PAPER *Special Section on Discrete Mathematics and Its Applications*

Scaling Algorithms for M-convex Function Minimization

Satoko MORIGUCHI[†], **Kazuo MUROTA**[††], *and* **Akiyoshi SHIOURA**[†††], *Nonmembers*

SUMMARY M-convex functions have various desirable properties as convexity in discrete optimization. We can find a global minimum of an M-convex function by a greedy algorithm, i.e., so-called descent algorithms work for the minimization. In this paper, we apply a scaling technique to a greedy algorithm and propose an efficient algorithm for the minimization of an M-convex function. Computational results are also reported.

key words: matroid, convex function, scaling algorithm, discrete optimization.

1. Introduction

The concept of convexity for sets and functions plays a central role in continuous optimization (or nonlinear programming with continuous variable). It has various applications in the areas of mathematical economics, engineering, operations research, etc. [2], [23], [25]. The importance of convexity relies on the fact that a local minimum of a convex function is also a global minimum. Due to this property, we can find a global minimum of a convex function by iteratively moving in descent directions, i.e., so-called descent algorithms work for the convex function minimization.

In the area of discrete optimization, on the other hand, discrete analogues of convexity, or "discrete convexity" for short, have been considered, with a view to identifying the discrete structure that guarantees the success of descent methods, so-called "greedy algorithms." Examples of discrete convexity are "discretely-convex functions" by Miller [12] and "integrally-convex functions" by Favati-Tardella [7]. It would be natural to expect that discrete convexity yields a theory of "discrete convex analysis," which covers discrete analogues of the fundamental concepts such as conjugacy, subgradients, duality, and separation theorems. Unfortunately, neither "discretely-convex functions" nor "integrally-convex functions" seem to be fully suitable for such a theory. This suggests that we must identify a more restrictive class of well-behaved "discrete convex functions."

Manuscript received
Manuscript revised

[†]Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Tokyo, 152-8552 Japan.
[††]Graduate School of Information Science and Technology, University of Tokyo, Tokyo 113-0033, Japan.
[†††]Graduate School of Information Sciences, Tohoku University, Sendai, 980-8579, Japan.

A clue has been found in the theory of matroids and submodular functions, which has successfully captured the combinatorial essence underlying the well-solved class of combinatorial optimization problems such as those on graphs and networks (cf. [6], [8], [10]).

The relationship between convex functions and submodular functions was made clear through the works of Frank, Fujishige, and Lovász in the eighties. In particular, Lovász [11] pointed out that a set function is submodular if and only if the so-called Lovász extension of that function is convex.

Independent of the development in the theory of submodular functions, in the nineties, Dress and Wenzel [4], [5] introduced the concept of valuated matroids, which turned out to provide a nice combinatorial framework to which optimization algorithms for matroids can be generalized. Variants of greedy algorithms work for optimizing a matroid valuation [4], [13], and the weighted matroid intersection algorithm can be extended to the valuated matroid intersection problem [14], [15]. The duality theorem was shown by Murota [14] several years later, and the relationship to discrete convexity was recognized [16].

The concept of M-convex functions was proposed by Murota [18], [19] in 1996 as a natural extension of the concept of valuated matroids. Let V be a finite set. A function $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ is said to be M-convex if it satisfies

(M-EXC) $\forall x, y \in \operatorname{dom} f$, $\forall u \in \operatorname{supp}^+(x-y)$, $\exists v \in \operatorname{supp}^-(x-y)$ such that

$$f(x) + f(y) \geqq f(x - \chi_u + \chi_v) + f(y + \chi_u - \chi_v),$$

where $\chi_w \in \{0,1\}^V$ is the characteristic vector of $w \in V$ and

$$\begin{aligned} \operatorname{dom} f &= \{x \in \mathbf{Z}^V \mid f(x) < +\infty\}, \\ \operatorname{supp}^+(x-y) &= \{w \in V \mid x(w) > y(w)\}, \\ \operatorname{supp}^-(x-y) &= \{w \in V \mid x(w) < y(w)\}. \end{aligned}$$

It is easy to see that $B = \operatorname{dom} f$ satisfies the following property:

(B-EXC) $\forall x, y \in B$, $\forall u \in \operatorname{supp}^+(x-y)$, $\exists v \in \operatorname{supp}^-(x-y)$ such that

$$x - \chi_u + \chi_v \in B, \; y + \chi_u - \chi_v \in B.$$

Note that (B-EXC) implies $\sum_{v\in V} x(v) = \sum_{v\in V} y(v)$ for any $x, y \in B$. A nonempty set $B \subseteqq \mathbf{Z}^V$ with (B-EXC) is called an M-convex set.

M-convexity is quite a natural concept appearing in many situations; linear and separable-convex functions are M-convex, and more general M-convex functions arise from the minimum cost flow problem with separable-convex cost functions. M-convex functions have various desirable properties as discrete convexity:
(i) local minimality leads to global minimality for M-convex functions,
(ii) M-convex functions can be extended to ordinary convex functions,
(iii) various duality theorems hold.
In particular, the property (i) shows that greedy algorithms (descent algorithms) work for the minimization of an M-convex function. A theory of "discrete convex analysis" [19]–[21] has been developed with the use of M-convex functions.

