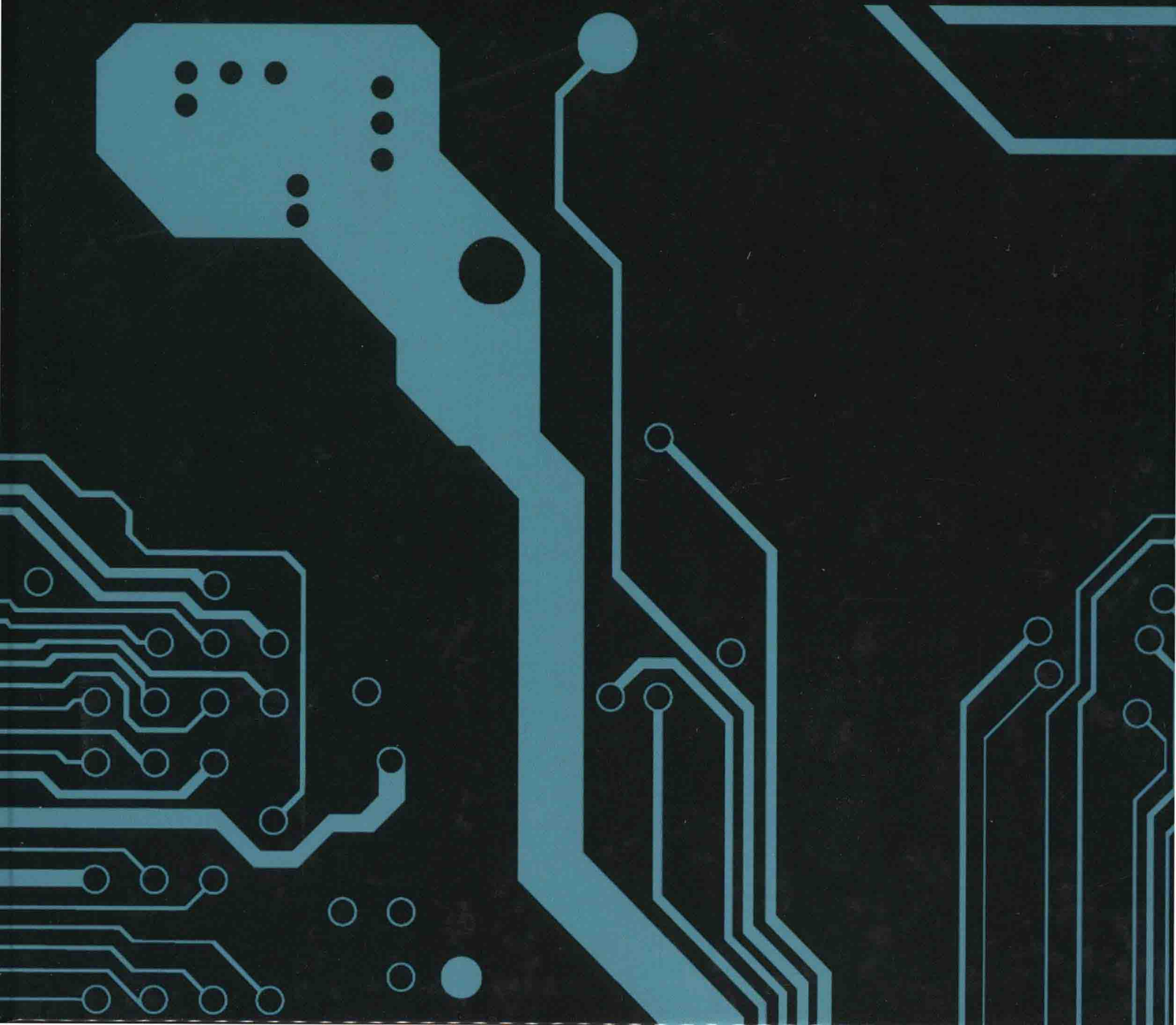# Handbook of
Electrical and
Computer Engineering

Volume III

Dewayne Hopper

Handbook of Electrical and Computer Engineering Volume III

Edited by **Dewayne Hopper**

NY RESEARCH
P R E S S

New York

Handbook of Electrical and Computer Engineering: Volume III
Edited by Dewayne Hopper

Handbook of Electrical and Computer Engineering

Volume III

Preface

The field of engineering that deals with the study and application of electronics and electricity is termed as electrical engineering. This field gained recognition as one of the biggest fields of engineering during the 19th century. Electric power has gained a foothold in the most basic activities of our lives. There is not a part of our daily lives that is not involved with electrical power in some way or the other. There are many subdivisions of electrical engineering like telecommunications, control systems, instrumentation and electronics, though electronics can be termed as a field in its own right. The skills required of an electrical engineer are variable and they work in a huge variety of industries. Computer engineering on the other hand is a discipline that combines various fields of electrical engineering and computer science and specializes in the development of computer hardware and software. Computer engineers can be involved in the many hardware and software facets of computing, like the design of microprocessors and personal computers to circuit design. It can be said that this field of study focuses on both how computer systems work as well as their integration in the larger picture. Both electrical and computer engineering are disciplines that can be counted as the biggest and fastest growing in the range of engineering fields that are there in the industry.

This book is an attempt to collate all current data and research on computer and electrical engineering. I am thankful to all the contributing authors for the hard work and effort put in these researches. I also wish to acknowledge the efforts of the publishing team who provided excellent technical assistance, whenever needed. Lastly, I wish to thank my friends and family who have supported me at every step in my life.

