A Discrete-Time Recurrent Neural Network for Shortest-Path Routing

Youshen Xia and Jun Wang

***Abstract*—This paper presents a discrete-time recurrent neural network, with a fixed step parameter, for solving the shortest path problem. The proposed discrete-time recurrent neural network with a simple architecture is proven to be globally convergent to exact optimal solutions and is suitable for hardware implementation. Furthermore, an improved network with a larger step size independent of the problem size is proposed to increase its convergence rate. The performance and operating characteristics of the proposed neural network are demonstrated by means of simulation results.**

***Index Terms*—Combinatorial optimization, discrete-time, neural networks, shortest-path routing.**

I. Introduction

The shortest path problem is concerned with finding the shortest path from a specified origin to a specified destination in a given network while minimizing the total cost associated with the path. The shortest path problem is an archetypical combinatorial optimization problem having widespread applications in a variety of settings. The applications of the shortest path problem include vehicle routing in transportation systems [1], traffic routing in communication networks [2], and path planning in robotic systems [3]. Furthermore, the shortest path problem also has numerous variations such as the minimum weight problem, the quickest path problem, the most reliable path problem, and so on.

The shortest path problem has been investigated extensively. The well-known algorithms for solving the shortest path problem include the $O(n^2)$ Bellman's dynamic programming algorithm for directed acycle networks, the $O(n^2)$ Dijkstra-like labeling algorithm, and the $O(n^3)$ Bellman–Ford successive approximation algorithm for network nonnegative cost coefficients only, where n denotes the number of vertices in the network. See [4] for a comprehensive coverage of these algorithms. Besides the classical methods, many new and modified methods have been developed during the past few years. For large-scale and real-time applications such as traffic routing and path planning, the existing series algorithms may not be effective and efficient due to the limitation of sequential processing in computational time. Therefore, parallel solution methods are more desirable.

With the advances in new technologies (especially VLSI technology), the dynamical systems approach to solving optimization problems using artificial neural networks has been greatly attracted due to the massively parallel operations of the computing units and very faster convergence properties. Since Hopfield and Tank's seminal work [5], various optimization continuous-time neural networks have been developed [6]–[10]. Some of these neural networks have been applied to solve the shortest path problem, and their investigations have shown the sufficient potentials for the neural network approach to the shortest path problem [11]–[17]. In many operations, however, discrete-time neural networks are preferable to their continuous-time counterparts because of the availability of design tools and the compatibility with computers and other digital devices. Although a discrete-time neural network model can be obtained from a continuous-time one by converting a differential equation into an appropriate difference equation though discretization, the resulting discrete-time model is usually not guaranteed to be globally convergent since step parameters may not be bounded in a small range and difficulties may arise in selecting very small step parameters.

This paper presents a discrete-time neural network with a fixed step parameter for solving the shortest path routing problem. The proposed neural network can solve the routing network problem with mixed-sign cost coefficients and is guaranteed to converge to an optimal solution globally. Furthermore, the proposed network is realizable in parallel digital circuits.

The remainder of this paper is organized as follows. In Section II, the problem statement and formulation are described. In Section III, the dynamic equation and network architecture of the discrete-time routing network are discussed. In Section IV, the global convergence of the discrete-time routing network is proven. In Section V, the bound of a large step for the discrete-time routing network is specified. In Section VI, operating characteristics of the proposed networks are demonstrated via simulation results.

II. Problem and Formulation

Let $V = \{i | i = 1, \ldots, n\}$ be an arbitrary finite set and let $S = \{(i, j) | i \in V, j \in V\}$ be the set of all ordered pairs of elements of V. Note that, in ordered pairs, (i, j) and (j, i) do not represent the same element unless $i = j$. The pair $G = (V, E)$ is called a directed graph where $E \subset S$. The elements of V are called vertices and the elements of E are called directed edges. G is called a weighted directed graph if a fixed cost c_{ij} is associated with the edge from vertex i to vertex j in G. In general, the cost coefficients matrix $[c_{ij}]$ is not necessarily symmetric; i.e., the cost from vertices i to j may not be equal to the cost from vertices j to i. Furthermore, the edges between some vertices may not exist; i.e., the number of edges may be less than that of the vertices. The values of cost coefficients for the nonexistent edges are defined as infinity. More generally, a cost coefficient can be either positive or negative. A positive cost coefficient represents a loss, whereas a negative one represents a gain.

There are many applications of the direct graph since it can be used to model a wide variety of real-world problems. For example, in a road network, the vertices can represent intersections, the edges can represent streets, and the physical meaning of the cost can be the distance between the vertices. In this paper, the shortest path problem to be discussed is: find the shortest (least costly) possible directed path from a specified origin vertex to a specified destination vertex. The cost of the path is the sum of the cost coefficients on the edges in the path and the shortest path is the minimum cost path.

For convenience, we consider the shortest path from vertex 1 to vertex n in a directed graph with n vertices, n edges, and a cost c_{ij} associated with each edge (i, j) in G. In order to formulate the shortest path problem, there are two typical path representations methods: vertex representation and edge representation. Because of the advantages of the edge representation over the vertex representation [4], the development of this paper is based on the edge path representation. Thus, the shortest path problem can be mathematically formulated as a linear integer program as follows [19]:

$$\text{minimize} \quad \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{1}$$

$$\text{subject to} \quad \sum_{k=1}^{n} x_{ik} - \sum_{l=1}^{n} x_{li} = \begin{cases} 1, & \text{if } i = 1 \\ 0, & \text{if } i = 2, 3, \ldots, n-1 \\ -1, & \text{if } i = n \end{cases}$$
$$x_{ij} \in \{0, 1\}, \qquad i, j = 1, 2, \ldots, n \tag{2}$$

where the objective function to be minimized, (1), is also the total cost for the path. The equality constraint coefficients and the right-hand

Manuscript received March 6, 1998; revised April 6, 1999, November 27, 1999, and April 19, 2000. Recommended by Associate Editor, M. Polycarpou. This work was supported by the Hong Kong Research Grants Council under Grant CUHK381/96E.

