MATHEMATICAL ENGINEERING TECHNICAL REPORTS

Dijkstra's Algorithm and L-concave Function Maximization

Kazuo MUROTA and Akiyoshi SHIOURA

METR 2012–05 March 2012

DEPARTMENT OF MATHEMATICAL INFORMATICS
GRADUATE SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY
THE UNIVERSITY OF TOKYO
BUNKYO-KU, TOKYO 113-8656, JAPAN

WWW page: http://www.keisu.t.u-tokyo.ac.jp/research/techrep/index.html

Dijkstra's Algorithm and L-concave Function Maximization

Kazuo MUROTA

Graduate School of Information Science and Technology
University of Tokyo, Tokyo 113-8656, Japan

and

Akiyoshi SHIOURA

Graduate School of Information Sciences
Tohoku University, Sendai 980-8579, Japan

March 2012

Abstract

Dijkstra's algorithm is a well-known algorithm for the single-source shortest path problem in a directed graph with nonnegative edge length. We investigate Dijkstra's algorithm from the viewpoint of discrete convex analysis, and point out that it can be regarded as an implementation of the steepest ascent algorithm for L-concave function maximization.

1 Introduction

The single-source shortest path problem in a directed graph with nonnegative edge length is a classical combinatorial optimization problem formulated as follows: given a directed graph $G = (V, E)$ with edge length $\ell(e) \geq 0$ $(e \in E)$ and a vertex $s \in V$ called a source, we want to compute the shortest path length from the source vertex s to each vertex $v \in V$. Among many algorithms for the shortest path problem, Dijkstra's algorithm [2] described below is most fundamental (see, e.g., [1, 9]).

Dijkstra's Algorithm

Step 0: Set $U := V$. Set $\pi(s) := 0$, $\pi(v) = +\infty$ $(v \in V \setminus \{s\})$.

Step 1: Set $W := \arg\min\{\pi(v) \mid v \in U\}$ and $X := U \setminus W$.

Step 2: If $X = \emptyset$, then stop; for $v \in V$, $\pi(v)$ is the shortest path length from s to v.

Step 3: Set $U := X$. For $v \in U$, set

$$\pi(v) := \min\{\pi(v), \min\{\pi(u) + \ell(u,v) \mid (u,v) \in E,\ u \in W\}\}.$$

Go to Step 1. □

In this note, we discuss Dijkstra's algorithm from the viewpoint of discrete convex analysis. Discrete convex analysis is a theoretical framework for well-solved combinatorial optimization problems introduced by Murota (see [5]; see also [6]), where the concept of discrete convexity called L$^\natural$-convexity plays a central role. As is well known, the single-source shortest path problem can be formulated as a linear programming (LP) problem. We observe first that the dual of the LP formulation can be seen as a special case of L$^\natural$-concave function maximization (see Section 2 for the definition of L$^\natural$-concavity). Then we point out that the steepest ascent algorithm for L$^\natural$-concave function maximization, when applied to the LP dual of the shortest path problem and implemented with some auxiliary variables, coincides exactly with Dijkstra's algorithm.

2 Review of L$^\natural$-convexity

In this section we review the concepts of L$^\natural$-convex sets and L$^\natural$-concave functions, and present some useful properties. See [6] for more account of these concepts.

2.1 L$^\natural$-convex Sets

Let V be a finite set. A set $S \subseteq \mathbb{Z}^V$ of integral vectors is said to be *L$^\natural$-convex* if it is nonempty and satisfies the following condition:

$$\text{if } p, q \in S, \text{ then } \left\lceil \frac{p+q}{2} \right\rceil, \left\lfloor \frac{p+q}{2} \right\rfloor \in S, \tag{2.1}$$

where for $x \in \mathbb{R}^V$, $\lceil x \rceil$ and $\lfloor x \rfloor$ denote, respectively, the integer vectors obtained from x by component-wise round-up and round-down to the nearest integers. The condition (2.1) is called the *discrete midpoint convexity* for a set.