Editor

Contents

Computation of Channel Capacity Based on Self-Concordant Functions

Da-gang Tian[1] and Yi-qun Huang[2]

[1] *Search Center for Complex System Science, University of Shanghai for Science and Technology, Shanghai 200093, China*
[2] *Shanghai Stock Communication Co., Ltd., Shanghai 200131, China*

Correspondence should be addressed to Da-gang Tian, tiandagang_2@163.com

The computation of channel capacity is a classical issue in information theory. We prove that algorithms based on self-concordant functions can be used to deal with such issues, especially when constrains are included. A new algorithm to compute the channel capacity per unit cost is proposed. The same view is suited to the computation of maximum entropy. All the algorithms are of polynomial time.

1. Introduction

The computation of channel capacity is a classical issue in information theory. Much attention has been paid to it [1–3]. Despite some progress [3], the Arimoto-Blahut algorithm [4–7] is still the most used method. This method has been developed into a general algorithm, alternative minimization method [8], and has found applications in various fields [9–11]. Recently, the computation of channel capacity with constraints, such as the computation of capacity-cost function, is becoming increasingly important [5, 12–15]. The Arimoto-Blahut algorithm is very limited in dealing with such issues. Due to lack of appropriate means to eliminate the Lagrange multipliers, the method in [5], which is based on Arimoto-Blahut algorithm, cannot obtain the optimal point.

The computation of channel capacity has always been an optimization problem, but no one optimization algorithm, such as the gradient method and Newton's method, and so forth, has become the mainstream method. One reason is that so far no one optimization algorithm can be as effective as the Arimoto-Blahut algorithm. However, with advances in optimization theory and with the increasing importance of constrained channel capacity issues, the situation is changing.

In this paper, some new algorithms based on self-concordant function are proposed. Besides the effectiveness—it is of polynomial time—the algorithm can compute constrained channel capacity easily. In Section 2, we introduced the definition of the self-concordant function and the main properties, as well as some algorithms for convex optimization problems. In Section 3, we give some new algorithms for channel capacity with or without constraints and for channel capacity per unit cost. Because it is based on the self-concordant function, these algorithms are of polynomial time. Some conclusions are given in Section 4.

2. Self-Concordant Function and Convex Programming

For convenience, in this section, we make a brief introduction to self-concordant functions, including the definition, main propositions, and algorithms for convex optimization.

2.1. Self-Concordant Function. Self-concordant function theory, published by Nesterov and Nemirovskii in 1994 [16, 17], is of landmark importance for establishing general interior point polynomial-time algorithms for convex programming. The theory reveals the main property of the interior point

method for linear programming initiated by Karmarkar in 1984 [18]: self-concordance of the barrier function.

Suppose that $f : D \to R$ is three times continuously differentiable and strictly convex, where the domain D is an open convex subset in R^n.

Definition 1 (see [19]). Let $n = 1$. A convex function $f : R \to R$ is self-concordant if

$$\left| f'''(x) \right| \leq 2 f''(x)^{3/2} \tag{1}$$

for all $x \in D$.

Definition 2. A function $f : R^n \to R$ is self-concordant if it is self-concordant along every line in its domain, that is, if the function $g(t) = f(x + tv)$ is a self-concordant function of t for all $x \in D$ and for all $v \in R^n$.

Since linear and convex quadratic functions have zero third derivatives, they are trivial self-concordant functions. The most common nontrivial self-concordant function is the logarithmic function $-\log x$.

We can get more self-concordant functions due to the following simple and useful combination rules [19].

Proposition 3. *If f_1, f_2 are self-concordant on D and $a_1 \geq 1$, $a_2 \geq 1$, then $a_1 f_1 + a_2 f_2$ is self-concordant on D.*

Proposition 4. *If f is self-concordant on $D \subset R^n$, and $A : R^m \to R^n$ is a linear map, $b \in R^n$, then $f(Ax + b)$ is self-concordant on $\{x \in R^m \mid Ax + b \in D\}$.*

Proposition 5. *If f_i is self-concordant on $D_i \subset R^{ni}$, $i = 1, 2, \ldots, m$. Let $D = D_1 \times \cdots \times D_m$. Then, $f(x_1, \ldots, x_m) = f_1(x_1) + \cdots + f_m(x_m)$ is self-concordant on D.*

2.2. Newton's Method for Self-Concordant Functions. Consider the convex optimization problem

$$\min_D f(x), \tag{2}$$

where $D \subset R^n$ is a bounded, closed, convex subset with a nonempty interior. Let $f(x)$ be a convex function.

Damped Newton Algorithm. Given an initial point $x_0 \in \text{int}(D)$, with tolerance $\varepsilon > 0$.

Repeat

  (1) Compute the Newton step

$$\Delta x_{nt} = -\nabla^2 f(x)^{-1} \nabla f(x), \tag{3}$$

    and Newton decrement

$$\lambda(f, x) = \left(\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x) \right)^{1/2}. \tag{4}$$

  (2) Stopping criterion. **Quit** if $\lambda(f, x)^2 / 2 \leq \varepsilon$.
  (3) If $\lambda(f, x) > 1/4$, then

$$x = x + \frac{1}{1 + \lambda(f, x)} \Delta x_{nt}, \tag{5}$$

else

$$x = x + \Delta x_{nt}. \tag{6}$$

If $f(x)$ is a self-concordant function on D, then the damped Newton algorithm above possesses the following good properties:

  (i) all $x \in \text{int}(D)$ if $x_0 \in \text{int}(D)$.
  (ii) if $\lambda(f, x) > 0.25$, then $f(x_i) - f(x_{i+1}) \geq 0.026$.
  (iii) if at some iteration i we have $\lambda(f, x_i) \leq 0.25$, then we are in the region of quadratic convergence of the method, that is, for every $j \geq i$,

$$\lambda\left(f, x_{j+1}\right) \leq 2\lambda^2\left(f, x_j\right) \leq \left(\frac{1}{2}\right)\lambda\left(f, x_j\right),$$

$$f\left(x_j\right) - p^* \leq \frac{\lambda^2\left(f, x_j\right)}{2\left(1 - \lambda\left(f, x_j\right)\right)}. \tag{7}$$

  (iv) the number of Newton steps required to find a point $x \in \text{int}(D)$ with $\lambda(f, x_i) \leq \varepsilon < 1$ is bounded by

$$C\left(\left(f\left(x^{(0)}\right) - p^*\right) + \log_2 \log_2\left(\frac{1}{\varepsilon}\right)\right) \tag{8}$$

for some constant C.

