アルゴリズム 65−5
(1998. 10. 28)

# 離散凸最適化における共役スケーリング法

岩田 覚 (大阪大学)　繁野 麻衣子 (筑波大学)

**梗概:** 離散凸関数の共役性を利用した新しいスケーリング技法を導入する. その結果, 離散凸関数を費用関数とする劣モジュラ流問題に対する弱多項式時間算法が得られる. この算法によって, 離散凸解析における Fenchel 型双対定理の両辺の最適解を多項式時間で求めることが可能となる.

# Conjugate Scaling Technique for Fenchel-type Duality in Discrete Convex Optimization

Satoru IWATA (Osaka University)　Maiko SHIGENO (Tsukuba University)

**Abstract:** This paper presents a polynomial time algorithm for solving submodular flow problems with a class of discrete convex cost functions. The algorithm adopts a new scaling technique that scales the discrete convex cost functions via the conjugacy relation. The algorithm can be used to find a pair of optima in the form of the Fenchel-type duality theorem in discrete convex analysis.

## 1 Introduction

The Fenchel-type duality concerning M- and L-convex/concave functions is of fundamental importance in the theory of discrete convex analysis [17, 18, 19]. This paper aims at an algorithmic approach to this duality framework.

Let $V$ be a finite set and $\chi_v$ denote the characteristic vector of $v \in V$. The characteristic vector of $X \subseteq V$ is denoted by $\chi_X$. We write $\mathrm{supp}^+(z) = \{v \mid v \in V,\, z(v) > 0\}$ and $\mathrm{supp}^-(z) = \{v \mid v \in V,\, z(v) < 0\}$ for a vector $z \in \mathbf{Z}^V$. For functions $g : \mathbf{Z}^V \to \mathbf{Z} \cup \{+\infty\}$ and $h : \mathbf{Z}^V \to \mathbf{Z} \cup \{-\infty\}$, we denote by $\mathrm{dom}_{\mathbf{Z}}\, g$ and $\mathrm{dom}_{\mathbf{Z}}\, h$ their effective domains, i.e., $\mathrm{dom}_{\mathbf{Z}}\, g = \{x \mid x \in \mathbf{Z}^V,\, g(x) < +\infty\}$ and $\mathrm{dom}_{\mathbf{Z}}\, h = \{x \mid x \in \mathbf{Z}^V,\, h(x) > -\infty\}$.

A function $g : \mathbf{Z}^V \to \mathbf{Z} \cup \{+\infty\}$ with nonempty effective domain is said to be M-convex [16, 17, 18] if it satisfies the following:

- $\forall x, y \in \mathrm{dom}_{\mathbf{Z}}\, g$, $\forall u \in \mathrm{supp}^+(x-y)$, $\exists v \in \mathrm{supp}^-(x-y)$:

$$g(x) + g(y) \geq g(x - \chi_u + \chi_v) + g(y + \chi_u - \chi_v).$$

It is not difficult to see that the effective domain of an M-convex function forms the set of integral points in an base polyhedron with an integral rank function. A function $h : \mathbf{Z}^V \to \mathbf{Z} \cup \{-\infty\}$ is called M-concave if $-h$ is an M-convex function. M-concave functions generalize valuations on matroids invented by Dress and Wenzel [2].

Let $\langle \cdot, \cdot \rangle$ designates the inner product of vectors, i.e., $\langle p, x \rangle = \sum\{p(v)x(v) \mid v \in V\}$. In particular, we denote $x(X) = \langle \chi_X, x \rangle$ for $X \subseteq V$. If $g$ is an M-convex function, $x(V)$ is constant for every $x \in \mathrm{dom}_{\mathbf{Z}} g$.

For a pair of vectors $p, q \in \mathbf{Z}^V$, let $p \vee q$ and $p \wedge q$ denote the vectors defined by $(p \vee q)(v) = \max\{p(v), q(v)\}$ and $(p \wedge q)(v) = \min\{p(v), q(v)\}$, respectively. We also denote by $\mathbf{1}$ the vector in $\mathbf{Z}^V$ with all its components being equal to one, i.e., the characteristic vector of $V$.

A function $f : \mathbf{Z}^V \to \mathbf{Z} \cup \{+\infty\}$ with nonempty effective domain is said to be L-convex [18] if it satisfies the following:

- $\exists r \in \mathbf{Z}$, $\forall p \in \mathbf{Z}^V$: $f(p+\mathbf{1}) = f(p) + r$;
- $\forall p, q \in \mathbf{Z}^V$: $f(p) + f(q) \geq f(p \vee q) + f(p \wedge q)$.

L-convex functions generalize the Lovász extensions of submodular functions [13]. They are in a close relation to submodular integrally convex functions of Favati and Tardella [4]. See Fujishige–Murota [8] for this connection. A function $h : \mathbf{Z}^V \to \mathbf{Z} \cup \{-\infty\}$ is called L-concave if $-h$ is an L-convex function.

These two notions of discrete convexity are conjugate each other. For a function $g : \mathbf{Z}^V \to \mathbf{Z} \cup \{+\infty\}$, we denote by $g^\bullet$ the convex conjugate function defined by

$$g^\bullet(p) = \sup\{\langle p, x\rangle - g(x) \mid x \in \mathbf{Z}^V\} \qquad (p \in \mathbf{Z}^V).$$

The convex conjugate function of an M-convex function is L-convex, and vice versa [18]. The concave conjugate function $h^\circ$ of $h : \mathbf{Z}^V \to \mathbf{Z} \cup \{-\infty\}$ is similarly defined by

$$h^\circ(p) = \inf\{\langle p, x\rangle - h(x) \mid x \in \mathbf{Z}^V\} \qquad (p \in \mathbf{Z}^V).$$

The concave conjugate function of an M-concave function is L-concave, and vice versa. This conjugacy framework is a discrete counterpart of the well-known conjugate duality in convex analysis [20].

Analogously to the Fenchel duality theorem in convex analysis, Murota [17] shows that any pair of an M-convex function $g$ and an M-concave function $h$ satisfies

$$\sup\{h(x) - g(x) \mid x \in \mathbf{Z}^V\} = \inf\{g^\bullet(p) - h^\circ(p) \mid p \in \mathbf{Z}^V\} \tag{1.1}$$

if $g(x) - h(x) \neq +\infty$ for some $x \in \mathbf{Z}^V$ or $g^\bullet(p) - h^\circ(p) \neq +\infty$ for some $p \in \mathbf{Z}^V$. Throughout this paper, we assume M-convex/concave functions to have bounded effective domains, and accordingly L-convex/concave functions to take finite values in $\mathbf{Z}^V$. The equality (1.1) always holds in this situation. The original proof by Murota [16] is based on an algorithm that solves submodular flow problems with M-convex cost functions, which we call discrete convex submodular flow problems. The time complexity of this algorithm is pseudopolynomial, i.e., polynomial in the input values, but not in the input size. See Fujishige–Murota [9] for an alternative shorter proof of this Fenchel-type duality theorem.

In this paper, we present a polynomial time algorithm for solving the discrete convex submodular flow problem. The new algorithm naturally provides an efficient method for finding both optima in (1.1). See Murota [18] for this connection.

