Dynamic flows with time-varying network parameters:
Optimality conditions and strong duality

S. Mehdi Hashemi · Ronald Koch ·
Ebrahim Nasrabadi · Martin Skutella

Abstract Dynamic network flow problems model the temporal evolution of flows over time and also consider changes of network parameters such as capacities, costs, supplies, and demands over time. These problems have been extensively studied in the past beacuse of their important role in real world applications such as transport, traffic, and logistics. This has led to many results, but the more challenging continuous time model still lacks some of the key features such as network related optimality conditions and algorithms that are available in the static case.

The aim of this paper is to advance the state of the art for dynamic network flows by developing the continuous time analogues of several well-known optimality conditions for static network flows. Specifically, we establish a reduced cost optimality condition, a negative cycle optimality condition, and a strong duality result for a very general class of dynamic network flows. The underlying idea is to construct a dual feasible solution that proves optimality when the residual network (with respect to a given flow) contains no dynamic cycles with negative cost. We also discuss a generic negative cycle-canceling algorithm resulting from the corresponding optimality criterion and point out promising directions for future research.

Keywords Dynamic network flows · Continuous linear programming · Augmenting paths and cycles · Optimality conditions · Duality · Cycle-canceling algorithm

Mathematics Subject Classification (2000) 90B10 · 49N05 · 05C38 · 90C46

1 Introduction

Network flows have applications in a wide range of fields, including chemistry, physics, most branches of engineering, manufacturing, scheduling and routing, telecommunica-

S. M. Hashemi
Department of Computer Science, Amirkabir University of Technology, 424 Hafez Ave., Tehran, Iran
E-mail: hashemi@aut.ac.ir

R. Koch · E. Nasrabadi · M. Skutella
Institut für Mathematik, Technische Universität Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany
E-mail: {koch,nasrabadi,skutella}@math.tu-berlin.de

tion, transportation and logistics (see for instance [1]). A crucial characteristic arising in various applications such as road traffic control, production systems, communication networks (e. g., the Internet), financial flows, and pipeline systems for transporting (e.g., crude oil) is flow variation over time. In other words, in such applications flow values on arcs are not constant but may change over time due to seasonal altering demands, supplies and arc capacities. Moreover, there is a second temporal dimension in these applications. Usually, flow does not travel instantaneously through a network but requires a certain amount of time to travel through each arc.

The above mentioned aspects of network flows are captured by *dynamic network flows* (also called *network flows over time*) which were introduced by Ford and Fulkerson [18, 19]. They include a temporal dimension and therefore provide a more realistic modeling tool for numerous real-world applications. In addition to the normal input for classical network flows, each arc also has a *transit time*. The transit time is the amount of time required to send flow from the tail to the head of that arc. In contrast to the classical case of static flows, a *dynamic flow* in such a network specifies a flow rate entering an arc for each point in time. In this setting, the capacity of an arc limits the rate of flow into the arc at each point in time.

Dynamic network flows have been traditionally considered in a purely static environment and the terminology "dynamic" has emphasized the fact that the movement of flow through the network over time is considered. In many practical applications such a static representation may be inadequate and it would be worthwhile if the model considers not only the time-varying nature of flow, but also of network parameters. However, the fact that network characteristics such as capacities, costs, demands, and supplies etc. may vary over time has not been reflected to an adequate extent in the literature so far. The main reason is that the resulting dynamic network flow problems are much harder to solve or analyze in detail, specifically when time is modeled as a continuum. In this paper we consider a general class of dynamic network flow problems with time-varying parameters and develop several optimaity conditions and a strong dualiy result for these problems.

1.1 Problem description

The class of dynamic network flows that we consider is as follows. We are given a directed graph G with node set $N = \{1, 2, \dots, n\}$ and arc set $A \subseteq N \times N$ and a time horizon $T > 0$. To simplify notation, we assume (without loss of generality) that every pair of nodes is connected by at most one arc. Each arc (i, j) is associated with two functions defined on the time interval $[0, T]$: *transit cost* $c_{i,j}$ and *transit capacity* $a_{i,j}$. The transit cost $c_{i,j}(t)$ gives the cost per flow unit for sending flow into arc (i, j) at time t and the transit capacity $a_{i,j}(t)$ gives an upper bound on the flow rate (i.e., amount of flow per time unit) that can enter arc (i, j) at time t. In addition, the arc (i, j) has an associated *transit time* $\lambda_{i,j}$. Thus flow entering arc (i, j) at time t needs $\lambda_{i,j}$ time units to travel through the arc and thus arrives at node j at time $t + \lambda_{i,j}$.

Each node i is associated with three functions defined on $[0, T]$: *supply/demand* r_i, *storage cost* d_i and *storage capacity* b_i. Here $r_i(t)$ denotes the available supply rate or required demand rate of flow at node i at time t, depending on whether $r_i(t) > 0$ or $r_i(t) < 0$. Moreover, $d_i(t)$ is the cost per time unit for storing one unit of flow at node i at time t and $b_i(t)$ is an upper bound on the amount of flow that can be stored at node i at time t.

