper and lower bound of optimal objective function value. When the problem needs to compute the minimum of objective function with the superdecreasing sequence $(c_j)$, two cases are considered. Firstly, we have described a type of problem when optimal solution can be obtained directly using a polynomial procedure. The second case needs two phases to calculate an optimal solution. The second phase relies on improving a feasible solution. The complexities of all the presented procedures are given.

## 1 Introduction

The most frequently met formulation of 0-1 optimization problem (PLB) is:

$$\max \sum_{j=1}^{n} c_j x_j \tag{1}$$

subject to

$$\sum_{j \in N_i} a_{ij} x_j \leq d_i, \ \ i = \overline{1,m} \tag{2}$$

$$x_j \in \{0,1\}, \ \ j \in N = \{1,2,\ldots,n\}, \ \ N_i \subset N \tag{3}$$

Marian Chudy
Military University of Technology, Faculty of Cybernetics, Warsaw, Poland
e-mail: mchudy@wat.edu.pl

To refer to the title of paper we are reminded that the sequence $(c_j)$ is called superincreasing when

$$\sum_{i=1}^{j-1} c_i < c_j \text{ for } j = 2,3,\ldots \tag{4}$$

We will consider sequences containing $n$ elements only and assume that for $n = 1$ a sequence is a superincreasing one.

For $m = 1$ and positive $a_{ij}, d_i, c_j$ the problem (1)-(3) becomes knapsack one and its special case with superincreasing parameters was effectively applied in the knapsack-type public key cryptosystem [5]. The decision type of knapsack problem with superincreasing parameters was shown in [4] to be P-complete. Short reference to the 0-1 optimization problems with superincreasing objective functions was presented in [3] and [1].

For two-constraint 0-1 knapsack problem, an exact algorithm was described in [6]. A general knapsack problem can be solved using Sbihi's new algorithm [7].

To the best of our knowledge, very few algorithms solving also PLB, are available. We will focus on these kinds of PLB which are solvable in polynomial time or their solutions can be estimated in polynomial time. Some of these cases were also presented in [2]. Now we attempt to extend this class.

## 2 Superincreasing Sequence and 0-1 Optimization Problem. General Remarks

For further considerations we will enumerate a few useful properties of superincreasing sequences:

1. Each subsequence of a superincreasing sequence is a superincreasing one,
2. Each increasing sequence containing only negative elements is a superincreasing one,
3. Each nondecreasing sequence $(c_j)$ such that $c_1 \neq c_2$ containing only negative elements is a superincreasing one,
4. Nondecreasing finite sequence of nonnegative elements contains some superincreasing subsequence.

**Proposition 1.** *If the problem (1)-(3) satisfies the following assumptions:*

- *the sequence $(c_j)$ is superincreasing and nonnegative,*
- *elements $a_{ij}$ are nonnegative ($a_{ij} \geq 0$),*

*then the optimal solution of the problem (1)-(3) is given by the following procedure*

$$x_j^* = \begin{cases} 1 \quad when \begin{cases} a_{1j} \le d_1 - \sum_{k \in N_j^+} a_{1k} \\ a_{2j} \le d_2 - \sum_{k \in N_j^+} a_{2k} \\ . \\ . \\ . \\ a_{mj} \le d_m - \sum_{k \in N_j^+} a_{mk} \end{cases} & j = n, n-1, ..., 1 \\ 0 \quad otherwise \end{cases} \tag{5}$$

*where $a_j$ is the j-th column of the constraint matrix (2)*

$$d = (d_1, d_2, ..., d_m)^T, N_n^+ = \phi$$

$$N_j^+ = \{k : x_k^* = 1,\ k \in \{n, n-1, ..., j+1\}\}.$$

The proof results from (4) and assumptions.

The complexity of procedure (5) is equal to $f_5(n) \in O(n^3)$. To calculate this function one should observe that calculating each element of $x_j^*$ for $j = n, n-1, ..., 1$ needs $n, 2n, ..., n \times n$ basic operations, respectively. The total sum of these operations gives us that function.

Proposition 1 allows us to solve 0-1 optimization problem in polynomial time, when the assumptions it needs are satisfied.

*Remark.* The following example shows that the assumption $a_{ij} \ge 0$ is significant.