The authors are with the Department of Automation and Computer-Aided Engineering, The Chinese University of Hong Kong, Hong Kong, China.

Publisher Item Identifier S 0018-9286(00)09999-2.

sides are -1, 0, or 1. Equation (2) ensures that a continuous path starts from a specified origin and ends at a specified destination. x_{ij} denotes the decision variable associated with the edge from vertices i to j, as defined below

$$x_{ij} = \begin{cases} 1, & \text{if the edge from vertices } i \text{ to } j \text{ is in the path} \\ 0, & \text{otherwise.} \end{cases}$$

Because of the total unimodulity property of the constraint coefficient matrix defined in (2) [19], replacing $x_{ij} = 0$ or 1, we still obtain an integer solution where the value of each variable is zero or one if an optimal solution exists and unique. Thus we may solve the above integer program as the following linear program [19]:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} \\ \text{subject to} \quad & \sum_{k=1}^{n} x_{ik} - \sum_{l=1}^{n} x_{li} = \delta_{i1} - \delta_{in} \\ & x_{ij} \geq 0, \qquad i, j = 1, 2, \ldots, n \end{aligned} \tag{3}$$

where δ_{ij} is the Kronecker delta function defined as $\delta_{ij} = 1$ $(i = j)$ and $\delta_{ij} = 0$ $(i \neq j)$. By the duality of convex program [16], we see that the dual shortest path problem is as follows:

$$\begin{aligned} \text{maximize} \quad & y_1 - y_n \\ \text{subject to} \quad & y_i - y_j \leq c_{ij}, \qquad i, j = 1, 2, \ldots, n \end{aligned} \tag{4}$$

where y_i denotes the dual decision variable associated with vertex i and $y_1 - y_i$ is the shortest distance from vertex 1 to vertex i at optimality.

III. Primal-Dual Routing Network

In this section, we propose a discrete-time neural network for solving shortest path routing and discuss its architecture.

It is well known that in order to design an recurrent neural network for optimization problems, one needs to construct an appropriate computational energy function so that the lowest energy state will correspond to the desired solutions. Based on the dual property of linear program [19], we consider an energy function for the primal-dual problem below

$$\begin{aligned} E[x, y] = {} & \frac{1}{2}\sum_{i=1}^{n}\left[\sum_{k=1}^{n}(x_{ik} - x_{ki}) - \delta_{i1} + \delta_{in}\right]^2 \\ & + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}[(-x_{ij})^+]^2 \\ & + \frac{1}{2}\left[\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} - y_1 + y_n\right]^2 \\ & + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}[(y_i - y_j - c_{ij})^+]^2 \end{aligned} \tag{5}$$

where $(s)^+ = \max\{0, s\}$, $s \in R$. The first term in (5) is for the equality constrains (2), the second term is for the nonnegativity, the third term is the squared duality gap, and the last term is for the inequality constraint in the dual problem.

By the well-known gradient method, we obtain the discrete-time version of the primal-dual shortest path routing network below: for $i, j = 1, 2, \ldots, n$

$$\begin{aligned} x_{ij}^{(k+1)} = x_{ij}^{(k)} - h\Bigg\{ & c_{ij}\left(\sum_{p=1}^{n}\sum_{q=1}^{n} c_{pq}x_{pq}^{(k)} - y_1^{(k)} + y_n^{(k)}\right) \\ & - \left(-x_{ij}^{(k)}\right)^+ + \sum_{p=1}^{n}\left(x_{ip}^{(k)} - x_{pi}^{(k)}\right) + \delta_{in} \\ & - \delta_{1i} - \left(\sum_{p=1}^{n}\left(x_{jp}^{(k)} - x_{pj}^{(k)}\right) - \delta_{jn} + \delta_{1j}\right)\Bigg\} \end{aligned} \tag{6}$$

$$\begin{aligned} y_i^{(k+1)} = y_i^{(k)} - h\Bigg\{ & \left(\sum_{p=1}^{n}\sum_{q=1}^{n} c_{pq}x_{pq}^{(k)} - y_1^{(k)} + y_n^{(k)}\right)(\delta_{in} - \delta_{i1}) \\ & + \sum_{p=1}^{n}\Bigg[\left(y_i^{(k)} - y_p^{(k)} - c_{ip}\right)^+ \\ & - \left(y_p^{(k)} - y_i^{(k)} - c_{pi}\right)^+\Bigg]\Bigg\} \end{aligned} \tag{7}$$

where $h > 0$ is a step parameter to be given. For convenience, let

$$\xi^{(k)} = \left(\sum_{p=1}^{n}\sum_{q=1}^{n} c_{pq}x_{pq}^{(k)} - y_1^{(k)} + y_n^{(k)}\right) \tag{8}$$

$$u_i^{(k)} = \sum_{p=1}^{n}\left(x_{ip}^{(k)} - x_{pi}^{(k)}\right) + \delta_{in} - \delta_{1i} \tag{9}$$

$$v_i^{(k)} = \sum_{p=1}^{n}\left[\left(y_i^{(k)} - y_p^{(k)} - c_{ip}\right)^+ - \left(y_p^{(k)} - y_i^{(k)} - c_{pi}\right)^+\right]. \tag{10}$$

Then (6) and (7) can be rewritten as: for $i, j = 1, 2, \ldots, n$

$$x_{ij}^{(k+1)} = x_{ij}^{(k)} - h\left[\xi^{(k)}c_{ij} - \left(-x_{ij}^{(k)}\right)^+ + u_i^{(k)} - u_j^{(k)}\right] \tag{11}$$

$$y_i^{(k+1)} = y_i^{(k)} - h\left[\xi^{(k)}(\delta_{in} - \delta_{i1}) + v_i^{(k)}\right]. \tag{12}$$

Fig. 1 illustrates the architecture of the discrete-time routing network defined in difference equations (11) and (12) [Fig. 1(a)] and auxiliary equations (8)–(10) [Fig. 1(b)–(d)]. It shows that the discrete-time network is composed of a number of adders, limiters, and time delays only. It is not difficult to see that the proposed discrete-time routing network contains $n^2 + n$ neurons and needs to perform $n(2n - 1) + 3$ multiplications and at most $9n^2 - 3n$ additions/subtractions per iteration. Thus, the proposed model complexity is $O(n^2)$ for multiplication/divisions and additions/subtractions.

