PERGAMON Computers and Mathematics with Applications 37 (1999) 125–133

An International Journal
computers & mathematics
with applications

TABU Search Methodology in Global Optimization

V. V. KOVAČEVIĆ-VUJČIĆ AND M. M. ČANGALOVIĆ
Laboratory for Operations Research
Faculty of Organizational Sciences
University of Belgrade, Yugoslavia

M. D. AŠIĆ
Department of Mathematics
The Ohio State University
Newark, OH 43055-1797, U.S.A.

L. IVANOVIĆ AND M. DRAŽIĆ
Faculty of Mathematics, University of Belgrade, Yugoslavia

Abstract—This paper investigates the application of TABU search methodology in global optimization. A general multilevel TABU search algorithm is proposed. The algorithm is applied to the problem of finding constrained global minima of a piecewise smooth function of the form $\Phi(x) = \max\{\varphi_1(x), \ldots, \varphi_m(x)\}$ subject to box constraints. The tests are performed on a special class of problems of this type arising from the synthesis of radar polyphase codes. It is shown that problems of this type are NP-hard.

Keywords—Global optimization, Heuristic methods, TABU search.

1. INTRODUCTION

In this paper we consider global optimization problems of the type

$$\operatorname*{global\,min}_{x \in X} \Phi(x), \tag{1}$$

where $\Phi : R^n \to R$ is a continuous function on an open set containing X and X is a compact set. A number of methods for global optimization problems have been proposed, both deterministic and nondeterministic (for a comprehensive bibliography see [1] or [2]). Nevertheless, there are no efficient mathematical methods to solve the problem in general. Neither does there exist widely available and efficient software which works on problems of higher dimensions. It is natural then, that in applications heuristic methods are often used. The purpose of this paper is to tackle the global optimization problem (1) with one of new heuristic methodologies—TABU search, which was originally proposed for solving large combinatorial optimization problems.

The organization of the paper is as follows. In Section 2 we describe a general TABU search algorithm for solving problem (1). In Section 3 we apply the algorithm to the problem of finding global minima of a piecewise smooth function arising in finite minimax problems. The developed algorithm is tested on a class of minimax problems appearing in the radar polyphase code design, which is described in Section 4. It is shown that such problems are NP-hard. Section 5 contains the details of numerical experiments and comparison with Monte Carlo and covering techniques.

0898-1221/99/$ - see front matter Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX
PII: S0898-1221(99)00064-4

2. GENERAL MULTILEVEL TABU SEARCH ALGORITHM

As it is well known [3], TABU search is an intelligent search procedure which starts from an initial feasible point x^0 and tries to reach an optimal solution by generating a finite sequence of feasible points $x^1, \ldots, x^k$. At the step k a subset $N'(x^k)$ of the neighborhood $N(x^k)$ of the current point x^k is constructed and the next point x^{k+1} is determined as the best one in $N'(x^k)$. As TABU search allows ascent moves, lists of tabu moves (so called TABU lists) are introduced in order to prevent cycling back to some previous points. Due to ascent moves the procedure avoids traps in local optima.

Here, a TABU search concept is applied to a continuous case—the global optimization problem (1), in order to find a satisfactory approximate solution. Following the general framework in [4] the basic notions of this concept are specified as described below.

(i) The neighborhood $N(x^k)$, $x^k \in X$, is defined in the usual Euclidean way,

$$N(x^k) = \{x \in X \mid \|x - x^k\| \leq \alpha_k\},$$

where $\alpha_k > 0$ is a radius which depends on the iteration number k.

(ii) In our concept several TABU lists can be introduced, but in each iteration only one of them is active. Namely, when x^k is an interior point of X, after performing a move from x^k, a pair (x^k, β_k) enters the currently active TABU list. This pair defines a TABU cube

$$C(x^k, \beta_k) = \{x \in X \mid |x_j - x^k| \leq \beta_k,\ j = 1, \ldots, n\},$$

where β_k, $0 < \beta_k < \alpha_k$, is the cube size which again depends on k. A TABU cube contains points which are forbidden to be visited in a prescribed number of future iterations with the same active TABU list. This number represents the length of the list. A point is tabu in an iteration if it belongs to the TABU region which is the union of all TABU cubes from the currently active TABU list.