In order to obtain a polynomial time bound, it is now standard to apply the scaling approach. However, a straightforward scaling scheme does not work for M-convex cost functions. For example, a function $g'$ defined by $g'(x) = \lceil g(x)/\alpha \rceil$ for an M-convex function $g$ and a positive integer $\alpha$ is not necessarily M-convex. Instead, we scale M-convex functions via the conjugacy relation, exploiting the fact that if $f$ is L-convex then so is $f'$ defined by $f'(p) = f(\alpha p)$. This conjugate scaling method employs polynomial time minimization algorithms for L- and M-convex functions respectively due to Favati–Tardella [4] and Shioura [21].

## 2 The Discrete Convex Submodular Flow Problem

Let $G = (V, A)$ be a directed graph with a vertex set $V$ and an arc set $A$. The initial and terminal vertices of an arc $a$ are denoted by $\partial^+ a$ and $\partial^- a$. For a vertex $v \in V$, we denote by $\delta^+ v$ and $\delta^- v$ the set of arcs leaving $v$ and those entering $v$, respectively. The boundary $\partial\varphi$ of a function $\varphi$ on the arc set $A$ is defined by

$$\partial\varphi(v) = \sum_{a\in\delta^+ v} \varphi(a) - \sum_{a\in\delta^- v} \varphi(a) \qquad (v \in V).$$

We denote by $n$ the cardinality of the vertex set $V$.

With the directed graph $G = (V, A)$ are associated functions $\overline{c} : A \to \mathbf{Z}\cup\{+\infty\}$ and $\underline{c} : A \to \mathbf{Z} \cup \{-\infty\}$ as upper and lower capacities. Let $\gamma : A \to \mathbf{Z}$ be a cost function on the arc set and $g : \mathbf{Z}^V \to \mathbf{Z}\cup\{+\infty\}$ an M-convex cost function such that $x(V) = 0$ for $x \in \mathrm{dom}_{\mathbf{Z}} g$. As a common generalization of the submodular flow problem [3, 6, 7] and the valuated matroid intersection [14, 15], Murota [16, 18] addresses the following generalized submodular flow problem with a nonseparable discrete convex cost function, which we call the discrete convex submodular flow problem:

$$\begin{array}{lll} \text{(DCSF)} & \text{Minimize} & g(\partial\varphi) + \sum_{a\in A} \gamma(a)\varphi(a) \\ & \text{subject to} & \underline{c}(a) \leq \varphi(a) \leq \overline{c}(a) \quad (a \in A), \\ & & \partial\varphi \in \mathrm{dom}_{\mathbf{Z}} g, \\ & & \varphi(a) \in \mathbf{Z} \quad (a \in A). \end{array}$$

This is nothing but the submodular flow problem if the M-convex cost function $g$ is constant. Thus there are efficient algorithms [5, 10, 22] to find a feasible solution, which will be referred to as a feasible flow.

For a vector $p \in \mathbf{Z}^V$, we denote by $\gamma_p$ the reduced cost function, i.e.,

$$\gamma_p(a) = \gamma(a) + p(\partial^+ a) - p(\partial^- a) \qquad (a \in A).$$

Partition $A$ into $A_p^+ = \{a \mid a \in A,\ \gamma_p(a) > 0\}$, $A_p^\circ = \{a \mid a \in A,\ \gamma_p(a) = 0\}$, and $A_p^- = \{a \mid a \in A,\ \gamma_p(a) < 0\}$. The following theorem of Murota [16, 18] characterizes the optimality for the discrete convex submodular flow problem.

**Theorem 2.1** *A feasible flow $\varphi : A \to \mathbf{Z}$ is optimal if and only if there exists a vector $p \in \mathbf{Z}^V$ that satisfies the following* (i)–(iii).

(i) $\forall a \in A_p^- : \varphi(a) = \overline{c}(a)$.

(ii) $\forall a \in A_p^+ : \varphi(a) = \underline{c}(a)$.

(iii) $\partial\varphi \in \arg\min\{g(x) - \langle p, x\rangle \mid x \in \mathbf{Z}^V\}$.

## 3 A Primal-Dual Algorithm

This section introduces a continuous version of the discrete convex submodular flow problem and presents an algorithm for solving it. The algorithm extends the primal-dual submodular flow algorithm of Cunningham–Frank [1].

We first extend the concept of L-convexity by saying that a function $f : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ is L-convex if it satisfies the following:

- $\exists r \in \mathbf{R}$, $\forall p \in \mathbf{Z}^V$: $f(p+\mathbf{1}) = f(p) + r$;
- $\forall p, q \in \mathbf{Z}^V$: $f(p) + f(q) \geq f(p \vee q) + f(p \wedge q)$.

Let $f$ be an L-convex function that satisfies $f(p+\mathbf{1}) = f(p)$ for any $p \in \mathbf{Z}^V$. The convex conjugate function $f^\bullet : \mathbf{R}^V \to \mathbf{R} \cup \{+\infty\}$ is now defined by

$$f^\bullet(x) = \sup\{\langle p, x\rangle - f(p) \mid p \in \mathbf{Z}^V\} \qquad (x \in \mathbf{R}^V).$$

We denote by $\mathrm{dom}_{\mathbf{R}} f^\bullet$ the effective domain of $f^\bullet$ in $\mathbf{R}^V$. Then $x(V) = 0$ for every $x \in \mathrm{dom}_{\mathbf{R}} f^\bullet$.

With a directed graph $G = (V, A)$ are associated upper and lower capacity functions $\underline{c} : A \to \mathbf{R} \cup \{-\infty\}$ and $\overline{c} : A \to \mathbf{R} \cup \{+\infty\}$ as well as an integral arc cost function $\gamma : A \to \mathbf{Z}$. The following continuous version of the discrete convex submodular flow problem will be referred to as CSF$(f, \gamma)$:

$$\begin{aligned} \text{Minimize} \quad & f^\bullet(\partial\varphi) + \sum_{a \in A} \gamma(a)\varphi(a) \\ \text{subject to} \quad & \underline{c}(a) \leq \varphi(a) \leq \overline{c}(a) \quad (a \in A), \\ & \partial\varphi \in \mathrm{dom}_{\mathbf{R}} f^\bullet, \\ & \varphi(a) \in \mathbf{R} \quad (a \in A). \end{aligned}$$

For an integral vector $p \in \mathbf{Z}^V$, let $B_p(f)$ denote a polyhedron defined by

$$B_p(f) = \{x \mid x \in \mathbf{R}^V,\ x(V) = 0,\ \forall X \subseteq V : x(X) \leq f(p + \chi_X) - f(p)\}.$$

Recall the partition of $A$ into $A_p^+ = \{a \mid a \in A,\ \gamma_p(a) > 0\}$, $A_p^\circ = \{a \mid a \in A,\ \gamma_p(a) = 0\}$, and $A_p^- = \{a \mid a \in A,\ \gamma_p(a) < 0\}$. An optimality criterion for CSF$(f, \gamma)$ is given by the following continuous version of Theorem 2.1.

**Theorem 3.1** *A feasible flow* $\varphi : A \to \mathbf{R}$ *is optimal if and only if there exists a function* $p : V \to \mathbf{Z}$ *that satisfies the following* (i)–(iii).

(i) $\forall a \in A_p^- : \varphi(a) = \overline{c}(a)$.