The aim of the *Continuous-time Dynamic Network Flow Problem (CDNFP)* is to find a dynamic flow that satisfies all demands and obeys all transit and storage capacity constraints over the time interval $[0, T]$, while minimizing the total transit and storage costs. This problem is formulated as an infinite-dimensional linear program with a network structure and arc time-delays as given below:

$$\begin{aligned}
\text{CDNFP: } \min \quad & \int_0^T c(t)^T x(t)\, dt + \int_0^T d(t)^T y(t)\, dt \\
\text{s.t.} \quad & \int_0^t \sum_{j:(i,j)\in A} x_{i,j}(s)\, ds - \int_0^t \sum_{j:(j,i)\in A} x_{j,i}(s - \lambda_{j,i})\, ds \\
& \qquad + y_i(t) = \int_0^t r_i(s)\, ds, \quad i \in N,\ t \in [0, T], \qquad (1) \\
& 0 \le x(t) \le a(t), \quad t \in [0, T], \qquad (2) \\
& 0 \le y(t) \le b(t), \quad t \in [0, T]. \qquad (3)
\end{aligned}$$

In this formulation, $x_{i,j}(t)$ gives the rate of flow (i.e., amount of flow per time unit) entering arc (i, j) at time t and $y_i(t)$ measures the amount of flow stored at node i at time t. Notice that any choice of flow $x(s)$, $s \in [0, t]$, will uniquely determine a storage function $y(t)$ by the *flow conservation constraints* (1). We say that flow x (with corresponding storage y) is *feasible* for CDNFP if x satisfies *transit capacity constraints* (2) and generates storage y satisfying the *storage capacity constraints* (3).

For technical reasons, we require that the components of b are continuous on $[0, T]$. Moreover, we require to work within the space $L_\infty([0, T])$ of essentially bounded measurable functions on $[0, T]$ in which functions that differ only on a set of measure zero are identified. In particular, the components of a, b, c, d, r, and x are assumed to be bounded measurable functions on $[0, T]$. Hence, the *feasible region* of CDNFP, denoted by F, is defined as

$$F := \left\{ x \in L_\infty^{|A|}[0, T] \mid x \text{ with corresponding storage } y \text{ is feasible for CDNFP} \right\}.$$

where $|A|$ denotes the number of arcs in the network G. Throughout the paper, it is assumed that F is not empty. This guarantees the existence of an optimum solution for CDNFP at an extreme point of F. This is because of the fact that F is convex, bounded, and closed in the weak topology $\sigma(L_\infty^{|A|}([0, T]), L_1^{|A|}([0, T]))$. Then, it follows from Alaoglua's Theorem (see, e.g., [15]) that F is compact in the weak topology on $L_\infty^{|A|}([0, T])$ and consequently is a convex hull of its extreme points by Krein-Milman's Theorem (see again [15]). Further, the objective function of CDNFP is $\sigma\big(L_\infty^{|A|}([0, T]), L_1^{|A|}([0, T])\big)$-continuous functional and hence will attain its minimum over F at an extreme point.

1.2 Literature

Since the seminal work of Ford and Fulkerson in the 1950s, a large number of authors have studied different features of dynamic network flow models (see [42] and the references therein). The research in this area has taken two approaches. One approach models time in discrete time steps. The other approach models time continuously. Research of the first type typically uses the time-expanded network, either explicitly in the algorithms, or implicitly in the proof, to produce theoretically or practically efficient algorithms. Research using the second approach usually considers networks with time-varying capacities, costs, supplies, and demands, and focuses on proving the existence of optimal solutions, investigating the structure of optimal solution and extending duality theory. In the following, we briefly review results on the continuous-time model, particularly those related to CDNFP.

A closely related problem to CDNFP is the continuous-time dynamic maximum flow problem, whose goal is to send as much flow as possible from a source to a sink within a given time interval in a network with transit times on the arcs and time-varying transit and storage capacities. This problem was first introduced by Philpott [30] and further studied by Anderson, Nash, and Philpott [6]. They introduce the concept of continuous-time cuts and establish a MaxFlow-MinCut theorem (see also [4]) for the case that transit times are zero and the transit capacities are bounded measurable. This result was later extended to arbitrary transit times by Philpott [32] and to a general model of dynamic network flows combining both discrete and continuous aspects in only one model by Koch, Nasrabadi, and Skutella [23].

The CDNFP problem was first introduced by Anderson [3], who characterizes extreme point solutions for the problem given rational transit times. Anderson and Philpott [8] survey results relating to dynamic network flows in the continuous-time model. They also introduce a dual problem for CDNFP with a corresponding definition of complementary slackness and prove a weak duality result.

In the absence of transit times, storage costs and storage capacities, CDNFP becomes a special type of *Separated Continuous Linear Programs (SCLP)*. The SCLP problem has been first introduced by Anderson [2] in order to model job-shop scheduling problems and has attracted most of the attention in the class of continuous-time linear programs[1] (CLP) due to its applications. Actually, problems of this kind arise in a number of engineering applications (see, e.g., [26,37,43]). Anderson, Nash, and Perold [5] characterize the extreme point solutions to SCLP and show the existence of optimum solutions with a finite number of breakpoints in certain cases. Since then a number of authors (including Pullan [34–36,39,40], Philpott and Craddock [33], Luo and Bertsimas [25], Fleischer and Sethuraman [17] and Weiss [43]) have studied SCLP from different points of view.