IV. Global Convergence

In this section, we will show that the discrete-time network with a constant step parameter is globally convergent to an exact solution to the primal-dual routing problem.

First, denote

$$y = (y_1, \ldots, y_n)^T$$
$$c = (c_{11}, \ldots, c_{1n}, c_{21}, \ldots, c_{2n}, \ldots, c_{n1}, \ldots, c_{nn})^T$$

and

$$x = (x_{11}, \ldots, x_{1n}, x_{21}, \ldots, x_{2n}, \ldots, x_{n1}, \ldots, x_{nn})^T$$

and let A be the $n \times n^2$ constraint matrix whose (i, j) column is $e_i - e_j$, e_i $(i = 1, \ldots, n)$ is a vector in R^n with the ith element being one and others being zero. Then the energy function corresponding to (5) can be rewritten as

$$\begin{aligned} E(x, y) = \frac{1}{2}\Bigg[& \left(c^T x - (e_1 - e_n)^T y\right)^2 + \|(-x)^+\|_2^2 \\ & + \left\|\left(A^T y - c\right)^+\right\|_2^2 + \|Ax + e_n - e_1\|_2^2\Bigg] \end{aligned} \tag{13}$$

where $\|\cdot\|_2$ denotes l_2-norm in R^n and $(-x)^+ = [(-x_1)^+, \ldots, (-x_n)^+]^T$. We then analyze the properties of $E(x, y)$.

Lemma 1: For any $u_0, v_0 \in R$, then

$$[(-u_0)^+]^2 \leq [(-v_0)^+]^2 - 2(-v_0)^+(u_0 - v_0) + (u_0 - v_0)^2. \tag{14}$$

Proof: Consider four cases as follows.

i) For both $v_0 \geq 0$ and $u_0 \geq 0$, $(-u_0)^+ = (-v_0)^+ = 0$ and $(u_0 - v_0)^2 \geq 0$. So (14) holds.

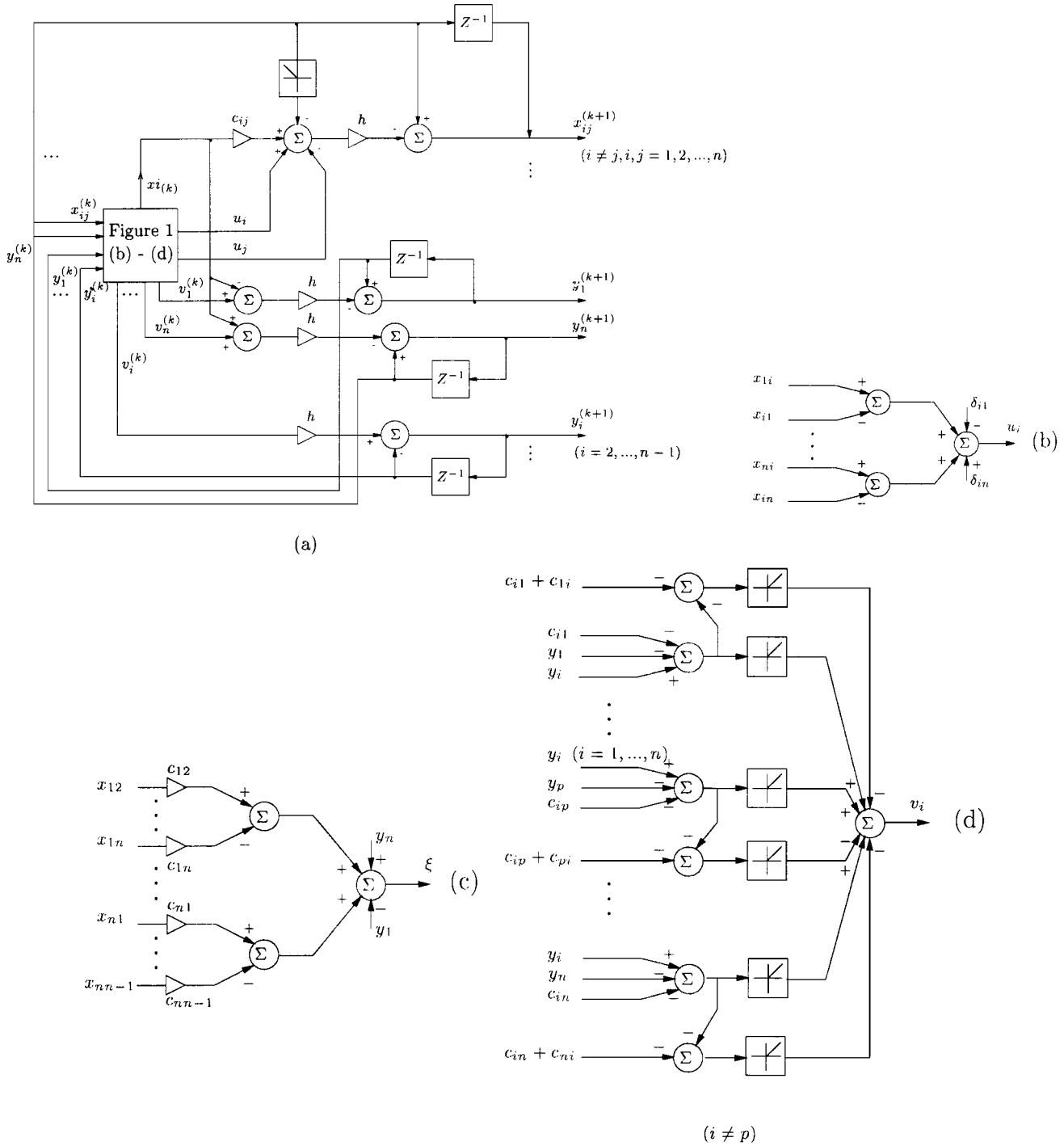

Fig. 1. Architecture of the discrete-time routing network.