(ii) $\forall a \in A_p^+ : \varphi(a) = \underline{c}(a)$.

(iii) $\partial\varphi \in B_p(f)$.

Note that the "if" part is rather trivial. The "only if" part follows from the validity of the primal-dual algorithm described below.

The primal-dual algorithm repeats the following process for a feasible flow $\varphi$ and a potential $p$ with $\partial\varphi \in B_p(f)$. Given such $\varphi$ and $p$, we denote $D_\varphi^+(v) = \{a \mid v = \partial^+ a,\ a \in A_p^-,\ \varphi(a) < \overline{c}(a)\}$, $D_\varphi^-(v) = \{a \mid v = \partial^- a,\ a \in A_p^+,\ \varphi(a) > \underline{c}(a)\}$, and $D_\varphi(v) = D_\varphi^+(v) \cup D_\varphi^-(v)$ for $v \in V$.

The algorithm picks up a vertex $v^*$ with nonempty $D_\varphi(v^*)$. If no such vertex exists, the current $\varphi$ and $p$ are optimal. Otherwise, with reference to the new upper and lower capacities defined by

$$\underline{c}^*(a) = \begin{cases} \varphi(a) & (a \in A_p^-) \\ \underline{c}(a) & (a \in A_p^\circ \cup A_p^+), \end{cases} \qquad \overline{c}^*(a) = \begin{cases} \varphi(a) & (a \in A_p^+) \\ \overline{c}(a) & (a \in A_p^\circ \cup A_p^-), \end{cases}$$

the algorithm solves the following maximum submodular flow problem:

$$\begin{array}{lll}\text{(MSF)} & \text{Maximize} & \psi(D^+_\varphi(v^*)) - \psi(D^-_\varphi(v^*)) \\ & \text{subject to} & \underline{c}^*(a) \le \psi(a) \le \overline{c}^*(a) \quad (a \in A) \\ & & \partial\psi \in B_p(f).\end{array}$$

A cut for (MSF) means a vertex subset that contains $v^*$. For each cut $W$, let $\Delta^+W$ and $\Delta^-W$ respectively denote the sets of arcs leaving $W$ and entering $W$. We now consider the cut capacity

$$\begin{aligned}\kappa_\varphi(W) &= \overline{c}^*(\Delta^-W \backslash D^-_\varphi(v^*)) - \underline{c}^*(\Delta^+W \backslash D^+_\varphi(v^*)) \\ &\quad + \overline{c}^*(D^+_\varphi(v^*) \backslash \Delta^+W) - \underline{c}^*(D^-_\varphi(v^*) \backslash \Delta^-W) + f(p+\chi_W) - f(p).\end{aligned}$$

Then it follows from [7, Theorem 5.11] that the optimal objective value of (MSF) is equal to the minimum cut capacity $\min\{\kappa_\varphi(W) \mid v^* \in W \subseteq V\}$ unless $D_\psi(v^*)$ becomes empty.

If $D_\psi(v^*)$ is empty, the algorithm updates $\varphi$ to $\psi$ without changing $p$. Otherwise, it finds a minimum capacity cut $W$ containing $v^*$. Since $\psi(W) = f(p+\chi_W) - f(p)$, it follows from $\partial\psi \in B_p(f)$ and Lemma 3.2 below that every $X \subseteq V$ satisfies

$$\begin{aligned}\partial\psi(X) &= \partial\psi(X \cup W) + \partial\psi(X \cap W) - \partial\psi(W) \\ &\le f(p+\chi_{W\cup X}) + f(p+\chi_{W\cap X}) - f(p) - f(p+\chi_W) \\ &\le f(p+\chi_W+\chi_X) - f(p+\chi_W),\end{aligned}$$

which means $\partial\psi \in B_{p+\chi_W}(f)$. Thus the algorithm updates $p$ to $p+\chi_W$ as well as $\varphi$ to $\psi$ without violating (iii).

The primal-dual algorithm repeats this process until (i) and (ii) get satisfied. Note that one iteration reduces at least by one the sum of $\max\{|\gamma_p(a)| \mid a \in D_\varphi(v)\}$ for those vertices with nonempty $D_\varphi(v)$. Since $\gamma_p$ is integral, the algorithm eventually terminates after a finite number of iterations. Thus Theorem 3.1 has been proved.

The following easy lemma, which has been referred to in the above argument, will also be used later in Section 4.

**Lemma 3.2** *If $f$ is an L-convex function, then $Y \subseteq Z \subseteq V$ implies*

$$f(p) + f(p+\chi_Y+\chi_Z) \ge f(p+\chi_Y) + f(p+\chi_Z)$$

*for any $p \in \mathbf{Z}^V$.*

## 4 Conjugate Scaling

This section presents a cost-scaling framework to solve the discrete convex submodular flow problem (DCSF).

Given a vector $y \in \mathrm{dom}_{\mathbf{Z}} g$, we can efficiently find an integer subgradient of $g$ at $y$, i.e., a vector $p \in \mathbf{Z}^V$ such that $g(x) - g(y) \ge \langle p, x-y\rangle$ holds for $x \in \mathbf{Z}^V$. Thus we henceforth assume without loss of generality that an initial submodular flow $\varphi$ satisfies (iii) in Theorem 2.1 for $p = \mathbf{0}$ by replacing $g$ appropriately.

Let $f_\alpha : \mathbf{Z}^V \to \mathbf{R} \cup \{+\infty\}$ with $\alpha \in \mathbf{Z}$ be an L-convex function defined by

$$f_\alpha(p) = \frac{g^\bullet(\alpha p)}{\alpha} \qquad (p \in \mathbf{Z}^V).$$

Recall here that $g^\bullet$ denotes the convex conjugate function of the M-convex function $g$. Our cost-scaling algorithm repeatedly applies the primal-dual algorithm concerning $f_\alpha$ with an integer parameter $\alpha$ as follows.

**Algorithm** CONJUGATE SCALING

**Step 0:** Let $\varphi$ be an initial feasible flow satisfying $\partial\varphi \in \arg\min\{g(x) \in x \in \mathbf{Z}\}$. Put $p^* \leftarrow \mathbf{0}$, $K \leftarrow \max\{|\gamma(a)| \mid a \in A\}$, and $\alpha \leftarrow 2^{\lceil \log_2 K \rceil}$.

**Step 1:** Repeat the following (1-1)–(1-4) while $\alpha \geq 1$.

- **(1-1)** $\xi(a) \leftarrow \lceil \gamma(a)/\alpha \rceil$ for $a \in A$.
- **(1-2)** Find an integer vector $p \in \mathbf{Z}^V$ that maximizes $\langle p, \partial\varphi \rangle - f_\alpha(p)$ subject to $2p^* \leq p \leq 2p^* + n\mathbf{1}$.
- **(1-3)** Solve CSF$(f_\alpha, \xi)$ by the primal-dual algorithm starting from $\varphi$ and $p$ to obtain an optimal flow $\varphi^*$ and an optimal potential $p^*$.
- **(1-4)** $\varphi \leftarrow \varphi^*$, $\alpha \leftarrow \alpha/2$.

Recall that the primal-dual algorithm requires initial $\varphi$ and $p$ with $\partial\varphi \in B_p(f_\alpha)$. We now intend to verify that the integer vector $p$ obtained in (2-2) satisfies this condition.