Pullan [38] examines a larger class of SCLP to include time-delays, so-called *Separated Continuous Linear Programs with Time-Delays (SCLPTD)*. CDNFP becomes a special case of SCLPTD when storage capacities are infinite. For the case that transit times are rational, Pullan [38] transforms SCLPTD into a larger problem which is very close to a special class of SCLP and extends some results of SCLP to SCLPTD.

The common approach to solving CLP as well as SCLP is to convert the original problem to a finite-dimensional approximation linear program by discretization of time. This approach, which is called *discretization*, was first taken by Buie and Abrham [13] for solving CLP and later used by Pullan [34], Philpott and Craddock [33], Luo and Bertsimas [25] for SCLP, who assumed that the problem data are piecewise constant/linear. This approach has attracted most of the attention for solving practical problems for the following reasons:

[1] Continuous-time linear programs were introduced by Bellman [11,12], who called them bottleneck problems.

1. Discretization of time leads to problems that typically can be solved by using traditional methods and are in general much easier to handle computationally. In fact, the techniques required to implement such algorithms are no more than finite-dimensional linear programming.
2. The solutions for discrete approximations converge to the solution for the original problem as the discretizations become finer.

Motivated by these advantages, Hashemi, Nasrabadi, and Skutella [21] have developed two discretization-based algorithms, so-called Descent Algorithm and Adaptive Discretization Algorithm, for CDNFP under some assumptions on the form of the problem data. Although both algorithms converge to the optimal value of CDNFP as the discretizations become finer, these algorithms, particularly the Descent Algorithm, may not be satisfactory in practice. More specifically, computations for small example instances in [21, 27] show that the solutions obtained by the Descent Algorithm have a huge number of breakpoints[2], many more than necessary. Furthermore, when this algorithm proceeds, the number of breakpoints increases further with little improvement in the objective function value. Apart from the slow convergence and long computation times, this can also obscure the structure of the optimal solution. The same serious problems have been already reported in [34, 40] for solving SCLP.

In general, discretization-based algorithms for solving continuous-time linear programs have the following major disadvantages:

1. The size of resulting discrete approximations is enormous, which leads to long computation times.
2. The solution is only approximate, and to obtain a good approximation it is necessary to divide the time interval into a large number of subintervals. On the other hand, a very fine level of discretization is likely to generate solutions with huge numbers of breakpoints, many more than necessary. The redundant breakpoints not only increase the size of the subproblems, leading to long computation times, but also cause serious numerical problems (see [27, 33]).
3. Sensitivity analysis plays an important role in optimization and it is most worthwhile to develop algorithms which allow to perform sensitivity analysis. Unfortunately, the discretization-based approaches are not suitable for performing sensitivity analysis,
4. The input functions must be piecewise constant/linear. For example, Descent Algorithm and Adaptive Discretization Algorithm presented in [27] for solving CDNFP rely on the assumption of piecewise linear c and b and piecewise constant d, r and a.

Consequently, a number of authors have attempted to generalize the simplex method to solve instances of CLP without discretization. This approach was initiated by Lehman [24] and continued later by Drews [16], Hartberger [20] and Segers [41]. Perold [28, 29] makes major progress in this direction with the specification of a simplex-like algorithm for CLP. Anstreicher [9] continues Perold's work in his thesis. However, the described algorithm is complicated and incomplete, reflecting the difficult nature of the problem. Recently, Weiss [43] examines SCLP under the assumption of piecewise linear problem data and develops a simplex algorithm that gives an exact solution after a finite number of iterations. Moreover, he characterizes the form of optimal solutions and establishes a strong duality result.

[2] The term *breakpoints* is used to refer to the points in time at which the solution changes.

The first attempt for developing a simplex algorithm for dynamic network flow problems in the continuous-time model is due to Anderson and Philpott [7]. They consider CDNFP with zero transit times on the arcs and piecewise constant/linear input functions. In particular, they assume that the transit times λ are zero and the transit cost functions c and the storage capacity functions b are piecewise linear. All remaining input functions are assumed to be piecewise constant, that is, supply and demand rates r, storage costs d, and transit capacities a. Then they discuss how the simplex method can be developed for CDNFP to directly produce an exact solution, rather than doing a discretization to get an approximation to the optimal solution. Unfortunately, there are no guarantees for the convergence of this algorithm and it often produces a sequence of solutions which converge to a suboptimal solution.

1.3 Our contribution

Despite many attempts on dynamic network flows, the continuous-time theory still lacks some of the key features (such as network related algorithms) that are available in static network flow theory. Most algorithms for static network flows are based on the duality theory and optimality conditions. Hence, an essential and crucial step is to develop these fundamental elements for CDNFP.