ii) For both $v_0 \leq 0$ and $u_0 \leq 0$, $u_0^2 = v_0^2 + 2v_0(u_0 - v_0) + (u_0 - v_0)^2$. So (14) holds equally.

iii) For both $v_0 \leq 0$ and $u_0 \geq 0$, $(-u_0)^+ = 0$ and

$$v_0^2 + 2v_0(u_0 - v_0) + (u_0 - v_0)^2 = u_0^2 \geq 0.$$

So (14) holds.

iv) For both $v_0 \geq 0$, $u_0 \leq 0$, $(-v_0)^+ = 0$ and $(u_0 - v_0)^2 \geq v_0^2$. So (14) holds.

Using Lemma 1, we easily get the following similar result [20].

Lemma 2: Let $\Phi(y) = \|(A^T y - c)^+\|_2^2$, where $A \in R^{n \times n^2}$ and $y \in R^n$. For any $y, z \in R^n$

$$\Phi(z) \leq \Phi(y) + \nabla\Phi(y)^T(z - y) + (z - y)^T AA^T(z - y).$$

Proof: First, for any $z, y \in R^n$, denote $v = -A^T y + c$ and $u = -A^T z + c$. Then $u - v = -A^T(y - x)$, $\Phi(z) = \|(-u)^+\|_2^2$, $\Phi(y) = \|(-v)^+\|_2^2$, and $\nabla\Phi(y) = A(-v)^+$. By Lemma 1, we have

$$\begin{aligned}
&\|(-u)^+\|_2^2 \\
&\quad \leq \sum_{i=1}^{n} \left[((-u_i)^+)^2 - 2u_i(v_i - u_i) + (v_i - u_i)^2\right] \\
&\quad = \|(-v)^+\|_2^2 + \nabla\|(-v)^+\|_2^2(u - v) + (v - u)^T(v - u) \\
&\quad = \Phi(y) + [(-v)^+]^T A^T(y - z) + (z - y)^T AA^T(z - y).
\end{aligned}$$

So

$$\Phi(z) \leq \Phi(y) + \nabla\Phi(y)^T(z - y) + (z - y)^T AA^T(z - y).$$

Lemma 3: Let $E(w) = E(x, y)$ be defined in (13). Then for any $w^{(1)} = (x^{(1)}, y^{(1)})^T, w^{(2)} = (x^{(2)}, y^{(2)})^T \in R^{n^2+n}$

$$E\left(w^{(1)}\right) \le E\left(w^{(2)}\right) + \nabla E\left(w^{(2)}\right)^T \left(w^{(1)} - w^{(2)}\right) + \tfrac{1}{2}\left(w^{(1)} - w^{(2)}\right)^T H \left(w^{(1)} - w^{(2)}\right)$$

where $\nabla \hat{E}(w)$ is the gradient of $\hat{E}(w)$ and

$$H = \begin{bmatrix} A^TA + cc^T + I & -cb^T \\ -bc^T & AA^T + bb^T \end{bmatrix}$$

where I is an $n^2 \times n^2$ matrix, A and c are defined in (13), and $b = e_1 - e_n = (1, 0, \ldots, 0, -1)^T \in R^n$.

Proof: Let $f(w^{(1)}) = \|(-^{(1)})^+\|_2^2 + \|(A^T y^{(1)} - c)^+\|_2^2$ and let $g(w^{(1)}) = [c^T x^{(1)} - (e_1 - e_n)^T y^{(1)}]^2 + \|Ax^{(1)} + e_n - e_1\|_2^2$. Then by Lemma 2, we have

$$f\left(w^{(1)}\right) \le f\left(w^{(2)}\right) + \nabla f\left(w^{(2)}\right)^T \left(w^{(1)} - w^{(2)}\right) + \left(y^{(1)} - y^{(2)}\right)^T \cdot AA^T \left(y^{(1)} - y^{(2)}\right) + \left(x^{(1)} - x^{(2)}\right)^T \left(x^{(1)} - x^{(2)}\right).$$

On the other hand, by using the second-order Taylor formula, we get

$$g\left(w^{(1)}\right) = g\left(w^{(2)}\right) + \nabla g\left(w^{(2)}\right)^T \left(w^{(1)} - w^{(2)}\right) + \tfrac{1}{2}\left(w^{(1)} - w^{(2)}\right)^T H_1 \left(w^{(1)} - w^{(2)}\right)$$

where

$$H_1 = \begin{bmatrix} A^TA + cc^T & -cb^T \\ -bc^T & bb^T \end{bmatrix}.$$

From the above inequalities, it follows:

$$E\left(w^{(1)}\right) \le E\left(w^{(2)}\right) + \nabla E\left(w^{(2)}\right)^T \left(w^{(1)} - w^{(2)}\right) + \tfrac{1}{2}\left(w^{(1)} - w^{(2)}\right)^T H \left(w^{(1)} - w^{(2)}\right).$$

