INFORMATICA, 1991, Vol.2, No.4, 539–551

THE CUTTING OFF ALGORITHMS FOR PSEUDOBOOLEAN OPTIMIZATION

Alexander ANTAMOSHKIN and Yurij KOSHKIN

Space Technology University,
P.O. Box 486, Krasnojarsk,
SU - 660014

Abstract. The local optimization techniques is the basis of majority of regular (exact) algorithms for the non-monoton pseudoboolean functions optimization as the most simple and, accordingly, the most universal method of the discrete optimization. However, the local optimization method does not guarantee the elimination of the total examination when the pseudoboolean optimization problem in a general state is solved. In the present paper the cutting off algorithms are suggested which guarantee the total examination elimination for any pseudoboolean optimization problem.

Key words: searchal pseudoboolean optimization, cutting off algorithms.

1. Introduction. The considering pseudoboolean optimization problem was submited by Antamoshkin A. and E. Semionkin (1991). Using by us in the sequel definitions and designations were introduced in this paper too. In addition we formulate several necessary lemmas.

Lemma 1.1 by Antamoshkin A. et al (1990).

$$\forall X \in B_2^n \wedge X^n \in O_n(X) : O_k(X) = O_{n-k}(X^n).$$

Lemma 1.2 by Antamoshkin A. et al (1990).

$$\forall X \in B_2^n : card\, O_k(X) = C_n^k.$$

Lemma 1.3 by Antamoshkin A. et al (1990).

$$\forall X^k \in O_k(X),\ X \in B_2^n,\ k = \overline{0,n},$$
$$card\left\{O_1(X^k) \cap O_{k-1}(X)\right\} = k,$$
$$card\left\{O_1(X^k) \cap O_{k+1}(X)\right\} = n - k.$$

COROLLARY 1.1 by Antamoshkin A. et al (1990).
$\forall k = \overline{1,n}$ among the points $X_j^k \in O_k(X)$, $X \in B_2^n$, $j = \overline{1, C_n^k}$, there are no neighbouring points.

Lemma 1.4 by Antamoshkin. et al (1990).

$$\forall X \in B_2^n : B_2^n = \bigcup_{k=0}^{n} O_k(X).$$

Lemma 1.5 by Antamoshkin A. et al (1990).
For any unimodal of different values on B_2^n function f for every point $X \in B_2^n \setminus \{X^*\}$ among the points $X_j^1 \in O_1(X)$, $j = \overline{1,n}$, there is at least one point X_j^1 such that $f(X_j^1) < f(X)$.

2. The cutting off algorithms

As appears from Lemma 1.4 the space B_2^n always may be represented as a structure in the capacity of the origin of which any point of B_2^n may be chosen. Denote this point X^0.

Evidently enough that any level $O_k(X^0)$, $k = \overline{1,n-1}$, separates B_2^n into two non-empty subsets consisting of the complete levels of the structure origin X^0:

$$V_1 = \bigcup_{i=1}^{k-1} O_i(X^0), \quad V_2 = \bigcup_{i=k+1}^{n} O_i(X^0). \qquad (2.1)$$

Denote as X_k^* the point of B_2^n which is defined from the condition $f(X_k^*) = \min\limits_{X \in O_k(X^0)} f(X)$. I.e., the point X_k^* supplies the minimal value of the function $f(X)$ on k-th level of X^0.

Lemma 2.1. *If f is an unimodal on B_2^n function and for a certain $\overline{X} \in O_1(X_k^*)$, $k = \overline{1, n-1}$, $f(\overline{X}) < f(X_k^*)$, $X_k^* \in O_k(X^0)$, is correct then the points $\overline{X}$ and X^* belong to alone subspace of B_2^n : V_1 or V_2.*

Proof. Presuppose the opposite: $\overline{X} \in V_1$, $X^* \in V_2$. And let $\overline{X} \in O_l(X^0)$, $l = \overline{0, k-1}$, $X^* \in O_p(X^0)$, $p = \overline{k+1, n}$. As f is unimodal function then for any point $\overline{X}$ there is the curve $W_-^{s+1}(\overline{X}, X^*) = \{\overline{X}, X^1, \dots, X^s, X^*\}$. By the curve definition the points $\overline{X}$ and $X^1, \dots, X^i$ and X^{i+1} $(i = \overline{2, s-1})$, X^s and X^* there are neighbouring points. And by Lemma 1.3 and Corollary 1.1 if $X \in O_i(X^0)$, $i = \overline{0, n}$, then all it's neighbouring points belong the level $O_{i-1}(X^0)$ or the level $O_{i+1}(X^0)$. Hence the curve $W_-^{s+1}(\overline{X}, X^*)$ will passe through every level $O_i(X^0)$, $i = \overline{l, p}$, including the level $O_k(X^0)$ as $l < k < p$ by the assumption. Then there is X' such that $X' \in W_-^{s+1}(\overline{X}, X^*)$ and $X' \in O_k(X^0)$.