(iii) The subneighborhood $N'(x^k)$ is a finite subset of $N(x^k)$ which is generated using a finite set $D(x^k)$ of "good" move directions. The choice of the directions in $D(x^k)$ in general depends on local characteristics of the objective function Φ at x^k. This set should be large enough to give the possibility of directing the search into different parts of X. We assume that all members of $D(x^k)$ are normalized.

Given the set $D(x^k)$, $N'(x^k)$ is generated as follows.

Procedure **Neighborhood** (x^k, α_k).

For each $d \in D(x^k)$, if $x^k + \alpha_k d \in X$, then the point $x^k + \alpha_k d$ is included into $N'(x^k)$. If $x^k + \alpha_k d \notin X$, then the point $x^k + \lambda d$ is included into $N'(x^k)$, where $\lambda > 0$ is such that $x^k + \lambda d \in \partial X$.

(iv) The best neighbor $x^{k+1} \in N'(x^k)$ is determined such that $\Phi(x^{k+1}) = \min \Phi(x)$ among all x satisfying $x \in N'(x^k)$ and x is not tabu.

(v) If either all the members of $N'(x^k)$ are tabu or x^k is a boundary point of X, i.e., $x^k \in \partial X$, then the following random restarting of the TABU search process is applied.

Procedure **Restart** (x^k, α_k).

A move direction d_r pointing into the interior of X is randomly generated and normalized. Then, the smallest integer j for which $x^k + j\alpha_k d_r$ is not tabu, is found. If $x^k + j\alpha_k d_r \in X$, then $x^{k+1} = x^k + j\alpha_k d_r$. If $x^k + j\alpha_k d_r \notin X$, then $x^{k+1} = x^k + \lambda d_r$, where $\lambda > 0$ is such that $x^k + \lambda d_r \in \partial X$.

The step α_k should be chosen to balance two contradictory requirements. It should be sufficiently large in order to provide the search of the entire set X and it should be small enough to attain a satisfactory solution. This balance can be achieved using a multilevel search strategy based on the following ideas: let $A_1 < \cdots < A_w$ be the w possible choices (levels) for the step size

during TABU search procedure. The level A_i is used in a periodical manner, after a prescribed number q_{i-1} of iterations with steps at levels $A_1, \ldots, A_{i-1}$. The level A_i can be forced earlier, if there was no improvement of the objective function in the last u_{i-1} iterations, where $u_{i-1} < q_{i-1}$. If such a strategy is used, it is natural to introduce w TABU lists $T_1, \ldots, T_w$, and thus, w TABU regions $TR_1, \ldots, TR_w$. Let $B_1 < \cdots < B_w$ be the corresponding TABU cube sizes. If $\alpha_k = A_i$, then the i^{th} TABU list is active at the k^{th} iteration and the pair (x^k, β_k) enters the list, where $\beta_k = B_i$. In the sequel, the lengths of $T_1, \ldots, T_w$ will be denoted by $L_1, \ldots, L_w$. It is assumed that TABU lists are updated using the usual FIFO principle.

The details of the outlined multilevel TABU search strategy are given by the MLTS algorithm below. The following notations will be used:

j	the current level,
k	the iteration number,
x^k	the current point,
x^{k+1}	the next point,
$\Phi_{\min}$	the current record for the objective function value,
$x_{\min}$	$\arg(\Phi_{\min})$,
$q\,\text{Count}(i)$	the number of consecutive iterations performed at levels not greater than i,
$u\,\text{Count}(i)$	the number of consecutive iterations performed at levels not greater than i without the improvement of $\Phi_{\min}$,
q_i	the upper bound for $q\,\text{Count}(i)$,
u_i	the upper bound for $u\,\text{Count}(i)$,
level	the next level.