*Example 1.*

$$\max \ x_1 + 2x_2 + 5x_3 + 10x_4$$

subject to

$$\begin{aligned} &x_1 - 2x_2 + x_3 + 8x_4 \le 6 \\ &2x_1 - x_2 - x_3 + 3x_4 \le 2 \\ &x_j \in \{0,1\}, \quad j = \overline{1,4} \end{aligned}$$

The coefficients $c_j$ form the superincreasing sequence $(c_j) = (1, 2, 5, 10)$. The optimal solution is $x^* = (0, 1, 0, 1)$ and optimal objective function value is equal to 12. On the other hand, using procedure (5) we obtain vector $\bar{x} = (1, 1, 1, 0)$ and the objective function value is equal to 8.

*Remark.* A simple example shows that assumption $c_j \ge 0,\ j = \overline{1,n}$ is also important.

*Example 2.*

$$\max \ -2x_1 + 3x_2 + 3x_3 + 5x_4$$

subject to

$$x_1 + 2x_2 + 4x_3 + 6x_4 \le 6$$
$$4x_1 + x_2 + x_3 + 5x_4 \le 5$$
$$x_j \in \{0,1\}, \quad j = \overline{1,4}$$

One can observe that the sequence $(c_j) = (-2,3,3,5)$ is superincreasing. Procedure (5) computes the vector $x = (0,0,0,1)$ and we obtain an objective function value of $(c|x) = 5$. However, the optimal solution is $x^* = (0,1,1,0)$ and the optimal objective function value is equal to $(c|x^*) = 6$.

To continue our considerations we renumber, if necessary, all variables of the 0-1 problem and assume that the sequence $(c_j)$ is **integer**, **nonnegative** and **non decreasing**.

Let us start with the following example.

*Example 3.*

$$\max \; x_1 + 2x_2 + 3x_3 + 5x_4 + 5x_5$$

subject to

$$x_1 + x_3 + x_4 + x_5 \le 2$$
$$x_2 + x_4 + x_5 \le 2$$
$$x_1 + x_3 + x_4 \le 3$$
$$x_j \in \{0,1\}, \quad j = \overline{1,5}$$

The optimal solution is $x^* = (0,1,1,1,0)$. The sequence $(c_j) = (1,2,3,5,5)$ is not superincreasing but we can indicate some superincreasing subsequence of $(c_j)$ which corresponds to some feasible solution. This feasible solution can be obtained by correctly selecting elements of vector $x^* = (0,1,1,1,0)$ which are equal to one.

The vectors we mentioned above are: $x^1 = (0,1,1,0,0)$, $x^2 = (0,1,0,1,0)$, $x^3 = (0,0,1,1,0)$. They correspond to superincreasing subsequence $(c_2,c_3) = (2,3)$, $(c_2,c_4) = (2,5)$, $(c_3,c_4) = (3,5)$ respectively. It leads to the following proposition.

**Proposition 2.** *If the coefficients of problem (1)-(3) are nonnegative, i.e., $c_j \ge 0$, $a_{ij} \ge 0$ for all $i, j$ and there exists a feasible solution $x \ne 0 = (0,0,...,0)$, then there exists a feasible solution $x^1$ having elements $x^1_{j_r} = 1$, $j_r \in N^1 \subset N^+ = \{j : x_j = 1\}$, $x^1_{j_r} = 0$ $j_r \in N \backslash N^1$ that correspond to the superincreasing subsequence $(c_{j_r})$ of the sequence $(c_j)$ which satisfies property IV.*

The proof results from (4), assumptions of this proposition and property IV.
Let us consider two superincreasing subsequences of sequence $(c_j)$:

1. Subsequence $(c_{i_1}, c_{i_2}, ..., c_{i_r})$ which corresponds to a feasible solution $x$

$$x_j = \begin{cases} 1 \text{ for } j = i_k \\ 0 \text{ for } j \ne i_k \end{cases} \quad k = \overline{1,r}, \tag{6}$$

2. Subsequence $(c_{j_1}, c_{j_2}, ..., c_{j_r})$ which corresponds to a feasible solution $y$

$$y_j = \begin{cases} 1 & \text{for } j = j_k \\ 0 & \text{for } j \neq j_k \end{cases} \quad k = \overline{1, r}. \tag{7}$$

**Proposition 3.** *If $c_{i_r} > c_{j_r}$, then such inequality holds $\sum_{j=1}^{n} c_j x_j > \sum_{j=1}^{n} c_j y_j$ i.e., solution $x$ is better than solution $y$. The proof results directly from the definition of the superincreasing sequence.*

One can describe this dependence using subsequences $(c_3, c_4), (c_2, c_3)$ from example 3.