For the convex optimization problem with equality constraints $Ax = b$, Newton's method and the bound (8) are still valid as long as the initial point $x_0 \in \text{int}(D)$ satisfies $Ax_0 = b$, and the Newton step Δx_{nt} and decrement $\lambda(f, x)$ are computed as follows:

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{nt} \\ w \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix},$$

$$\lambda(f, x) = \left(\Delta x_{nt}{}^T \nabla^2 f(x) \Delta x_{nt} \right)^{1/2}. \tag{9}$$

For general convex optimization problems are

$$\begin{aligned} \min \quad & f_0(x), \\ \text{s.t} \quad & f_i(x) \leq 0, \quad i = 1, \ldots, m, \\ & Ax = b, \end{aligned} \tag{10}$$

where $f_0, \ldots, f_m : R^n \to R$ are convex and twice continuously differentiable, and A is a $p \times n$ matrix with rank $(A) = p < n$. By means of the logarithmic barrier function, (10) can be formulated approximately as an equality constrained problem as follows:

$$\begin{aligned} \min \quad & t f_0(x) + \phi(x), \\ \text{s.t} \quad & Ax = b, \end{aligned} \tag{11}$$

where

$$\phi(x) = -\sum_{i=1}^{m} \log(-f_i(x)) \tag{12}$$

is called the logarithmic barrier function for (10). Let $x^*(t)$ be the optimal point of (11) and let p^* be the optimal value of (10), then $f_0(x^*(t)) - p^* \leq m/t$. Therefore, we can simply take $t = m/\varepsilon$ and solve the equality constrained problem (11) using Newton's method to get an ε-solution of (10), and if the objective function of (11) is self-concordant, then the number of Newton's iterate is bounded by (8). However, this method usually requires a large t, which may bring about a large $f(x^{(0)}) - p^*$ and a large number of Newton's iterate, thus it is rarely used. A commonly used method is the path-following method as follows.

Path-Following Method. Given strictly feasible $x, t = t_0 > 0$, $\mu > 1$, with tolerance $\varepsilon > 0$.

Repeat

(1) Centering step.

Compute $x^*(t)$ by minimizing $tf_0(t) + \varphi$, subject to $Ax = b$, starting at x.

(2) Update. $x = x^*(t)$.

(3) Stopping criterion. **Quit** if $m/t < \varepsilon$.

(4) Increase t. $t = \mu t$.

If for any $t > 0$, $f(x) = tf_0(x) + \varphi(x)$ is self-concordant, then the total number of Newton steps in the path-following method, not counting the initial centering step, is

$$N = \left\lceil \frac{\log(m/(t_0 \varepsilon))}{\log \mu} \right\rceil \times \left(m(\mu - 1 - \log \mu) + \log_2 \log_2 \left(\frac{1}{\varepsilon} \right) \right). \tag{13}$$

The term in brackets is the iterate number of t from t_0 to m/ε, and the term in parentheses is the iterate number of Newton's method per centering step.

3. Algorithms Based on Self-Concordant Functions for Channel Capacity

In this section, it is proved that the channel capacity, with or without constrains, can be computed by the path-following method in polynomial time. Meanwhile, we prove that channel capacity per unit cost is a single-peak function of the expected cost.

3.1. Channel Capacity without Constrains. Let $X = \{x_1, \ldots, x_m\}$ and $Y = \{y_1, \ldots, y_n\}$ be the input and output alphabets, respectively. Let $p(x) = (p(x_1), \ldots, p(x_m))$ and $q(y) = (q(y_1), \ldots, q(y_n))$ be the distributions of X and Y, respectively. Let $T = (p(y_j \mid x_i))_{m \times n} = (p_{ij})_{m \times n}$ be a transition matrix from X to Y. Hence,

$$q(y_j) = p(x_1)p_{1j} + \cdots + p(x_m)p_{mj}, \quad j = 1, \ldots, n. \tag{14}$$

If $I(X; Y)$ is the mutual information between X and Y, then

$$I(X; Y) = \sum_{i,j} p(x_i)p(y_j \mid x_i) \log \frac{p(y_j \mid x_i)}{q(y_j)} \tag{15}$$

$$= \sum_{i,j} p(x_i)p(y_j \mid x_i) \log \frac{p(x_i \mid y_j)}{p(x_i)}. \tag{16}$$

Channel capacity is defined as follows:

$$C = \max_{p(x)} I(X; Y). \tag{17}$$

Equation (16) is a function of $p(x)$ and $p(x|y)$ and we denote it by $I(p(x); p(x|y))$.