In this paper, we consider the problem of minimizing an M-convex function. Although an M-convex function can be minimized by a descent algorithm, it may require exponential time. A steepest descent algorithm (see §4), a faster version of a descent algorithm, terminates in pseudo-polynomial time. The domain reduction-type polynomial time algorithm of Shioura [24] has the time complexity $\mathrm{O}(n^4(\log L)^2)$, where

$$n = |V|, \qquad L = \max\{\|x - y\|_\infty \mid x, y \in \mathrm{dom}\, f\}.$$

Although the domain reduction-type algorithm has polynomial time complexity, our numerical experiments show that it does not run fast in practice.

The objective of this paper is to propose faster polynomial time algorithms for the minimization of an M-convex function by using a scaling technique. Scaling is a fundamental technique used extensively in polynomial time algorithms for combinatorial optimization problems. Indeed, scaling-based algorithms achieve better time complexities for the resource allocation problem [9], the minimum cost flow problem [1], etc.

We propose efficient minimization algorithms for functions in the class of M-convex functions closed under the scaling operation. Some fundamental classes of M-convex functions such as separable convex functions and quadratic M-convex functions are closed under the scaling operation, although this is not the case with general M-convex functions. We apply the scaling technique to a steepest descent algorithm to obtain faster algorithms.

In order to compare the performance of our new scaling algorithms to those of the previously proposed algorithms, we make numerical experiments with randomly generated test problems. It is observed from numerical results that our new scaling algorithms are much faster than the previously proposed algorithms from the viewpoint of both theory and practice.

2. Scaling of M-convex Functions

For $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$, a positive integer α and a vector $b \in \mathbf{Z}^V$, define a function $f^{\alpha,b} : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ by

$$f^{\alpha,b}(x) = f(\alpha x + b) \qquad (x \in \mathbf{Z}^V).$$

This operation is called *scaling*. Even if f is an M-convex function, $f^{\alpha,b}$ is not necessarily M-convex in general. We can still identify a number of subclasses of M-convex functions that are closed under the scaling operation.

Example 2.1 (Separable convex functions)**:**
For $S \subseteqq V$, we define $x(S) = \sum_{v\in S} x(v)$. For a family of convex functions $f_i : \mathbf{Z} \to \mathbf{R}$ indexed by $i \in V$ and an integer β, the (separable convex) function $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ defined by

$$f(x) = \begin{cases} \displaystyle\sum_{i=1}^{n} f_i(x_i) & \text{if } x(V) = \beta, \\ +\infty & \text{otherwise} \end{cases}$$

is M-convex.

Since $f^{\alpha,b}(x) = \sum_{i=1}^{n} f_i(\alpha x_i + b_i)$ is also a separable convex function, the class of separable convex functions is closed under the scaling operation.

Example 2.2 (Quadratic M-convex functions)**:**
Let $A = (a_{ij}) \in \mathbf{R}^{n\times n}$ be a symmetric matrix. A quadratic function $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ given by

$$f(x) = \begin{cases} \frac{1}{2} x^{\mathrm{T}} A x & \text{if } x(V) = 0, \\ +\infty & \text{otherwise} \end{cases}$$

is M-convex if and only if

$$\forall i, j, k, l \in V \text{ with } \{i, j\} \cap \{k, l\} = \emptyset,$$
$$a_{ij} + a_{kl} \geqq \min\{a_{ik} + a_{jl}, a_{il} + a_{jk}\}$$

(see [21], [22]). For a quadratic M-convex function f, the function $f^{\alpha,b}$ is written as

$$\begin{aligned} f^{\alpha,b}(x) &= \frac{1}{2}(\alpha x + b)^{\mathrm{T}} A(\alpha x + b) \\ &= \frac{\alpha^2}{2} x^{\mathrm{T}} A x + \alpha b^{\mathrm{T}} A x + \frac{1}{2} b^{\mathrm{T}} A b. \end{aligned}$$

This expression shows that the function $f^{\alpha,b}$ is M-convex. Therefore, the class of quadratic M-convex functions is closed under the scaling operation.

Example 2.3 (Laminar convex functions)**:**
A nonempty family $\mathcal{T}$ of subsets of V is called a *laminar family* if it satisfies the following property:

$$\forall X, Y \in \mathcal{T} : X \cap Y = \emptyset \text{ or } X \subseteqq Y \text{ or } X \supseteqq Y.$$

Given a laminar family $\mathcal{T}$, a family of convex functions $f_X : \mathbf{Z} \to \mathbf{R}$ indexed by $X \in \mathcal{T}$, and an integer β, define a function $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ by

$$f(x) = \begin{cases} \sum_{X \in \mathcal{T}} f_X(x(X)) & \text{if } x(V) = \beta, \\ +\infty & \text{otherwise.} \end{cases}$$

We call a function f of this type a *laminar convex function*. We show that laminar convex functions constitute a class of M-convex functions closed under the scaling operation (see also [3], [21]).