Lemma 4: Let $E(x, y)$ be defined in (13). Then $E(x, y)$ is continuously differentiable and convex in R^{n^2+n}, and $E(x, y) = 0$ if and only if (x^*, y^*) is an optimal solution to the primal-dual problem (3) and (4). Moreover, let $w^* = (x^*, y^*)$; then for all $w = (x, y) \in R^{n^2+n}$

$$\nabla E(w)^T (w^* - w) \le -E(w).$$

Proof: Clearly, $E(x, y) \ge 0$. It is not difficult to see that $[c^T x - (e_1 - e_n)^T y]^2$, $\|(-x)^+\|_2^2$, and $\|(A^T y - c)^+\|_2^2$ are continuously differentiable and convex. Thus the function $E(x, y)$ is also continuously differentiable and convex. By the Kuhn–Tucker conditions [16], we know that (x^*, y^*) is an optimal solution to the primal-dual problem if and only if (x^*, y^*) satisfies: $c^T x - (e_1 - e_n)^T y = 0$; $A^T y \le c$; $Ax + e_n - e_1 = 0$; and $x \ge 0$, which is equivalent to $E(x^*, y^*) = 0$. Again, from the property of convex function [21], we know that

$$(w^* - w)^T \nabla E(w) \le E(w^*) - E(w), \qquad \forall\, w \in R^{n^2+n}.$$

Since $E(w^*) = 0$, $\nabla E(w)^T (w^* - w) \le -E(w)$.

The following lemma shows that the maximum eigenvalue of AA^T is $2n$.

Lemma 5: Let A be defined in (13). Then AA^T has $n - 1$ zero eigenvalues and has a nonzero eigenvalue $2n$.

Proof: Since A is an $n \times n^2$ matrix whose (i, j) column is $e_i - e_j$ for $i, j = 1, 2, \ldots, n$, then

$$AA^T = \sum_{i=1}^n \sum_{j=1}^n (e_i - e_j)(e_i - e_j)^T = \sum_{i=1}^n \sum_{j=1}^n \left(e_i e_i^T + e_j e_j^T - e_i e_j^T - e_j e_i^T\right)$$

$$= 2\sum_{i=1}^n \sum_{j=1}^n e_j e_j^T - \sum_{i=1}^n \sum_{j=1}^n \left(e_i e_j^T + e_j e_i^T\right) = 2n \sum_{j=1}^n e_j e_j^T - 2\sum_{i=1}^n e_i \sum_{j=1}^n e_j^T = 2nI - 2ee^T$$

where $e = (1, \ldots, 1)^T \in R^n$ and I is an $n \times n$ identity matrix. Note that $\det(AA^T - \mu I) = \det(2ee^T - (2n - \mu I))$, and $n-1$ eigenvalues of $2ee^T$ is 0 and one eigenvalue of $2ee^T$ is $2n$. Then AA^T has $n - 1$ zero eigenvalues and has a nonzero eigenvalue $2n$.

Using the mentioned properties about $E(x, y)$ and Lemma 5, we establish the main results for the discrete-time network.

Theorem 1: If $h < 1/\lambda_H$, where λ_H is a maximum eigenvalue of the matrix H defined in Lemma 3, then sequence $\{(x^{(k)}, y^{(k)})^T\}$ generated by the discrete-time network is globally convergent to a point $(\hat{x}, \hat{y})^T$, in which $\hat{x}, \hat{y}$ corresponds to an optimal solution of the primal-dual routing problem (3) and (4), respectively.

Proof: Let $\{w^{(k)}\} = \{(x^{(k)}, y^{(k)})^T\}$. First, by Lemma 3, we have

$$E\left(w^{(k+1)}\right) \le E\left(w^{(k)}\right) + \nabla E\left(w^{(k)}\right)^T \left(w^{(k+1)} - w^{(k)}\right) + \tfrac{1}{2}\left(w^{(k+1)} - w^{(k)}\right)^T H \left(w^{(k+1)} - w^{(k)}\right).$$

Since $w^{(k+1)} - w^{(k)} = -h\nabla E(w^{(k)})$

$$0 \le E\left(w^{(k)}\right) + \nabla E\left(w^{(k)}\right)^T \left(-h\nabla E\left(w^{(k)}\right)\right) + \frac{h^2}{2} \nabla E\left(w^{(k)}\right)^T H \nabla E\left(w^{(k)}\right).$$

Then

$$-E\left(w^{(k)}\right) \le -h\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2 + \frac{h^2\lambda_H}{2}\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2$$

and thus

$$-2hE\left(w^{(k)}\right) \le -2h^2\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2 + h^3\lambda_H\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2.$$

Therefore, from Lemma 4, we obtain

$$\begin{aligned}
&\left\|w^{(k+1)} - w^*\right\|_2^2 \\
&= \left\|w^{(k)} - w^*\right\|_2^2 - 2h\left(w^{(k)} - w^*\right)^T \nabla E\left(w^{(k)}\right) + h^2\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2 \\
&\le \left\|w^{(k)} - w^*\right\|_2^2 - 2hE\left(w^{(k)}\right) + h^2\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2 \\
&\le \left\|w^{(k)} - w^*\right\|_2^2 - h^2\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2 + h^3\lambda_H\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2 \\
&= \left\|w^{(k)} - w^*\right\|_2^2 - h^2\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2 (1 - h\lambda_H)
\end{aligned}$$

where $w^* = (x^*, y^*)^T$ is an optimal solution to the primal and dual problem. Since

$$H = \begin{bmatrix} A^TA + I & 0 \\ 0 & AA^T \end{bmatrix} + \begin{bmatrix} c \\ -b \end{bmatrix}\left[c^T, -b^T\right]$$

