Journal of Computational Mathematics, Vol.24, No.1, 2006, 25–32.

A NEW FILLED FUNCTION METHOD FOR INTEGER PROGRAMMING *1)

Ming-ming Li
(*Department of Mathematics, Shanghai University, Shanghai 200444, China; Department of Mathematics, Shanghai University of Engineering Science, Shanghai 200336, China*)
You-lin Shang
(*Department of Mathematics, Shanghai University, Shanghai 200444, China; Department of Mathematics, Henan University of Science and Technology, Luoyang 471003, China*)
Lian-sheng Zhang
(*Department of Mathematics, Shanghai University, Shanghai 200444, China*)

Abstract

The Filled Function Method is a class of effective algorithms for continuous global optimization. In this paper, a new filled function method is introduced and used to solve integer programming. Firstly, some basic definitions of discrete optimization are given. Then an algorithm and the implementation of this algorithm on several test problems are showed. The computational results show the algorithm is effective.

Mathematics subject classification: 90C10, 90C30, 65K10
Key words: Global minimization, Integer programming, Filled function method

1. Introduction

An integer programming problem is very difficult to be solved. In [1], it is pointed out that the integer programming with a linear objective function and quadratic constrained is algorithmically unsolved, that is, there no algorithms can be created to solve them. Because of these reasons, the approximate algorithms have been rapidly developed in recent years(see [2]). In publishing literatures, approximate algorithms for integer programming can be sorted into two categories: stochastic approach(see [3-5]) and deterministic approach(see [6-8]). The filled function method applied for continuous global optimization is proposed firstly by Ge in [9]. This method is consist of two stages: 1. finding a local minimizer, x_1^*, of the original continuous global optimization by any local minimization method; 2. constructing a filled function and then minimizing this filled function to get another local minimizer of original problem whose objective function value is smaller than that of x_1^*. Repeat the above two steps to find the global minimizer of original problem. The key of this method is to construct a filled function. Now we extend the filled function method to solve the integer programming problem. In this paper, a new filled function is introduced. At the same time, we show a new filled function method for discrete global optimization by this new filled function. This is an approximate and direct approach. By solving some testing problems, it is showed to be an effective and efficient method. In [10], Tian and Zhang give a filled function with two parameters. But the filled function in this paper has only one parameter. Moreover this parameter is chosen easily.

We consider integer programming problem given by**(P)**

$$\begin{cases} min \;\; f(x) \\ s.t. \;\; x \in \Omega \cap R_I^n \end{cases}$$

* Received September 1, 2004; Final revised July 12, 2005 .
1) Foundation item: the National Science Foundation of China(7A14178); the Science Fund of Shanghai University of Engineering Science for Young Scholars(2005Q23); the Natural Science Foundation of Education Commission of Shanghai (No.05NZ07).

where Ω is a bounded closed box with all vertices integral, R_I^n is the set of integer points in R^n. Furthermore, we suppose that $f(x)$ is coercive, that is, $f(x) \to +\infty$ as $\|x\| \to +\infty$. So all local minimizers of $f(x)$ which values are more small can't be on the boundary of $\Omega \cap R_I^n$ if Ω is sufficiently large. In the following of this paper, we let

$$X_I = \Omega \cap R_I^n.$$

The paper is organized as follows: Section 2 gives the preliminary knowledge about discrete optimization; A new filled function for discrete optimization is proposed in section 3; We show an algorithm and the numerical experiments in section 4; Some conclusions are in section 5.

2. Preliminaries

In this section, we firstly introduce some basic definitions and method for discrete optimization.

Definition 2.1 *The set of all directions in discrete analysis is defined by*

$$D = \{e_i, -e_i; i = 1, 2, ..., n\}.$$

where e_i is the $i-th$ unit vector which is the $n-dimensional$ vector with the $i-th$ component equal to one and all other components equal to zero.