In this paper we are concerned with the development of continuous-time analogues to those concepts and techniques which are the cornerstones of static network flows. Specifically, several network based optimality conditions analogous to that found in static network flows are developed for CDNFP with piecewise analytic input functions and rational transit times. A strong duality result is then derived from these optimality conditions. Previously, strong duality was developed by Pullan [36,38] for SCLP given piecewise analytic problem data and for SCLPTD with rational transit times and piecewise constant/linear input functions. However, we do not follow the approach taken by Pullan but we make use of ideas from the area of static network flows for proving a strong duality result.

The remainder of this paper is organized as follows. Section 2 presents preliminaries and some earlier results that are required for the purpose of the paper. In particular, a dual problem for CDNFP is introduced and a notation of complementary slackness is derived from a weak duality result. In Section 3 we introduce the concept of augmenting paths and cycles and prove the existence of shortest augmenting paths. This result is used in Section 4 to establish some optimality conditions for CDNFP. More precisely, it is shown that the shortest path labels define a dual solution which satisfies complementary slackness conditions together with a given flow when the network has no augmenting cycles with negative cost. This leads to the development of a reduced cost optimality condition, a negative cycle optimality condition, and a strong duality result for CDNFP. The optimality conditions allow us to develop algorithms for CDNFP analogous to that known for the static minimum cost flow problem. We present a generic version of such an algorithm and discuss several promising directions for future research in Section 5.

2 Dual formulation

The concept of duality plays a central role in the theory of linear programming and is at the heart of the simplex algorithm for static network flows. Thus to generalize this algorithm to CDNFP, it would be necessary to establish a similar duality theory. In this section we present some results on the duality theory of CDNFP that will be useful for our main results. Let us first rewrite CDNFP in the following equivalent form:

$$
\begin{aligned}
\text{CDNFP: min } & \int_0^T c(t)^T x(t)\,dt + \int_0^T d(t)^T y(t)\,dt \\
\text{s.t. } & \int_0^t \sum_{j:(i,j)\in A} x_{i,j}(s)\,ds - \int_0^t \sum_{j:(j,i)\in A} x_{j,i}(s-\lambda_{j,i})\,ds \\
& \qquad + y_i(t) = \bar{r}_i(t), \quad i \in N,\ t \in [0,T], \qquad (4) \\
& -\int_0^t \sum_{j:(i,j)\in A} x_{i,j}(s)\,ds + \int_0^t \sum_{j:(j,i)\in A} x_{j,i}(s-\lambda_{j,i})\,ds \\
& \qquad \le b_i(t) - \bar{r}_i(t), \quad i \in N,\ t \in [0,T], \qquad (5) \\
& x(t) \le a(t), \quad t \in [0,T], \qquad (6) \\
& x(t) \ge 0,\ y(t) \ge 0, \quad t \in [0,T].
\end{aligned}
$$

Here $\bar{r}_i(t)$ denotes the total supply or demand at node i up to time t, i.e., $\bar{r}_i(t) := \int_0^t r_i(s)\,ds$. Moreover for the ease of notation, in what follows we assume that the storage costs d are zero. This assumption imposes no loss of generality because we can transform a general instance of CDNFP to an instance where the storage costs are zero. Now by introducing the dual variables u, v, and w associated to the constraints (4), (5) and (6), respectively, a dual problem $\text{CDNFP}^{*'}$ for CDNFP can be given as follows:

$$
\begin{aligned}
\text{CDNFP}^{*'} : \text{max } & \int_0^T \bar{r}(t)^T u(t)\,dt + \int_0^T \{b(t) - \bar{r}(t)\}^T v(t)\,dt + \int_0^T a(t)^T w(t)\,dt \\
\text{s.t. } & \int_t^T \left(u_i(s) - u_j(s+\lambda_{i,j})\right) ds - \int_t^T \left(v_i(s) - v_j(s+\lambda_{i,j})\right) ds \\
& \qquad + w_{i,j}(t) \le c_{i,j}(t), \quad (i,j) \in A, \quad t \in [0,T], \\
& u(t) \le 0,\ v(t) \le 0,\ w(t) \le 0, \quad t \in [0,T].
\end{aligned}
$$

The form of this dual problem is based on the dual formulation of CLP proposed by Bellman [11, 12]. Simple examples can be constructed such that CDNFP has an optimal solution, but there is no optimal solution for $\text{CDNFP}^{*'}$, even for the case of zero transit times (see, e.g., Example 4.2 in Pullan [37]). For this reason it would be necessary to consider a more general dual problem. In particular, we consider the following dual problem CDNFP*:

$$\begin{aligned}\text{CDNFP}^* : \max \ & -\int_0^T \bar{r}(t)^T \, d\eta(t) - \int_0^T \{b(t) - \bar{r}(t)\}^T \, d\mu(t) + \int_0^T a(t)^T \rho(t) \, dt \\ \text{s.t. } & \eta_i(t) - \eta_j(t + \lambda_{i,j}) + \mu_i(t) - \mu_j(t + \lambda_{i,j}) \\ & \qquad + \rho_{i,j}(t) \le c_{i,j}(t), \quad (i,j) \in A, \ t \in [0,T], \\ & \eta \text{ and } \mu \text{ monotonic increasing and right continuous} \\ & \qquad \text{on } [0,T] \text{ with } \eta(T) = \mu(T) = 0, \\ & \rho(t) \le 0, \quad t \in [0,T].\end{aligned}$$