Discrete midpoint convexity (2.1) implies the following property. For $p, q \in \mathbb{R}^V$, we denote by $p \vee q$ and $p \wedge q$, respectively, the vectors of component-wise maximum and minimum of p and q, i.e.,

$$(p \vee q)(v) = \max\{p(v), q(v)\}, \quad (p \wedge q)(v) = \min\{p(v), q(v)\} \quad (v \in V).$$

Proposition 2.1 (cf. [6, Chapter 5]). *Let $S \subseteq \mathbb{Z}^V$ be an $L^\natural$-convex set. If $p, q \in S$, then it holds that $p \vee q, p \wedge q \in S$.*

This property implies, in particular, that a maximal vector in a bounded L$^\natural$-convex set is uniquely determined.

The following proposition gives a polyhedral description of L$^\natural$-convex sets.

Proposition 2.2 (cf. [6, Chapter 5]). *A set $S \subseteq \mathbb{Z}^V$ is an $L^\natural$-convex set if and only if S is a nonempty set represented as*

$$S = \{p \in \mathbb{Z}^V \mid p(v) - p(u) \leq a(u,v) \ (u, v \in V, u \neq v), \\ b(v) \leq p(v) \leq c(v) \ (v \in V)\}$$

with some $a(u,v) \in \mathbb{Z} \cup \{+\infty\}$ $(u, v \in V,\ u \neq v)$, $b(v) \in \mathbb{Z} \cup \{-\infty\}$ $(v \in V)$, and $c(v) \in \mathbb{Z} \cup \{+\infty\}$ $(v \in V)$.

2.2 L$^\natural$-concave Functions

Let $g : \mathbb{Z}^V \to \mathbb{R} \cup \{-\infty\}$ be a function defined on the integer lattice points, and denote $\operatorname{dom} g = \{p \in \mathbb{Z}^V \mid g(p) > -\infty\}$. We say that g is an *$L^\natural$-concave function* if $\operatorname{dom} g \neq \emptyset$ and it satisfies the following condition:

$$g(p) + g(q) \leq g\left(\left\lceil \frac{p+q}{2} \right\rceil\right) + g\left(\left\lfloor \frac{p+q}{2} \right\rfloor\right) \qquad (\forall p, q \in \operatorname{dom} g).$$

In the maximization of an L$^\natural$-concave function $g : \mathbb{Z}^V \to \mathbb{R} \cup \{-\infty\}$, a maximizer of g can be characterized by a local optimality.

Theorem 2.3. *Let $g : \mathbb{Z}^V \to \mathbb{R} \cup \{-\infty\}$ be an $L^\natural$-concave function. A vector $p \in \operatorname{dom} g$ is a maximizer of g if and only if $g(p) \geq g(p + \chi_X)$ $(\forall X \subseteq V)$ and $g(p) \geq g(p - \chi_X)$ $(\forall X \subseteq V)$.*

A maximizer of g can be computed by the following steepest ascent algorithm. For $X \subseteq V$, we denote by $\chi_X \in \{0, +1\}^V$ the characteristic vector of X. We suppose that an initial vector $p_0 \in \operatorname{dom} g$ is given in advance.

Algorithm 0

(Steepst Ascent Algorithm for L$^\natural$-concave Function Maximization)

Step 0: Set $p := p_0$.

Step 1: Find $\varepsilon \in \{+1, -1\}$ and $X \subseteq V$ that maximizes $g(p + \varepsilon\chi_X)$.

Step 2: If $g(p) \geq g(p + \varepsilon\chi_X)$, then stop; p is a maximizer of g.