Definition 2.2 *For any $x \in R_I^n$, the neighborhood of x is defined as*

$$N(x) = \{x, x \pm e_i : i = 1, 2, ..., n\}.$$

Definition 2.3 *An integer point $x_1^* \in X_I$ is called a discrete local minimizer of $f(x)$ if $f(x) \geq f(x_1^*)$, for all $x \in N(x_1^*) \cap X_I$. Furthermore, if $f(x) > f(x_1^*)$ for all $x \in (N(x_1^*) \cap X_I) \backslash x_1^*$, then x_1^* is called a strict discrete local minimizer of $f(x)$.*

Suppose x_1^* is a (strict) discrete local minimizer of $f(x)$, then x_1^* is a (strict) discrete local maximizer of $-f(x)$.

Definition 2.4 *An integer point $x^* \in X_I$ is called a discrete global minimizer of $f(x)$ if $f(x) \geq f(x^*)$, for all $x \in X_I$. Furthermore, if $f(x) > f(x^*)$ for all $x \in X_I \backslash x^*$, then x^* is called a strict discrete global minimizer of $f(x)$.*

Suppose x^* is a (strict) discrete global minimizer of $f(x)$, then x^* is a (strict) discrete global maximizer of $-f(x)$.

Definition 2.5 *Suppose x_1^* is a discrete local minimizer of $f(x)$. A function $P(x, x_1^*)$ is said to be a discrete filled function of $f(x)$ at x_1^* if $P(x, x_1^*)$ satisfies the following properties:*

1. *x_1^* is a strict discrete local maximizer of $P(x, x_1^*)$;*
2. *if $f(x) \geq f(x_1^*)$ and $f(x + d_i) \geq f(x_1^*), \forall d_i \in D$, $x \neq x_1^*$, then x is not a local minimizer of $P(x, x_1^*)$;*
3. *for x_1, x_2, if $\|x_1 - x_1^*\| > \|x_2 - x_1^*\| > 0$ and $f(x_1)$, $f(x_2) \geq f(x_1^*)$, then $P(x_1, x_1^*) < P(x_2, x_1^*)$;*
4. *for x_1, x_2, if $\|x_1 - x_1^*\| > \|x_2 - x_1^*\| > 0$ and $f(x_2) \geq f(x_1^*) > f(x_1)$, then $P(x_1, x_1^*) > P(x_2, x_1^*)$.*

Remark. The 2-th item of this definition notes: if there exists $d_{i_0} \in D$ such that $f(x + d_{i_0}) < f(x_1^*)$, then let $x_0 = x + d_{i_0}$ to find another discrete local minimizer x_2^* of $f(x)$ which holds $f(x_2^*) < f(x_1^*)$.

Definition 2.6 *For any $x \in X_I$, $d \in D$ is said to be a descent direction of $f(x)$ at x if $x+d \in X_I$ and $f(x+d) < f(x)$.*

Now we present the discrete local search method for finding a discrete local minimizer of problem **(P)** from a given initial point $x_0 \in X_I$.

Algorithm 2.1 *(discrete local search method)*
1. Give an initial integer point $x_0 \in X_I$;
2. If x_0 is a discrete local minimizer of $f(x)$ over X_I, stop; otherwise, a descent direction, d, of $f(x)$ at x_0 over X_I can be found;
3. Let $x_0 = x_0 + d$, go to 2.

3. A New Filled Function

We extend the filled function for continuous global optimization to the discrete case in this section. And this function is proved to be a filled function of $f(x)$. Then, a new filled function method for discrete global optimization is given.

Now, we construct a filled function

$$P(x, x_1^*, r) = -\|x - x_1^*\|^2 h_r(f(x) - f(x_1^*)) \tag{1}$$

where x_1^* , a minimizer of $f(x)$, has been found.

$$(*) \qquad h_r(t) = \begin{cases} c, & t \geq 0 \\ -\frac{2ct^3}{r^3} - \frac{3ct^2}{r^2} + c, & -r \leq t \leq 0 \\ 0, & t \leq -r \end{cases}$$

Parameter c can be any positive constant. Therefore, $P(x, x_1^*, r)$ has only one parameter r in fact. Now we prove $P(x, x_1^*, r)$ is a discrete filled function of $f(x)$ at x_1^*.