This problem was introduced by Anderson and Philpott [8] and is based on that given by Pullan [34] for the dual of SCLP. Here the notation $\int_u^v g(t)\, df(t)$ denotes the Lebesgue-Stieltjes integral of function g with respect to function f from u to v when the integral exists. Thus by integration by parts (see Theorem 7.6 in [10]) and the fact that $\bar{r}(0) = \eta(T) = \mu(T) = 0$, the objective function of CDNFP* can be written in the following equivalent form:

$$\int_0^T r(t)\{\eta(t) - \mu(t)\} \, dt - \int_0^T b(t) \, d\mu(t) + \int_0^T a(t)\rho(t) dt.$$

It is easy to see that CDNFP* is a generalization of CDNFP$^{*'}$ because any feasible solution u, v, w for CDNFP$^{*'}$ generates one for CDNFP* with the same objective function value by defining

$$\eta(t) = \int_t^T u(s) \, ds, \qquad \mu(t) = \int_t^T v(s) \, ds, \qquad \rho(t) = w(t).$$

Conversely if η, μ, ρ is feasible for CDNFP* in which η and μ are absolutely continuous on $[0,T]$, then

$$u(t) = -\dot{\eta}(t), \qquad v(t) = -\dot{\mu}(t), \qquad w(t) = \rho(t),$$

is feasible for CDNFP$^{*'}$ and again the two solutions have the same objective function value.

Anderson and Philpott [8] show that CDNFP* has an alternative equivalent formulation in an analogous manner to that described for static network flows. They also introduce the concept of complementary slackness for CDNFP deriving from a weak duality result. We present these results in the rest of the section.

Given a feasible solution η, μ, ρ for CDNFP*, we define the *potential function* π on the time interval $[0,T]$ by

$$\pi(t) = \eta(t) - \mu(t), \qquad t \in [0,T]. \tag{7}$$

It is clear that π is of bounded variation because it is the difference between two monotonic increasing functions. Then there exist functions $\pi^{(+)}$ and $\pi^{(-)}$, known as the *Jordan decomposition* of π, that are monotonic increasing on $[0,T]$ with $\pi^{(+)}(T) = \pi^{(-)}(T) = 0$ and $\pi(t) = \pi^{(+)}(t) - \pi^{(-)}(t)$ for $t \in [0,T]$. These functions are defined by

$$\pi^{(+)}(t) = V(t) - V(T), \qquad \pi^{(-)}(t) = V(t) - \pi(t) - V(T), \qquad t \in [0,T], \tag{8}$$

where $V(t) = V(\pi; [0,t])$ measures the total variation of π within the time interval $[0,t]$ (see, e.g., Chapter 6 in [10]). The functions $\pi^{(+)}$ and $\pi^{(-)}$ are called the positive and negative part of π, respectively.

Let us give some properties of the Jordan decomposition $\pi^{(+)}$ and $\pi^{(+)}$ of π that will be useful for our discussion. We first need to give the concept of a function strictly increasing at a point.

We say that a monotonic increasing function $f : [u,v] \to \mathbb{R}$ is *strictly increasing* at $t \in (u,v)$ if $f(t_1) < f(t_2)$ for any $t_1, t_2 \in [u,v]$ with $t \in (t_1,t_2)$, f is *strictly increasing* at u if $f(u) < f(t)$ for every $t \in (u,v]$, and f is *strictly increasing* at v if $f(t) < f(v)$ for every $t \in [u,v)$. A function f of bounded variation on $[u,v]$ is said to be *strictly increasing* at $t \in [u,v]$ if $f^{(+)}$ is strictly increasing at t, similarly f is *strictly decreasing* at t if $f^{(-)}$ is strictly increasing at t.

The following lemma follows from some basic results in measure theory.

Lemma 1 *Let η, μ, ρ be a feasible solution for CDNFP* and π be given by (7).*

1. *If π is strictly increasing (decreasing) at some t, then η (μ) is also strictly increasing at t.*
2. *The functions $\eta - \pi^{(+)}$ and $\mu - \pi^{(-)}$ are monotonic increasing on $[0,T]$.*

We can now establish the following result.

Lemma 2 *Suppose that η, μ, ρ is an optimal solution for CDNFP* and π is given by (7). Let $\pi^{(+)}$ and $\pi^{(-)}$ be the Jordan decomposition of π, given by (8). Then*

$$\eta^* = \pi^{(+)}, \qquad \mu^* = \pi^{(-)}, \qquad \rho^* = \rho,$$

is also an optimal solution for CDNFP*.