MLTS algorithm

Initialization:
$j := 1$,
$k := 0$,
$q\,\text{Count}(i) := 0,\ i = 1, w$,
$u\,\text{Count}(i) := 0,\ i = 1, w$,
$x^0 :=$ starting point,
$T_i := \emptyset,\ i = 1, w$,
$x_{\min} := x^0$,
$\Phi_{\min} := \Phi(x^0)$,

Loop:
while$(j \le w)$
  $x^{k+1} :=$ **Step**(x^k, j)
  $\Phi^{k+1} := \Phi(x^{k+1})$
  $q\,\text{Count}(i) := 0,\ i = 1, j-1$
  $q\,\text{Count}(i) := q\,\text{Count}(i) + 1,\ i = j, w$
  if $\Phi^{k+1} < \Phi_{\min}$
    $\Phi_{\min} := \Phi^{k+1}$
    $x_{\min} := x^{k+1}$
    $u\,\text{Count}(i) := 0,\ i = 1, w$
  else
    $u\,\text{Count}(i) := 0,\ i = 1, j-1$
    $u\,\text{Count}(i) := u\,\text{Count}(i) + 1,\ i = j, w$
  end if
  level=1

```
        for (i = w, j − 1)
            if(u Count(i) ≥ u_i or q Count(i) ≥ q_i)
                level := i + 1
                exit
            end if
        end for
        j := level
        k := k + 1
    end while
```

Output: $k, \Phi_{\min}, x_{\min}$

The procedure Step, used in MLTS algorithm in order to generate the next point, is based on already described principles and can be formalized as follows.

Procedure **Step** (x^k, j):

```
α_k := A_j
β_k := B_j
if x^k ∈ ∂X
    x^{k+1} := Restart (x^k, α_k)
else
    Generate D(x^k)
    N'(x^k) := Neighborhood (x^k, α_k)
    if N'(x^k) \ TR_j ≠ ∅
        x^{k+1} := arg min Φ(x), x ∈ N'(x^k) \ TR_j
    else
        x^{k+1} := Restart (x^k, α_k)
    end if
    Update TABU list T_j
end if
Return
```

As it was already pointed out, the main feature of the MLTS algorithm is a dynamic change of the step size combined with several TABU lists. The expected effect is that larger steps allow diversification inducing the search of new subregions of X, while smaller steps regionally intensify the search.

3. MLTS FOR FINITE MINIMAX PROBLEMS

The concept described in Section 2 is rather general and can be used for any continuous global optimization problem. Here we shall adapt the MLTS algorithm for the case of the finite minimax problem

$$\begin{gathered} \operatorname*{global\,min}_{x \in X} \Phi(x) = \max\{\varphi_1(x), \ldots, \varphi_m(x)\}, \\ X = \{(x_1, \ldots, x_n) \in R^n \mid a_j \le x_j \le b_j,\ j = 1, \ldots, n\}, \end{gathered} \tag{2}$$

where $\varphi_i : R^n \to R$, $i = 1, \ldots, m$, are defined and continuously differentiable on an open set containing the feasible set. This problem has various applications, such as L_1 and L_∞ approximation problems, solution of systems of nonlinear equations, engineering design problems, etc. Let us note that the objective function in problem (2) is nondifferentiable. In fact, the finite minimax problem has been one of the main motivations for the development of nondifferentiable optimization theory.

For a given $x \in X$ let $\mathbf{A}(x) = \{i \in \{1, \ldots, m\} \mid \Phi(x) = \varphi_i(x)\}$. If $i \in \mathbf{A}(x)$ we shall say that φ_i is active at x. Generally speaking, it could be expected that at the optimal solution

$\mathbf{A}(x)$ is not a singleton. Therefore, during TABU search we shall try to generate points such that $\mathbf{A}(x^k) \neq \mathbf{A}(x^{k+1})$, i.e., to make moves which change active functions. We can then expect that the segment $[x^k, x^{k+1}]$ contains 'valley' points with low objective function values. Such points can be obtained by a suitable discretization of the segment $[x^k, x^{k+1}]$.

Having this in mind, a search direction at x^k can be considered as 'good' if it decreases active and 'almost active' functions and/or increases functions which are not active. For example, suppose that at the point x^k the function φ_1 is active, functions $\varphi_2, \ldots, \varphi_s$ are 'almost active' (i.e., $\varphi_1(x^k) - \varphi_i(x^k) \leq \varepsilon$, $i = 2, \ldots, s$) and $\varphi_{s+1}, \ldots, \varphi_m$ are not active (i.e., $\varphi_1(x^k) - \varphi_i(x^k) > \varepsilon$, $i = s+1, \ldots, m$), where $\varepsilon > 0$ is a given parameter. Let