We observe that Propositions 2 and 3 can be applied to improve some given feasible solution.

It results in the fact that for many 0-1 optimization problems we can construct suitable 0-1 optimization problems with superincreasing objective functions and use them to compute, in polynomial time, high quality upper and lower bounds of optimal objective function values.

## 3 Superincreasing Sequence and Upper Bound

To obtain the upper bound of optimal objective function value, we have to introduce several new objects. Denote by:

- $H^n$ – the set of all finite superincreasing integer sequences $(h_j)$, $j = \overline{1, n}$,
- $A^n = \left\{ h \in H^n : h_j \geq c_j, j = \overline{1, n} \right\}$ – the set of finite superincreasing sequences with integer elements no smaller than suitable elements of the sequence $(c_j)$.

Remembering that $(c_j)$ is nondecreasing we form the following definition.

**Definition 1.** A superincreasing sequence $h^* = (h_j^*)$ is called the nearest up to the sequence $(c_j)$ when

$$h^* \in A^n \text{ and } \|c - h^*\| = \min_{h \in A^n} \|c - h\| = \min_{h \in A^n} \sum_{j=1}^{n} \left| c_j - h_j \right| \tag{8}$$

For a given $(c_j)$ we can compute the sequence $h^* = (h_j^*)$ in the following way:

$$h_1^* = c_1 \tag{9}$$

and for $j = \overline{2, n}$

$$h_j^* = \sum_{k=1}^{j-1} h_k^* + 1 \quad \text{when} \quad c_j \leq \sum_{k=1}^{j-1} h_k^* \tag{10}$$

$$h_j^* = c_j \quad \text{when} \quad c_j > \sum_{k=1}^{j-1} h_k^*. \tag{11}$$

We notice that $(h_j^*) = (c_j)$ when $(c_j)$ is a superincreasing sequence.

To compute all elements of $h^*$, the following numbers of basic operations are needed: 0 for $j = 1$, 2 for $j = 2$, 3 for $j = 3$, ..., $n$ for $j = n$, respectively. Hence, the complexity of the procedure (9)-(11) is equal to $f_{h^*}(n) \in O(n^2)$.

The upper bound of optimal objective function value for the PLB is given by

$$\sum_{j=1}^{n} h_j^* x_j \geq \sum_{j=1}^{n} c_j x_j^* \tag{12}$$

where:

- $x = (x_j),\, j = \overline{1,n}$ denotes a feasible solution computed by procedure (5) when we set the sequence $(h_j^*)$ instead of the sequence $(c_j)$ in PLB,
- $x^* = (x_j^*),\, j = \overline{1,n}$ denotes an optimal solution of the problem (1)-(3), under the assumption $a_{ij} \geq 0, c_j \geq 0$.

*Example 4.*

$$\max\ \ x_1 + 4x_2 + 5x_3 + 6x_4$$

subject to

$$\begin{aligned} &x_1 + x_2 + x_3 + 6x_4 \leq 6 \\ &2x_1 + x_2 + x_3 + x_4 \leq 7 \\ &x_j \in \{0,1\}\,, \quad j = \overline{1,4} \end{aligned}$$

The vector $x^* = (1,1,1,0)$ is the optimal solution of this problem and the optimal objective function value $(c|x^*)$ is equal to 10.

According to procedure (9)-(11) the vector $h^* = (h_1^*, h_2^*, h_3^*, h_4^*) = (1,4,6,12)$ is superincreasing and the nearest up to the vector $c = (c_1, c_2, c_3, c_4) = (1,4,5,6)$ which is not superincreasing.

Setting the sequence $(h_j^*)$ instead of $(c_j)$ and applying procedure (5) we obtain a feasible solution of $x = (0,0,0,1)$ and objective function value of $(c|x) = 6$.

The upper bound of the optimal value $(c|x^*) = 10$, based on (12), is equal to $(h^*|x) = 12$.

At this point we should underline a very important fact: procedure (5), in every case, produces an upper bound of optimal function value when we use vector $h^*$ instead of vector $c$. But procedure (5) cannot compute the upper bound without setting $h^*$ instead of $c$. From example 4 results we obtain the same vector $x = (0,0,0,1)$ as when we use procedure (5) and keep the vector $c = (c_1, c_2, c_3, c_4) = (1,4,5,6)$.

## 4 Superincreasing Sequence and Lower Bound

To improve an assessment of optimal objective function value we propose to compute a lower bound of it.