The Arimoto-Blahut algorithm utilized (16) as follows. In (16), both $p(x)$ and $p(x|y)$ are unknown, but given $p(x)$, (16) can get its maximum at

$$p(x_i \mid y_j) = \frac{p(x_i)p(y_j \mid x_i)}{\sum_{k=1}^{m} p(x_k)p(y_j \mid x_k)}, \tag{18}$$

$$i = 1, \ldots, m. \ j = 1, \ldots, n,$$

and given $p(x|y)$, (16) can get its maximum at

$$p(x_i) = \frac{\prod_j p(x_i \mid y_j)^{p(y_j|x_i)}}{\sum_{i=1}^{m} \prod_j p(x_i \mid y_j)^{p(y_j|x_i)}}, \quad i = 1, \ldots, m. \tag{19}$$

If $p^0(x) = (1/m, \ldots, 1/m)$, then the channel capacity can be approximated by computing $p^0(x|y)$, $p^1(x)$, $p^1(x|y)$, ... through (18) and (19), alternatively. Let $C(N + 1, N)$ be the mutual information $I(p^{N+1}(x); p^N(x|y))$, then,

$$C - C(N + 1, N) \leq \frac{\log m}{N}. \tag{20}$$

In order to get an algorithm based on the self-concordant function, we utilize (15). Let

$$c_i = -\sum_{j=1}^{n} p(y_j \mid x_i) \log p(y_j \mid x_i). \tag{21}$$

By (15), the channel capacity can be expressed as follows:

$$\min \quad \sum_{i=1}^{m} c_i p_i + \sum_{j=1}^{n} q_j \log q_j,$$

$$\text{s.t} \quad q_j = \sum_{i=1}^{m} p_i p(y_j \mid x_i), \quad j = 1, \ldots, n, \tag{22}$$

$$p_1 + \cdots + p_m = 1,$$

$$p_1 > 0, \ldots, p_m > 0.$$

It is a standard convex optimization problem. Unfortunately, the objective function of (22) is not self-concordant, even when the logarithmic barrier $-\log p_i$ is added.

It seems that we should add $q_j > 0$ as constraints for (22) since there are $\log q_j$ in the objective function of (22). But the q_j are not independent variables. It can be computed by (14) and maintain positive as long as all the $p_i > 0$, $i = 1, 2, \ldots, m$, hold. Nevertheless, we still add the logarithmic barrier $-\log q_j$ in the objective of (22) to get a self-concordant objective function.

For $t > 0$, consider the convex optimization problem as follows:

$$
\min \quad t\sum_{i=1}^{m} c_i p_i + t\sum_{j=1}^{n} q_j \log q_j - \sum_{i=1}^{m} \log p_i - \sum_{j=1}^{n} \log q_j,
$$

$$
\text{s.t} \quad q_j = \sum_{i=1}^{m} p_i p(y_j \mid x_i), \quad j = 1, \ldots, n, \tag{23}
$$

$$
p_1 + \cdots + p_m = 1.
$$

In order to show the self-concordance of the objective function of (23), we need Proposition 6.

Proposition 6. *For any $t > 0$, $f(x) = tx\log x - \log x$ is a self-concordant function on $x > 0$.*

Proof. We have

$$
f'(x) = t\log x + t - \frac{1}{x},
$$

$$
f''(x) = \frac{t}{x} + \frac{1}{x^2}, \qquad |f'''(x)| = \frac{t}{x^2} + \frac{2}{x^3}. \tag{24}
$$

Thus

$$
\frac{|f'''(x)|}{f''(x)^{3/2}} = \frac{(tx+2)}{(tx+1)^{3/2}} \leq 2. \tag{25}
$$

□

By Propositions 3–6, the objective function of (23) is self-concordant, so we can solve it by the path following method, and the number of Newton iterations is bounded by (13).

Equation (13) shows that solving (23) by the path following method is a polynomial time algorithm. In addition, a main advantage is that the algorithm can deal with constrains very easily.

3.2. Channel Capacity with Constrains and Channel Capacity Per Unit Cost. Consider the channel capacity with constraints as follows:

$$
\max \quad I(X; Y),
$$

$$
\text{s.t} \quad \sum_{j=1}^{m} a_{ij} p_j \leq E_i, \quad i = 1, \ldots, r, \tag{26}
$$

$$
p_1 + \cdots + p_m = 1,
$$

$$
p_1 \geq 0, \ldots, p_m \geq 0.
$$

For $r = 2$, [5] gave some algorithms based on the Arimoto-Blahut method. Because one cannot eliminate the Lagrange multiplier, the algorithms in [5] cannot get the optimal solution for (26).

By adding logarithmic barriers for the inequality constraints in the objective of (26) as follows, we can get a polynomial time algorithm for (26):

$$
\min \quad t\sum_{i=1}^{m} c_i p_i + t\sum_{j=1}^{n}\sum_{i=1}^{m} p_i p(y_j \mid x_i) \log \sum_{i=1}^{m} p_i p(y_j \mid x_i)
$$

$$
- \sum_{i=1}^{m} \log p_i - \sum_{j=1}^{n} \log \sum_{i=1}^{m} p_i p(y_j \mid x_i)
$$

$$
- \sum_{k=1}^{r} \log\left(E_k - \sum_{i=1}^{m} p_i a_{ri}\right),
$$

$$
\text{s.t} \quad p_1 + \cdots + p_m = 1. \tag{27}
$$

The objective function of (27) is self-concordant, so we can solve (27) by the path-following method.

Channel capacity per unit cost is one of the typical problems of channel capacity with constraints [15, 20–22]. By [15], channel capacity per unit cost can be computed by

$$
C = \sup_{\beta > 0} \frac{C(\beta)}{\beta}, \tag{28}
$$

where $C(\beta)$ is the capacity-cost function, which is the solution of the following constrained problem [12]:

$$
\max \quad I(X; Y),
$$

$$
\text{s.t} \quad \sum_{j=1}^{m} a_j p_j \leq \beta, \tag{29}
$$

$$
p_1 + \cdots + p_m = 1,
$$

$$
p_1 \geq 0, \ldots, p_m \geq 0.
$$

Equation (29) is a special case of (26) ($r = 1$). Here we discuss the channel capacity per unit cost in a more general manner.