Let $q$ be a minimal integer vector that maximizes $\langle q, \partial\varphi \rangle - f_\alpha(q)$ subject to $q \geq 2p^*$. Denote by $d$ the minimum positive integer that is not equal to $q(v) - 2p^*(v)$ for any $v \in V$, and consider a vertex subset $U = \{u \mid q(u) - 2p^*(u) > d\}$. Note that $q(v) = 2p^*(v)$ holds for some $v \in V$ because $f_\alpha(p) = f_\alpha(p + \mathbf{1})$ for any $p \in \mathbf{Z}^V$.

**Lemma 4.1** *The vertex subset $U$ is empty.*

*Proof.* We first claim

$$f_\alpha(2p^* + 2\chi_U) - f_\alpha(2p^* + \chi_U) \leq f_\alpha(q) - f_\alpha(q - \chi_U). \tag{4.2}$$

Put $\ell = \max\{q(v) - 2p^*(v)\}$, and consider $Y_i = \{v \mid q(v) - 2p^*(v) \geq i\}$ for $i = 1, \ldots, \ell$. We also denote $q_j = 2p^* + \sum_{i=1}^{j} \chi_{Y_i}$ for $j = 0, 1, \ldots, \ell$. Note that $q_0 = 2p^*$, $q_\ell = q$, and $Y_d = Y_{d+1} = U$ hold. Since $Y_i \supseteq U$ for $i = 1, \ldots, d$, Lemma 3.2 implies that $f_\alpha(q_{j-1} + 2\chi_U) - f_\alpha(q_{j-1} + \chi_U) \leq f_\alpha(q_j + 2\chi_U) - f_\alpha(q_j + \chi_U)$ for $j = 1, \ldots, d-1$. Since $Y_i \subseteq U$ for $i = d, \ldots, \ell$, Lemma 3.2 also implies that $f_\alpha(q_j) - f_\alpha(q_j - \chi_U) \leq f_\alpha(q_{j+1}) - f_\alpha(q_{j+1} - \chi_U)$ for $j = d+1, \ldots, \ell - 1$. Thus, by $q_{d-1} + \chi_U = q_d = q_{d+1} - \chi_U$, we obtain (4.2).

The current $\varphi$, obtained by the primal-dual algorithm in the previous scaling phase, satisfies $\partial\varphi(U) \leq f_{2\alpha}(p^* + \chi_U) - f_{2\alpha}(p^*) = \{f_\alpha(2p^* + 2\chi_U) - f_\alpha(2p^*)\}/2$. The L-convexity of $f_\alpha$ implies $f_\alpha(2p^* + \chi_U) - f_\alpha(2p^*) = f_\alpha(2p^* + \chi_U + \mathbf{1}) - f_\alpha(2p^* + \mathbf{1}) \leq f_\alpha(2p^* + 2\chi_U) - f_\alpha(2p^* + \chi_U)$. Therefore, $\partial\varphi(U) \leq f_\alpha(2p^* + 2\chi_U) - f_\alpha(2p^* + \chi_U) \leq f_\alpha(q) - f_\alpha(q - \chi_U)$, where the last inequality follows from (4.2). Hence $\langle q, \partial\varphi\rangle - f_\alpha(q) \leq \langle q - \chi_U, \partial\varphi\rangle - f_\alpha(q - \chi_U)$, which contradicts the definition of $q$ unless $U$ is empty. □

As a consequence of Lemma 4.1, we have $q \leq 2p^* + n\mathbf{1}$. Hence the integer vector $p$ obtained in (2-2) in fact maximizes $\langle p, \partial\varphi\rangle - f_\alpha(p)$ over $\mathbf{Z}^V$. In particular, $\langle p + \chi_X, \partial\varphi\rangle - f_\alpha(p + \chi_X) \leq \langle p, \partial\varphi\rangle - f_\alpha(p)$ holds for every $X \subseteq V$, which implies $\partial\varphi \in B_p(f_\alpha)$.

We now discuss the time complexity, provided that an evaluation oracle for the M-convex function $g$ is available. The algorithm performs $\mathrm{O}(\log K)$ scaling phases. In each scaling phase, $f_\alpha$ is computed in polynomial time by an M-convex function minimization algorithm of Shioura [21]. The maximization problem in (1-2) is solved in polynomial time with the aid of the ellipsoid method because it is equivalent, by a result of Fujishige–Murota [8], to minimizing a submodular integrally convex function in the sense of Favati–Tardella [4]. The number of iterations in the primal-dual algorithm in (2-3) is at most $\sum_v \max\{|\gamma_p(a)| \mid a \in D_\varphi(v)\}$, where the summation is taken over those vertices adjacent to arcs violating (i) or (ii) in Theorem 3.1, and hence bounded by $\mathrm{O}(n^2)$. Each iteration solves one maximum submodular flow problem in polynomial time. Thus we have the following theorem.

**Theorem 4.2** *The algorithm* CONJUGATE SCALING *solves the discrete convex submodular flow problem* (DCSF) *in polynomial time.*

# 5 Conclusion

We have devised a polynomial time algorithm for the discrete convex submodular flow problem by scaling the convex cost function via the conjugacy relation. The resulting algorithm is an extension of the primal-dual algorithm of Cunningham–Frank [1]. It may be interesting to know if other polynomial time submodular flow algorithms [11, 12] extend to this general framework.


# Acknowledgements

The authors are grateful to Kazuo Murota for suggesting the scaling approach via the conjugacy relation and for careful reading of the manuscript. Thanks are also due to Tom McCormick for helpful comments on the manuscript.


# 参考文献


[1] W. H. Cunningham and A. Frank: A primal-dual algorithm for submodular flows, *Math. Oper. Res.*, **10** (1985), pp. 251–262.

[2] A. W. M. Dress and W. Wenzel: Valuated matroids, *Adv. Math.*, **93** (1992), pp. 214–250.

[3] J. Edmonds and R. Giles: A min-max relation for submodular functions on graphs, *Ann. Discrete Math.*, 1 (1977), pp. 185–204.

[4] P. Favati and F. Tardella: Convexity in nonlinear integer programming, *Ricerca Operativa*, **53** (1990), pp. 3–44.

[5] A. Frank: Finding feasible vectors of Edmonds–Giles polyhedra, *J. Combinatorial Theory*, **B36** (1984), pp. 221–239.

[6] A. Frank and É. Tardos: Generalized polymatroids and submodular flows, *Math. Programming*, **42** (1988), 489–563.

[7] S. Fujishige: *Submodular Functions and Optimization*, North-Holland, 1991.

[8] S. Fujishige and K. Murota: On the relationship between L-convex functions and submodular integrally convex functions, RIMS Preprint 1152, Kyoto University, 1997.

[9] S. Fujishige and K. Murota: Short proofs of the separation theorems for L-convex/concave and M-convex/concave functions, RIMS Preprint 1167, Kyoto University, 1997.

[10] S. Fujishige and X. Zhang: New algorithms for the intersection problem of submodular systems, *Japan J. Indust. Appl. Math.*, **9** (1992), pp. 369–382.

[11] S. Iwata: A capacity scaling algorithm for convex cost submodular flows, *Math. Programming*, **76** (1997), 299–308.

