Minimization of an M-convex Function*

Akiyoshi SHIOURA†

Abstract

We study the minimization of an M-convex function introduced by Murota. It is shown that any vector in the domain can be easily separated from a minimizer of the function. Based on this property, we develop a polynomial time algorithm.

Keywords: matroid, base polyhedron, convex function, minimization.

1 Introduction

M-convex function, recently introduced by Murota [8, 9, 10], is an extension of valued matroid due to Dress and Wenzel [1, 2] as well as a quantitative generalization of (the integral points of) the base polyhedron of an integral submodular system [4]. 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 problems with separable-convex cost functions. M-convex function enjoys several nice properties which persuade us to regard it as “convexity” in combinatorial optimization. Let \( V \) be a finite set with cardinality \( n \). A function \( f : \mathbb{Z}^V \to \mathbb{R} \cup \{+\infty\} \) is said to be M-convex if it satisfies

\[
(M\text{-EXC}) \quad \forall x, y \in \text{dom} f, \forall u \in \text{supp}^+(x - y), \exists v \in \text{supp}^-(x - y) \text{ such that } f(x) + f(y) \geq f(x - \chi_u + \chi_v) + f(y + \chi_u - \chi_v),
\]

where \( \text{dom} f = \{x \in \mathbb{Z}^V \mid f(x) < +\infty\} \), \( \text{supp}^+(x - y) = \{w \in V \mid x(w) > y(w)\} \), \( \text{supp}^-(x - y) = \{w \in V \mid x(w) < y(w)\} \), and \( \chi_w \in \{0, 1\}^V \) is the characteristic vector of \( w \in V \). For an M-convex function \( f \) with \( \text{dom} f \subseteq \{0, 1\}^V \), \(-f\) is a valuation on a matroid in the sense of [1, 2]. The property (M-EXC) implies that \( \text{dom} f \) is a base polyhedron.

In this paper, we consider the problem of minimizing an M-convex function. While the concept of M-convexity is quite new and no efficient algorithm is known yet, several polynomial

---

*Research Report No. 20, Department of Mechanical Engineering, Sophia University, 1997. (version: 17/11/97)
†Department of Mechanical Engineering, Sophia University, 7-1 Kioi-cho, Chiyoda-ku, Tokyo 102, Japan, shioura@kea.me.sophia.ac.jp.
time algorithms are proposed for special cases of M-convex functions. It is well-known that a
linear function can be easily minimized over a base polyhedron by a simple greedy algorithm
(see [4]). A strongly-polynomial time algorithm was proposed by Fujishige [3] for a separable-
convex quadratic function, and weakly-polynomial time algorithms were given by Groenevelt [6]
and Hochbaum [7] for a general separable-convex function. It was reported that there is no
strongly-polynomial time algorithm for a general separable-convex function [7].

The aim of this paper is to develop an efficient algorithm for minimizing an M-convex function.
Since the local optimality is equal to the global optimality, an optimal solution can be found by
a descent method, which does not necessarily terminate in polynomial time. Instead, we propose
a different approach based on the property that any vector in the domain can be efficiently
separated from a minimizer of the function, which is shown later. Each iteration finds a certain
vector in the current domain, and divides the domain so that the vector and an optimal solution
are separated. By a clever choice of the vector, the size of the domain reduces in a certain ratio
iteratively, which leads to a weakly-polynomial time algorithm.

2 Theorems

Throughout the paper we suppose $f : Z^V \to R \cup \{+\infty\}$ is an M-convex function with bounded
domain. The global minimality of an M-convex function is characterized by the local minimality.

**Theorem 2.1 ([8, 10])** For any $x \in \text{dom } f$, $f(x) \leq f(y)$ ($\forall y \in Z^V$) if and only if $f(x) \leq f(x - \chi_u + \chi_v)$ ($\forall u, v \in V$).

Any vector in dom $f$ can be easily separated from some minimizer of $f$.

**Theorem 2.2** (i) For $x \in \text{dom } f$ 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 $x^* \in \text{arg min } f$ with $x^*(u) \leq x'(u)$.

(ii) For $x \in \text{dom } f$ 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 $x^* \in \text{arg min } f$ with $x^*(v) \geq x'(v)$.