Theorem 3.1 *x_1^* is a strict discrete local maximizer of $P(x, x_1^*, r)$.*

Proof. Because x_1^* is a discrete local minimizer of $f(x)$, for any $d \in D$, $x_1^* + d \in X_I$ and $f(x_1^* + d) \geq f(x_1^*)$.
Hence $h_r(f(x_1^* + d) - f(x_1^*)) = c$.
Therefore

$$\begin{aligned} P(x_1^* + d, x_1^*, r) &= -\|x_1^* + d - x_1^*\|^2 h_r(f(x_1^* + d) - f(x_1^*)) && (2) \\ &= -\|d\|^2 \cdot c \\ &= -c \\ &< 0 \\ &= P(x_1^*, x_1^*, r) && (3) \end{aligned}$$

Thus x_1^* is a strict discrete local maximizer of $P(x, x_1^*, r)$.

Theorem 3.2 *For x_1, x_2, if $\|x_1 - x_1^*\| > \|x_2 - x_1^*\| > 0$ and $f(x_1)$, $f(x_2) \geq f(x_1^*)$, then $P(x_1, x_1^*, r) < P(x_2, x_1^*, r)$.*

Proof. For x_1, x_2, since $f(x_1) \geq f(x_1^*)$,$f(x_2) \geq f(x_1^*)$, it has

$$h_r(f(x_1) - f(x_1^*)) = h_r(f(x_2) - f(x_1^*)) = c. \tag{4}$$

Thus

$$\begin{aligned} P(x_1, x_1^*, r) &= -\|x_1 - x_1^*\|^2 h_r(f(x_1 + d) - f(x_1^*)) \\ &= -\|x_1 - x_1^*\|^2 \cdot c \\ &< -\|x_2 - x_1^*\|^2 \cdot c && (5) \\ &= P(x_2, x_1^*, r). \end{aligned}$$

Corollary 3.1 *For any $x \neq x_1^*, x \in X_I$, if $f(x) \geq f(x_1^*)$, then $P(x, x_1^*, r) < 0 = P(x_1^*, x_1^*, r)$.*

Proof. Because $f(x) \geq f(x_1^*)$, $h_r(f(x) - f(x_1^*)) = c$.
So

$$\begin{aligned} P(x, x_1^*, r) &= -\|x - x_1^*\|^2 h_r(f(x) - f(x_1^*)) && (6)\\ &= -\|x - x_1^*\|^2 \cdot c && (7)\\ &< 0 \\ &= P(x_1^*, x_1^*, r). \end{aligned}$$

Theorem 3.3 *For any $x \neq x_1^*, x \in X_I$, if $f(x) \geq f(x_1^*)$ and $f(x + d_i) \geq f(x_1^*), \forall d_i \in D$, then x is not a discrete local minimizer of $P(x, x_1^*, r)$.*

Proof. Let $d_{i_0} \in D$ satisfying $\|x + d_{i_0} - x_1^*\| > \|x - x_1^*\|$ for $x \neq x_1^*, x \in X_I$.

Because $f(x) \geq f(x_1^*)$ and $f(x + d_i) \geq f(x_1^*), \forall d_i \in D$, obviously $f(x + d_{i_0}) \geq f(x_1^*)$.
Therefore, we have

$$\begin{aligned} P(x + d_{i_0}, x_1^*, r) &= -\|x + d_{i_0} - x_1^*\|^2 h_r(f(x + d_{i_0}) - f(x_1^*)) && (8)\\ &= -\|x + d_{i_0} - x_1^*\|^2 \cdot c \\ &< -\|x - x_1^*\|^2 \cdot c && (9)\\ &= -\|x - x_1^*\|^2 h_r(f(x) - f(x_1^*)) && (10)\\ &= P(x, x_1^*, r). \end{aligned}$$

That is, there exists a descent direction $d_{i_0} \in D$ for x. So x is not a discrete local minimizer of $P(x, x_1^*, r)$.