Step 3: Set $p := p + \varepsilon\chi_X$. Go to Step 1. □

It is noted that Step 1 can be done in (strongly) polynomial time by using any of polynomial-time algorithms for submodular set function minimization [3, 8] since the set functions $\rho^+, \rho^- : 2^V \to \mathbb{R} \cup \{+\infty\}$ defined by

$$\rho^+(X) = g(p) - g(p + \chi_X), \quad \rho^-(X) = g(p) - g(p - \chi_X) \quad (X \subseteq V)$$

are submodular functions with $\rho^+(\emptyset) = \rho^-(\emptyset) = 0$.

The steepest ascent algorithm above terminates in a finite number of iterations if $\mathrm{dom}\, g$ is a finite set. The obtained vector p is indeed a maximizer of g by Theorem 2.3. See [4, 7] for the time complexity of the algorithms of this type.

3 Shortest Path Problem and L$^\natural$-convexity

We show the connection of the single-source shortest path problem with L$^\natural$-convex sets and L$^\natural$-concave functions.

A linear programming formulation of the single-source shortest path problem is given as follows:

$$\textbf{(P)} \left| \begin{array}{ll} \text{Minimize} & \displaystyle\sum_{(u,v)\in E} \ell(u,v)x(u,v) \\ \text{subject to} & \sum\{x(u,s) \mid (u,s) \in E, u \in V\} \\ & \qquad - \sum\{x(s,u) \mid (s,u) \in E,\ u \in V\} = -(n-1), \\ & \sum\{x(u,v) \mid (u,v) \in E, u \in V\} \\ & \qquad - \sum\{x(v,u) \mid (v,u) \in E,\ u \in V\} = 1 \quad (v \in V \setminus \{s\}), \\ & x(u,v) \geq 0 \qquad ((u,v) \in E). \end{array} \right.$$

This LP can be seen as a minimum-cost flow problem, where a unit of flow is sent from the source vertex s to each vertex $v \in V \setminus \{s\}$, and the flow cost on edge $(u,v) \in E$ is given by $\ell(u,v)$.

The LP dual of (P) is given as follows:

$$\left| \begin{array}{ll} \text{Maximize} & \displaystyle\sum_{v\in V\setminus\{s\}} \{p(v) - p(s)\} \\ \text{subject to} & p(v) - p(u) \leq \ell(u,v) \qquad ((u,v) \in E), \\ & p(v) \in \mathbb{R} \qquad (v \in V). \end{array} \right.$$

In this LP, we can fix $p(s) = 0$ without loss of generality, which yields the following LP:

$$\textbf{(D)} \left| \begin{array}{ll} \text{Maximize} & \displaystyle\sum_{v\in V\setminus\{s\}} p(v) \\ \text{subject to} & p(v) - p(u) \leq \ell(u,v) \qquad ((u,v) \in E), \\ & p(s) = 0, \\ & p(v) \in \mathbb{R} \qquad (v \in V \setminus \{s\}). \end{array} \right.$$

This problem will be the main object of our discussion.

We denote by $S \subseteq \mathbb{R}^V$ the feasible region of (D), i.e.,

$$S = \{p \in \mathbb{R}^V \mid p(v) - p(u) \leq \ell(u, v)\ ((u, v) \in E),\ p(s) = 0\}. \tag{3.1}$$

By Proposition 2.2, S is an L$^\natural$-convex set. Hence, the problem (D) can be seen as maximization of a linear function with positive coefficients over an L$^\natural$-convex set.

We assume that there exists a directed path from s to every $v \in V \setminus \{s\}$. Then, (P) has a feasible (and optimal) solution, and by LP duality, the optimal value of (D) is finite. Hence, the set S is bounded from above, and Proposition 2.1 implies that S has a unique maximal vector p_*, which is an optimal solution of (D). It is also noted that the zero vector $\mathbf{0}$ is contained in S since $\ell(u, v) \geq 0$ for $(u, v) \in E$.