**Definition 2.** Let $(c_j)$ be a non decreasing integer sequence.

A superincreasing sequence $h^o = (h_j^o)$ is called the nearest down to the sequence $c = (c_j)$ when

$$\|c - h^o\| = \min_{h \in B^n} \|c - h\| = \min_{h \in B^n} \sum_{j=1}^{n} \left| c_j - h_j \right| \tag{13}$$

where

$$B^n = \left\{ h \in H^n : \; h_j \le c_j, \; j = \overline{1,n} \right\}$$

For the given $c = (c_j)$ a sequence $h^o = (h_j^o)$ can be computed according to the following procedure:

1. For $n = 1$, $h_1^o = c_1$,
2. For $n = 2$, ; $h_2^o = c_2$,

$$h_1^o = \begin{cases} c_1 & \text{if} \quad c_1 < h_2^o = c_2 \\ c_1 - 1 & \text{if} \quad c_1 = c_2 = h_2^o \end{cases} \tag{14}$$

3. For $n \ge 3$, ; $h_j^o = c_j$, ; $j = \overline{n,2}$ and first element $h_1^o$ needs a special recurrence formula to compute:

$$h_1^{k-2} = \begin{cases} c_1 & \text{for } k = n \\ h_1^{k-1} & \text{if} \quad h_1^{k-1} + \sum_{i=2}^{k} c_i < h_{k+1}^o = c_{k+1}, \\ h_1^{k-1} - A & \text{if} \quad h_1^{k-1} + \sum_{i=2}^{k} c_i \ge h_{k+1}^o = c_{k+1} \end{cases} \quad k = \overline{n-1,2} \tag{15}$$

where $A = (c_{k+1} - (h_1^{k-1} + \sum_{i=2}^{k} c_i)) + 1$

*Example 5.*

$$1. \begin{cases} c = (1,1,1,1,4) \\ h^o = (-2,1,1,1,4) \end{cases} \quad 2. \begin{cases} c = (-1,-1,-1,-1,4) \\ h^o = (-2,-1,-1,-1,4) \end{cases}$$

$$3. \begin{cases} c = (-3,-1,3,4,4) \\ h^o = (-3,-1,3,4,4) \end{cases} \quad 4. \begin{cases} c = (-1,2,5,7,8,8) \\ h^o = (-15,2,5,7,8,8) \end{cases}$$

To evaluate the complexity of computing $(h_j^o)$ for the given $(c_j)$, it will be enough to take into account expression (15). The recurrence structure of this formula leads us to the following evaluation of the complexity: $f_{h^o}(n) \in O(n^2)$.

## 5 Some Useful Properties of $(h_j^*)$ and $(h_j^o)$

1. From Definitions 1 and 2 it follows that if a sequence $(c_j)$ is superincreasing then
$$c_j = h_j^* = h_j^o, \quad j = \overline{1,n}. \tag{16}$$
2. For each non decreasing $(c_j)$ the following inequalities hold
$$h_j^* \geq c_j \geq h_j^o, \quad j = \overline{1,n}. \tag{17}$$
3. If some non decreasing sequence $(c_j)$ satisfies
$$h_j^* = h_j^o, \quad j = \overline{1,n} \tag{18}$$
then $(c_j)$ is superincreasing.
4. For each vector $x$ such that $x \in S = \{x \in E^n : ;\ (2),(3)\text{hold}\}$ we can obtain from (17) the following evaluation
$$(h^*|x) \geq (c|x) \geq (h^o|x). \tag{19}$$
5. The previous properties allow us to formulate
$$(c|x^*) \geq (c|x) \geq (h^o|x) \quad x^*, x \in S. \tag{20}$$
It means that value $(c|x), x \in S$ is not worse a lower bound of $(c|x^*)$ than $(h^o|x)$.

## 6 Superdecreasing Sequence and 0-1 Optimization Problem

Some 0-1 optimization problems have the following form:
$$\min \sum_{j=1}^{n} c_j x_j = \min(c|x) \tag{21}$$
subject to
$$\sum_{j \in N_i} a_{ij} x_j \geq d_i,\ \ i = \overline{1,m} \tag{22}$$
$$x_j \in \{0,1\},\ \ j \in N = \{1,2,...,n\},\ \ N_i \subset N \tag{23}$$
This problem needs a different approach than the approach to (1)-(3).

**Definition 3.** A sequence $(c_j)$ is called a superdecreasing one when
$$c_j > \sum_{i=j+1}^{n} c_i,\ \ j = 1,...,n-1 \tag{24}$$

and for $n = 1$, $(c_j)$ is superdecreasing.