If follows from the lemma condition and the curve definition that $f(X') < f(X_k^*)$. But from the definition of X_k^* we have that $f(X') \geqslant f(X_k^*)$. The obtained contradiction proves the lemma.

Corollary 2.1. If for an unimodal function f and certain points $\overline{X}_1, \overline{X}_2 \in O_k(X^0)$:

$$\begin{aligned} f(\overline{X}_1) &< f(X_{k-1}^*), \\ f(\overline{X}_2) &< f(X_{k+1}^*), \end{aligned} \tag{2.2}$$

then X^* is defined from the condition

$$f(X^*) = \min\{f(\overline{X}_1),\ f(\overline{X}_2)\}.$$

Proof. Determine the point X' from the condition $f(X') = \min\{f(\overline{X}_1), f(\overline{X}_2)\}$. Without losing generality assume that $X' = \overline{X}_1$. According to Lemma 1.3 all neighbouring to X' points lie on the levels $O_{k-1}(X^0)$ and $O_{k+1}(X^0)$.

By the condition $f(X') < f(X)$ $\forall X \in O_1(X') \cap O_{k-1}(X^0)$. And $\forall X \in O_1(X') \cap O_{k+1}(X^0)$ we have $f(X') \leqslant f(\overline{X}_2) < f(X^*_{k+1}) \leqslant f(X)$. Thus the point X' is a local minimum point, i.e., $X^* = X'$.

COROLLARY 2.2. If for a pseudoboolean function f there are the points $\overline{X}_1 \in O_{k-1}(X^0) \cap O_1(X^*_k)$ and $\overline{X}_2 \in O_{k+1}(X^0) \cap O_1(X^*_k)$ such that $f(\overline{X}_1) < f(X^*_k)$, $f(\overline{X}_2) < f(X^*_k)$, then the function f has at least two local minima X^*_1 and X^*_2 where $X^*_1 \in V_1$ and $X^*_2 \in V_2$ (the subsets V_1 and V_2 are defined (2.1)).

Proof. Directly follows from the lemma.

The proved lemma permits the following scheme of cutting off algorithms for case of unimodal of different values pseudoboolean functions to be proposed.

1. The point $X^0 \in B^n_2$ and a certain it's level $O_k(X^0)$, $k = \overline{1, n-1}$, are chosen arbitrarily. Suppose $l = 0$, $L = n$.

2. X^*_k and $\overline{X}_k$ are determined from the conditions:

$$f(X^*_k) = \min_{X \in O_k(X^0)} f(X), \tag{2.3}$$

$$f(\overline{X}_k) < f(X^*_k), \quad \overline{X}_k \in O_1(X^*_k). \tag{2.4}$$

If there is no $\overline{X}_k$ then $X^* = X^*_k$ and pass to item 5.

3. If $\overline{X}_k \in O_{k-1}(X^0)$ then $L = k$, $k = k - i$ $(i = \overline{1, k-l-1})$.

If $\overline{X}_k \in O_{k+1}(X^0)$ then $l = k$, $k = k + i$ $(i = \overline{1, L-k-1})$.

4. If $L - l = 2$ then from the condition

$$f(X^*) = \min \{f(\overline{X}_L, f(\overline{X}_l)\}$$

we determine X^*. Otherwise pass to item 2.

5. Stop.

Here l and L are numbers of first and last levels of the considered on step subspace.

Explain the scheme.

On first step the space B_2^n is devided into two subspaces (item 2). According to Lemma 2.1 we determine in which the subspace there is X^* (item 3). The obtained subspace similarly is devided into two ones (item 2. 3). And in this way until the subspace containing X^* will consist of one level (it is possible of course that the minimum will have been located before). After that X^* is determined by Corollary 2.1.

For the of undifferent values functions the scheme is analogous but the going out of constancy sets strategy is added.

The freedom of choice of the "cutting" level $O_k(X^0)$ for every step permits to construct the number of algorithms distinguishing the rule of determination of k in first and third items of the scheme. When an a priori information on the object function is absent it is natural to consider the "middle" levels and the zero point as X^0. As "middle" level we understande the level number of which is equal to the arithmetic mean (entire part) of the numberes of l and L or the level which devides the considering on the step subspace into two equivalent ones.