We define a function $g : \mathbb{Z}^V \to \mathbb{R} \cup \{-\infty\}$ by

$$g_{\mathrm{D}}(p) = \begin{cases} \displaystyle\sum_{v \in V \setminus \{s\}} p(v) & (\text{if } p \in S), \\ -\infty & (\text{otherwise}). \end{cases} \tag{3.2}$$

We see that the maximization of g_{D} is equivalent to the problem (D). Since S satisfies the discrete midpoint convexity (2.1), g_{D} satisfies the inequality

$$g_{\mathrm{D}}(p) + g_{\mathrm{D}}(q) \leq g_{\mathrm{D}}\left(\left\lceil \frac{p+q}{2} \right\rceil\right) + g_{\mathrm{D}}\left(\left\lfloor \frac{p+q}{2} \right\rfloor\right)$$

for all $p, q \in \mathrm{dom}\, g_{\mathrm{D}}$; in fact, the inequality above holds with equality. This means that g_{D} is an L$^\natural$-concave function. Hence, the problem (D) can be seen as a special case of L$^\natural$-concave function maximization.

4 Dijkstra's Algorithm and Steepest Ascent Algorithm

4.1 Steepest Ascent Algorithm Applied to Shortest Path Problem

We apply the steepest ascent algorithm in Section 2 to the maximization of the L$^\natural$-concave function g_{D} in (3.2) associated with the shortest path problem, where the zero vector $\mathbf{0} \in S$ is used as the initial vector p_0. Then, we observe the following properties.

Proposition 4.1.
(i) *The condition* $g_{\mathrm{D}}(p) \geq g_{\mathrm{D}}(p - \chi_Y)$ $(\forall Y \subseteq V)$ *holds in each iteration, and therefore we may assume* $\varepsilon = +1$ *in Step 1.*
(ii) *In Step 1, we have*

$$X \in \arg\max\{|Y| \mid Y \subseteq V,\ p + \chi_Y \in S\}. \tag{4.1}$$

In particular, in Step 2 it holds that $g_{\rm D}(p) \geq g_{\rm D}(p+\chi_X)$ if and only if $X=\emptyset$.
(iii) *Denote by X_k the set X found in Step 1 of the k-th iteration. Then, it holds that $X_k \subseteq X_{k-1}$ for all $k \geq 2$.*

Proof. [Proof of (i)] The vector p is always contained in S in each iteration. Hence, we have $g_{\rm D}(p) = \sum_{v \in V \setminus \{s\}} p(v)$. If $p - \chi_Y \notin S$, then $g_{\rm D}(p-\chi_Y) = -\infty < g_{\rm D}(p)$. If $p-\chi_Y \in S$, then

$$g_{\rm D}(p-\chi_Y) = \sum_{v \in V\setminus\{s\}} p(v) - |Y \setminus \{s\}| \leq g_{\rm D}(p).$$

[Proof of (ii)] Since $X \in \arg\max\{g(p+\chi_Y) \mid Y \subseteq V,\ p+\chi_Y \in S\}$, we have $s \notin X$. For $Y \subseteq V \setminus \{s\}$ with $p+\chi_Y \in S$, it holds that

$$g_{\rm D}(p+\chi_Y) = \sum_{v \in V\setminus\{s\}} p(v) + |Y| = g_{\rm D}(p) + |Y|.$$

Hence, the equation (4.1) follows. Then the latter statement is obvious.

[Proof of (iii)] For a fixed $k \geq 2$, let $p' = \sum_{i=1}^{k-2} \chi_{X_i}$. Since p' and $p'+\chi_{X_{k-1}}+\chi_{X_k}$ are in S, the discrete midpoint convexity (2.1) for S implies that $p'+\chi_{X_{k-1}\cup X_k} \in S$. By the choice of X_{k-1}, we have $|X_{k-1} \cup X_k| = |X_{k-1}|$ (see the claim (ii)), implying that $X_k \subseteq X_{k-1}$. □

From the observation above, the steepest ascent algorithm in Section 2 applied to the function $g_{\rm D}$ in (3.2) can be rewritten as follows with a variable U and a step size λ.