Without loss of generality, assume $V \in \mathcal{T}$. Otherwise, we can add V to $\mathcal{T}$ and put $f_V(\alpha) = 0$ ($\forall \alpha \in \mathbf{Z}$). For $X \subseteqq V$, we denote by $\mathcal{T}(X)$ the family of all maximal proper subsets of X in $\mathcal{T}$. For any $x \in \mathbf{Z}^V$ and $X \subseteqq V$, we have

$$\begin{aligned} x(X) = \; & \sum \{x(Y) \mid Y \in \mathcal{T}(X)\} \\ & + \sum \{x(v) \mid v \in X \setminus \bigcup_{Y \in \mathcal{T}(X)} Y\}. \end{aligned} \tag{1}$$

Take any $x, y \in \operatorname{dom} f$ and $u \in \operatorname{supp}^+(x - y)$. To prove (M-EXC), it suffices to show that there exists some $v \in \operatorname{supp}^-(x - y)$ satisfying

$$u \in X,\ v \notin X,\ X \in \mathcal{T} \implies x(X) > y(X) \tag{2}$$

and

$$u \notin X,\ v \in X,\ X \in \mathcal{T} \implies x(X) < y(X). \tag{3}$$

Let X_0 be the unique minimal set in $\mathcal{T}$ satisfying $u \in X$ and $x(X) \leqq y(X)$. By the minimality of X_0 and (1), there are two cases:

(i) $\exists v \in X_0 \setminus \bigcup_{Y \in \mathcal{T}(X_0)} Y : x(v) < y(v)$,

(ii) $\exists X_1 \in \mathcal{T}(X_0) : x(X_1) < y(X_1)$.

In case of (i), this v satisfies (2) and (3). In case of (ii), from (1) follows

(i) $\exists v \in X_1 \setminus \bigcup_{Y \in \mathcal{T}(X_1)} Y : x(v) < y(v)$, or

(ii) $\exists X_2 \in \mathcal{T}(X_1) : x(X_2) < y(X_2)$.

Repeating this argument, we reach the case (i). Therefore, a laminar convex function is M-convex.

Moreover,

$$f^{\alpha,b}(x) = \sum_{X \in \mathcal{T}} f_X(\alpha x(X) + b(X))$$

is a laminar convex function. Therefore the class of laminar convex functions is closed under the scaling operation.

3. Theorems on the Minimizers of M-convex Functions

In this section, we show properties of the minimizers of M-convex functions.

Global minimality of an M-convex function is characterized by local minimality.

Theorem 3.1 ([18], [19])**:** Let $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ be a function with (M-EXC). For $x \in \operatorname{dom} f$, $f(x) \leqq f(y)$ ($\forall y \in \mathbf{Z}^V$) if and only if $f(x) \leqq f(x - \chi_u + \chi_v)$ ($\forall u, v \in V$). ■

Any vector in $\operatorname{dom} f$ can be easily separated from some minimizer of f.

Theorem 3.2 ([24])**:** Let $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ be a function with (M-EXC). Assume $\arg\min f \neq \emptyset$.

(i) For $x \in \operatorname{dom} f$ and $v \in V$, let $u \in V$ satisfy $f(x - \chi_u + \chi_v) = \min_{s \in V} f(x - \chi_s + \chi_v)$. Set $x' = x - \chi_u + \chi_v$. Then, there exists $x^* \in \arg\min f$ with $x^*(u) \leqq x'(u)$.

(ii) For $x \in \operatorname{dom} f$ and $u \in V$, let $v \in V$ satisfy $f(x - \chi_u + \chi_v) = \min_{t \in V} f(x - \chi_u + \chi_t)$. Set $x' = x - \chi_u + \chi_v$. Then, there exists $x^* \in \arg\min f$ with $x^*(v) \geqq x'(v)$. ■

Corollary 3.3 ([24])**:** Let $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ be an M-convex function. Assume $\arg\min f \neq \emptyset$. Let $x \in \operatorname{dom} f$ with $x \notin \arg\min f$, and $u, v \in V$ satisfy

$$f(x - \chi_u + \chi_v) = \min_{s,t \in V} f(x - \chi_s + \chi_t).$$

Then, there exists $x^* \in \arg\min f$ with $x^*(u) \leqq x(u) - 1$, $x^*(v) \geqq x(v) + 1$. ■

Let α be a positive integer, and $x_\alpha \in \operatorname{dom} f$. We call x_α an α-local minimum of f if it satisfies

$$f(x_\alpha) \leqq f(x_\alpha - \alpha(\chi_u - \chi_v)) \ (\forall u, v \in V).$$

The following is a "proximity theorem," showing that a global minimizer of an M-convex function exists in the neighborhood of an α-local minimum.