Consider the algorithm for which the "middle" level on step is determined by the numberes of first and last levels.

Algorithm 1

1. The point $X^0 \in B_2^n$ and it's k-th level $O_k(X^0)$, $k = \lceil n/2 \rceil$, are chosen arbitrarily. Suppose $l = 0$, $L = n$.

2. From the conditions (2.3) and (2.4) X_k^* and $\overline{X}_k$ are determined.

If there is no $\overline{X}_k$ then $X^* = X_k^*$ and pass to item 5.

3. If $\overline{X}_k \in O_{k-1}(X^0)$ then $L = k$,

$$k = \begin{cases} \lceil (k-l)/2 \rceil & \text{for } l < \lceil n/2 \rceil, \\ \lceil (k-l)/2 \rceil & \text{for } l \geqslant \lceil n/2 \rceil; \end{cases}$$

If $\overline{X}_k \in O_{k+1}(X^0)$ then $l = k$,

$$k = \begin{cases} \lceil (k+l)/2 \rceil & \text{for } l < \lceil n/2 \rceil, \\ \lceil (k+L)/2 \rceil & \text{for } l \geqslant \lceil n/2 \rceil. \end{cases}$$

4. If $L - l = 2$ then X^* from the condition: $f(X^*) = \min \{f(\overline{X}_L), f(\overline{X}_l)\}$ is determined. Otherwise pass to item 2.
5. Stop.

Here (and in the sequel) as $\lfloor a \rfloor$ a nearest to a integer, which is less than or equal to a, is denoted and analogously as $\lceil a \rceil$ a nearest to a integer, which is more than or equal to a, is denoted.

Note the simplicity of determination of a "cutting" level by the given algorithm. However that is poorly the "cutting" level devides a considered subspace into two non-equivalent ones (except the first subdivision for even n). It is connected with the binomial distribution of the points of B_2^n an levels (see Lemma 1.2). Therefore the points of the different subspaces have different rights in suspicion on minimum.

For next algorithm a "cutting" level devides the subspace into equivalent ones.

Algorithm 2

The items 1 and 2 coincide with the corresponding items of Algorithm 1.

3. If $\overline{X}_k \in O_{k-1}(X^0)$ then $L = k$, if $\overline{X}_k \in O_{k+1}(X^0)$ the $l = k$.

The number of next "cutting" level is found among of values $k = \overline{l+1, L-1}$ according to the levels cardinalities table for a given n (the triangle of Pascal).

The items 4 and 5 coincide with the Algorithm's items 4 and 5 too.

Estimate the Algorithm 1 effectiveness.

Lemma 2.2. *Locating of the minimum point X^* for an unimodal of different values on B_2^n function by Algorithm 1* *requires for the worst case (the estimate on top) the construction of $U_1(n)$ "cutting" levels,*

$$U_1(n) = \lfloor \log_2(n-1) \rfloor, \quad n > 1. \tag{2.5}$$

Proof. Conduct the induction by n. It is obvious that for small n the statement is correct. Let us assume that it is correct for certain $n-1$ too, i.e., $U_1(k) = \lfloor \log_2(k-1) \rfloor$ for $k = \overline{2, n-1}$. Now we show the correctness of (2.4) for n.

Devide the space B_2^n into two subspaces according to item 1 of Algorithm 1. The first "cutting" level has number $\lceil n/2 \rceil$. One of the subspaces consists of $\lceil n/2 \rceil + 1$ levels (including "cutting" level) another subspace consists of $\lfloor n/2 \rfloor$ levels.

Consider the subspace with greater number of levels. As for our algorithm the cardinality of levels is not important we may consider the chosen subspace as a space with dimension $\lceil n/2 \rceil$. By assumption for this space $U_1(\lceil n/2 \rceil) = \lfloor \log_2(\lceil n/2 \rceil - 1) \rfloor$ is correct. By the algorithm $U_1(n) = U_1(\lceil n/2 \rceil) + 1$, from which $U_1(n) = \lfloor \log_2(n/2 - 1) \rfloor + 1 = \lfloor \log_2(n-1) \rfloor$.

Theorem 2.1. *Locating of the minimum point of an unimodal of different values on B_2^n function by Algorithm 1 requires computating of it's values not less than in $C_n^{\lceil n/2 \rceil} + n$ and not more than in $T_1(n)$ points of $B_2^n (n > 4)$.*

$$T_1(n) = \sum_{i=1}^{\lfloor \log_2(n-1) \rfloor} C_n^{j_i} + n \lfloor \log_2(n-1) \rfloor, \tag{2.6}$$

where $j_0 = 0$, $j_1 = \lceil n/2 \rceil$, $j_i = \lceil (j_{i-1} + j_1)/2 \rceil$, $i = \overline{2, \lfloor \log_2(n-1) \rfloor}$.