$$G = \left\{ -\frac{\nabla\varphi_1(x^k)}{\|\nabla\varphi_1(x^k)\|}, -\frac{\nabla\varphi_1(x^k)}{\|\nabla\varphi_1(x^k)\|} - \frac{\nabla\varphi_2(x^k)}{\|\nabla\varphi_2(x^k)\|}, \ldots, -\frac{\nabla\varphi_1(x^k)}{\|\nabla\varphi_1(x^k)\|} - \frac{\nabla\varphi_s(x^k)}{\|\nabla\varphi_s(x^k)\|}, \right.$$
$$\left. -\frac{\nabla\varphi_1(x^k)}{\|\nabla\varphi_1(x^k)\|} + \frac{\nabla\varphi_{s+1}(x^k)}{\|\nabla\varphi_{s+1}(x^k)\|}, \ldots, -\frac{\nabla\varphi_1(x^k)}{\|\nabla\varphi_1(x^k)\|} + \frac{\nabla\varphi_m(x^k)}{\|\nabla\varphi_m(x^k)\|} \right\}, \quad \text{and}$$

$$G_N = \left\{ \frac{g}{\|g\|} \mid g \in G \right\}.$$

Then, $D(x^k)$ can consist of all directions $d \in G_N$ which satisfy the condition $A(x^k) \neq A(x^k + \alpha_k d)$. If there is no such d, then we can take $D(x^k) = G_N$.

Let x_s^{k+1} be the discretization point of the segment $[x^k, x^{k+1}]$ with the minimum objective function value. Then, the MLTS algorithm can be slightly modified using x_s^{k+1} (instead of x^{k+1}) to update the best objective function value $\Phi_{\min}$, i.e., if $\Phi(x_s^{k+1}) < \Phi_{\min}$ then $\Phi_{\min} = \Phi(x_s^{k+1})$ and $x_{\min} = x_s^{k+1}$. It should be pointed out that the point x_s^{k+1} is an auxiliary point generated only to improve $\Phi_{\min}$ and that the move from x^k is still performed to x^{k+1}.

The described modification of the MLTS algorithm will be applied to a special class of finite minimax problems which will be introduced in Section 4. The numerical results will be discussed in Section 5.

4. A CLASS OF NP-HARD GLOBAL OPTIMIZATION PROBLEMS

A natural field of applications of global optimization algorithms are problems of optimal design. Such an engineering problem arises in the spread spectrum radar polyphase code design [6,7]. The problem can be formulated as follows:

$$\begin{aligned} &\operatorname*{global\,min}_{x \in X} \Phi(x) = \max\{\varphi_1(x), \ldots, \varphi_{2m}(x)\}, \\ &X = \{x_1, \ldots, x_n) \in R^n \mid 0 \leq x_j \leq 2\pi,\ j = 1, \ldots, n\}, \end{aligned} \tag{3}$$

where $m = 2n - 1$ and

$$\varphi_{2i-1}(x) = \sum_{j=i}^{n} \cos\left(\sum_{k=|2i-j-1|+1}^{j} x_k \right), \qquad i = 1, \ldots, n,$$

$$\varphi_{2i}(x) = 0.5 + \sum_{j=i+1}^{n} \cos\left(\sum_{k=|2i-j-1|+1}^{j} x_k \right), \qquad i = 1, \ldots, n-1,$$

$$\varphi_{m+i}(x) = -\varphi_i(x), \qquad i = 1, \ldots, m.$$

We shall analyze the computational complexity of the problem which is somewhat more general than (3). Let

$$\psi_i(x) = c_{io} + \sum_{j=1}^{a} c_{ij} \cos(L_{ij}(x)) + \sum_{j=a+1}^{b} c_{ij} \sin(L_{ij}(x)), \qquad i = 1, \ldots, s, \tag{4}$$

where all c_{ij} are integers and L_{ij} are linear forms with integer coefficients. Consider the optimization problem

$$\begin{aligned} &\operatorname*{global\,min}_{x\in Y} \Psi(x) = \max\{\psi_1(x),\dots,\psi_s(x)\}, \\ &Y = \left\{(x_1,\dots,x_n)\in R^n \mid -\frac{\pi}{2}\le x_j\le\frac{\pi}{2},\ j=1,\dots,n\right\}. \end{aligned} \tag{5}$$

The following decision problem is obviously easier than problem (5).