Theorem 3.4: Let $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ be an M-convex function and α be any positive integer. Suppose that $x_\alpha \in \operatorname{dom} f$ satisfies $f(x_\alpha) \leqq f(x_\alpha - \alpha(\chi_u - \chi_v))$ for all $u, v \in V$. Then, there exists some $x_* \in \arg\min f$ such that

$$|x_\alpha(v) - x_*(v)| \leqq (n - 1)(\alpha - 1) \qquad (v \in V). \tag{4}$$

■

Proof. It suffices to show that for any $\gamma > \inf f$ there exists some $x_* \in \operatorname{dom} f$ satisfying $f(x_*) \leqq \gamma$ and (4).

Let $x_* \in \operatorname{dom} f$ satisfy $f(x_*) \leqq \gamma$, and suppose that x_* minimizes the value $\|x_* - x_\alpha\|_1$ among all such vectors. In the following, we fix $v \in V$ and prove $x_\alpha(v) - x_*(v) \leqq (n - 1)(\alpha - 1)$. The inequality $x_*(v) - x_\alpha(v) \leqq (n - 1)(\alpha - 1)$ can be shown similarly.

We may assume $x_\alpha(v) > x_*(v)$. We first prove the following two claims. Let $k = x_\alpha(v) - x_*(v)$.

Claim 1: There exist $w_1, w_2, \cdots, w_k \in V \setminus \{v\}$ and $y_0 (= x_\alpha), y_1, \cdots, y_k \in \operatorname{dom} f$ such that

$$y_i = y_{i-1} - \chi_v + \chi_{w_i},\ f(y_i) < f(y_{i-1}) \ (i = 1, \cdots, k).$$

[Proof of Claim 1] We show the claim by induction on i. Suppose $y_{i-1} \in \operatorname{dom} f$. By (M-EXC) applied to y_{i-1}, x_*, and $v \in \operatorname{supp}^+(y_{i-1} - x_*)$, we have some $w_i \in \operatorname{supp}^-(y_{i-1} - x_*) \subseteqq \operatorname{supp}^-(x_\alpha - x_*) \subseteqq V \setminus \{v\}$ such that $f(x_*) + f(y_{i-1}) \geqq f(x_* - \chi_{w_i} + \chi_v) + f(y_{i-1} + \chi_{w_i} - \chi_v)$. By the choice of x_*, we have $f(x_* + \chi_v - \chi_{w_i}) > f(x^*)$ since $\|(x_* + \chi_v - \chi_{w_i}) - x_\alpha\|_1 < \|x_* - x_\alpha\|_1$. Therefore,

$f(y_i) = f(y_{i-1} - \chi_v + \chi_{w_i}) < f(y_{i-1})$. [End of Proof for Claim 1]

Claim 2: For any $w \in V \setminus \{v\}$ with $y_k(w) > x_\alpha(w)$ and $\mu \in [0, y_k(w) - x_\alpha(w) - 1]$, we have

$$f(x_\alpha - (\mu+1)(\chi_v - \chi_w)) < f(x_\alpha - \mu(\chi_v - \chi_w)). \quad (5)$$

[Proof of Claim 2] We prove (5) by induction on μ. Put $x' = x_\alpha - \mu(\chi_v - \chi_w)$ for $\mu \in [0, y_k(w) - x_\alpha(w) - 1]$, and suppose $x' \in \mathrm{dom}\, f$ by induction hypothesis. Let j_* $(1 \leqq j_* \leqq k)$ be the largest index such that $w_{j_*} = w$. Then, $y_{j_*}(w) = y_k(w) > x'(w)$ and $\mathrm{supp}^-(y_{j_*} - x') = \{v\}$. (M-EXC) implies that $f(x') + f(y_{j_*}) \geqq f(x' - \chi_v + \chi_w) + f(y_{j_*} + \chi_v - \chi_w)$. By Claim 1, we have $f(y_{j_*} + \chi_v - \chi_w) > f(y_{j_*})$. Hence, (5) follows. [End of Proof for Claim 2]

The α-local minimality of x_α implies $f(x_\alpha - \alpha(\chi_v - \chi_w)) \geqq f(x_\alpha)$, which, combined with Claim 2, implies $y_k(w) - x_\alpha(w) \leqq \alpha - 1$ for all $w \in V \setminus \{v\}$. Thus,

$$\begin{aligned} x_\alpha(v) - x_*(v) &= x_\alpha(v) - y_k(v) \\ &= \sum_{w \in V \setminus \{v\}} \{y_k(w) - x_\alpha(w)\} \\ &\leqq (n-1)(\alpha - 1), \end{aligned}$$

where the second equality is by $x(V) = y(V)$ $(\forall x, y \in \mathrm{dom}\, f)$. □

4. Minimization Algorithms of an M-convex Function

In the previous section we gave some theorems on the minimizers of an M-convex function. Based on these theorems, we obtain several algorithms for the minimization of an M-convex function. In this section we describe previous algorithms and propose new algorithms based on a scaling technique.