Theorem 7. *Let $D \subseteq R^n$ be a convex set. Let $F(x)$ be a concave function on D and $G(x)$ a convex function on D. Let $a = \inf\{G(x) | x \in D\}$, $b = \sup\{G(x) | x \in D\}$, and suppose $a \geq 0$. Let $C(\beta) = \sup\{F(x) | G(x) \leq \beta\}$. Then,*

(1) *$C(\beta)$ is a concave increasing function,*

(2) *let*

$$
\rho(\beta) = \frac{C(\beta)}{\beta}, \tag{30}
$$

then $\rho(\beta)$ is either a unimodal function or a monotone function.

$$
\sup_{D} \frac{F(x)}{G(x)} = \sup_{a < \beta < b} \frac{C(\beta)}{\beta}. \tag{31}
$$

Proof. (1) Let $\beta_1 > \beta_2$, then $\{x|G(x) \leq \beta_2\} \subseteq \{x|G(x) \leq \beta_1\}$. Therefore, $C(\beta_2) \leq C(\beta_1)$, For any $\varepsilon > 0$, there are x_1 such that $G(x_1) \leq \beta_1$ and $C(\beta_1) = F(x_1) + \varepsilon$, and there are x_2 such that $G(x_2) \leq \beta_2$ and $C(\beta_2) = F(x_2) + \varepsilon$. Since $G(x)$ is convex and $F(x)$ is concave, we get

$$G(\alpha x_1 + (1-\alpha)x_2) \leq \alpha G(x_1) + (1-\alpha)G(x_2) \leq \alpha\beta_1$$
$$+ (1-\alpha)\beta_2,$$

$$\alpha C(\beta_1) + (1-\alpha)C(\beta_2) = \alpha F(x_1) + (1-\alpha)F(x_2) + \varepsilon$$
$$\leq F(\alpha x_1 + (1-\alpha)x_2) + \varepsilon$$
$$\leq C(\alpha\beta_1 + (1-\alpha)\beta_2) + \varepsilon. \quad (32)$$

Since ε is arbitrary, $C(\beta)$ is concave.

(2) Without loss of generality, we can assume $C(\beta)$ is differentiable. If not, we can approach it by a differentiable function to any precision [23]. Suppose there is a $\beta_0 \in (a, b)$, such that

$$\rho'(\beta)\big|_{\beta=\beta_0} = 0, \text{ that is } C'(\beta_0)\beta_0 - C(\beta_0) = 0. \quad (33)$$

Let $\beta_1 > \beta_0$; since $C(\beta)$ is concave, we have $C'(\beta_1) \leq C'(\beta_0)$. Let

$$y_1(\beta) = C'(\beta_1)(\beta - \beta_0) + C(\beta_0). \quad (34)$$

It is the equation of a straight line that has slope $C'(\beta_1)$ and passes through the $(\beta_0, C(\beta_0))$. Since $C(\beta)$ is increasing and concave, we get $y_1(\beta_1) \leq C(\beta_1)$ and

$$C'(\beta_1) - \frac{C(\beta_1)}{\beta_1} \leq C'(\beta_1) - \frac{y_1(\beta_1)}{\beta_1}$$
$$= C'(\beta_1)\frac{\beta_0}{\beta_1} - \frac{C(\beta_0)}{\beta_1} \leq C'(\beta_0)\frac{\beta_0}{\beta_1} - \frac{C(\beta_0)}{\beta_1}$$
$$= 0, \quad (35)$$

that is, $\rho'(\beta_1) \leq 0$.

Let $\beta_2 < \beta_0$, since $C(\beta)$ is concave, $C'(\beta_2) \geq C'(\beta_0)$. Let

$$y_0(\beta) = C'(\beta_0)(\beta - \beta_0) + C(\beta_0). \quad (36)$$

It is the equation of a straight line that has slope $C'(\beta_0)$ and passes through $(\beta_0, C(\beta_0))$. Since $C(\beta)$ is increasing and concave, we get $y_0(\beta_2) \geq C(\beta_2)$ and

$$C'(\beta_2) - \frac{C(\beta_2)}{\beta_2} \geq C'(\beta_2) - \frac{y_0(\beta_2)}{\beta_2} \geq C'(\beta_0) - \frac{y_0(\beta_2)}{\beta_2}$$
$$= C'(\beta_0)\frac{\beta_0}{\beta_2} - \frac{C(\beta_0)}{\beta_2} = 0, \quad (37)$$

that is, $\rho'(\beta_2) \geq 0$.

Therefore, β_0 is a maximum point of function $\rho(\beta) = C(\beta)/\beta$.