Problem P1

INSTANCE. Let $n, a, b, s \in N$, $0 < a < b$, the function $\Psi(x)$ and the set Y as in (5).

QUESTION. Does there exist $x \in Y$ such that $\Psi(x) \le 0$?

In order to show NP-hardness of P1 we will reduce the 3-SAT problem to a problem of the form P1. As it is well known [8], 3-SAT problem is the following.

Problem 3-SAT

INSTANCE. Let $n, s \in N$, a set $E = \{E_1,\dots,E_s\}$ of expressions of the form $\hat{p}\vee\hat{q}\vee\hat{r}$, where $p,q,r\in\{p_1,\dots,p_n\}$ and $\hat{p}$ is either p or $\bar{p}$ (similarly for $\hat{q}$ and $\hat{r}$).

QUESTION. Does there exist a truth value assignment for the variables $p_1,\dots,p_n$ which makes all expressions in E take on the value 'true'?

In the sequel we shall use 1 and 0 instead of 'true' and 'false', respectively. Let us note that any disjunction E_i will take on the value 1 at 7 out of 8 possible truth value assignments. For example, $\bar{p}_6\vee p_2\vee\bar{p}_{23}$ has value 0 only if $p_2=0$, $p_6=1$ and $p_{23}=1$.

THEOREM. *The problem P1 is NP-hard.*

PROOF. Let us consider the function

$$f(x,y,z) = \max\{\cos x,\cos y,\cos z,4(1+\sin x)(1+\sin y)(1+\sin z)\},$$

where the domain is $[-\pi/2,\pi/2]^3$. Since

$$\begin{aligned} 4(1+\sin x)(1+\sin y)(1+\sin z) &= 4+4\sin x+4\sin y+4\sin z+2\cos(x-y)+2\cos(x-z) \\ &\quad +2\cos(y-z)-2\cos(x+y)-2\cos(x+z)-2\cos(y+z) \\ &\quad +\sin(x+y-z)+\sin(x-y+z)+\sin(-x+y+z)-\sin(x+y+z), \end{aligned}$$

the function f has the same form as $\Psi(x)$ in (5). It is clear that f is positive on the entire domain except for seven out of eight vertices, where its value is zero. The only vertex where the value is positive is $(\pi/2,\pi/2,\pi/2)$.

To each (Boolean) variable p_i we shall assign a (continuous) variable x_i, where $x_i\in[-\pi/2,\pi/2]$. The values $\pi/2$ and $-\pi/2$ will correspond to the values 0 and 1, respectively, (the values in $(-\pi/2,\pi/2)$ are not candidates for optima, since the function f is strictly positive there).

For each disjunction E_i we construct an auxiliary function f_i as follows: If, for example, E_i contains variables p_2,p_6,p_{23} then, f_i will contain $\pm x_2$, $\pm x_6$, and $\pm x_{23}$. The sign '$-$' is used if and only if the Boolean variable appears in E_i with the negation sign. For example, the disjunction $E_i=\bar{p}_6\vee p_2\vee\bar{p}_{23}$ above would generate the function $f_i=f(-x_6,x_2,-x_{23})$. It is clear that under this transformation it follows that $f_i=0$ if and only if the disjunction E_i has the value 1.

Consider the following global optimization problem of the form (5)

$$\operatorname*{global\,min}_{x\in Y} F(x) = \max\{f_1(x),\dots,f_s(x)\}.$$

Since $f_i \geq 0$, if the optimal value is 0 at some point $(x_1^*, \ldots, x_n^*)$, then $f_i(x_1^*, \ldots, x_n^*) = 0$, $i = 1, \ldots, s$. However, under the correspondence described above, the values of all disjunctions have to be 1 and the answer to the instance of 3-SAT problem is 'yes'. The converse is also true: if we have a truth value assignment which makes all disjunctions equal to 1, the corresponding values of $x_1, \ldots, x_n$ will make all functions f_i equal to 0, which means that F will also have the minimum value of zero.

It follows that any algorithm which would solve the problem P1 in polynomial time, would also solve 3-SAT in polynomial time, which proves that P1 is NP-hard. ∎

In order to apply the theorem to problem (3), we need also to consider the problem P2, in which the objective function $\Psi(x)$ is composed only of cosine functions.