**Proof.** We prove the first claim only. Let $x^* \in \text{arg min } f$ with the minimum value of $x^*(u)$,
and to the contrary suppose $x^*(u) > x'(u)$. By (M-EXC), there exists $w \in \text{supp}^{-}(x^* - x')$ such
that $f(x^*) + f(x') \geq f(x^* - \chi_u + \chi_w) + f(x + \chi_v - \chi_w)$. The assumptions for $x^*$ and $x'$ imply
$x^* - \chi_u + \chi_w \in \text{arg min } f$, a contradiction.

**Corollary 2.3** Let $x \in \text{dom } f$ with $x \not\in \text{arg min } f$, 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 $x^* \in \text{arg min } f$ with $x^*(u) \leq x(u) - 1$, $x^*(v) \geq x(v) + 1$.

Let $B \subseteq Z^V$ be a base polyhedron, i.e., $B$ satisfies the next property:

**(B-EXC)** $\forall x, y \in \text{dom } f$, $\forall u \in \text{supp}^+(x - y)$, $\exists v \in \text{supp}^{-}(x - y)$ such that $x - \chi_u + \chi_v$, $y + \chi_u - \chi_v \in B$.  

2
Assume $B$ is bounded. We define the narrowed base polyhedron $N_B(\subseteq B)$ of $B$ as follows. For each $w \in V$, define

$$l_B(w) = \min_{y \in B} \{ y(w) \}, \quad u_B(w) = \max_{y \in B} \{ y(w) \}, \quad (1)$$

$$l'_B(w) = [(1 - 1/n)l_B(w) + (1/n)u_B(w)], \quad u'_B(w) = [(1/n)l_B(w) + (1 - 1/n)u_B(w)]. \quad (2)$$

Then, $N_B$ is defined as $N_B = \{ y \in B \mid l'_B(w) \leq y(w) \leq u'_B(w) \ (\forall w \in V) \}$. We see from definition that $N_B$ is a base polyhedron if it is not empty.

**Theorem 2.4** $N_B \neq \emptyset$.

**Proof.** Let $\rho : 2^V \to \mathbb{Z}$ be the submodular function with $\rho(\emptyset) = 0$ and $B = \{ y \in \mathbb{Z}^V \mid y(X) \leq \rho(X) \ (\forall X \subseteq V), y(V) = \rho(V) \}$. Note that $l_B(w) = \rho(V) - \rho(V - w)$, $u_B(w) = \rho(w)$ ($\forall w \in V$). It suffices to show the following (see [4, Theorem 3.8]):

(i) $l'_B(X) \leq \rho(X)$ ($\forall X \subseteq V$),

(ii) $u'_B(X) \geq \rho(V) - \rho(V - X)$ ($\forall X \subseteq V$).

Since (ii) can be shown similarly, we prove (i) only. Let $X \subseteq V$ with cardinality $k$. We claim

$$k\rho(X) + \sum_{v \in X} \{ \rho(V - v) - \rho(V) \} \geq \sum_{v \in X} \{ \rho(v) + \rho(V - v) - \rho(V) \}. \quad (3)$$

Indeed, we have

$$\text{LHS} = k\rho(X) + \sum_{v \in X} \sum_{u \in X - v} \{ \rho(V - u) - \rho(V) \} + \sum_{v \in X} \{ \rho(V - v) - \rho(V) \}$$

$$\geq k\rho(X) + \sum_{v \in X} \{ \rho(V - (X - v)) - \rho(V) \} + \sum_{v \in X} \{ \rho(V - v) - \rho(V) \} \geq \text{RHS},$$

where the inequalities are by the submodularity of $\rho$. Since the LHS is nonnegative, $k$ in (3) can be replaced by $n (\geq k)$. Thus,

$$\rho(X) \geq (1 - 1/n) \sum_{v \in X} \{ \rho(V) - \rho(V - v) \} + (1/n) \sum_{v \in X} \rho(v) \geq l'_B(X). \quad \blacksquare$$

For $x \in B$ and $u, v \in V$, define

$$\tilde{c}_B(x, u, v) = \max\{ \alpha : \alpha \in \mathbb{Z}, x + \alpha(\chi_v - \chi_u) \in B \} \geq 0,$$

which is called the exchange capacity associated with $x$, $v$ and $u$. For any $\alpha$ with $0 \leq \alpha \leq \tilde{c}_B(x, u, v)$, we have $x + \alpha(\chi_v - \chi_u) \in B$. The next theorem shows that a vector in $N_B$ can be computed efficiently by using the exchange capacity.