Algorithm 1 (Steepest Ascent Algorithm for (D))

Step 0: Set $p := \mathbf{0}$, $U := V$.

Step 1: Let $X \in \arg\max\{|Y| \mid Y \subseteq U,\ p+\chi_Y \in S\}$.

Step 2: If $X = \emptyset$, then stop; p is an optimal solution of (D).

Step 3: Set $p := p + \lambda\chi_X$ with $\lambda = \max\{\mu \in \mathbb{Z}_+ \mid p+\mu\chi_X \in S\}$. Set $U := X$. Go to Step 1. □

It is noted that if $v \in U$, the value $p(v)$ may possibly be incremented in the following iterations, and if $v \in V \setminus U$, the value $p(v)$ remains the same in the following iterations. We also have $s \notin U$ in each iteration, except for the first iteration.

Remark 4.2. Algorithm 1 can be applied to the following more general problem:

$$\text{Maximize} \sum_{v \in V} w(v)p(v) \text{ subject to } p \in S,$$

where $w \in \mathbb{R}^V$ is a positive vector and $S \subseteq \mathbb{Z}^V$ is an L$^\natural$-convex set containing the zero vector. □

4.2 Implementation with Auxiliary Variables

We present an implementation of Algorithm 1 by using auxiliary variables. This reveals the connection between the steepest ascent algorithm for L$^\natural$-concave function maximization and Dijkstra's algorithm.

In Steps 1 and 2 of Algorithm 1, we need to compute a set X and a step size λ. This can be done easily by using auxiliary variables $\pi(v)$ $(v \in V)$ that satisfy the following conditions at the beginning of each iteration:

$$\pi(s) = 0, \tag{4.2}$$

$$\pi(v) = \min\{p(u) + \ell(u, v) \mid (u, v) \in E,\ u \in V \setminus U\} \quad (v \in U \setminus \{s\}). \tag{4.3}$$

Proposition 4.3. *Suppose that $\pi(v)$ $(v \in V)$ satisfy the conditions* (4.2) *and* (4.3) *with respect to $p \in S$ and $U \subseteq V$.*
(i) *$p(v) \le \pi(v)$ holds for all $v \in U$.*
(ii) *For $Y \subseteq U$ and $\mu \in \mathbb{Z}_+$, we have $p + \mu\chi_Y \in S$ if and only if $\mu \le \min\{\pi(v) - p(v) \mid v \in Y\}$.*

Proof. [Proof of (i)] Since $p \in S$, we have $p(s) = 0$ and $p(u)+\ell(u, v) \ge p(v)$ for $(u, v) \in E$ with $u \in V \setminus U$ and $v \in U \setminus \{s\}$. Hence, we have $p(s) = \pi(s)$ and $p(v) \le \pi(v)$ for all $v \in U \setminus \{s\}$ by (4.2) and (4.3).

[Proof of (ii)] If $s \in Y$, then we have $p + \mu\chi_Y \in S$ if and only if $\mu = 0$, which is equivalent to

$$\mu \le \min\{\pi(v) - p(v) \mid v \in Y\} = \pi(s) - p(s) = 0.$$

Hence, we consider the case with $s \notin Y$. Put $q = p + \mu\chi_Y$. We have $q \in S$ if and only if

$$q(v) - q(u) \le \ell(u, v) \qquad (\forall (u, v) \in E,\ u \in Y,\ v \in V \setminus Y), \tag{4.4}$$

since, for other edges (u, v), it holds that $q(v) - q(u) \le p(v) - p(u) \le \ell(u, v)$. For $(u, v) \in E$ with $u \in Y$ and $v \in V \setminus Y$, we have $q(v)-q(u) = p(v)-p(u)+\mu$, and therefore the condition (4.4) can be rewritten as