Remark. By the above theorem, if x satisfies the conditions of theorem, x is not a discrete local minimizer of $P(x, x_1^*, r)$. Thus, if x is a discrete local minimizer of $P(x, x_1^*, r)$ and $f(x) \geq f(x_1^*)$, $x \neq x_1^*, x \in X_I$, then there must exist $d_{i_0} \in D$ for x satisfying $f(x+d_{i_0}) < f(x_1^*)$; let $x_0 = x+d_{i_0}$ and start from this point to minimize $f(x)$ to obtain its another discrete local minimizer x_2^* of $f(x)$, and $f(x_2^*) < f(x_1^*)$.

Theorem 3.4 *If x_1^* is not a discrete global minimizer of $f(x)$, then there must exist another discrete local minimizer $\bar{x}$ of $f(x), \bar{x} \in X_I$ satisfying $P(\bar{x}, x_1^*, r) = 0$ and $f(\bar{x}) < f(x_1^*) - r$.*

Proof. Let $L = \{f(x) | f(x)$ is a discrete local minimum of $f(x)\}$, $M = \{f(x) | f(x) \in L$ and $f(x) < f(x_1^*)\}$.

Because x_1^* is not a discrete global minimizer of $f(x)$, M is not empty.
Let

$$r < \alpha_0 = \begin{cases} max \ \ (f(x_1^*) - f(x)) \\ s.t. \ \ f(x) \in M \end{cases} \tag{11}$$

Since M is not empty, there must exist $\bar{x} \in X_I$ satisfying

$$f(x_1^*) - f(\bar{x}) = \alpha_0. \tag{12}$$

Therefore $\quad f(x_1^*) - f(\bar{x}) > r$.
That is $\quad f(\bar{x}) - f(x_1^*) < -r$ i.e. $f(\bar{x}) < f(x_1^*) - r$.
So $\quad h_r(f(\bar{x}) - f(x_1^*)) = 0$.
Thus

$$P(\bar{x}, x_1^*, r) = -\|\bar{x} - x_1^*\|^2 h_r(f(\bar{x}) - f(x_1^*)) = 0. \tag{13}$$

Remark. If there exists $x \neq x_1^*$ satisfying $P(x, x_1^*, r) = 0$, then $f(\bar{x}) \leq f(x_1^*) - r$, $f(\bar{x}) < f(x_1^*)$.

Corollary 3.2 *If x_1^* is not a discrete global minimizer of $f(x)$ and r satisfying*

$$r < \begin{cases} min \ \ (f(x_1^*) - f(x)) \\ s.t. \ \ f(x) < f(x_1^*) \end{cases} \tag{14}$$

then $P(x, x_1^*, r) = 0$ for x satisfying $f(x) < f(x_1^*)$.

If r is chosen according to Corollary 3.2, then for x_1, x_2 satisfying $\|x_1 - x_1^*\| > \|x_2 - x_1^*\| > 0$ and $f(x_2) \geq f(x_1^*) > f(x_1)$, it must have $P(x_1, x_1^*, r) = 0 > -\|x_2 - x_1^*\|c = P(x_2, x_1^*, r)$. That is, during minimizing the discrete filled function $P(x, x_1^*, r)$ along the direction d, if $P(x_{k+1}, x_1^*, r) > P(x_k, x_1^*, r)$, then $f(x_{k+1}) < f(x_1^*)$(obviously $f(x_k) \geq f(x_1^*)$, $\|x_{k+1} - x_1^*\| > \|x_k - x_1^*\| > 0$). If not (that is, $f(x_{k+1}) \geq f(x_1^*)$), then $P(x_{k+1}, x_1^*) < P(x_k, x_1^*)$ by Theorem 3.2.

From the above theorems, We can know $P(x, x_1^*, r)$ is the discrete filled function of $f(x)$ at x_1^*. Moreover $c > 0$ is any constant and $r > 0$ can be chosen sufficiently small by Theorem 4, so $P(x, x_1^*, r)$ can be regarded a discrete filled function without parameter.