[12] S. Iwata, S. T. McCormick, and M. Shigeno: A faster algorithm for minimum cost submodular flows, *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms* (1998), pp. 167–174.

[13] L. Lovász: Submodular functions and convexity, *Mathematical Programming — The State of the Art* (A. Bachem, M. Grötschel, and B. Korte, eds.), Springer-Verlag, 1983, pp. 235–257.

[14] K. Murota: Valuated matroid intersection, I: Optimality criteria, II: Algorithms, *SIAM J. Discrete Math.*, **9** (1996), pp. 545–576.

[15] K. Murota: Fenchel-type duality for matroid valuations, *Math. Programming*, **82** (1998), pp. 357–375.

[16] K. Murota: Submodular flow problem with a nonseparable cost function, RIMS Preprint 1061, Kyoto University, 1996.

[17] K. Murota: Convexity and Steinitz's exchange property, *Adv. Math.*, **124** (1996), pp. 272–311.

[18] K. Murota: Discrete convex analysis, *Math. Programming*, **83** (1998), pp. 313–371.

[19] 室田: 離散凸解析, 離散構造とアルゴリズム V (藤重 編), 近代科学社, 1998, pp. 51–100.

[20] R. T. Rockafellar: *Convex Analysis*, Princeton University Press, 1970.

[21] A. Shioura: Minimization of an M-convex function, *Discrete Appl. Math.*, **84** (1998), pp. 215–220.

[22] É. Tardos, C. A. Tovey, and M. A. Trick: Layered augmenting path algorithms, *Math. Oper. Res.*, **11** (1986), pp. 362–370.

アルゴリズム 65−6
(1998. 10. 28)

# 確率的故障に耐える疎なネットワーク

山田敏規　　上野修一
東京工業大学　電子物理工学科
〒152–8552　東京都目黒区大岡山 2–12–1
E-mail: {*yamada, ueno*}*@ss.titech.ac.jp*

**概要**

小文では，$N$個のプロセッサから成る任意のネットワーク $G$ に対して，各プロセッサが定確率で独立に故障してたとしても，故障していない部分が高い確率で $G$ を含んでいるような $O(N)$ 個のプロセッサから成る耐故障ネットワークを構成するための統一的な手法を与える．この構成方法に基づいて，$N$個のプロセッサと $M$本の通信回線から成る循環ネットワーク，ハイパーキューブ，*de Bruijn* ネットワーク，シャッフル交換ネットワーク，$CCC$ に対して，$O(N)$ 個のプロセッサと $O(M \log N)$ 本の通信回線から成る耐故障ネットワークを構成できることも示す．

# Sparse Networks Tolerating Random Faults

Toshinori Yamada and Shuichi Ueno

Department of Physical Electronics, Tokyo Institute of Technology

2–12–1 O-okayama, Meguro-ku, Tokyo 152–8552, Japan
E-mail: {*yamada, ueno*}*@ss.titech.ac.jp*

**Abstract**

*This paper proposes a general method to construct a fault-tolerant network $G^*$ for any network $G$ with $N$ processors such that $G^*$ has $O(N)$ processors and contains a fault-free isomorphic copy of $G$ with high probability even if processors fail independently with constant probability. Based on the construction, we also show that we can construct such fault-tolerant networks with $O(N)$ processors and $O(M \log N)$ communication links for a circulant network, hypercube, de Bruijn network, shuffle-exchange network, and cube-connected-cycles with $N$ processors and $M$ communication links.*

## 1 Introduction

This paper considers the following problem in connection with the design of fault-tolerant interconnection networks for multiprocessor systems: Given an $N$-vertex graph $G$, construct an $O(N)$-vertex graph $G^*$ with a minimum number of edges such that even after deleting vertices from $G^*$ independently with constant probability, the remaining graph contains $G$ as a subgraph, with probability converging to 1, as $N \to \infty$. $G^*$ is called an RFT (random-fault-tolerant) graph for $G$. Let $V(G)$ and $E(G)$ be the vertex set and edge set of a graph $G$, respectively. Fraigniaud, Kenyon, and Pelc showed that for any $N$-vertex graph $G$, there exists an RFT graph for $G$ with $O(|E(G)| \cdot \log^2 N)$ edges, and that there exists a graph $G$ such that any RFT graph for $G$ has $\omega(|E(G)|)$ edges. It is also known that for an $N$-vertex path[1, 2], cycle[2], and tree with bounded vertex degree[3], there exist RFT graphs with $O(N)$ edges; for an $N$-vertex mesh and torus[5], there exist RFT graphs with $O(N \log\log N)$ edges; and for an $N$-vertex tree, there exists an RFT graph with $O(N \log N)$ edges [2].

In this paper, we propose a general method to construct an RFT graph for any graph. Based on the construction, we show that if $G$ is an $N$-vertex circulant graph, hypercube, de Bruijn graph, shuffle-exchange graph, or cube-connected-cycles, we can construct an RFT graph for $G$ with $O(|E(G)| \cdot \log N)$ edges.

## 2 General Construction

For any positive integer $k$, let $[k] = \{0, 1, \ldots, k-1\}$. For any set of $S$, a collection $\mathcal{S} = \{S_0, S_1, \ldots, S_{k-1}\}$ of subsets of $S$ is a partition of $S$ if $\bigcup_{i\in[k]} S_i = S$ and $S_i \cap S_j = \emptyset$ for any $i \neq j$.

Let $G$ be any $N$-vertex graph. For any partition $\mathcal{V} = \{V_0, V_1, \ldots, V_{k-1}\}$ of $V(G)$, define

$$\Lambda(G, \mathcal{V}) = \{(i,j) |\ \exists (u,v) \in E(G)(u \in V_i, v \in V_j)\}$$

and

$$\lambda(G, \mathcal{V}) = |\Lambda(G, \mathcal{V})|.$$

Let $0 < p < 1$ be the probability for each vertex to be deleted. A deleted and undeleted vertex are said to be faulty and fault-free, respectively.

Let $\mathcal{V} = (V_0, V_1, \ldots, V_{k-1})$ be any partition of $V(G)$ such that $|V_i| \leq \alpha \ln N$ and $k \leq \beta N/\ln N$ for some fixed positive numbers $\alpha$ and $\beta$. Let $V_0^*, V_1^*, \ldots$, and $V_{k-1}^*$ be $k$ sets such that $|V_i^*| = \lceil \gamma \ln N \rceil$ for any $i \in [k]$ and $V_i^* \cap V_j^* = \emptyset$ for any $i \neq j$, where

$$\gamma = \frac{(\sqrt{2\alpha+1}+1)^2}{2(1-p)}.$$

Note that $\gamma$ is fixed since $\alpha$ and $p$ are fixed. Then, $G^*[\mathcal{V}]$ is the graph defined as follows:

$$\begin{aligned} V(G^*[\mathcal{V}]) &= V_0^* \cup V_1^* \cup \cdots \cup V_{k-1}^*; \\ E(G^*[\mathcal{V}]) &= \left\{ (u^*, v^*) \;\middle|\; \begin{array}{l} u^* \in V_i^*,\ v^* \in V_j^*, \\ (i,j) \in \Lambda(G, \mathcal{V}) \end{array} \right\}. \end{aligned}$$

**Theorem 1** *Let $G$ be any $N$-vertex graph, and let $\mathcal{V} = \{V_0, V_1, \ldots, V_{k-1}\}$ be any partition of $V(G)$ such that $|V_i| = O(\ln N)$ and $k = O(N/\ln N)$. Then $G^*[\mathcal{V}]$ is an RFT graph for $G$ with $O(\lambda(G, \mathcal{V}) \cdot \log^2 N)$ edges.*

**Proof :** We prove the theorem by a series of lemmas. It is easy to see the following two lemmas.

**Lemma 1** $|V(G^*[\mathcal{V}])| \leq \frac{\beta N}{\ln N} \cdot \lceil \gamma \ln N \rceil$. ∎

**Lemma 2** $|E(G^*[\mathcal{V}])| \leq \lambda(G, \mathcal{V}) \cdot \lceil \gamma \ln N \rceil^2$. ∎

Now we prove that $G^*[\mathcal{V}]$ is an RFT graph for $G$. We need a few probabilistic notations and lemmas.