Problem P2

INSTANCE. Let $n, a, b, s \in N$, $0 < a = b$, the function $\Psi(x)$ and the set Y as in (5).

QUESTION. Does there exist $x \in Y$ such that $\Psi(x) \leq 0$?

NP-hardness of P2 is proved by the following result.

COROLLARY. *The problem P2 is NP-hard.*

PROOF. It suffices to show that the sine terms in (4) can be eliminated. For example, to eliminate $\sin(x - y + z)$ we replace $\sin(x - y + z)$ by $\cos(x - y + z - t)$, and insert a new term $\cos t$ into the objective function Ψ, where $-\pi/2 \leq t \leq \pi/2$. Since $\cos t = 0$ for $t = -\pi/2$ or $t = \pi/2$, the domain is symmetric and cosine is an even function, there is no loss of generality in assuming $t = \pi/2$. Since $\cos(u - \pi/2) = \sin u$, we obtain the original sine function again. ∎

Since the set X can be reduced to Y using a linear transformation of coordinates, the corollary shows that problems of type (3) are special instances of a class of NP-hard problems.

5. NUMERICAL RESULTS

The power of the MLTS algorithm for finite minimax problems was tested on the problem of spread spectrum radar polyphase code design with dimensions $n = 2, \ldots, 10$. For the sake of comparison we also present the results obtained by an exact covering technique and the standard Monte Carlo method. The tests were performed on a 486 PC.

As the exact covering method we used a variant of the implicit enumeration technique described in [5]. At each iteration of this technique the set of certain subregions of the feasible set X is considered and those subregions not containing the global optimum are excluded from further consideration, while some of the remaining ones are refined. The results are summarized in Table 1. It contains the optimal value of the objective function, the number of iterations needed to attain this value and the average number of subregions considered at one iteration. Although the results in [5] show that the exact method was very successful on problems of linearly constrained separable concave minimization (even for $n = 20$), Table 1 demonstrates that this approach is not suitable for problem (3). Namely, here the number of subregions grows rapidly with the dimensions and the results within reasonable time were obtained only for $n \leq 5$. This is caused by the fact that it is not possible to get good upper and lower bounds of the minimum of the objective function on subregions.

Table 1.

n	Optimal Value	N^0 of Iterations	Average N^0 of Subregions
2	0.3852	486	30
3	0.2610	624	29
4	0.0560	8321	388
5	0.3371	97496	3768

The parameters for the MLTS algorithm were chosen as follows: For $n \leq 5$ we applied a two-level search strategy, while for $n \geq 6$ we used three levels. The tolerance ε, used in definition of the set $D(x^k)$, was set to 0.00001. The discretization of the segment $[x^k, x^{k+1}]$ was uniform, with ten discretization points. Many experiments with different values of the remaining parameters have been performed and we will quote here only the most interesting details. Experiments have shown that, independently of the dimensions, the reasonable choices for TABU lists lengths are, in the case of two-level strategy, $L_1 = 100$, $L_2 = 20$, while for three-level strategy $L_1 = 200$, $L_2 = 100$, $L_3 = 20$. The size of the TABU cube was always chosen to be a half of the corresponding step size ($B_i = A_i/2$). The initial point was in all cases the center of the feasible set X, i.e., $x_j^0 = (a_j + b_j)/2$, $j = 1, \ldots, n$, and the total number of iterations was 10000.

Different values for A_i and q_i have been used in order to determine the best multilevel search strategy. In all cases u_i is taken to be 40% of q_i. Tables 2 and 3 show the values of parameters for which the best objective function value is obtained, then this value and the corresponding iteration number. The last column contains the results obtained by the standard Monte Carlo method (with 10000000 points).