4.1 Previous Algorithms

Let $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ be a function such that $\mathrm{dom}\, f$ is a nonempty bounded set, and put

$$L = \max\{\|x - y\|_\infty \mid x, y \in \mathrm{dom}\, f\}.$$

We can compute the value L in $\mathrm{O}(n^2 \log L)$ time by $\mathrm{O}(n^2)$-time evaluation of the exchange capacity. For $x \in B$ and $u, v \in V$, the exchange capacity associated with x, v and u is defined as $\tilde{c}_B(x, v, u) = \max\{\alpha \mid x - \alpha(\chi_u - \chi_v) \in B\}$. The exchange capacity can be computed in $\mathrm{O}(\log L)$ time. See [24] for details.

Assume (M-EXC) for f. Then, Theorem 3.1 immediately leads to the following algorithm.

Algorithm Descent
S0: Let x be any vector in $\mathrm{dom}\, f$.
S1: If $f(x) = \min_{s,t \in V} f(x - \chi_s + \chi_t)$ then stop [x is a minimizer of f].
S2: Find $u, v \in V$ with $f(x - \chi_u + \chi_v) < f(x)$.
S3: Set $x := x - \chi_u + \chi_v$. Go to S1. □

This algorithm always terminates since the function value of x decreases strictly in each iteration. However, there is no guarantee for the polynomiality of the number of iterations. The number of iterations is at most $|\mathrm{dom}\, f| \leqq (L+1)^{n-1}$ since the algorithm generates distinct x in each iteration.

The following is a faster version of Algorithm Descent which exploits Corollary 3.3.

Algorithm Steepest_Descent
S0: Let x be any vector in $\mathrm{dom}\, f$. Set $B := \mathrm{dom}\, f$.
S1: If $f(x) = \min_{s,t \in V} f(x - \chi_s + \chi_t)$ then stop [x is a minimizer of f].
S2: Find $u, v \in V$ with $x - \chi_u + \chi_v \in B$ satisfying

$$f(x - \chi_u + \chi_v) = \min\{f(x - \chi_s + \chi_t) \mid s, t \in V, x - \chi_s + \chi_t \in B\}.$$

S3: Set

$$B := B \cap \{y \in \mathbf{Z}^V \mid y(u) \leqq x(u) - 1,\ y(v) \geqq x(v) + 1\}$$

and $x := x - \chi_u + \chi_v$. Go to S1. □

By Corollary 3.3, the set B always contains a minimizer of f. Hence, Algorithm Steepest_Descent finds a minimizer of f. To analyze the number of iterations, we consider the value

$$\sum_{w \in V} \{\max_{y \in B} y(w) - \min_{y \in B} y(w)\}.$$

This value is bounded by nL and decreases at least by two in each iteration. Therefore, Steepest_Descent terminates in $\mathrm{O}(nL)$ iterations. Each iteration can be done in $\mathrm{O}(n^2)$ time. Therefore, Algorithm Steepest_Descent finds a minimizer of f in $\mathrm{O}(n^3 L)$ time, i.e., Steepest_Descent is a pseudo-polynomial time algorithm. In particular, if $\mathrm{dom}\, f \subseteq \{0,1\}^V$ then the number of iterations is $\mathrm{O}(n)$.

We propose the following modified version of Algorithm Steepest_Descent by exploiting Theorem 3.2 (i).

Algorithm MODIFIED_STEEPEST_DESCENT
S0: Let x be any vector in $\mathrm{dom}\, f$. Set $B := \mathrm{dom}\, f$.
S1: Choose $u \in V$ such that $x - \chi_u + \chi_v \in B$ for some $v \in V \setminus \{v\}$. If there is no such u then stop [x is a minimizer of f].
S2: For u, find $v \in V$ with $x - \chi_u + \chi_v \in B$ satisfying

$$f(x - \chi_u + \chi_v) = \min\{f(x - \chi_u + \chi_t) \mid t \in V,\ x - \chi_u + \chi_t \in B\}.$$

S3: Set $B := B \cap \{y \in \mathbf{Z}^V \mid y(v) \geqq x(v) + 1\}$ and $x := x - \chi_u + \chi_v$. Go to S1. □

Although the number of iterations of Algorithm MODIFIED_STEEPEST_DESCENT is equal to that of Algorithm STEEPEST_DESCENT, each iteration of MODIFIED_STEEPEST_DESCENT can be done in $\mathrm{O}(n)$ time, while each iteration of STEEPEST_DESCENT can be done in $\mathrm{O}(n^2)$ time. MODIFIED_STEEPEST_DESCENT is also a pseudo-polynomial time algorithm.