$$\mu \le p(u) + \ell(u, v) - p(v) \qquad (\forall (u, v) \in E,\ u \in Y,\ v \in V \setminus Y),$$

which is equivalent to $\mu \le \min\{\pi(v) - p(v) \mid v \in Y\}$ by (4.3). □

Proposition 4.4. *Suppose that $\pi(v)$ $(v \in V)$ satisfy the conditions* (4.2) *and* (4.3) *at the beginning of an iteration of Algorithm 1. Then, we have $X = \{v \mid v \in U,\ p(v) < \pi(v)\}$ in Step 1, and $\lambda = \min\{\pi(v) - p(v) \mid v \in X\}$ in Step 3.*

Proof. Let $Z = \{v \mid v \in U,\ p(v) < \pi(v)\}$. By Proposition 4.3 (ii), we have $p + \chi_Y \in S$ if and only if $Y \subseteq Z$. Hence, X in Step 1 is equal to Z. By Proposition 4.3 (ii) again, we have $p+\mu\chi_X \in S$ if and only if $\mu \le \min\{\pi(v)- p(v) \mid v \in X\}$. Hence, λ in Step 3 is equal to $\min\{\pi(v) - p(v) \mid v \in X\}$. □

Based on Proposition 4.4, Algorithm 1 can be implemented by using the auxiliary variables $\pi(v)$ $(v \in V)$ as follows. As shown in Proposition 4.5 below, $\pi(v)$ $(v \in V)$ satisfy the conditions (4.2) and (4.3) at the beginning of each iteration, which guarantees that Algorithm 2 computes an optimal solution of (D).

Algorithm 2 (Implementation of Algorithm 1 with auxiliary variables)

Step 0: Set $p := \mathbf{0}$, $U := V$. Set $\pi(s) := 0$, $\pi(v) = +\infty$ $(v \in V \setminus \{s\})$.

Step 1: Set $X := \{v \mid v \in U,\ p(v) < \pi(v)\}$ and $W := U \setminus X$.

Step 2: If $X = \emptyset$, then stop; p is an optimal solution of (D).

Step 3: Set $p := p + \lambda\chi_X$ with $\lambda = \min\{\pi(v) - p(v) \mid v \in X\}$. Set $U := X$. For $v \in U$, set

$$\pi(v) := \min\{\pi(v), \min\{\pi(u) + \ell(u,v) \mid (u,v) \in E,\ u \in W\}\}.$$

Go to Step 1. □

Proposition 4.5. *In Step 1 of each iteration in Algorithm 2, the following properties hold:*
(i) *The values $p(v)$ $(v \in U)$ are the same.*
(ii) *Auxiliary variables $\pi(v)$ $(v \in V)$ satisfy the conditions* (4.2) *and* (4.3)*.*

Proof. We prove the claims inductively in each iteration.

We consider the first iteration. Since $U = V$, the right-hand side of (4.3) is given as $+\infty$. Hence, (i) and (ii) follow from the setting of p and π in Step 0. We note that $p(s) = \pi(s)$ holds at the beginning of the first iteration, and therefore we have $s \notin X$; this means that s is never contained in the set U in the following iterations. Hence, the condition (4.2) is always satisfied until the algorithm terminates.

Suppose that (i) and (ii) hold at the beginning of some iteration. We show that p and π satisfy these properties at the end of this iteration (i.e., after Step 3 is executed). We denote by p', π', and U', respectively, the vectors p, π, and the set U after the update in Step 3, i.e., $p' = p + \lambda\chi_X$, $U' = X$, and

$$\pi'(v) = \min\{\pi(v), \min\{\pi(u) + \ell(u,v) \mid (u,v) \in E,\ u \in W\}\}\ (v \in U'). \quad (4.5)$$

Since $p' = p + \lambda\chi_X$ and $U' = X$, the property (i) for p and U implies that (i) holds for p' and U' at the end of the iteration.