Proof It is clear that η^*, μ^*, ρ^* is feasible for CDNFP*. So it is sufficient to show that[3]

$$V[\text{CDNFP}^*, \eta^*, \mu^*, \rho^*] \geq V[\text{CDNFP}^*, \eta, \mu, \rho],$$

or equivalently

$$\int_0^T b(t)^T \, d(\mu(t) - \pi^{(-)}(t)) \geq 0.$$

The above inequality easily follows from that fact that b is nonnegative and $\mu - \pi^{(-)}$ is monotonic increasing on $[0,T]$. □

Having Lemma 2, we can replace μ with $\pi^{(-)}$ in the dual problem CDNFP* and rewrite CDNFP* in the following equivalent form:

$$\begin{aligned}
\text{CDNFP}^*: \ \max \ & \int_0^T r(t)^T \pi(t)\, dt - \int_0^T b(t)^T \, d\pi^{(-)}(t) + \int_0^T a(t)^T \rho(t) dt \\
\text{s.t. } & \pi_i(t) - \pi_j(t + \lambda_{i,j}) + \rho_{i,j}(t) \leq c_{i,j}(t), \quad (i,j) \in A, \ t \in [0,T], \\
& \pi \text{ of bounded variation and right continuous} \\
& \qquad \text{on } [0,T] \text{ with } \pi(T) = 0, \\
& \rho(t) \leq 0, \quad t \in [0,T].
\end{aligned}$$

[3] Throughout the paper, we use the notation $V[OP, x]$ to denote the objective function value of an optimization problem OP for a given feasible solution x and use the notation $V[\text{OP}]$ to denote the optimal value of OP.

The dual variable ρ can be eliminated from the CDNFP* problem since it appears in the objective function integrated with the transit capacity function a which is non-negative on $[0,T]$, and hence at an optimum solution each component of ρ should be made as large as possible. This observation implies that if we know optimal values for the dual variables π_i, $i \in N$, we can compute the optimal values for $\rho_{i,j}$ by

$$\rho_{ij}(t) = \min\left\{0, c_{ij}(t) - \pi_i(t) + \pi_j(t+\lambda_{i,j})\right\}, \quad (i,j) \in A,\ t \in [0,T]. \tag{9}$$

Summarizing the above discussion, the dual problem CDNFP* can be simplified as

$$\begin{aligned} \text{CDNFP}^* : \max \quad & \int_0^T r(t)^T \pi(t)\,dt - \int_0^T b(t)^T\,d\pi^{(-)}(t) \\ & + \int_0^T \sum_{(i,j)\in A} a_{i,j}(t) \min\left\{0, c_{ij}(t) - \pi_i(t) + \pi_j(t+\lambda_{i,j})\right\} dt \\ \text{s.t.} \quad & \pi \text{ of bounded variation and right continuous} \\ & \qquad \text{on } [0,T] \text{ with } \pi(T) = 0. \end{aligned}$$

The following result is now easily established.

Theorem 1 (Anderson and Philpott [8]) *Weak duality holds between CDNFP and CDNFP*.*

Proof Assume that x is feasible for CDNFP with corresponding storage y derived from (1) and π is feasible for CDNFP* with corresponding ρ given by (9). By integrating by parts, we have

$$\begin{aligned} &\int_0^T r(t)^T \pi(t)\,dt = -\int_0^T \bar{r}(t)^T\,d\pi(t) \\ &= -\int_0^T \sum_{i\in N} \left(\int_0^t \sum_{j:(i,j)\in A} x_{i,j}(s)\,ds - \int_0^t \sum_{j:(j,i)\in A} x_{j,i}(s-\lambda_{j,i})ds + y_i(t) \right) d\pi_i(t) \\ &= \int_0^T \sum_{(i,j)\in A} x_{i,j}(t)\pi_i(t)\,dt - \int_0^T \sum_{j:(j,i)\in A} x_{j,i}(s-\lambda_{j,i})\pi_i(t)\,dt - \int_0^T y(t)^T\,d\pi(t) \\ &= \int_0^T \sum_{(i,j)\in A} x_{i,j}(t)\left(\pi_i(t) - \pi_j(t-\lambda_{i,j})\right) dt - \int_0^T y(t)^T\,d\pi(t). \end{aligned}$$

Then by comparing the objective function values of CDNFP and CDNFP* for x and π, respectively, we obtain

$$\begin{aligned} & V[\text{CDNFP}, x] - V[\text{CDNFP}^*, \pi] \\ & \qquad = \int_0^T \sum_{(i,j)\in A} x_{i,j}(t)\left(c_{i,j}(t) - \rho_{i,j}(t) - \pi_i(t) + \pi_j(t-\lambda_{i,j})\right) dt \\ & \qquad - \int_0^T \rho(t)^T (a(t) - x(t))\,dt + \int_0^T y(t)^T\,d\pi^{(+)}(t) \\ & \qquad + \int_0^T (b(t) - y(t))^T\,d\pi^{(-)}(t). \end{aligned}$$

The result now follows by the fact that each of the above integrals is nonnegative due to feasibility of x, y for CDNFP and feasibility of π, ρ for CDNFP*. □

The weak duality result motivates the notion of complementary slackness for CDNFP as the following result.

Corollary 1 *Suppose that x, y is feasible for CDNFP and π, ρ is feasible for CDNFP***. If*

$$\int_0^T \sum_{(i,j)\in A} x_{i,j}(t)\left(c_{i,j}(t) - \rho_{i,j}(t) - \pi_i(t) + \pi_j(t - \lambda_{i,j})\right) dt = 0, \tag{10}$$

$$\int_0^T \rho(t)^T (a(t) - x(t))\, dt = 0, \tag{11}$$

$$\int_0^T y(t)^T\, d\pi^{(+)}(t) = 0, \tag{12}$$

$$\int_0^T (b(t) - y(t))^T\, d\pi^{(-)}(t) = 0, \tag{13}$$

then x and π are optimal for CDNFP and CDNFP**, respectively. Moreover, strong duality holds between CDNFP and CDNFP**.