Table 2.

n	A_1	A_2	q_1	q_2	The Best Value	Iteration Number	Monte Carlo
2	0.02	0.9	199	10000	0.3852	7581	0.3857
3	0.02	1.0	499	10000	0.2610	5280	0.2687
4	0.03	1.5	99	10000	0.0599	2839	0.0823
5	0.04	2.0	199	10000	0.3418	3123	0.3988

The comparison of the exact optimal values (Table 1) and the best values in Table 2 shows that the number of accurate decimal places for $n = 2, 3, 4, 5$ is, respectively, 4,4,2,2, which means that the obtained solutions are quite satisfactory. For $n \geq 6$ the optimal values have not been found by the exact method, but it is evident from Table 3 that the MLTS algorithm gives much better results than the Monte Carlo method with respect both to the objective function value and to the computational effort. Namely, it is easy to calculate that the computational effort per one MLTS step is about $5n + 10$ times greater than the effort per step of Monte Carlo. This means that, for example, in the case $n = 10, 10000$ MLTS steps are equivalent to 600000 Monte Carlo interations.

Table 3.

n	A_1	A_2	A_3	q_1	q_2	q_3	The Best Value	Iteration Number	Monte Carlo
6	0.002	0.02	2.0	4	499	10000	0.4603	5307	0.5000
7	0.002	0.02	2.0	4	499	10000	0.4985	1919	0.6772
8	0.02	0.2	2.0	4	499	10000	0.4288	7385	0.6179
9	0.002	0.2	2.5	4	499	10000	0.3820	7297	0.8802
10	0.02	0.2	3.0	4	999	10000	0.4615	7175	0.9106

The experiments have shown that the trajectory of the search greatly depends on the choice of the parameters, which means that even small variations of parameters produce completely different trajectories. For example, if in the case $n = 9$, instead of $A_1 = 0.002$, the value $A_1 = 0.02$ is used, a completely different terminal point is obtained, with the objective function value 0.6412. However, if the remaining parameters are also altered, the results can be improved. For example, if $A_1 = 0.02$, $A_2 = 0.2$, $A_3 = 4$, $B_1 = 0.01$, $B_2 = 0.1$, $B_3 = 1.25$, $q_1 = 4$, $q_2 = 999$, $q_3 = 10000$, $L_1 = 200$, $L_2 = 100$, $L_3 = 50$, the terminal objective function value is 0.3494. Having this in mind, given a particular problem, our recommendation is to run the

MLTS algorithm several times with different values of parameters in order to get the best possible results.

Let us note, at the end, that the described MLTS algorithm can be combined with local search in order to improve the terminal point. In fact, the MLTS algorithm with one level and a small step can itself be used as a local search procedure. Table 4 illustrates the improvement of the results in Table 3 with 500 iterations of one-level MLTS.

Table 4.

n	A_1	The Improved Value
6	0.005	0.4588
7	0.005	0.4976
8	0.01	0.3871
9	0.01	0.3492
10	0.01	0.4342

4. CONCLUSIONS

The paper proposes a new global optimization algorithm. The algorithm is based on a multilevel TABU search strategy, where each level is characterized by a different size of the move step. The algorithm is tested on real-life problems arising from the synthesis of radar polyphase codes and satisfactory results are obtained with the two- and three-level strategy for dimensions not greater than 10. The future research will be oriented to the implementation of the multilevel TABU search strategy to problems of higher dimensions.

REFERENCES

1. A. Tom and A. Žilinskas, *Global Optimization*, p. 350, Springer-Verlag, (1987).
2. P.M. Pardalos and J.B. Rosen, *Constrained Global Optimization: Algorithms and Applications*, p. 268, Springer-Verlag, (1987).
3. F. Glover and M. Laguna, Tabu search, In *Modern Heuristic Techniques for Combinatorial Problems*, (Edited by C.R. Reeves), pp. 70–150, Blackwell Scientific, (1993).
4. F. Glover, Tabu search for nonlinear and parametric optimization, *Discrete Applied Mathematics* (1993).
5. M.D. Ašić and V.V. Kovačević-Vujčić, An implicit enumeration method for global optimization problems, *Computers Math. Applic.* **21** (6/7), 191–201 (1991).
6. M.L. Dukić and Z.S. Dobrosavljević, A method of a spread-spectrum radar polyphase code design, *IEEE Journal on Selected Areas in Comm.* (5), 743–749 (1990).
7. M.L. Dukić and Z.S. Dobrosavljević, A method of spread spectrum radar polyphase code design by nonlinear programming, *European Transactions on Telecommunications and Related Techniques* (to appear).
8. M.R. Garey and D.S. Johnson, *Computers and Intractability*, Freeman, San Francisco, (1979).