For any event $E$, let Prob$[E]$ denote the probability of $E$. For any random variable $X$ and real number $r$, let $\{X \leq r\}$ denote the event that $X \leq r$. The probability of $\{X \leq r\}$ is denoted by Prob$[X \leq r]$ instead of Prob$[\{X \leq r\}]$. The following inequality is well-known as Chernoff Bound.

**Lemma 3** [4] *Let $X$ be the binomial variable with parameters $m$ and $q$, that is, the number of successes in $m$ Bernoulli trials with probabilities $q$ for success and $1-q$ for failure. Then, for any constant $0 < \epsilon < 1$,*

$$\text{Prob}[X \leq (1-\epsilon)qm] \leq \exp(-\frac{1}{2}\epsilon^2 qm).$$

∎

**Lemma 4** *Let $Y_i$ be the number of fault-free vertices of $V_i^*$. Then, for any $i \in [k]$,*

$$\text{Prob}[Y_i \leq \alpha \ln N] \leq \frac{1}{N}.$$

*Moreover,*

$$\text{Prob}[\bigcup_{i=0}^{k-1} \{Y_i \leq \alpha \ln N\}] \leq \frac{\beta}{\ln N}.$$

**Proof :** Set $\epsilon = 2/(\sqrt{2\alpha+1}+1)$, $q = 1-p$, and $m = \lceil \gamma \ln N \rceil$. Since $0 < \epsilon < 1$,

$$\begin{aligned} &(1-\epsilon)qm \\ &\geq \frac{\sqrt{2\alpha+1}-1}{\sqrt{2\alpha+1}+1} \cdot (1-p) \cdot \frac{(\sqrt{2\alpha+1}+1)^2}{2(1-p)} \cdot \ln N \\ &= \frac{2\alpha}{(\sqrt{2\alpha+1}+1)^2} \cdot \frac{(\sqrt{2\alpha+1}+1)^2}{2} \cdot \ln N \\ &= \alpha \ln N, \end{aligned}$$

and

$$\begin{aligned} &\frac{1}{2}\epsilon^2 qm \\ &\geq \frac{1}{2} \cdot \frac{4}{(\sqrt{2\alpha+1}+1)^2} \cdot (1-p) \cdot \frac{(\sqrt{2\alpha+1}+1)^2}{2(1-p)} \cdot \ln N \\ &= \ln N, \end{aligned}$$

we obtain by Lemma 3 that

$$\begin{aligned} \text{Prob}[Y_i \leq \alpha \ln N] &\leq \text{Prob}[Y_i \leq (1-\epsilon)qm] \\ &\leq \exp(-\frac{1}{2}\epsilon^2 qm) \\ &\leq \frac{1}{N}. \end{aligned}$$

Moreover,

$$\begin{aligned} \text{Prob}[\bigcup_{i=0}^{k-1} \{Y_i \leq \alpha \ln N\}] &\leq \sum_{i=0}^{k-1} \text{Prob}[Y_i \leq \alpha \ln N] \\ &\leq \frac{\beta}{\ln N}. \end{aligned}$$

∎

**Lemma 5** *$G^*[\mathcal{V}]$ is an RFT graphs for $G$.*

**Proof :** Let $\phi$ be a one-to-one mapping from $V(G)$ to $V(G^*[\mathcal{V}])$ such that $\phi(v)$ is a fault-free vertex of $V_i^*$ for any $v \in V_i$. By Lemma 4, such $\phi$ exists with probability at least $1 - (\beta/\ln N)$.

Now we show that $(\phi(u), \phi(v)) \in E(G^*[\mathcal{V}])$ for any $(u, v) \in E(G)$. Let $u \in V_i$ and $v \in V_j$. Then, $(i,j) \in \Lambda(G, \mathcal{V})$. Since $\phi(u) \in V_i^*$ and $\phi(v) \in V_j^*$, we conclude that $(\phi(u), \phi(v)) \in E(G^*[\mathcal{V}])$. Hence $G^*[\mathcal{V}]$ is an RFT graphs for $G$. ∎

This completes the proof of Theorem 1. ∎

Since $\lambda(G, \mathcal{V}) \leq |E(G)|$, we obtain the following corollary.

**Corollary 1** [2] *Let $G$ be any $N$-vertex graph, and let $\mathcal{V} = \{V_0, V_1, \ldots, V_{k-1}\}$ be any partition of $V(G)$ such that $|V_i| = O(\ln N)$ and $k = O(N/\ln N)$. $G^*[\mathcal{V}]$ is an RFT graph for $G$ with $O(|E(G)| \cdot \log^2 N)$ edges.*

# 3 RFT Graphs for Circulant Graphs

Let $N$ be a positive integer and let $S \subset [N]$. The $N$-vertex circulant graph with connection set $S$, denoted by $C_N(S)$, is the graph defined as follows:

$$\begin{aligned} V(C_N(S)) &= [N]; \\ E(C_N(S)) &= \{(u,v) | \exists s \in S(v = (u \pm s) \bmod N)\}. \end{aligned}$$

An edge $(u,v)$ is said to be of offset $s$ if $v = (u \pm s) \bmod N$.

It is easy to see the following lemma.

**Lemma 6** *Let $S' = \{s | s \in S$ and $s \leq N/2\} \cup \{N - s | s \in S$ and $s > N/2\}$. Then $C_N(S')$ is isomorphic to $C_N(S)$. Moreover,*

$$|E(C_N(S))| = \begin{cases} (2|S'|-1)N/2 & \textit{if } N/2 \in S, \\ |S'|N & \textit{otherwise.} \end{cases}$$

∎

Let $c_N = \lceil \log N \rceil$ and $k_N = \lceil N/c_N \rceil$. Define $U_i = \{v \in [N] | \lfloor v/c_N \rfloor = i\}$ for any $i \in [k_N]$. Then, $\mathcal{U}_N = (U_0, U_1, \ldots, U_{k_N-1})$ is a partition of $[N]$ such that $|U_i| \leq c_N \leq \log N + 1$ for any $i \in [k_N]$ and $|\mathcal{U}_N| = k_N = \lceil N/c_N \rceil \leq (N/\log N) + 1$.