The following properties of a superdecreasing sequence take place:

1. Each subsequence of the superdecreasing sequence $(c_j)$ is superdecreasing,
2. Each of the decreasing sequence $(c_j)$containing only negative elements is superdecreasing,
3. Each of the non increasing sequence $(c_j)$containing only negative elements and satisfying $c_1 \neq c_2$ is superdecreasing,
4. Each of the non increasing sequence $(c_j)$ with only negative elements contains a superdecreasing subsequence.

We are able to select several cases when problems (21)-(23) are easily solvable.

**Proposition 4.** *Consider the problem (21)-(23) with superdecreasing $(c_j)$ and let the following conditions hold: $c_j \geq 0,\ \ a_{ij} \geq 0$ and there exists $j$ such that $a_{ij} \geq d_i,\ \ i = \overline{1,m}$, then an optimal solution has the form:*

$$x^*_{j_*} = 1 \ \textit{and}\ x^*_j = 0 \ \textit{for}\ \ j \neq j_*$$

*when there exists $i$ such that*

$$\sum_{j=j_*+1}^{n} a_{ij} < d_i,\ \ i \in \{1,2,...,m\} \ \ \textit{and}\ \ j_* < n$$

$$j_* = \max\left\{ j : a_{ij} \geq d_i,\ \ i = \overline{1,m} \right\} \tag{25}$$

*or*

$$j_* = n$$

The proof results from Definition 3 and the conditions presented above.

*Example 6.*

$$\min \ \ 10x_1 + 5x_2 + 2x_3 + x_4$$

subject to

$$\begin{aligned} &x_1 + 2x_2 + 7x_3 + 8x_4 \geq 6 \\ &2x_1 + x_2 + 5x_3 + 2x_4 \geq 4 \\ &x_j \in \{0,1\},\ \ j = \overline{1,4} \end{aligned}$$

In this problem we have $j_* = 3, x^* = (0,0,1,0)$ which satisfy all the conditions that Proposition 4 requires.

To obtain an optimal solution using the procedure described in Proposition 4, we need to execute the following numbers of basic operations:

1. At most $n^2$ to compute $j_*$,

2. To check if there exists $I$ such that $];\sum_{j=j_*+1} a_{ij} < d_i,\ \ i \in \{1,2...,m\}$, the worst case takes place for $j_* = 1$ and the number of basic operations equals at most $(n-1)n$.

Hence, the complexity of the procedure is equal to $f_{\min}(n) = O(n^2)$.

**Proposition 5.** *Let us consider the problem (21)-(23) with a superdecreasing* $(c_j)$ *and* $c_j \geq 0,\ \ a_{ij} \geq 0$.

*Assume that there is no* $j$ *such that* $a_{ij} \geq d_i,\ \ i = \overline{1,m}$. *Then the following expressions*

$$x_j^o = \begin{cases} 1 & for\ \ j = n, n-1, ..., j_o \\ 0 & otherwise \end{cases} \tag{26}$$

$$j_o = \max\left\{ j : \sum_{k=j}^{n} a_{ik} \geq d_i,\ \ i = \overline{1,m} \right\} \tag{27}$$

*give us the upper bound* $(c|x^o)$ *of optimal objective function value* $(c|x^*)$ *of the problem (21)-(23). The proof we obtain from (26) and Definition 3.*

*The complexity of this procedure is determined in (27). In the worst case it needs at most* $n^3$ *basic operations. Hence, the complexity equals* $f_{up}(n) \in O(n^3)$.

*Example 7.*

$$\min\ \ 10x_1 + 5x_2 + 2x_3 + x_4$$

subject to

$$\begin{aligned} &x_1 + 2x_2 + x_3 + 8x_4 \geq 6 \\ &2x_1 + 2x_2 + x_3 + 2x_4 \geq 4 \\ &x_j \in \{0,1\},\ \ j = \overline{1,4} \end{aligned}$$

The sequence $(c_j) = (10,5,2,1)$ is superdecreasing.

Using procedure (26), (27) we obtain $j_o = 2$ and the feasible solution $x = (0,1,1,1)$ that gives $(c|x) = 8$. This is not an optimal solution. The optimal solution is $x^* = (0,1,0,1)$ and gives $(c|x^*) = 6$. We can improve the feasible solution $x = (0,1,1,1)$ applying the procedure given below.