It is shown in [24] that the minimization of an M-convex function can be done in polynomial time by the domain reduction method explained below. Given a bounded M-convex set B, we define the set $N_B \subseteqq B$ as follows. For $w \in V$, define

$$l_B(w) = \min_{y \in B} y(w), \qquad u_B(w) = \max_{y \in B} y(w),$$

$$l'_B(w) = \left\lfloor (1 - \frac{1}{n}) l_B(w) + \frac{1}{n} u_B(w) \right\rfloor,$$

$$u'_B(w) = \left\lceil \frac{1}{n} l_B(w) + (1 - \frac{1}{n}) u_B(w) \right\rceil.$$

Then, N_B is defined as

$$N_B = \{y \in B \mid l'_B(w) \leqq y(w) \leqq u'_B(w)\ (\forall w \in V)\}.$$

Theorem 4.1 ([24])**:** N_B is a (nonempty) M-convex set. ■

The next algorithm maintains a set $B \subseteqq \mathrm{dom}\, f$ which is an M-convex set containing a minimizer of f. It reduces B iteratively by exploiting Corollary 3.3 and finally finds a minimizer.

Algorithm DOMAIN_REDUCTION
S0: Set $B := \mathrm{dom}\, f$.
S1: Find a vector $x \in N_B$.
S2: If $f(x) = \min_{s,t \in V} f(x - \chi_s + \chi_t)$ then stop [x is a minimizer of f].
S3: Find $u, v \in V$ with $x - \chi_u + \chi_v \in B$ satisfying

$$f(x - \chi_u + \chi_v) = \min\{f(x - \chi_s + \chi_t) \mid s, t \in V,\ x - \chi_s + \chi_t \in B\}.$$

S4: Set

$$B := B \cap \{y \in \mathbf{Z}^V \mid y(u) \leqq x(u) - 1,\ y(v) \geqq x(v) + 1\}.$$

Go to S1. □

Theorem 4.2 ([24])**:** If a vector in $\mathrm{dom}\, f$ and the value L are given, Algorithm DOMAIN_REDUCTION finds a minimizer of f in $\mathrm{O}(n^4 (\log L)^2)$ time. ■

4.2 Scaling Algorithms

In this section, we propose efficient algorithms with a scaling technique. We apply a scaling technique to Algorithm STEEPEST_DESCENT to obtain a faster algorithm. To the end of this section we assume that f is an M-convex function which is closed under the scaling operation.

Algorithm SCALING_STEEPEST_DESCENT
S0: Put $\alpha := 2^{\lceil \log(L/4n) \rceil}$, $B := \mathrm{dom}\, f$. Let $x_{2\alpha}$ be any vector in $\mathrm{dom}\, f$.
S1: [α-scaling phase]Define $\tilde{f} : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ by

$$\tilde{f}(y) = \begin{cases} f(x_{2\alpha} + \alpha y) & \text{if } x_{2\alpha} + \alpha y \in B, \\ +\infty & \text{if } x_{2\alpha} + \alpha y \notin B. \end{cases}$$

Compute a minimizer y_* of $\tilde{f}$ by applying Algorithm STEEPEST_DESCENT.
Set $x_\alpha = x_{2\alpha} + \alpha y_*$.
S2: If $\alpha = 1$ then stop [x_α is a minimizer of f].
S3: Put

$$B := B \cap \{y \in \mathbf{Z}^V \mid x_\alpha(w) - (n-1)(\alpha - 1) \leqq y(w) \leqq x_\alpha(w) + (n-1)(\alpha - 1)\ (\forall w \in V)\}$$

and $\alpha := \alpha/2$. Go to S1. □

We analyze the time complexity of Algorithm SCALING_STEEPEST_DESCENT for a function closed under the scaling operation. The number of scaling phases is $\lceil \log(L/4n) \rceil$. Since the number of iterations in each scaling phase is $(4n\alpha \times n)/\alpha$, each scaling phase terminates in $\mathrm{O}((4n\alpha \times n)/\alpha \times n^2) = \mathrm{O}(n^4)$ time. We can compute the value L in $\mathrm{O}(n^2 \log L)$ time. Here, we have the following theorem.

Theorem 4.3: Suppose that $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ satisfies (M-EXC) and is closed under the scaling operation. If a vector in $\mathrm{dom}\, f$ is given, Algorithm SCALING_STEEPEST_DESCENT finds a minimizer of f in $\mathrm{O}(n^4 \log(L/n))$ † time. ■

Algorithm SCALING_STEEPEST_DESCENT above can be improved further by using MODIFIED_STEEPEST_DESCENT in place of STEEPEST_DESCENT in each scaling phase. We refer to the algorithm resulting from this modification as SCALING_MODIFIED_STEEPEST_DESCENT. Each scaling phase of SCALING_MODIFIED_STEEPEST_DESCENT terminates in $\mathrm{O}(n^3)$ time, and therefore, its overall time complexity for finding a minimizer of f is $\mathrm{O}(n^3 \log(L/n))$. Thus the replacement of STEEPEST_DESCENT by MODIFIED_STEEPEST_DESCENT results in an $\mathrm{O}(n)$ improvement upon SCALING_STEEPEST_DESCENT.