H is a symmetric positive semidefinite matrix. Thus $\lambda_H > 0$. So, when $\nabla E(w^{(k)}) \ne 0$ (i.e., $w^{(k)}$ is not an optimal solution) and $h < 1/\lambda_H$, it follows that

$$\left\|w^{(k+1)} - w^*\right\|_2 < \left\|w^{(k)} - w^*\right\|_2 \tag{15}$$

and $\{w^{(k)}\}$ is bounded. On the other hand, we have

$$h^2\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2 (1 - 2h\lambda_H) \le \left\|w^{(k)} - w^*\right\|_2^2 - \left\|w^{(k+1)} - w^*\right\|_2^2.$$

Thus

$$\sum_{k=1}^{\infty}\left\|\nabla E\left(w^{(k)}\right)\right\|_2^2<+\infty$$

and hence $\lim_{k\to\infty}\|\nabla E(w^{(k)})\|_2=0$. Since $\{w^{(k)}\}$ is bounded, there is a sequence $\{k_i\}$ such that $\lim_{i\to\infty}w^{(k_i)}=\hat{w}$. Then

$$\lim_{i\longrightarrow\infty}\left\|\nabla E\left(w^{(k_i)}\right)\right\|_2=\|\nabla E(\hat{w})\|_2=0$$

and thus $E(\hat{w})=0$. From Lemma 4, we see that $\hat{w}=(\hat{x},\,\hat{y})^T$ is an optimal solution to the primal and dual problem. Finally, for given any $\epsilon>0$, there exists m such that $\|w^{(k_i)}-\hat{w}\|_2<\epsilon$ for all $i\geq m$. Similar to the above analysis, we still get

$$\left\|w^{(k)}-\hat{w}\right\|_2\leq\left\|w^{(k_i)}-\hat{w}\right\|_2<\epsilon\qquad\forall k\geq k_m.$$

So $\lim_{k\to\infty}w^{(k)}=\hat{w}$.

Theorem 2: If $h<1/(2n+\|c\|_2^2+3)$, then sequence $\{w^{(k)}\}$ generated by the discrete-time network is globally convergent to an optimal solution of the primal-dual routing problem.

Proof: Since

$$H=\begin{bmatrix}A^TA+I & 0\\ 0 & AA^T\end{bmatrix}+\begin{bmatrix}c\\ -b\end{bmatrix}\begin{bmatrix}c^T,\,-b^T\end{bmatrix}$$

and H is the sum of two symmetric matrices, according to the Courant–Fischer minmax theorem [22], we see that the maximum eigenvalue of H satisfies $\lambda_H\leq\lambda+\|c\|_2^2+2$, where λ is the maximum eigenvalue of

$$\begin{bmatrix}A^TA+I & 0\\ 0 & AA^T\end{bmatrix}.$$

Since A^TA and AA^T have the same nonzero eigenvalues, by Lemma 4, we have $\lambda=2n+1$. Hence

$$\lambda_H\leq 2n+\|c\|_2^2+3.$$

Then from Theorem 1, we can complete the proof.

V. Parameter Prescaling

From the preceding section, we see that the convergence rate of the discrete-time routing network depends upon the step parameter h, and thus upon the size of the shortest path routing problem. When h is small, the convergence rate of the primal-dual network decreases. In order to increase the convergence rate, we need to scale A, b, and c properly beforehand. An improved design is as follows: for $i,\,j=1,\,2,\,\ldots,\,n$

$$\hat{x}_{ij}^{(k+1)}=\hat{x}_{ij}^{(k)}-h\left[\hat{\xi}^{(k)}(\beta c_{ij}/\alpha)-\left(-\hat{x}_{ij}^{(k)}\right)^+ +\alpha^2\left(\hat{u}_i^{(k)}-\hat{u}_j^{(k)}\right)\right]\tag{16}$$

$$\hat{y}_i^{(k+1)}=\hat{y}_i^{(k)}-h\left[\hat{\xi}^{(k)}\left(\delta_{in}-\delta_{i1}\right)+\alpha^2\hat{v}_i^{(k)}\right)\Big]\tag{17}$$

where $\hat{\xi}^{(k)}=\alpha^2\sum_{i=1}^n[\sum_{j=1}^n(\beta c_{ij}/\alpha)\hat{x}_{ij}^{(k)}-\hat{y}_1^{(k)}+\hat{y}_n^{(k)}]$, $\hat{v}_i=\sum_{p=1}^n[(\hat{y}_i^{(k)}-\hat{y}_p^{(k)}-\beta c_{ip}/\alpha)^+-(\hat{y}_p^{(k)}-\hat{y}_i^{(k)}-\beta c_{pi}/\alpha)^+]$, $\hat{u}_i^{(k)}=\sum_{p=1}^n(\hat{x}_{ip}^{(k)}-\hat{x}_{pi}^{(k)})+\delta_{in}-\delta_{1i}$, and α and β are positive parameters, respectively. Note that $\beta c_{ij}/\alpha$ can be prescaled, compared with the original dynamic equations in (11) and (12), and the scaled dynamic equation adds only n^2+2n+1 amplifiers. More important, in the improved network, step h is independent of the size of the shortest path routing problem.

On the global convergence of the prescaled network in (16) and (17), we have the following result.

Theorem 3: If $\beta=1/(\sqrt{2}\|c\|_2)$, $\alpha=1/2\sqrt{(n+1)}$, and $h<1$, then the sequence $\{(x^{(k)},\,y^{(k)})^T\}$ generated by the discrete-time network (16), (17) is globally convergent to an optimal solution of the shortest path problem.