4. Algorithm and Numerical Experiments

Algorithm 4.2 *(Discrete Filled Function Method)*

1. *initialization:*
   $x_1^0 \in X_I, M_0 = 1, k = 0, D = \{\pm e_i, i = 1, 2, \cdots, n\}, k_0 = 2n, r_0 = 10^{-1}, N = 11;$

2. *use* discrete local search method *to minimize $f(x)$ from x_1^0 in X_I and obtain x_1^*, which is a discrete local minimizer of $f(x)$;*
   $r = r_0,\ M = M_0;$

3. *construct* $P(x, x_1^*, r) = -\|x - x_1^*\|^2 h_r(f(x) - f(x_1^*));$

4. *i=1;*

5. *set* $x_1 = x_1^* + d_i, \quad d_i \in D,$
   let $D_1 = \{d \in D : x_1 + d \in X_I\};$

6. *if there exists* $d \in D_1$ *satisfying* $f(x_1 + d) < f(x_1^*)$, *then set* $x_1^0 = x_1 + d, k = k + 1$, *go to 2;*

7. *let* $D_2 = \{d \in D_1 : ||x_1 + d - x_1^*|| > ||x_1 - x_1^*||\};$

8. *if* D_2 *is empty, go to 10;*

9. *if there exists* $d \in D_2$ *satisfying: when minimizing* $P(x, x_1^*, r)$ *starting from an initial point* $x_2^0 = x_1 + d$, *we can obtain a local minimizer* x_2^* *of* $P(x, x_1^*, r)$ *which satisfies* $x_2^* + d \in X_I$ *for any* $d \in X_I$, *then set* $x_1^0 = x_2^* + d_i$, *where* $f(x_2^* + d_i) < f(x_1^*), d_i \in D$; $k = k + 1$, *go to 2;*

10. $i = i + 1$, *if* $i \leq k_0$, *go to 5;*

11. $M = M + 1$, *if* $M \leq N$, *then* $r = 10^{-1}r$, *go to 3; otherwise,* x_1^* *is a discrete global minimizer of the original problem.*

In the algorithm, e_i is an $n-$dimensional vector with unit length whose $i-th$ coordinate is 1. N is the terminal rule. That is, when $N = 11$, r has been changed to be very small, the algorithm can be terminated and the current minimizer is regarded to be the discrete global minimizer of the original problem. In the following, computational results of some test problems using the above algorithm are summarized. $h_r(t)$ is chosen $(*)$. The used computer is equipped with Intel Pentium 2.5GHz. The symbols are used in the tables are noticed as follows:

x_1^0:	the initial point
k:	the iteration number
CPU:	the CPU time in seconds to obtain the final result
$CPUS$:	the CPU time in seconds to stop
NF:	the number of computing $f(x)$ to obtain the final result
NFS:	the number of computing $f(x)$ to stop
NP:	the number of computing $P(x, x_1^*, r)$ to obtain the final result
NPS:	the numberh of computing $P(x, x_1^*, r)$ to stop

Problem 1(in [11]).

$$minf(x) = x_1^4 + x_2^4 + 16[x_1x_2 + (4 + x_2)^2]$$
$$s.t.\ \ |x_i| \leq 10,\ \ x_i : integer,\ \ i = 1, 2$$

This problem has 441 feasible points. Its global minimal value is 17. We use five initial points: $(0, 0)$, $(1, 1)$, $(-1, -1)$, $(5, 5)$, $(-5, -5)$. For every experiment, the above discrete filled function method succeeds in finding the discrete global minimizer $x^* = (2, -3)$ with $f(x^*) = 17$. The maximum CPU time to reach the discrete global minimum is 0.0620 seconds. Results are given in Table 1.