Theorem 4.4: Suppose that $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ satisfies (M-EXC) and is closed under the scaling operation. If a vector in $\mathrm{dom}\, f$ is given, Algorithm SCALING_MODIFIED_STEEPEST_DESCENT finds a minimizer of f in $\mathrm{O}(n^3 \log(L/n))$ time. ■

† $\mathrm{O}(n^4 \max\{\log(L/n), 1\})$, to be more precise. Similarly for Theorem 4.4.

5. Numerical Experiments

We here mainly compare the performance of our new algorithms SCALING_STEEPEST_DESCENT (SSD) and SCALING_MODIFIED_STEEPEST_DESCENT (SMSD) to those of STEEPEST_DESCENT (SD) and DOMAIN_REDUCTION (DR). We observe from numerical experiments that our algorithms are much faster than the previous algorithms. Section 5.1 briefly explains test problems used in numerical experiments and our implementation. Section 5.2 compares new scaling algorithms to the previously proposed algorithms.

5.1 Test Problems and Implementation

As test problems we consider the minimization of a quadratic laminar convex function of the following form:

$$\begin{aligned} \text{minimize} \quad & \sum_{X \in \mathcal{T}} \{a_X x(X)^2 + b_X x(X) + c_X\} \\ \text{subject to} \quad & \sum_{i=1}^{n} x(i) = L, \\ & x_i \geqq 0, \ \text{integer}, \ i = 1, \ldots, n. \end{aligned}$$

For each n and L fixed (dimension of the variable x and the sum of $x(i)$, respectively), we generate ten test problems with randomly chosen real variables $0 \leqq a_X,\ b_X,\ c_X \leqq 1000$ ($X \in \mathcal{T}$) and laminar families $\mathcal{T}$. The C language function `random()` is used to generate these pseudo-random numbers. We measure the execution time and present average execution times of ten generated test problems for each size. The two main parameters n and L have a strong influence on the execution time. We make experiments with test problems of various sizes by changing n and L.

In our implementation, we tailored DR for the minimization of a laminar convex function, in which the following algorithm is used to find a vector x in N_B.

Algorithm FIND_VECTOR_IN_N_B
S1: For each $w \in V$, compute $l'_B(w)$ and $u'_B(w)$.
S2: For $w = 1, 2, \cdots, n$, put

$$x(w) = \begin{cases} u'_B(w) \\ \quad \text{if } \displaystyle\sum_{i=1}^{w-1} x(i) + u'_B(w) + \sum_{i=w+1}^{n} l'_B(i) \leqq L, \\ L - \displaystyle\sum_{i=1}^{w-1} x(i) - \sum_{i=w+1}^{n} l'_B(i) \\ \quad \text{otherwise.} \end{cases}$$

□

Algorithm FIND_VECTOR_IN_N_B finds a vector in N_B in $\mathrm{O}(n)$ time. The time complexity of the specialized DR is $\mathrm{O}(n^4 \log L)$ while those of DR mentioned in Section 4 is $\mathrm{O}(n^4 (\log L)^2)$.

Also, in our implementations of SD, DR, SSD and SMSD, it takes $\mathrm{O}(n)$ time to evaluate the function value. Hence, the execution time in our numerical experiments is $\mathrm{O}(n)$ times larger than the theoretical time complexity.

Each of SD, DR, SSD and SMSD is written in the C language, compiled under a personal computer with the CPU Pentium III 450MHz and 256 MB of memory under Vine Linux (a Linux distribution based on Red Hat Linux) V1.1 using the compiler pgcc 2.95.2 with the option -mcpu=pentiumpro -march=pentiumpro -O9 -funroll-loops.

5.2 Computational Results

Our numerical results are summarized in Figures 1 and 2. Figure 1 shows the relationship between the computation time T and the dimension n for the case of $L = 50000$. In all the four algorithms the relationship is linear in $\log T$ and $\log n$, which implies $T = \mathrm{O}(n^p)$ for some p. Our results show the following:

Algorithm	SD	DR	SSD	SMSD
Time T	$n^{2.16}$	$n^{3.89}$	$n^{3.70}$	$n^{2.96}$

Figure 2 shows the relationship between the computation time T and the size of the effective domain L for the case of $n = 100$. L is given in log scale whereas time T is on linear scale in this graph. It is verified that $T = \mathrm{O}(\log L)$ in SSD and SMSD, $T = \mathrm{O}((\log L)^2)$ in DR and $T = \mathrm{O}(L)$ in SD.

Fig. 1 The execution time in the case $L = 50000$.

Fig. 2 The execution time in the case $n = 100$.

The table below shows the standard deviations of execution times in the case of $L = 50000$ and $n = 100$, which is the case of the largest problems in our numerical experiments.

Algorithm	SD	DR	SSD	SMSD
Stand. Dev.	2.770	10.25	0.9075	0.2357

By numerical experiments with randomly generated test problems, we can conclude that our scaling algorithms are faster than the previously proposed algorithms. In particular, Algorithm SMSD is the fastest algorithm among algorithms we considered.