**Theorem 2** *$C_N^*(S)[\mathcal{U}_N]$ is an RFT graph for $C_N(S)$ with $O(|E(C_N(S))| \cdot \log N)$ edges.*

**Proof :** By Lemma 6, we may assume that if $s \in S$ then $s \leq N/2$. Thus, By Theorem 1 and Lemma 6, it suffices to prove that $\lambda(C_N(S), \mathcal{U}_N) = O(|S|N/\log N)$.

Consider any edge $(u,v) \in E(C_N(S))$ of offset $s \in S$. Assume without loss of generality that $v = (u + s) \bmod N$. Let $u = ic_N + a$ and $v = jc_N + b$, where $0 \leq a, b < c_N$. Then $u \in U_i$ and $v \in U_j$. We have the following two cases:

(i) $u < N - s$: We have $v = u + s$. Then, $jc_N + b = ic_N + a + s$, and so $j = i + (s + a - b)/c_N$. Since $i$ and $j$ are integers, $l = (s + a - b)/c_N$ is an integer. Thus, $\lfloor s/c_N \rfloor = l + \lfloor (b-a)/c_N \rfloor$. that is $l = \lfloor s/c_N \rfloor - \lfloor (b-a)/c_N \rfloor$. Since $0 \leq a, b < c_N$, $\lfloor (b-a)/c_N \rfloor = 0$ or $-1$, and so $l = \lfloor s/c_N \rfloor$ or $l = \lfloor s/c_N \rfloor + 1$. Hence $j = i + \lfloor s/c_N \rfloor$ or $j = i + \lfloor s/c_N \rfloor + 1$.

(ii) $u \geq N - s$: We have $v = u + s - N$. Then, $jc_N + b = ic_N + a + s - N$, and so $j = i + (a - b + s - N)/c_N$. Since $N = k_N c_N - d$ by the definition of $k_N$, we obtain $j = i - k_N + (a - b + d + s)/c_N$, where $0 \leq d < c_N$. Since $i$ and $j$ are integers, $l = (a - b + d + s)/c_N$ is an integer. Thus, $\lfloor s/c_N \rfloor = l + \lfloor (b - a - d)/c_N \rfloor$, that is $l = \lfloor s/c_N \rfloor - \lfloor (b-a-d)/c_N \rfloor$. Since $0 \leq a, b, d < c_N$, $\lfloor (b - a - d)/c_N \rfloor = 0$, $-1$, or $-2$, and so we have $l = \lfloor s/c_N \rfloor$, $l = \lfloor s/c_N \rfloor + 1$, or $l = \lfloor s/c_N \rfloor + 2$. Hence $j = (i + \lfloor s/c_N \rfloor) \bmod k_N$, $j = (i + \lfloor s/c_N \rfloor + 1) \bmod k_N$, or $j = (i + \lfloor s/c_N \rfloor + 2) \bmod k_N$.

Thus,

$$\begin{aligned} &\Lambda(C_N, \mathcal{V}) \\ &\quad \subseteq \{(i,j) | j = (i \pm \lfloor s/c_N \rfloor) \bmod k_N\} \\ &\qquad \cup \{(i,j) | j = (i \pm \lfloor s/c_N \rfloor + 1) \bmod k_N\} \\ &\qquad \cup \{(i,j) | j = (i \pm \lfloor s/c_N \rfloor + 2) \bmod k_N\}, \end{aligned}$$

and we have

$$\lambda(C_N(S), \mathcal{U}_N) \leq 3|S|k_N \leq \frac{3|S|N}{\log N} + 3|S| = O(\frac{|S|N}{\log N}).$$

∎

# 4 RFT Graphs for Hypercubic Graphs

For any $v = [v_n, v_{n-1}, \ldots, v_1] \in [2]^n$, let

$$\begin{aligned} \sigma(v) &= [v_{n-1}, \ldots, v_1, v_n], \\ \chi_i(v) &= [v_n, \ldots, v_{i+1}, \overline{v_i}, v_{i-1}, \ldots, v_1], \text{ and} \\ \rho_i(v) &= [v_i, \ldots, v_1], \end{aligned}$$

where $\overline{v_i}$ denotes the complement of $v_i$, that is $\overline{v_i} = 1$ if $v_i = 0$, and $\overline{v_i} = 0$ otherwise.

Let

$$V_x = \{v \in [2]^n | \rho_{n - \lceil \log n \rceil}(v) = x\}$$

for any $x \in [2]^{n - \lceil \log n \rceil}$ and let

$$\mathcal{V}_n = \{V_x | x \in [2]^{n - \lceil \log n \rceil}\}.$$

Then $\mathcal{V}_n$ is a partition of $[2]^n$ such that $|V_x| \leq 2 \log N$ for any $x \in [2]^{n - \lceil \log n \rceil}$ and $|\mathcal{V}_n| \leq N/\log N$, where $N = |[2]^n| = 2^n$.

## 4.1 RFT Graphs for Hypercubes

The $n$-cube ($n$-dimensional cube) $Q(n)$ is the graph defined as follows:

$$\begin{aligned} V(Q(n)) &= [2]^n; \\ E(Q(n)) &= \{(u,v) | v = \chi_i(u),\ 1 \leq i \leq n\}. \end{aligned}$$

It is easy to see that $|V(Q(n))| = N$ and $|E(Q(n))| = N \log N/2$, where $N = 2^n$. An edge $(u,v)$ is called an $i$-edge if $v = \chi_i(u)$. A graph $G$ is called a hypercube if $G$ is isomorphic to $Q(n)$ for some $n$.

**Theorem 3** *$Q^*(n)[\mathcal{V}_n]$ is an RFT graph for $Q(n)$ with $O(N \log^2 N)$ edges.*

**Proof :** By Theorem 1, it suffices to prove that $\lambda(Q(n), \mathcal{V}_n) = O(N)$.

Consider any $i$-edge $(u, v) \in E(Q(n))$. Let $x = \rho_{n-\lceil\log n\rceil}(u)$ and $y = \rho_{n-\lceil\log n\rceil}(v)$. It is easy to see that $y = \chi_i(x)$ if $1 \leq i \leq n - \lceil\log n\rceil$, and $x = y$ otherwise. Thus,

$$\Lambda(Q(n), \mathcal{V}_n) \subseteq \{(x,y) \mid y = \chi_i(x) \text{ or } x = y\},$$

and we have

$$\begin{aligned}\lambda(Q(n), \mathcal{V}_n) &\leq \{\frac{1}{2}(n - \lceil\log n\rceil) + 1\} \cdot 2^{n-\lceil\log n\rceil} \\ &\leq \{\frac{1}{2}(\log N - \log\log N) + 1\} \cdot \frac{N}{\log N} \\ &= O(N).\end{aligned}$$

∎

## 4.2 RFT Graphs for de Bruijn Graphs

The $n$-dimensional de Bruijn graph $dB(n)$ is the graph defined as follows:

$$\begin{aligned}V(dB(n)) &= [2]^n; \\ E(dB(n)) &= \{(u,v) \mid v = \sigma(u) \text{ or } u = \sigma(v)\} \\ &\quad \cup \left\{ (u,v) \;\middle|\; \begin{array}{c} v = \chi_1(\sigma(u)) \\ \text{or} \\ u = \chi_1(\sigma(v)) \end{array} \right\}\end{aligned}$$

It is easy to see that $|V(dB(n))| = N$ and $|E(dB(n))| = 2N$, where $N = 2^n$.