(3) If

$$\sup_{D}\frac{F(x)}{G(x)} = +\infty, \quad (38)$$

then for any $N > 0$, there are x_0, such that

$$\frac{F(x_0)}{G(x_0)} > N. \quad (39)$$

Let $G(x_0) = \beta_0$, then $C(\beta_0) = \sup\{F(x)|G(x) \leq \beta_0\} \geq F(x_0)$, so

$$\sup_{a<\beta<b}\frac{C(\beta)}{\beta} \geq \frac{C(\beta_0)}{\beta_0} \geq \frac{F(x_0)}{G(x_0)} > N. \quad (40)$$

Since N is arbitrary, we get

$$\sup_{a<\beta<b}\frac{C(\beta)}{\beta} = +\infty. \quad (41)$$

In turn, if

$$\sup_{a<\beta<b}\frac{C(\beta)}{\beta} = +\infty, \quad (42)$$

then for any $N > 0$, there are β_0, $a < \beta_0 < b$, such that

$$\frac{C(\beta_0)}{\beta_0} > N. \quad (43)$$

Since

$$\sup_{D}\frac{F(x)}{G(x)} \geq \sup_{G(x)\leq\beta_0}\frac{F(x)}{G(x)} \geq \sup_{G(x)\leq\beta_0}\frac{F(x)}{\beta_0} = \frac{C(\beta_0)}{\beta_0} > N \quad (44)$$

so

$$\sup_{D}\frac{F(x)}{G(x)} = +\infty. \quad (45)$$

Therefore, if

$$\sup_{D}\frac{F(x)}{G(x)} < +\infty \quad (46)$$

is valid, then so is

$$\sup_{a<\beta<b}\frac{C(\beta)}{\beta} < +\infty \quad (47)$$

and vice versa.

Suppose (46) is valid, for any $\varepsilon > 0$, there are $x_0 \in D$, such that

$$\frac{F(x_0)}{G(x_0)} + \varepsilon = \sup_{D}\frac{F(x)}{G(x)}. \quad (48)$$

Let $G(x_0) = \beta_0$, then

$$\sup_{D}\frac{F(x)}{G(x)} = \frac{F(x_0)}{G(x_0)} + \varepsilon \leq \frac{C(\beta_0)}{\beta_0} + \varepsilon \leq \sup_{a<\beta<b}\frac{C(\beta)}{\beta} + \varepsilon. \quad (49)$$

Since ε is arbitrary, we get

$$\sup_{D} \frac{F(x)}{G(x)} \leq \sup_{a<\beta<b} \frac{C(\beta)}{\beta}. \tag{50}$$

On the other hand, since (47) is valid too, for any $\varepsilon > 0$, there are $a < \beta_0 < b$, such that

$$\frac{C(\beta_0)}{\beta_0} + \varepsilon = \sup_{a<\beta<b} \frac{C(\beta)}{\beta}. \tag{51}$$

Let $x_0 \in D$, such that $G(x_0) \leq \beta_0$ and $F(x_0) + \varepsilon\beta_0 = \sup\{F(x)|G(x) \leq \beta_0\} = C(\beta_0)$, then

$$\sup_{D} \frac{F(x)}{G(x)} \geq \sup_{G(x) \leq \beta_0} \frac{F(x)}{G(x)} \geq \frac{F(x_0)}{\beta_0} = \sup_{a<\beta<b} \frac{C(\beta)}{\beta} - 2\varepsilon. \tag{52}$$

Since ε is arbitrary, we get

$$\sup_{D} \frac{F(x)}{G(x)} \geq \sup_{a<\beta<b} \frac{C(\beta)}{\beta}. \tag{53}$$

$\square$

If $F(x) = I(X, Y)$ is the average mutual information defined by (15) and $G(x) = E[b(X)]$, then (3) of Theorem 7 is Theorem 2 in [15].

In addition to average mutual information, entropy is another frequently used function in applications [24]. It is obvious that Theorem 7 is valid when $F(x)$ is an entropy function (as a function of the distribution of input symbols). It is useful to point out that Theorem 7 is still valid when $G(x)$ is a positive definite quadratic function [25].

Making use of (2) of Theorem 7, we can write an algorithm to compute the $\sup(F(x)/G(x))$ as follows. The algorithm is based on the 0.618 method used to locate the optimal point of a single peak function.

Algorithm 8 (0.618 method). Let $a = \inf\{G(x)|x \in D\}$, $b = \sup\{G(x)|x \in D\}$, and assume $a \geq 0$.

(1) Given search interval $[a, b]$, and tolerance $\varepsilon > 0$. compute λ and μ as follows:

$\lambda = a + 0.382(b - a)$, $\mu = a + 0.618(b - a)$; evaluate $C(\lambda)/\lambda$ and $C(\mu)/\mu$. Let $k = 1$.

(2) If $C(\lambda)/\lambda > C(\mu)/\mu$, then go to 3, else go to 4.

(3) If $b - \lambda \leq \varepsilon$, then output μ and $C(\mu)/\mu$, else, let $a = \lambda$, $\lambda = \mu$, $C(\lambda)/\lambda = C(\mu)/\mu$, $\mu = a + 0.618(b - a)$; evaluate $C(\mu)/\mu$, go to 5.

(4) If $\mu - a \leq \varepsilon$, then output λ and $C(\lambda)/\lambda$, else, let $b = \mu$, $\mu = \lambda$, $C(\mu)/\mu = C(\lambda)/\lambda$, $\lambda = a + 0.382(b - a)$; evaluate $C(\lambda)/\lambda$, go to 5.

(5) $k = k + 1$, go to 2.

Even if $a = 0$, that is when there are zero cost input symbols, the algorithm is still valid.

3.3. Maximum Entropy. A similar but simpler problem is the computation of maximum entropy. The typical form is as follows:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} p_i \log \frac{p_i}{\pi_i}, \\ \text{s.t} \quad & \sum_{j=1}^{m} a_{ij} p_j = E_i, \quad i = 1, \ldots, r, \\ & p_1 + \cdots + p_m = 1, \\ & p_1 \geq 0, \ldots, p_m \geq 0, \end{aligned} \tag{54}$$

where π_i are known, and E_i can be computed by sample data, $i = 1, \ldots, r$. The GIS algorithm [26–28] is one of the most common algorithms for (54), which is only suited to linear constraints. In spite of that there is a good discussion on the computation in [27]; however, we cannot ensure that GIS is a polynomial time algorithm.