**Theorem 2.5** (cf. [4, Theorem 3.27]) A vector in $N_B$ can be obtained by evaluating the exchange capacity associated with $B$ at most $n^2$ times, provided a vector in $B$ is given.
Proof. Suppose we are given a vector \( x_0 \in B \) with either \( x_0(u) < l_B'(u) \) or \( x_0(u) > u_B'(u) \) for some \( u \in V \). It suffices to show that the following algorithm finds \( x \in B \) such that

\[
l_B'(w) \leq x(w) \leq u_B'(w) \text{ if } l_B'(w) \leq x_0(w) \leq u_B'(w) \quad (\forall w \in V - u), \quad l_B'(u) \leq x(u) \leq u_B'(u)
\]

by evaluating the exchange capacity at most \( n \) times. Assume w.l.o.g. that \( x_0(u) > u_B'(u) \), \( n \geq 2 \) and \( V = \{u, v_1, v_2, \ldots, v_{n-1}\} \).

Step 0: Set \( x := x_0 \), \( i := 1 \).
Step 1: If \( x(v_i) < u'_B(v_i) \), set \( \alpha := \min\{\hat{c}_B(x, v_i, u), x(u) - u'_B(u), u'_B(v_i) - x(v_i)\} \),
\[
x := x + \alpha(\chi_{v_i} - \chi_u).
\]
Step 2: If \( i = n - 1 \) or \( x(u) = u'_B(u) \) then stop; otherwise \( i := i + 1 \) and go to Step 1.

To the contrary assume \( x(u) > u'_B(u) \) for the vector \( x \) obtained by the algorithm. Let \( x_* \) be any vector in \( NB \). Since \( x(u) > u'_B(u) \geq x_*(u) \), (B-EXC) implies that the existence of \( v_i \in V - u \) with \( x' = x - \chi_u + \chi_{v_i} \in B \) holds for some \( i \) with \( x(v_i) < x_*(v_i) \leq u'_B(v_i) \). Let \( x_i \) be the vector \( x \) after Step 1 of the \( i \)-th iteration. Then, it holds \( x'(u) < x_i(u), x'(w) \geq x_i(w) \) \( (\forall w \in V - u) \) and \( x'(v_i) > x_i(v_i) \). Hence, \( \hat{c}_B(x_i, v_i, u) = 0 \). On the other hand, we have \( x_i + \chi_{v_i} - \chi_u \in B \) by applying (B-EXC) to \( x' \), \( x_i \) and \( v_i \), a contradiction. 

The values \( l_B(w) \) and \( u_B(w) \) defined by (1) can be computed in the similar way.

Theorem 2.6 For any \( w \in V \), the values \( l_B(w) \) and \( u_B(w) \) can be computed by evaluating the exchange capacity associated with \( B \) at most \( n \) times, provided a vector in \( B \) is given.

3 Algorithms

Theorem 2.1 immediately leads to the following algorithm.

**Algorithm Steepest-Descent**

Step 0: Let \( x \) be any vector in dom \( f \).
Step 1: If \( f(x) = \min_{s, t \in V} \{f(x - \chi_s + \chi_t)\} \) then stop. \( x \) is a minimizer.
Step 2: Find \( u, v \in V \) with \( f(x - \chi_u + \chi_v) = \min_{s, t \in V} \{f(x - \chi_s + \chi_t)\} \).
Step 3: Set \( x := x - \chi_u + \chi_v \). Go to Step 1.

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 next algorithm maintains a set \( B \) (\( \subseteq \) dom \( f \)) which is a base polyhedron containing a minimizer of \( f \). It reduces \( B \) iteratively by exploiting Corollary 2.3 and finally finds a minimizer.

**Algorithm Domain-Reduction**

Step 0: Set \( B := \text{dom}\ f \).
Step 1: Find a vector \( z \in N_B \).

Step 2: If \( f(z) = \min_{x \in V} \{ f(x - \chi_x + \chi_y) \} \) then stop.

Step 3: Find \( u, v \in V \) with \( f(x - \chi_u + \chi_v) = \min_{x \in V} \{ f(x - \chi_x + \chi_y) \} \).