By feasibility of x, y for CDNFP and π, ρ for CDNFP*, and also by using some basic results from real analysis, the integral equations (10)-(13) can be simplified to derive the notion of complementary slackness for CDNFP in an analogous manner to that described for static network flows.

Suppose that x with corresponding storage y is feasible for CDNFP and that π is a function of bounded variation on $[0, T]$. We say that the function π is *complementary slack* with x if the following conditions are met:

(CS1) if $c_{i,j}(t) - \pi_i(t) + \pi_j(t + \lambda_{i,j}) > 0$, then $x_{i,j}(t) = 0$;
(CS2) if $c_{i,j}(t) - \pi_i(t) + \pi_j(t + \lambda_{i,j}) < 0$, then $x_{i,j}(t) = a_{i,j}(t)$;
(CS3) if π_i is strictly increasing at t, then $y_i(t) = 0$;
(CS4) if π_i is strictly decreasing at t, then $y_i(t) = b_i(t)$.

We refer to the above conditions as *complementary slackness* conditions.

Lemma 3 (Complementary Slackness Optimality Conditions) *Let x be feasible for CDNFP and π be complementary slack with x. If π is feasible for CDNFP***, then x and π are optimal for CDNFP and CDNFP***, respectively.*

Proof Let ρ be given by (9) with respect to π. Then conditions (CS1) and (CS2) imply that

$$\begin{aligned} x_{i,j}(t)\left(c_{i,j}(t) - \rho_{i,j}(t) - \pi_i(t) + \pi_j(t - \lambda_{i,j})\right) &= 0, \\ \rho_{i,j}(t)(a_{i,j}(t) - x_{i,j}(t)) &= 0, \end{aligned}$$

for every arc $(i, j) \in A$ and all $t \in [0, T]$, and obviously the integral equations (10) and (11) are satisfied. Moreover, by Lemma 3.3 in [36], we can show that the conditions (CS3) and (CS4) imply that the integral equations (12) and (13) hold. The result now follows from Corollary 1. □

So far we have seen that weak duality holds between CDNFP and CDNFP* and that conditions (CS1)-(CS4) are sufficient for optimality. It is of great interest to conjecture whether a strong duality result can be established whereby $V[\text{CDNFP}] = V[\text{CDNFP}^*]$

and these values are attained in each program. As noted previously, the feasible region F of CDNFP is compact in the weak topology on $L^n_\infty([0,T])$ and this is sufficient to guarantee the existence of an optimal solution x, say, for CDNFP. Thus we are left with the task to prove the existence of a dual feasible solution π for which $V[\text{CDNFP}, x] = V[\text{CDNFP}^*, \pi]$. In general, strong duality may not hold, even for the special case that all transit times are zero (see [37] for some examples). However, we shall show that strong duality can be derived for CDNFP under the following assumptions.

Assumption 1 *The transit times λ are all rational, as is the time horizon T.*

Assumption 2 *The input functions a, b, c, and r are all piecewise analytic*[4] *on* $[0,T]$.

Assumptions 1 and 2 are supposed to hold throughout the rest of the paper, which guarantee the existence of a piecewise analytic optimal solution for CDNFP.

Theorem 2 (Pullan [38]) *If F is nonempty, then CDNFP has an optimal solution which is also piecewise analytic on* $[0,T]$.

3 Shortest Augmenting Paths

The basic approach to derive strong duality for CDNFP is to go along the same lines as in the static network flows. A key step of establishing strong duality for the static minimum cost flow problem is the fact that starting from some feasible flow we can construct a dual solution if the network contains no augmenting cycles with negative cost. More precisely, the shortest distance labels from one specified node to the other nodes in the residual network define a dual feasible solution which is complementary slackness with the given feasible flow. Here the concept of residual network as well as the notation of augmenting paths and cycles play a central role. We recall that the residual network has a backward arc for each original arc. The residual capacity of an original arc is defined as the difference between between the capacity and the flow on the arc and the residual capacity of a backward arc is defined as the flow on the original arc. The residual network contains only those arcs with positive residual capacities. The residual capacity of a path (it may contain backward arcs) is defined as the minimum of all residual capacity of the arcs in the path and a path is called an augmenting path if its residual capacity is positive. So we first need to find a similar characterization of augmenting paths for CDNFP.

For each arc $(i,j) \in A$ we create a *backward arc* (j,i). Notice that $(i,j) \in A$ implies $(j,i) \notin A$ due to the assumption that there is at most one arc between any pair of nodes in G. For each backward arc (j,i) with $(i,j) \in A$ we associate a transit time $\lambda_{j,i} := -\lambda_{i,j}$ and a cost function $c_{j,i}(t) := -c_{i,j}(t - \lambda_{i,j})$, $t \in [0,T]$. We denote the set of all backward arcs by $\overleftarrow{A}$ and let $\overleftrightarrow{A} := A \cup \overleftarrow{A}$.