In [25] a maximum entropy problem with nonlinear constraints is considered. It is as follows:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} p_i \log \frac{p_i}{\pi_i}, \\ \text{s.t} \quad & \sum_{i=1}^{m} p_i a_i - \sum_{i,j} \sigma_{ij} p_i p_j \geq b, \\ & p_1 + \cdots + p_m = 1, \\ & p_1 \geq 0, \ldots, p_m \geq 0, \end{aligned} \tag{55}$$

where $(\sigma_{ij})_{m \times m}$ is a positive definite matrix.

We can solve (54) and (55) in polynomial time. In fact, for (54), we add logarithmic barriers $-\log p_i$ in its objective function, for (55), we add logarithmic barrier $-\log p_i$ and

$$-\log\left(\sum_{i=1}^{m} p_i a_i - \sum_{i,j} \sigma_{ij} p_i p_j - b \right) \tag{56}$$

in its objective function, and the problems can be transformed into (3). It is good to know that the function in (56) is self-concordant. Therefore, the path following method is valid.

4. An Example

Let transmission matrix P be as follows:

$$P = \begin{pmatrix} 0.9 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.02 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.01 & 0.02 & 0.01 & 0.9 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 \\ 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0 & 0.1 & 0.1 & 0 & 0.1 & 0.1 & 0 & 0.1 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.4 & 0.1 & 0.1 & 0 & 0.4 \\ 0.1 & 0.2 & 0.3 & 0.4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.2 & 0 & 0.2 & 0 & 0.2 & 0.2 & 0 & 0.2 \\ 0 & 0.1 & 0.5 & 0 & 0.2 & 0 & 0.1 & 0.1 & 0 & 0 \end{pmatrix}. \tag{57}$$

Let $\varepsilon = 0.00001$. Suppose that the initial distribution is discrete uniform, with penalty factor $\mu = 2$ and initial penalty parameter $t_0 = 10$. Iterating 70 times by the path-following method, we get channel capacity $C = 1.370656$ and the optimal distribution: (0.0893518, 0.2414400, 0.1743530, 0.1548483, 0.0000003, 0.0811754, 0.1946234, 0.0000004, 0.0642042, 0.0000080).

With the Arimoto-Blahut algorithm, iterate 180 times, we get the same results.

Let $b = (1, 2, 5, 7, 6, 8, 3, 9, 3, 4)^T$ be cost vector of the input symbols. Let $G(p) = b^T \cdot p$ be the expect cost of the input symbols. The curve of $C(\beta)/\beta$ is shown in Figure 1; By Algorithm 8, the channel capacity per unit cost is 0.5801, and it is attained at $\beta = 1.8325$. The search interval is $[1, 5.8]$. The number of iterations of the path-following is 17. The number of iterations of the 0.618 method is 19. The number of iterations of Newton's algorithm is 5.

Let $G(x) = b'x + x'Dx$ be a positive definite quadratic function, where

$$D = \begin{pmatrix} 4.2793 & 3.6390 & 3.3299 & 3.1505 & 2.5939 & 3.2977 & 3.1777 & 3.0131 & 3.2207 & 4.7107 \\ 3.6390 & 4.4536 & 3.6885 & 3.6644 & 2.2566 & 3.7725 & 3.3912 & 3.1615 & 2.1636 & 3.7311 \\ 3.3299 & 3.6885 & 4.8436 & 2.8852 & 2.9607 & 3.9994 & 1.9509 & 2.6599 & 2.7674 & 4.1990 \\ 3.1505 & 3.6644 & 2.8852 & 4.1004 & 2.2773 & 2.8920 & 3.8454 & 2.6106 & 2.6786 & 3.3013 \\ 2.5939 & 2.2566 & 2.9607 & 2.2773 & 3.6362 & 2.2764 & 2.0494 & 1.3858 & 2.4984 & 3.0855 \\ 3.2977 & 3.7725 & 3.9994 & 2.8920 & 2.2764 & 5.4158 & 2.9241 & 3.2113 & 3.0580 & 3.0860 \\ 3.1777 & 3.3912 & 1.9509 & 3.8454 & 2.0494 & 2.9241 & 6.0292 & 3.8679 & 3.2317 & 3.4582 \\ 3.0131 & 3.1615 & 2.6599 & 2.6106 & 1.3858 & 3.2113 & 3.8679 & 4.9280 & 3.4204 & 3.5231 \\ 3.2207 & 2.1636 & 2.7674 & 2.6786 & 2.4984 & 3.0580 & 3.2317 & 3.4204 & 5.1856 & 3.6543 \\ 4.7107 & 3.7311 & 4.1990 & 3.3013 & 3.0855 & 3.0860 & 3.4582 & 3.5231 & 3.6543 & 6.5953 \end{pmatrix}. \tag{58}$$

The curve of $C(\beta)/\beta$ is shown in Figure 2. The maximum is 0.2156, and it is attained at $\beta = 5.9093$. The number of iterations of the path-following is 17. The number of iterations of the 0.618 method is 19. The number of iterations of Newton's algorithm is 4.