Step 4: Set \( B := B \cap \{ y \in Z^V \mid y(u) \leq x(u) - 1, y(v) \geq x(v) + 1 \} \). Go to Step 1.

We analyze the number of iterations of the algorithm. Denote by \( B_i \) the set \( B \) in the \( i \)-th iteration, and let \( l_i(w) = l_{B_i}(w) \), \( u_i(w) = u_{B_i}(w) \) for each \( w \in V \). It is clear that \( u_i(w) - l_i(w) \) is monotonically nonincreasing w.r.t. \( i \). Furthermore, we have the following:

**Lemma 3.1** \( u_{i+1}(w) - l_{i+1}(w) < (1 - 1/n) \{ u_i(w) - l_i(w) \} \) for \( w \in \{ u, v \} \), where \( u, v \in V \) are the elements found in Step 3.

**Proof.** We show the case \( w = u \). Let \( x \in N_{B_i} \) be the vector chosen in Step 1. Then,

\[
u_{i+1}(u) - l_{i+1}(u) \leq x(u) - 1 - l_i(u) \leq [(1/n)l_i(u) + (1 - 1/n)u_i(u) - 1 - l_i(u)] = (1 - 1/n) \{ u_i(u) - l_i(u) \}.
\]

The proof for the case \( w = v \) is similar and omitted.

Let \( L = \max_{w \in V} \{ u_1(w) - l_1(w) \} \).

**Lemma 3.2** The algorithm \textsc{Domain Reduction} terminates in \( O(n^2 \log L) \) iterations.

**Proof.** Since the value \( u_i(w) - l_i(w) \) \((w \in V)\) is a nonnegative integer, the algorithm stops if \( u_i(w) - l_i(w) < 1 \) for all \( w \in V \). Let \( k \) be the minimum integer with \((1 - 1/n)^k \{ u_1(w) - l_1(w) \} < 1 \). Suppose \( u_1(w) \neq l_1(w) \) and \( n \geq 2 \). Then,

\[
k \leq -\ln \{ u_1(w) - l_1(w) \} / \ln(1 - 1/n) + 1 \leq n \ln \{ u_1(w) - l_1(w) \} + 1.
\]

by a well-known inequality \( \ln z \leq z - 1 \) \((\forall z > 0)\). Thus the claim follows.

In the following, we explain how to perform each step, especially how to find a vector in \( N_B \).

We assume that a vector \( x_0 \in \text{dom} f \) and the value \( L \) are given in advance.

We maintain the set \( B \) by using two vectors \( a, b \) with \( -a(w), b(w) \in Z \cup \{ +\infty \} \) \((\forall w \in V)\) as \( B = \text{dom} f \cap \{ y \in Z^V \mid a(w) \leq y(w) \leq b(w) \} \). Maintenance of \( a, b \) is easy: initially set \( a(w) = -\infty, b(w) = +\infty \) \((\forall w \in V)\), and update only the values \( a(v) \) and \( b(u) \) to \( x(v) + 1 \), \( x(u) - 1 \), respectively in Step 4 of each iteration.

When finding a vector in \( N_B \), we first compute the values \( l_B(w), u_B(w) \) \((\forall w \in V)\) defined by (1), which can be done by \( O(n^2) \)-time evaluation of the exchange capacity associated with \( B \) from Theorem 2.6. The exchange capacity can be computed in \( O(\log L) \) time by the binary search since \( 0 \leq \check{c}_B(x, u, v) \leq L \) \((\forall x \in B, \forall u, v \in V)\). Then, we compute \( l'_B(w), u'_B(w) \) \((\forall w \in V)\) defined by (2) by using floor and ceiling operations. Note that floor and ceiling operations can be performed easily since \( n \) is the denominator of each value for which floor or ceiling is operated.
After computing the values $l_B'(w), u_B'(w)$ we can find $x \in N_B$ by $O(n^2)$-time evaluation of the exchange capacity. Thus, Step 1 can be performed in $O(n^2 \log L)$ time.

The other steps require $O(n^2)$-time evaluation of $f$.

**Theorem 3.3** If a vector in $\text{dom } f$ and the value $L$ are given, the algorithm **DOMAIN REDUCTION** finds a minimizer of $f$ in $O(n^4 \log^2 L)$ time.

**References**