Since (ii) holds in Step 1, we have $p(v) \le \pi(v)$ holds for all $v \in U$ by Proposition 4.3 (i). Hence, the set $W = U \setminus X$ is given as

$$W = \{v \mid v \in U,\ p(v) = \pi(v)\}. \quad (4.6)$$

Let $v \in U'$. It holds that

$$\begin{aligned}\pi'(v) &= \min\{\pi(v), \min\{p(u) + \ell(u,v) \mid (u,v) \in E,\ u \in W\}\} \\ &= \min\{\min\{p(u) + \ell(u,v) \mid (u,v) \in E,\ u \in V \setminus U\}, \\ &\qquad \min\{p(u) + \ell(u,v) \mid (u,v) \in E,\ u \in W\}\} \\ &= \min\{p'(u) + \ell(u,v) \mid (u,v) \in E,\ u \in V \setminus U'\},\end{aligned}$$

where the first equality is by (4.5) and (4.6), the second by (ii) for π, and the third by $V \setminus U' = (V \setminus U) \cup W$ and $p'(u) = p(u)$ for $u \in V \setminus U'$. Hence, the condition (4.3) holds at the end of this iteration. □

In Algorithm 2, Step 1 can be rewritten as follows:

Step 1: Set $W := \arg\min\{\pi(v) \mid v \in U\}$ and $X := U \setminus W$.

This is possible due to the following facts:

- values in $\{p(v) \mid v \in U\}$ are the same (by Proposition 4.5 (i)),
- the inequality $p(v) \leq \pi(v)$ holds for $v \in U$ (by Proposition 4.3 (i)),
- $W = \{v \mid v \in U,\ p(v) = \pi(v)\}$ (by (4.6) in the proof of Proposition 4.5).

Moreover, the following property implies that the variables $p(v)$ $(v \in V)$ are not needed to compute an optimal solution of (D).

Proposition 4.6. *We have $\pi(v) = p(v)$ for all $v \in V$ when Algorithm 2 terminates.*

Proof. In Step 1 of each iteration, we have $\pi(v) = p(v)$ for all $v \in W$ (cf. (4.6)). All elements in W are deleted from the set U in Step 3 of the iteration, and for any $v \in V$, the value $p(v)$ does not change once v is deleted from U. Hence, the claim follows. □

Hence, the variables $p(v)$ $(v \in V)$ can be eliminated from Algorithm 2, and the resulting algorithm coincides with Dijkstra's algorithm described in Section 1. That is, Dijkstra's can be recognized as an algorithm which implicitly computes an optimal solution of the $\mathrm{L}^\natural$-concave maximization problem (D) in a greedy way.

References

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. Network Flows: Theory, Algorithms, and Applications. Prentice Hall, 1993.

[2] E.W. Dijkstra. A note on two problems in connexion with graphs. Numerische Mathematik 1 (1959) 269–271.

[3] S. Iwata, L. Fleischer, and S. Fujishige. A combinatorial, strongly polynomial-time algorithm for minimizing submodular functions. Journal of the ACM 48 (2001) 761–777.

[4] V. Kolmogorov and A. Shioura. New algorithms for convex cost tension problem with application to computer vision. Discrete Optimization 6 (2009) 378–393.

[5] K. Murota. Discrete convex analysis. Mathematical Programming 83 (1998) 313–371.

[6] K. Murota. Discrete Convex Analysis. Society for Industrial and Applied Mathematics, 2003.

[7] K. Murota. On steepest descent algorithms for discrete convex functions. SIAM Journal on Optimization 14 (2003) 699–707.

[8] A. Schrijver. A combinatorial algorithm minimizing submodular functions in strongly polynomial time. Journal of Combinatorial Theory Series B 80 (2000) 346–355.

[9] A. Schrijver. Combinatorial Optimization: Polyhedra and Efficiency. Springer, Berlin, 2003.