5. Conclusion

By means of self-concordant function theory, the computation of channel capacity, especially, when there are constraints and when constraints are nonlinear, becomes
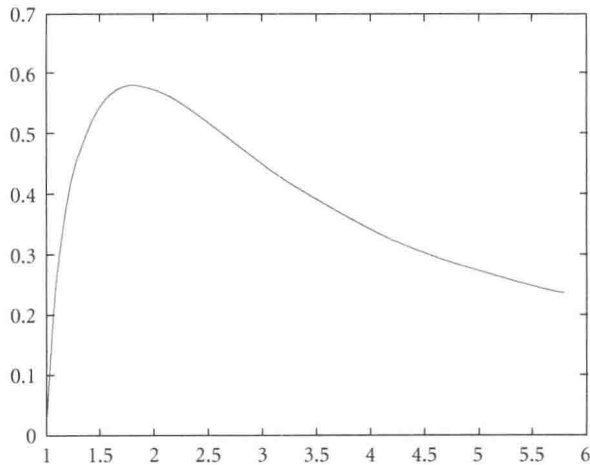
FIGURE 1: Curve of $C(\beta)/\beta$, where $G(x)$ is the expected cost of the input symbol.
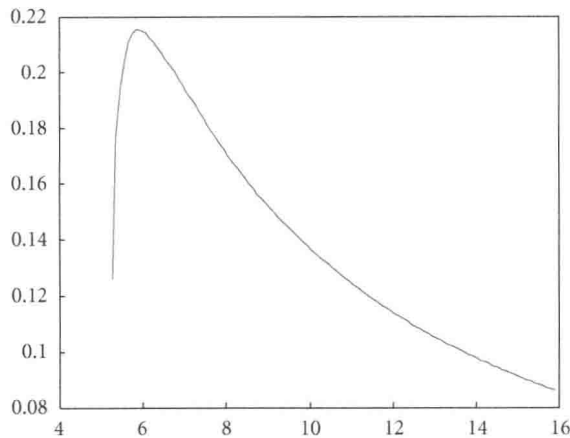

FIGURE 2: Curve of $C(\beta)/\beta$, where $G(x)$ is a positive definite quadratic function.

very simple. When the numerator is a general concave function and the denominator is a general convex function, the formula about channel capacity per unit cost is still valid. Furthermore, the function $C(\beta)/\beta$ is single peak, hence we can get some new algorithms for channel capacity per unit cost.

References

[1] M. Chiang and S. Boyd, "Geometric programming duals of channel capacity and rate distortion," *IEEE Transactions on Information Theory*, vol. 50, no. 2, pp. 245–258, 2004.

[2] A. Ben-Tal and M. Teboulle, "Extension of some results for channel capacity using a generalized information measure," *Applied Mathematics and Optimization*, vol. 17, no. 2, pp. 121–132, 1988.

[3] X. B. Liang, "An algebraic, analytic, and algorithmic investigation on the capacity and capacity-achieving input probability distributions of finite-input-finite-output discrete memoryless channels," *IEEE Transactions on Information Theory*, vol. 54, no. 3, pp. 1003–1023, 2008.

[4] S. Arimoto, "Algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Transactions on Information Theory*, vol. 18, no. 1, pp. 14–20, 1972.

[5] R. E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp. 460–473, 1972.

[6] F. Dupuis, W. Yu, and F. M. J. Willems, "Blahut-Arimoto algorithms for computing channel capacity and rate-distortion with side information," in *Proceedings of IEEE International Symposium on Information Theory*, p. 179, July 2004.

[7] Y. Yu, "Squeezing the arimotoblahut algorithm for faster convergence," *IEEE Transactions on Information Theory*, vol. 56, no. 7, Article ID 5484972, pp. 3149–3157, 2010.

[8] I. Csiszar and G. Tusnady, "Information geometry and alternating minimization procedures," *Statistics and Decisions*, supplement 1, pp. 205–237, 1984.

[9] U. Niesen, D. Shah, and G. W. Wornell, "Adaptive alternating minimization algorithms," *IEEE Transactions on Information Theory*, vol. 55, no. 3, pp. 1423–1429, 2009.

[10] W. Byrne, "Alternating minimization and Boltzmann machine learning," *IEEE Transactions on Neural Networks*, vol. 3, no. 4, pp. 612–620, 1992.

[11] Y. Wang, J. F. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.

[12] R. W. Yeung, *Information Theory and Network Coding*, Springer, Berlin, Germany, 2008.

[13] A. S. Khayrallah and D. L. Neuhoff, "Coding for channels with cost constraints," *IEEE Transactions on Information Theory*, vol. 42, no. 3, pp. 854–867, 1996.

[14] F. Alajaji and N. Whalen, "The capacity-cost function of discrete additive noise channels with and without feedback," *IEEE Transactions on Information Theory*, vol. 46, no. 3, pp. 1131–1140, 2000.

[15] S. Verdu, "On channel capacity per unit cost," *IEEE Transactions on Information Theory*, vol. 36, no. 5, pp. 1019–1030, 1990.

[16] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Methods in Convex Programming*, Society for Industrial and Applied Mathematics, 1994.

[17] J. Renegar, *A Mathematical View of Interior-Point Methods in Convex Optimization*, Society for Industrial and Applied Mathematics, 2001.

[18] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica*, vol. 4, no. 4, pp. 373–395, 1984.

[19] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

[20] R. Lav, "Varshney, Variations On Channel Capacity per Unit Cost," 2005, http://web.mit.edu/lrv/www/writing/cap_cost.pdf.

[21] M. Mitzenmacher, "A survey of results for deletion channels and related synchronization channels," *Probability Surveys*, vol. 6, pp. 1–33, 2009.

[22] M. Kleiner and B. Rimoldi, "On fidelity per unit cost," in *IEEE International Symposium on Information Theory (ISIT '09)*, pp. 1639–1643, July 2009.

[23] K. A. Kopotun, D. Leviatan, and I. A. Shevchuk, "Convex polynomial approximation in the uniform norm: conclusion," *Canadian Journal of Mathematics*, vol. 57, no. 6, pp. 1224–1248, 2005.

[24] M. Dudík, S. J. Phillips, and R. E. Schapire, "Maximum entropy density estimation with generalized regularization and