**Proposition 6.** *Let* $x^o = (x_j^o),\ \ j = \overline{1,n}$ *be the feasible solution of problem (21)-(23) which was obtained using procedure (26), (27) under the assumptions:* $(c_j)$ *is superdecreasing,* $c_j \geq 0$ *,* $a_{ij} \geq 0$.

*Defining auxiliary parameters:*

$$N_o^+ = \left\{ j : x_j^o = 1,\ \ x_j^o \text{ that satisfies } (26),(27) \right\} = \{n, n-1, ..., j_o\},$$

$$N_{j_o+1}^- = \begin{cases} \phi & when\ \ \sum_{k \in N_o^+ \setminus \{j_o+1\}} a_{ik} < d_i,\ \ i = \overline{1,m} \\ \{j_o+1\} & when\ \ \sum_{k \in N_o^+ \setminus \{j_o+1\}} a_{ik} \geq d_i,\ \ i = \overline{1,m} \end{cases}$$

$$N_j^- = \begin{cases} N_{j-1}^- & when\ \sum_{k\in N_o^+\setminus(N_{j-1}^-\bigcup\{j\})} a_{ik} < d_i,\ \ i=\overline{1,m} \\ N_{j-1}^-\bigcup\{j\} & when\ \sum_{k\in N_o^+\setminus(N_{j-1}^-\bigcup\{j\})} a_{ik} \geq d_i,\ \ i=\overline{1,m} \end{cases} \qquad j=\overline{j_o+2,n}$$

*the optimal solution can be expressed in the following way*

$$x_j^* = \begin{cases} 0 & when\ \sum_{k\in N_o^+\setminus\{j\}} a_{ik} \geq d_i,\ \ i=\overline{1,m} \\ 1 & otherwise \end{cases} \qquad j=j_o+1 \tag{28}$$

$$x_j^* = \begin{cases} 0 & when\ \sum_{k\in N_o^+\setminus N_j^-} a_{ik} \geq d_i,\ \ i=\overline{1,m} \\ 1 & otherwise \end{cases} \qquad j=\overline{j_o+2,n} \tag{29}$$

$$x_j^* = 0\ \ for\ \ j=\overline{1,j_o-1}. \tag{30}$$

The essence of the procedure (28)-(30) relies on reducing to zero, if possible, these elements $x_j^o$ of vector $x^o$ which are equal to one and $c_j$ is large. It is also essence of the proof.

The similarity between (26) and (28), (29) allows us to write $f_{cor}(n) \in O(n^3)$ as the complexity of the vector $x^o$ improving .

In example 7, we can correct $x=(0,1,1,1)$ to the form $x^*=(0,1,0,1)$ using procedure (28)-(30), because the sum of the second and fourth column satisfies (28), i.e. $\begin{bmatrix}2\\2\end{bmatrix} + \begin{bmatrix}8\\2\end{bmatrix} \geq \begin{bmatrix}6\\4\end{bmatrix}$ .

## 7 Conclusions

The results we have obtained are applicable to:

- solving 0-1 optimization problems with a superincreasing and superdecreasing objective function, if the indicated assumptions hold,
- computing upper bounds and lower bounds of optimal objective function value for 0-1 optimization problems under suitable assumptions,
- improving given feasible solution of 0-1 optimization problem using some properties of superincreasing and superdecreasing sequences.

It is worth underlining that all of these procedures are polynomial. The practical application area of 0-1 optimization problems is very broad. There are no reasons to exclude these results from this area.

## References


1. R. Alfonsin: On variations of the subset sum problem. *Discrete Applied Mathematics* **81** (1998) pp. 1–7
2. M. Chudy: Some properties of 0-1 optimization problems. In: *Proceedings of the 11th International Conference on System Modelling Control (SMC)*, pp. 61–67. EXIT (2005)
3. B. Jenner: Knapsack problems for NL. *Information Processing Letters* **54** (1995) pp. 167–174
4. H. J. Karloff, W. L. Ruzzo: The iterated mod problem. *Information and Computation* **80** (1989) pp. 193–204
5. N. Koblitz: *A Course in Number Theory and Cryptography*: Springer-Verlag, New York (1994)
6. S. Martello, P. Toth: An exact algorithm for two-constraint 0-1 knapsack problem. *Operations Research* **51** (2003) pp. 826–835+837
7. A. Sbihi: A best first search algorithm for the multiple-choice multidimensional knapsack problem. *Journal of Combinatorial Optimization* **13** (2007) pp. 337–351