Proof. The statement first part directly follows from the algorithm scheme and Lemma 1.2.

Prove the estimate $T_1(n)$ correctness. Without losing generality we will presuppose that $X^* \in \bigcup_{i=0}^{\lceil n/2 \rceil} O_i(X^0)$. Note that the difficult by the computations number case when $X^* \in O_{\lceil n/2 \rceil - 1}(X^0)$. According to Algorithm 1 and Lemma 2.1 in this case it is required to look over the maximal number $\lfloor \log_2(n-1) \rfloor$ of levels having numbers j_i which are defined in the following way:

$$j_0 = 0,\ j_1 = \lceil n/2 \rceil,\ j_i = \lceil (j_{i-1} + j_1)/2 \rceil,\ i = \overline{2, \lfloor \log_2(n-1) \rfloor}.$$

Moreover the neighbouring points to $X^*_{j_i}$, $i = \overline{1, \lfloor \log_2(n-1) \rfloor}$, are looked over. I.e., not more $n \lfloor \log_2(n-1) \rfloor$ computations of the object function are requires some more. Whence the estimate $T_1(n)$ follows.

COROLLARY 2.3.

$$\lim_{n \to \infty} \frac{card\, B_2^n}{T_1(n)} = \infty.$$

Proof. Consider the function $T_1'(n) = \lfloor \log_2(n-1) \rfloor C_n^{\lceil n/2 \rceil}$. For any $n > 7$ $T_1(n) < T_1'(n)$ is correct. By the formula of Stirling we have

$$C_n^{\lceil n/2 \rceil} = \begin{cases} \frac{2}{\sqrt{2\pi n}} & \text{for even } n, \\ \frac{2^{n+1}}{\sqrt{2\pi n}} \left(\frac{n}{n+1} \right) \epsilon & \text{for odd } n. \end{cases}$$

Whence

$$\lim_{n \to \infty} \frac{card\, B_2^n}{T_1(n)} > \lim_{n \to \infty} \frac{card\, B_2^n}{T_1'(n)} = \sqrt{\pi/2} \lim_{n \to \infty} \frac{\sqrt{n}}{\lfloor \log_2(n-1) \rfloor} = \infty.$$

COROLLARY 2.4. For $n > 13$

$$T_1(n) < 2^{n-1}. \qquad (2.7)$$

Thus already for small n Algorithm 1 excels the total examination in the convergence rate more than in twice. When n increases the algorithm advantage appreciably rises (see Fig. 1).

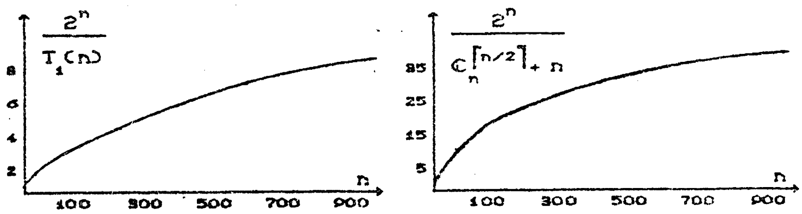

Fig. 1. Dependence of relation of the cutting off algorithms effectiveness to the total examination one from n. n – dimension, $T_1(n)$ – upper estimate, $C_n^{\lceil n/2 \rceil} + n$ – lower estimate.

For Algorithm 2 the estimates on top will not exceed the according estimates for Algorithm 1. Show it.

In Algorithm 2 by virtne of the binomial distribution of points on levels the second "cutting" level will be near to $\lceil n/2 \rceil$th one for any n.

Denoted as α_1 the number of levels between $\lceil n/2 \rceil$th level and next "cutting" one.

By Algorithm 2 the number of "cutting" level is determined by rule: the cardinality of nearest to $\lceil n/2 \rceil$th level subspace ought to be not less than the cardinality of other one and moreover the cardinalities of these subspace ought to aim at equivalence. Thus the α_1 such that

$$\sum_{i=0}^{\lceil n/2\rceil-\alpha_1-1} C_n^i \leqslant \sum_{i=\lceil n/2\rceil-\alpha_1}^{\lceil n/2\rceil-1} C_n^i. \qquad (2.8)$$

Now let n is even. Then the inequality (2.8) accepts from

$$\frac{2^n - C_n^{n/2}}{2} - \sum_{i=1}^{\alpha_1+1} C_n^{n/2-i} \leqslant \sum_{i=1}^{\alpha_1} C_n^{n/2-i}.$$