Table 1

x_1^0	k	CPU ($CPUS$)	NF (NFS)	NP (NPS)
(0, 0)	0	0.0160 (0.9220)	19 (239)	0 (6138)
(1, 1)	0	0.0470 (0.9840)	34 (254)	0 (6138)
(-1, -1)	0	0.0470 (1)	16 (236)	0 (6138)
(5, 5)	0	0.0470 (1)	66 (286)	0 (6138)
(-5, -5)	0	0.0620 (1.0150)	20 (240)	0 (6138)

Problem 2(in [11]).

$$minf(x) = \sum_{i=1}^{10}(x_i^4 - 4.9x_i^2)$$
$$s.t. -5 \leq x_i \leq 5,\ \ x_i : integer,\ \ i = 1, 2, ..., 10$$

This problem has $2.5937 \times e^{10}$ feasible points. Its global minimal value is -39. We use five initial points: $(0, 0, ..., 0)$, $(2, 2, ..., 2)$, $(-2, -2, ..., -2)$, $(4, 4, ..., 4)$, $(-4, -4, ..., -4)$. For every experiment, the above discrete filled function method succeeds in finding the discrete global minimum $f(x^*) = -39$. Starting from initial points: $(0, 0, ..., 0)$, $(2, 2, ..., 2)$ and $(4, 4, ..., 4)$, the method obtains the discrete global minimizer $(1, 1, ..., 1)$. Starting from initial points: $(-2, -2, ..., -2)$ and $(-4, -4, ..., -4)$, the method obtains the discrete global minimizer $(-1, -1, ..., -1)$. The maximum CPU time to reach the discrete global minimum is 0.0940 seconds. Table 2 gives the numerical Results.

Table 2

x_1^0	k	CPU ($CPUS$)	NF (NFS)	NP (NPS)
(0,0,...,0)	0	0.0150 (197.7650)	76 (4696)	0 (1208570)
(2,2,...,2)	0	0.0160 (198.0790)	176 (4796)	0 (1208570)
(-2,-2,...,-2)	0	0.0630 (246.5470)	76 (4696)	0 (1497430)
(4,4,...,4)	0	0.0940 (198.5160)	486 (5106)	0 (1208570)
(-4,-4,...,-4)	0	0.0630 (246.2190)	186 (4806)	0 (1497430)

Problem 3.

$$\begin{aligned} minf(x) &= 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 \\ &\quad + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1) \\ s.t. &- 10 \le x_i \le 10, \quad x_i : integer, \quad i = 1, 2, 3, 4 \end{aligned}$$

This problem has about 194481 feasible points. Its discrete global minimizer is $x^* = (1, 1, 1, 1)$ with $f(x^*) = 0$. We use four initial points: $(0, 0, 0, 0)$, $(-1, -1, -1, -1)$, $(-2, -2, -2, -2)$, $(-5, -5, -5, -5)$. For every initial point, the above discrete filled function method can obtain the discrete global minimum $f(x^*) = 0$. The maximum CPU time to reach the discrete global minimum is 0.4840 seconds. Results as follows:

Table 3

x_1^0	k	CPU ($CPUS$)	NF (NFS)	NP (NPS)
(0,0,0,0)	2	0.4680	31	1656
		(12.3590)	(841)	(85124)
(-1,-1,-1,-1)	2	0.4840	41	1656
		(12.3590)	(851)	(85124)
(-2,-2,-2,-2)	2	0.4510	51	1656
		(12.3280)	(861)	(85124)
(-5,-5,-5,-5)	1	0.4070	81	1656
		(12.3130)	(891)	(85124)

Problem 4.

$$\begin{aligned} minf(x) &= \sum_{i=1}^{n} x_i^4 + \Big(\sum_{i=1}^{n} x_i\Big)^2 \\ s.t. &- 5 \le x_i \le 5, \quad x_i : integer, \quad i = 1, 2, ..., n \end{aligned}$$

This problem has about 11^n feasible points. Its discrete global minimizer for any n is $x^* = (0, 0, ..., 0)$ with $f(x^*) = 0$. We use four initial points: $(1, 1, ..., 1)$, $(-1, -1, ..., -1)$, $(3, 3, ..., 3)$, $(-3, -3, ..., -3)$ for $n = 4, n = 8, n = 16$ respectively. For every experiment, the discrete filled function method can obtain the discrete global minimum $f(x^*) = 0$. The maximum CPU time to reach the discrete global minimum is 0.5150, 21.4370, 1193.2 seconds for $n = 4, n = 8, n = 16$ respectively. Computational results are summarized in Table 4, Table 5, Table 6.