**Theorem 4** *$dB^*(n)[\mathcal{V}_n]$ is an RFT graph for $dB(n)$ with $O(N \log N)$ edges.*

**Proof :** By Theorem 1, it suffices to prove that $\lambda(dB(n), \mathcal{V}_n) = O(N/\log N)$.

Consider any edge $(u, v) \in E(dB(n))$. Assume without loss of generality that $v = \sigma(u)$ or $v = \chi_1(\sigma(u))$. Let $x = \rho_{n-\lceil\log n\rceil}(u)$ and $y = \rho_{n-\lceil\log n\rceil}(v)$. It is easy to see that $y = \sigma(x)$ or $y = \chi_1(\sigma(x))$. Thus,

$$\begin{aligned}\Lambda(dB(n), \mathcal{V}_n) &\subseteq \{(x,y) \mid y = \sigma(x) \text{ or } x = \sigma(y)\} \\ &\quad \cup \left\{ (x,y) \;\middle|\; \begin{array}{c} y = \chi_1(\sigma(x)) \\ \text{or} \\ x = \chi_1(\sigma(y)) \end{array} \right\},\end{aligned}$$

and we have

$$\lambda(dB(n), \mathcal{V}_n) \leq 2^{n-\lceil\log n\rceil+1} \leq \frac{2N}{\log N} = O(\frac{N}{\log N}).$$

∎

## 4.3 RFT Graphs for Shuffle-Exchange Graphs

The $n$-dimensional shuffle-exchange graph $SE(n)$ is the graph defined as follows:

$$\begin{aligned}V(SE(n)) &= [2]^n; \\ E(SE(n)) &= \{(u,v) \mid v = \sigma(u) \text{ or } u = \sigma(v)\} \\ &\quad \cup \{(u,v) \mid v = \chi_1(u)\}.\end{aligned}$$

It is easy to see that $|V(SE(n))| = N$ and $|E(SE(n))| = 3N/2$, where $N = 2^n$.

**Theorem 5** *$SE^*(n)[\mathcal{V}_n]$ is an RFT graph for $SE(n)$ with $O(N \log N)$ edges.*

**Proof :** By Theorem 1, it suffices to prove that $\lambda(SE(n), \mathcal{V}_n) = O(N/\log N)$.

Consider any edge $(u, v) \in E(SE(n))$. Let $x = \rho_{n-\lceil\log n\rceil}(u)$ and $y = \rho_{n-\lceil\log n\rceil}(v)$. If $v = \sigma(u)$ then we have $y = \sigma(x)$ or $y = \chi_1(\sigma(x))$. If $v = \chi_1(u)$ then $x = y$. Thus,

$$\begin{aligned}\Lambda(SE(n), \mathcal{V}_n) &\subseteq \{(x,y) \mid y = \sigma(x) \text{ or } x = \sigma(y)\} \\ &\quad \cup \left\{ (x,y) \;\middle|\; \begin{array}{c} y = \chi_1(\sigma(x)) \\ \text{or} \\ x = \chi_1(\sigma(y)) \end{array} \right\} \\ &\quad \cup \{(x,y) \mid x = y\},\end{aligned}$$

and we have

$$\lambda(SE(n), \mathcal{V}_n) \leq 3 \cdot 2^{n-\lceil\log n\rceil} \leq \frac{3N}{\log N} = O(\frac{N}{\log N}).$$

∎

## 4.4 RFT Graphs for CCC's

The $n$-dimensional cube-connected-cycles(CCC), denoted by $CCC(n)$, is the graph defined as follows:

$$\begin{aligned}V(CCC(n)) &= [2]^n \times [n]; \\ E(CCC(n)) &= \{([v,i],[v,j]) \mid j = (i \pm 1) \bmod n\} \\ &\quad \cup \{([u,i],[v,i]) \mid v = \chi_{i+1}(u)\},\end{aligned}$$

where $u, v \in [2]^n$ and $i, j \in [n]$. It is easy to see that $|V(CCC(n))| = N$ and $|E(CCC(n))| = 3N/2$, where $N = n2^n$.

Let

$$V'_{[x,i]} = \{[u,i] \in V(CCC(n)) \mid \rho_{n-\lceil\log n\rceil}(u) = x\}$$

for any $x \in [2]^{n-\lceil\log n\rceil}$ and $i \in [n]$ and let

$$\mathcal{V}'_n = \{V_{[x,i]} \mid x \in [2]^{n-\lceil\log n\rceil},\ i \in [n]\}.$$

It is easy to see that $\mathcal{V}'_n$ is a partition of $V(CCC(n))$ such that $|V_{[x,i]}| \leq 2\log N$ for any $x \in [2]^{n-\lceil\log n\rceil}$ and $i \in [n]$, and $|\mathcal{V}'_n| \leq 2N/\log N$.

**Theorem 6** *$CCC^{*}(n)[\mathcal{V}'_n]$ is an RFT graph for $CCC(n)$ with $O(N \log N)$ edges.*

**Proof :** By Theorem 1, it suffices to prove that $\lambda(CCC(n), \mathcal{V}_n) = O(N/\log N)$.

Consider any edge $([u,i],[v,j]) \in E(CCC(n))$. Let $x = \rho_{n-\lceil \log n \rceil}(u)$ and $y = \rho_{n-\lceil \log n \rceil}(v)$. If $u = v$ then $x = y$. If $v = \chi_{i+1}(u)$ then $y = \chi_{i+1}(x)$ or $x = y$. Thus,

$$\begin{aligned}
&\Lambda(CCC(n), \mathcal{V}'_n) \\
&\subseteq \{([x,i],[y,j] |\ x = y, j = (i \pm 1) \bmod n\} \\
&\quad \cup \{[x,i],[y,j] |\ y = \chi_{i+1}(x), i = j\} \\
&\quad \cup \{[x,i],[y,j] |\ x = y, i = j\},
\end{aligned}$$

and we have

$$\begin{aligned}
\lambda(CCC(n), \mathcal{V}'_n) &\leq \frac{5}{2} n 2^{n-\lceil \log n \rceil} \\
&= \frac{5N}{\log N} \\
&= O(\frac{N}{\log N}).
\end{aligned}$$

■

**Acknowledgements:** The authors are grateful to Professors Y. Kajitani for his encouragement. The research is a part of CAD21 Project at TIT.

# References

[1] N. Alon and F. R. K. Chung. Explicit construction of linear sized tolerant networks. *Discrete Math.*, Vol. 72, pp. 15–19, 1988.

[2] Pierre Fraigniaud, Claire Kenyon, and Andrzej Pelc. Finding a target subnetwork in sparse networks with random faults. *Information Processing Letters*, Vol. 48, pp. 297–303, 1993.

[3] J. Friedman and N. Pippenger. Expanding graphs contains all small trees. *Combinatorica*, Vol. 7, No. 1, pp. 71–76, 1987.

[4] Torben Hagerup and Christine Rüb. A guided tour of Chernoff bounds. *Information Processing Letters*, Vol. 33, pp. 305–308, 1989/1990.

[5] Hisao Tamaki. Construction of the mesh and the torus tolerating a large number of faults. *Proc. 6th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA '94)*, pp. 268–277, 1994.