Using the formula of Stirling for $C_n^{n/2}$ and equality $C_n^m = C_n^{m-1}\frac{n-m+1}{m}$ we will have

$$2^n \leqslant C_n^{n/2}\Big(1 + 2\prod_{s=1}^{\alpha_1+1} \frac{n-2(s-1)}{n+2s} + 4\sum_{i=1}^{\alpha_1}\prod_{s=1}^{i} \frac{n-2(s-1)}{n+2s}\Big).$$

or

$$1.25\sqrt{n} \leqslant 1 + 2\prod_{s=1}^{\alpha_1+1} \frac{n-2(s-1)}{n+2s} + 4\sum_{i=1}^{\alpha_1}\prod_{s=1}^{i} \frac{n-2(s-1)}{n+2s}. \qquad (2.9)$$

For odd n we have

$$\frac{2^n - 2C_n^{\lceil n/2\rceil}}{2} - \sum_{i=2}^{\alpha_1+1} C_n^{\lceil n/2\rceil-i} \leqslant \sum_{i=1}^{\alpha_1} C_n^{\lceil n/2\rceil-i},$$

from which analogously (2.9)

$$1.25\sqrt{n} \leqslant \Big(4\sum_{i=1}^{\alpha_1}\prod_{s=1}^{i}\frac{n-2s+3}{n+2s-1} + 2\sum_{s=1}^{\alpha_1+1}\frac{n-2s+3}{n+2s-1} - 2\Big)\Big(\frac{n}{n+1}\Big)^{n+1}\epsilon \quad (2.10)$$

From (2.9), (2.10) α_1 and according dimension n are found easy. So for odd n $\alpha = 4$ if $n < 135$, for even n $\alpha = 4$ if $n < 196$. For large n may be found by following rough formules:

$$\begin{aligned} &1.25\sqrt{n} < 4\alpha_1 + 3, &&\text{for even } n \\ &1.25\sqrt{n} < 4\alpha_1, &&\text{for odd } n \end{aligned} \quad (2.11)$$

It follows from the said above and Theorem 2.1 that the number of "cutting" levels of Algorithm 2 does not exceed $U_2(n) = \lfloor\log_2 \alpha_1\rfloor + 2 = \lfloor\log_2(4\alpha_1)\rfloor)$, where α_1 is found from (2.9), (2.10).

$\alpha_2, \alpha_3, \ldots,$ according to subsequent "cutting" levels are determined similarly and similarly by the case of $T_1(n)$ the estimate on top of the number of computations of the objective function for Algorithm 2 $T_2(n)$ is determined:

$$T_2(n) = \sum_{i=1}^{t(n)} C_n^{j_i(n)} + nt(n),$$

where $j_i(n)$ are the numberes and $t(n)$ is number of the "cutting" levels defined for every n (see above).

As it was noted before the estimate $T_2(n)$ would not be worse than the estimate $T_1(n)$. As to the relation $card\, B_2^n / T_2(n)$ it will be more than two already for $n > 7$.

With regard for Corollary 2.2 the cutting off algorithms scheme for polymodal case is constructed similarly to considered above scheme. But the case of polymodal pseudoboolean functions requires additional detailed researches.

Conclusions. The proposed cutting off algorithms eliminate the total examination for the case of unimodal pseudoboolean functions. As to the case of polymodal functions by Antamoshkin A. and L.Lytkina (1990) it was proved that there were the polymodal pseudoboolean functions (consisting of the alternating levels of minima and maxima) the optimization of which was possible by the total examination only.

In comparison with the local optimization method it may be contended that the cutting off algorithms effectiveness is not less.

REFERENCES

Antamoshkin, A., and E. Semionkin (1991). Optimization of polymodal locally monotone pseudoboolean functions. *Informatica*, **2**(3), 331–351.

Antamoshkin, A. V. N. Saraev and E. S. Semionkin (1990). Optimization of unimodal monotone pseudoboolean functions. *Kybernetika*, **26**(5), 432–442.

Antamoshkin, A., and L. Lytkina (1990). The total examination elimination for the pseudoboolean optimization. In *Abs. of 15th Int. Conf. on Mathematical Optimization – Theory and Application*, Technische Hochschule, Ilmenau. pp. 8–11.

Received Jule 1991

A. Antamoshkin in 1977 maintained a candidate thesis at the Tomsk State University. At present he is a professor of Krasnoyarsk Space Technology University. His research interests are optimization, control and related topics.

Y. Koshkin is an assistant professor of Krasnoyarsk Space Technology University. The research interests field is the discreate optimization, the brainware, the complex object control models.