Proof: Similar to the discussion in Section IV, we consider the following energy function with parameters α and β:

$$\hat{E}(x,\,y)=\frac{1}{2}\Bigg[\left(\beta c^Tx-\alpha(e_1-e_n)^Ty\right)^2+\|(-x)^+\|_2^2+\left\|\left(\alpha A^Ty-\beta c\right)^+\right\|_2^2+\|\alpha Ax+\alpha(e_n-e_1)\|_2^2\Bigg].$$

Similar to the proof of Theorem 2, we can obtain the following inequality:

$$\left\|\hat{w}^{(k+1)}-w^*\right\|_2^2\leq\left\|\hat{w}^{(k)}-w^*\right\|_2^2-h^2\left\|\nabla\hat{E}\left(\hat{w}^{(k)}\right)\right\|_2^2(1-h\lambda_{\overline{H}})$$

where $\nabla\hat{E}(w)$ is the gradient of $\hat{E}(w)$, $\hat{w}^{(k)}=(\hat{x}^{(k)},\,\hat{y}^{(k)})^T$,

$$\overline{H}=\begin{bmatrix}\alpha^2A^TA+I & 0\\ 0 & \alpha^2AA^T\end{bmatrix}+\begin{bmatrix}\beta c\\ -\alpha b\end{bmatrix}\begin{bmatrix}\beta c^T,\,-\alpha b^T\end{bmatrix}$$

and $\lambda_{\overline{H}}$ is a maximum eigenvalue of the matrix $\overline{H}$. Because $\beta=1/(\sqrt{2}\|c\|_2)$ and $\alpha=1/(2\sqrt{n+1})$, $\|\beta c\|_2^2+2(n+1)\alpha^2=1$. By the Courant–Fischer minimax theorem [22] and Theorem 2, we obtain

$$\lambda_{\overline{H}}\leq\alpha^2(2n)+\beta^2\|c\|_2^2+2\alpha^2=1$$

since the maximum eigenvalue of A^TA is $2n$. Thus $h\lambda_{\overline{H}}<1$. From Theorem 1, it follows the conclusion.

Remark 1: Theorem 3 provides a sufficient condition for global convergence. Because it is not necessary condition, the discrete-time network could still converge when $h\geq 1$. Furthermore, since $\hat{w}^{(k+1)}-\hat{w}^k=-h\nabla\hat{E}(\hat{w}^{(k)})$

$$\hat{E}\left(\hat{w}^{(k+1)}\right)-\hat{E}\left(\hat{w}^k\right)\leq-h\left\|\nabla\hat{E}\left(\hat{w}^{(k)}\right)\right\|_2^2+\frac{h^2}{2}\left\|\nabla\hat{E}\left(\hat{w}^{(k)}\right)\right\|_2^2$$

and

$$\hat{E}\left(\hat{w}^{(k+1)}\right)\leq\hat{E}\left(\hat{w}^k\right)-h\left\|\nabla\hat{E}\left(\hat{w}^{(k)}\right)\right\|_2^2+\left\|\nabla\hat{E}\left(\hat{w}^{(k)}\right)\right\|_2^2$$

when $h\leq\sqrt{2}$. We thus conclude that the larger iteration step h is, the faster the convergence rate of the discrete-time network is in this case.

Remark 2: Because of the gradient descent nature of the recurrent neural network, the convergence rate near an optimum is unavoidably slow. Since the optimal solution is always binary, a threshold logic unit may be added to each neurons for x_{ij} as an output unit to expedite the convergence near an optimum so that the output variable

$$\overline{x}_{ij}=\begin{cases}0, & \text{if } x_{ij}<0.5\\ 1, & \text{if } x_{ij}>0.5.\end{cases}$$

VI. Simulation Results

Consider a problem in oil transport technology from the Black Gold Petroleum Company [18]. From oil storage to final destinations there are 42 possible sites for each substation. In order to minimize the attendant manufacturing costs, this problem is to find the shortest transmission line between the source node 1 and the terminal node 44. The shortest path of this problem is $\{e_{1,3},\,e_{3,9},\,e_{9,14},\,e_{14,20},\,e_{20,26},\,e_{26,32},\,e_{32,38},\,e_{38,44}\}$, where $e_{i,j}$ $(i=1,\,2,\,\ldots,\,n)$ denotes the distance between node i and

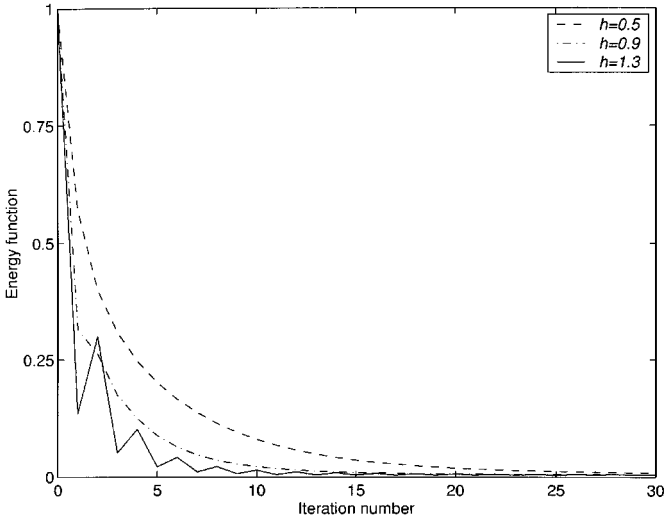

Fig. 2. Energy function over iterations for the primal-dual routing network with different step lengths.