6. Conclusion

Although our scaling algorithms run in polynomial time only for a restricted class of M-convex functions, our scaling approach can be polished up to a polynomial time algorithm applicable to general M-convex functions by Tamura [26].

The authors thank Yoshitsugu Yamamoto for a stimulating comment.

References

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, Network Flows – Theory, Algorithms, and Applications, Prentice Hall, New Jersey, 1993.
[2] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty, Nonlinear Programming: Theory and Algorithm (Second Edition), John Wiley and Sons, New York, 1993.
[3] V. Danilov, G. Koshevoy, and K. Murota, "Discrete convexity and equilibria in economies with indivisible goods and money," Math. Social Sciences, vol.41, pp.251–273, 2001.
[4] A.W.M. Dress and W. Wenzel, "Valuated matroid: a new look at the greedy algorithm," Appl. Math. Lett., vol.3, pp.33–35, 1990.
[5] A.W.M. Dress and W. Wenzel, "Valuated matroids," Adv. Math., vol.93, pp.214–250, 1992.
[6] U. Faigle, "Matroids in combinatorial optimization," in Combinatorial Geometries, ed. N. White, pp.161–210, Cambridge University Press, London, 1987.
[7] P. Favati and F. Tardella, "Convexity in nonlinear integer programming," Ricerca Operativa, vol.53, pp.3–44, 1990.
[8] S. Fujishige, Submodular Functions and Optimization, Annals of Discrete Math., vol.47, North-Holland, Amsterdam, 1991.
[9] D.S. Hochbaum, "Lower and upper bounds for the allocation problem and other nonlinear optimization problems," Math. Oper. Res., vol.19, pp.390–409, 1994.
[10] E.L. Lawler, Combinatorial Optimization: Networks and Matroids, Holt, Rinehart and Winston, New York, 1976.
[11] L. Lovász, "Submodular functions and convexity," in Mathematical Programming — The State of the Art, eds. A. Bachem, M. Grötschel, and B. Korte, pp.235–257, Springer, Berlin, 1983.
[12] B.L. Miller, "On minimizing nonseparable functions defined on the integers with an investory application," SIAM J. Appl. Math., vol.21, pp.166–185, 1971.
[13] K. Murota, "Finding optimal minors of valuated bimatroids," Appl. Math. Lett., vol.8, pp.37–42, 1995.
[14] K. Murota, "Valuated matroid intersection, I: optimality criteria," SIAM J. Discrete Math., vol.9, pp.545–561, 1996.
[15] K. Murota, "Valuated matroid intersection, II: algorithms," SIAM J. Discrete Math., vol.9, pp.562–576, 1996.
[16] K. Murota, "Fenchel-type duality for matroid valuations," Math. Programming, vol.82, pp.357–375, 1998.
[17] K. Murota, "Submodular flow problem with a nonseparable cost function," Combinatorica, vol.19, pp.87–109, 1999.
[18] K. Murota, "Convexity and Steinitz's exchange property," Adv. Math., vol.124, pp.272–311, 1996.
[19] K. Murota, "Discrete convex analysis," Math. Programming, vol.83, pp.313–371, 1998.
[20] K. Murota, "Discrete convex analysis — Exposition on conjugacy and duality," in Graph Theory and Combinatorial Biology, eds. L. Lovász et al., pp.253–278, The János Bolyai Mathematical Society, 1999.
[21] K. Murota, Discrete Convex Analysis—An Introduction, Kyoritsu Publishing Co., Tokyo, 2001. [In Japanese]
[22] K. Murota and A. Shioura, "Quadratic M-convex and L-convex functions," RIMS Preprint No. 1326, Kyoto University, 2001.
[23] R.T. Rockafellar, Convex Analysis, Princeton University Press, Princeton, 1970.
[24] A. Shioura, "Minimization of an M-convex function," Discrete Appl. Math., vol.84, pp.215–220, 1998.
[25] J. Stoer and C. Witzgall, Convexity and Optimization in Finite Dimension I, Springer-Verlag, Berlin, 1970.
[26] A. Tamura, "Coordinatewise domain scaling algorithm for M-convex function minimization," RIMS Preprint No. 1324, Kyoto University, 2001.

Satoko Moriguchi received Bachelor and Master Degrees of Engineering from Sophia University in 1999 and 2001, respectively. She is presently studying towards Doctor Degree of Science in Department of Mathematical and Computing Sciences, Tokyo Institute of Technology. She is interested in combinatorial optimization.

Kazuo Murota received Bachelor, Master and Doctor Degrees of Engineering from the University of Tokyo in 1978, 1980 and 1983, respectively. He is presently a professor at Department of Mathematical Informatics, Graduate School of Information Science and Technology, University of Tokyo. His research interest centers around discrete mathematical methods in engineering.

Akiyoshi Shioura received Bachelor, Master and Doctor Degrees of Science from Tokyo Institute of Technology in 1993, 1995 and 1998, respectively. He is presently an Associate Professor at Graduate School of Information Sciences, Tohoku University. His major research interests are in mathematical programming and combinatorial optimization.