From the above computational results, we can know the algorithm can't stop until it is forced to stop even though the (approximate)global minimizer is found. The reason is that the filled function method has no good terminal rule.

Table 4(n=4)

x_1^0	k	CPU ($CPUS$)	NF (NFS)	NP (NPS)
(1,1,1,1)	1	0.1550	49	0
		(8.0310)	(841)	(42240)
(-1,-1,-1,-1)	1	0.5150	27	1326
		(8.3910)	(846)	(43566)
(3,3,3,3)	1	0.2510	107	425
		(8.1880)	(908)	(42665)
(-3,-3,-3,-3)	1	0.4050	55	1309
		(8.2810)	(874)	(43549)

Table 5(n=8)

x_1^0	k	CPU ($CPUS$)	NF (NFS)	NP (NPS)
(1,1,...,1)	3	2.7490	240	9005
		(113.9370)	(3056)	(586285)
(-1,-1,...,-1)	3	21.4370	386	57164
		(122.4840)	(3202)	(634444)
(3,3,...,3)	2	6.1870	441	21135
		(115.6250)	(3552)	(598415)
(-3,-3,...,-3)	3	21.3470	253	56256
		(122.6250)	(3551)	(633536)

Table 6(n=16)

x_1^0	k	CPU ($CPUS$)	NF (NFS)	NP (NPS)
(1,1,...,1)	7	251.7060	1478	462315
		(1790.5)	(12742)	(8741355)
(-1,-1,...,-1)	6	1193.2	2781	1764407
		(2052.2)	(14045)	(10043447)
(3,3,...,3)	4	365.0950	2899	758374
		(1843.7)	(14163)	(9037414)
(-3,-3,...,-3)	6	1174.5	3377	1742966
		(2045.4)	(14641)	(10022006)

5. Conclusions

In this paper, we propose the new discrete filled function. And it proves that $P(x, x_1^*, r)$ is truly the discrete filled function of $f(x)$ at the point x_1^* according to our definition for the filled function. By solving the above problems, the algorithm constructed by $P(x, x_1^*, r)$ is effective.

References

[1] R.G.Jeroslow, There cannot be any algorithm for integer programming with quadratic constraints, *Oper.Res.*, **21** (1973), 221-224.

[2] S.H.Zanakis, J.R.Evans, A.A.Vazacopoulous, Heuristic methods and applications: a categorized survey, *Eur.J. Oper. Res.*, **43** (1989), 88-110.

[3] B.M.Bertocchi et.al., A Monte-Carlo approach for 0-1 programming problems, *Computing*, **48** (1992), 259-274.

[4] W.Conley, Computer Optimization Techniques, Pertrocelli Books, 1980.

[5] Q.Zheng, D.Zhuang,Testing integral global algorithms for constrained and discrete minimization problems, Working Paper, Mount Saint Vincent University, 1994.

[6] E.Girlich, M.M.Kowaljow, Nichtlineare Discrete Optimization, Akademie-Verlag, Berlin, 1981.

[7] B.L.Miller, On minimizing nonseparable functions defined on the integers with an inventory application, *SIAM J.Appl.Math.*, **21**:1 (1971).

[8] V.Vassilev, K.Genova, An approximate algorithm for nonlinear integer programming, *Eur. J. Oper. Res.*, **74** (1994), 170-178.

[9] Renpu Ge, A Filled Function Method for Finding a Global Minimizer of a Function of Several Variables, *Mathematical Programming*, **46** (1990), 191-204.

[10] Weiwen Tian, Liansheng Zhang, An algorithm for finding global minimum of nonlinear integer programming, *Jour. of Computational Math.*, **22**:1 (2004), 69-78.

[11] Renpu Ge, A Continuous Approach to Nonlinear Integer Programming, *Applied Math. and Computation*, **34** (1989), 39-60.