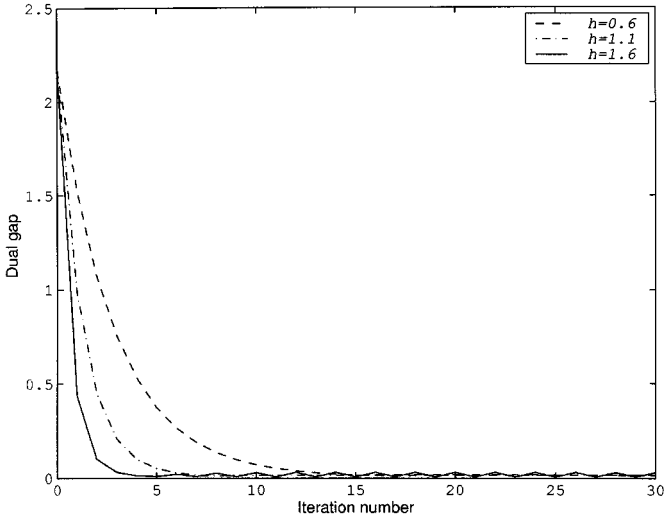

Fig. 3. Duality gap over iterations for the primal-dual routing network with different step lengths.

node j in the network. The total cost of the optimal path is equal to 33. We now solve this problem by the proposed discrete network. Fig. 2 illustrates, under the same initial point $x(0) = 0 \in R^{1936}$ and $y(0) = [1, \ldots, 1]^T/44 \in R^{44}$, the transient behavior of the energy function of the discrete-time routing network with three different step lengths. Specifically, the solid line decreasing at the fastest rate corresponds to $h = 1.3$, the slowest line $h = 0.5$, and in-between $h = 0.9$. Fig. 3 illustrates, under the same initial point $x(0) = [1, \ldots, 1]^T \in R^{1936}$ and $y(0) = [1, \ldots, 1]^T/44 \in R^{44}$, the transient behavior of the dual gap of the discrete-time routing network with three different step lengths. Specifically, the solid line decreasing at the fastest rate corresponds to $h = 1.6$, the slowest line $h = 0.6$, and in-between $h = 1.1$. All the values of the energy function and duality gap converge to zero rapidly.

VII. Concluding Remarks

In this paper, a discrete-time primal-dual routing network with a fixed step parameter is presented. The proposed primal-dual routing network is proven to be capable of obtaining shortest path routing for directed networks with arbitrary cost coefficients, unlike some existing numerical algorithms. Compared with other discrete-time neural networks for optimization, the present one is guaranteed to be globally convergent to exact solutions. Compared with continuous-time neural networks for the shortest path problem [11]–[17], the proposed discrete-time network is suitable for digital implementation using widely available design tools. The tradeoff is that the network size is slightly larger than the primal or dual routing network. Furthermore, the convergence rate of the primal-dual routing network is nondecreasing with respect to the size of the shortest path problem and can be expedited by properly scaling design parameters. These features make the primal-dual routing network suitable for solving large-scale shortest path problems in real-time applications.

References

[1] L. Bodin, B. L. Golden, A. Assad, and M. Ball, "Routing and scheduling of vehicles and crews: The state of the art," *Comput. Oper. Res.*, vol. 10, pp. 63–211, 1983.

[2] A. Ephremides and S. Verdu, "Control and optimization methods in communication network problems," *IEEE Trans. Automat. Contr.*, vol. 34, pp. 930–942, 1989.

[3] S. Jun and K. G. Shin, "Shortest path planning in distributed workspace using dominance relation," *IEEE Trans. Robot. Automat.*, vol. 7, pp. 342–350, 1991.

[4] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*. New York: Holt, Rinehart, and Winston, 1976, pp. 59–108.

[5] J. J. Hopfield and D. W. Tank, "Neural computation of decisions in optimization problems," *Biol. Cybern.*, vol. 52, pp. 141–152, 1985.

[6] D. W. Tank and J. J. Hopfield, "Simple neural optimization networks, an A/D converter, signal decision circuit, and a linear programming circuit," *IEEE Trans. Circuits Syst.*, vol. 33, pp. 533–541, 1986.

[7] A. Cichocki and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*, London, U.K.: Wiley, 1993.

[8] J. Wang, "A deterministic annealing neural network for convex programming," *Neural Networks*, vol. 7, pp. 629–641, 1994.

[9] Y. Xia, "A new neural network for solving linear programming problems and its applications," *IEEE Trans. Neural Networks*, vol. 7, pp. 525–529, 1996.

[10] ——, "A new neural network for solving linear and quadratic programming problems," *IEEE Trans. Neural Networks*, vol. 7, pp. 1544–1547, 1996.

[11] H. E. Rauch and T. Winarske, "Neural networks for routing communication traffic," *IEEE Control Syst. Mag.*, vol. 8, pp. 26–30, 1988.

[12] S.-L. Lee and S. Chang, "Neural networks for routing communication networks with unreliable components," *IEEE Trans. Neural Networks*, vol. 4, pp. 854–863, 1993.

[13] M. Ali and F. Kamoun, "Neural networks for shortest path computation and routing in computer networks," *IEEE Trans. Neural Networks*, vol. 4, pp. 941–954, 1993.

[14] J. Wang, "A recurrent neural network for solving the shortest path problem," *IEEE Trans. Circuits Syst. I*, vol. 43, pp. 482–486, 1996.

[15] ——, "Primal-dual neural networks for solving the shortest path routing," *IEEE Trans. Syst., Man, Cybern.*, vol. 28, pp. 854–859, 1998.

[16] L. Zhang and S. C. A. Thomopoulos, "Neural network implementation of the shortest path algorithm for traffic routing in communication network," in *Proc. Int. Joint Conf. Neural Networks*, vol. II, Washington, D.C., June 18–22, 1989, p. 591.

[17] S. C. A. Thomopoulos, L. Zhang, and C.-D. Wann, "Neural network implementation of the shortest path algorithm for traffic routing in communication network," in *Proc. Int. Joint Conf. Neural Networks (IJCNN'91)*, Singapore, Nov 18–21, 1991, pp. 2693–2702.

[18] D. T. Phillips and A. Garcia-Diaz, *Fundamentals of Network Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1981.

[19] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, *Linear Programming and Network Flows*, 2nd ed. New York: Wiley, 1990.

[20] X. Lu, "An approximate Newton method for linear programming," *J. Numer. Comput. Appl.*, vol. 15, no. 2, pp. 93–104, 1994.

[21] J. M. Ortega and W. G. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*. New York: Academic, 1970.

[22] G. H. Golub and C. F. Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1996.