Following Philpott [31], we use the term *node-time pair* (NTP) to refer to a particular node at a particular time instance, i.e., a member of $N \times [0,T]$. We say that

[4] A function $f : [0,T] \to \mathbb{R}$ is said to *piecewise analytic* if there exists a partition $\{t_0, t_1, \dots, t_m\}$ of $[0,T]$, $\epsilon > 0$, and g_k analytic on $(t_{k_1} - \epsilon, t_k + \epsilon)$ with $g_k(t) = f(t)$ for $t \in [t_{k_1}, t_k)$, $k = 1, \dots, m$. It follows from this definition that a piecewise analytic function is right-continuous but not necessarily left-continuous, and in particular may be discontinuous at a finite number of points.

NTP (i,α) is *arc-linked* to NTP (j,β) if $(i,j) \in \overleftrightarrow{A}$ and $\beta = \alpha + \tau_{i,j}$. We also say that NTP (i,α) is *node-linked* to NTP (j,β) if $i = j$. In this case, it is assumed that $\alpha \neq \beta$. A *continuous-time dynamic walk* from NTP (i,α) to NTP (j,β) is defined as a sequence of NTPs as

$$P : (i,\alpha) = (i_1,t_1),(i_2,t_2),\ldots,(i_q,t_q) = (j,\beta),$$

with consecutive members either arc- or node-linked. Here it is supposed that $i_k \neq i_{k+1}$ if $i_{k-1} = i_k$ for $k = 2,\ldots,q-1$. The sequence P is called a *continuous-time dynamic path* if all NTPs are distinct and is called a *continuous-time dynamic cycle* if $q \geq 3$, $(i,\alpha) = (j,\beta)$, and all other NTPs are distinct. For reasons of brevity, hereafter, the term "continuous-time dynamic" is omitted when referring to a continuous-time dynamic walk, path, or cycle.

Let $P : (i_1,t_1),\ldots,(i_q,t_q)$ be a path (or cycle) and $Q : (i_\ell,t_\ell),\ldots,(i_r,t_r)$ be a subsequence of consecutive NTPs in P. We shall refer to Q as an *arc-subpath* of P if any pair of consecutive NTPs in Q are arc-linked, i.e., $(i_k,i_{k+1}) \in \overleftrightarrow{A}$ for $k = \ell,\ldots,r-1$. In this case, Q can be seen as the sequence $(i_\ell,i_{\ell+1}),\ldots,(i_{r-1},i_r)$ of arcs in $\overleftrightarrow{A}$ together with starting time t_ℓ from node i_ℓ. If in addition, $i_{\ell-1} = i_\ell$ or $i_\ell = i_1$ and $i_r = i_{r+1}$ or $i_r = i_q$, then Q is called a *maximal arc-subpath* of P. Assume that Q is an arc-subpath of P. For a point in time $\alpha \in [0,T]$, we define a path $P|_Q(\alpha)$ as

$$P|_Q(\alpha) \ : (i_1,t_1),\ldots,(i_{\ell-1},t_{\ell-1}),(i_\ell,\alpha_\ell),\ldots,(i_r,\alpha_r),(i_{r+1},t_{r+1})\ldots,(i_q,t_q) \tag{14}$$

where $\alpha_\ell := \alpha$ and $\alpha_{k+1} := \alpha_k + \lambda_{i_k,i_{k+1}}$ for $k = \ell,\ldots,r-1$. Roughly speaking, $P|_Q(\alpha)$ is constructed from P by changing the starting time of arc-subpath Q from t_k to α.

Suppose that $P : (i_1,t_1),\ldots,(i_q,t_q)$ is a path (or cycle) from NTP (i,α) to NTP (j,β). For each arc $(i,j) \in \overleftrightarrow{A}$, we let $\mathbf{v}^P_{i,j}$ denote the corresponding *incidence vector* whose entries are the times that arc (i,j) is used in P. The incidence vector $\mathbf{v}^P_{i,j}$ is defined to be empty (of length 0) if arc (i,j) is not used at any point in time along P. Notice that the entries of $\mathbf{v}^P_{i,j}$ are ordered according to the times at which arc (i,j) appears along path P. Thus the path P can be identified by a family $\left\{\mathbf{v}^P_{i,j} \mid (i,j) \in \overleftrightarrow{A}\right\}$ of incidence vectors. For each $\epsilon > 0$, the ϵ*-neighborhood* $N(P,\epsilon)$ of P is defined as the set of all paths P' from (i,α) to (j,β) for which

$$|\mathbf{v}^P_{i,j}| = |\mathbf{v}^{P'}_{i,j}| \qquad \text{and} \qquad ||\mathbf{v}^P_{i,j} - \mathbf{v}^{P'}_{i,j}||_\infty < \epsilon \qquad \forall (i,j) \in \overleftrightarrow{A}.$$

Notice that for a vector $\mathbf{v} = (\mathbf{v}_1,\ldots,\mathbf{v}_m) \in \mathbb{R}^m$, the notations $|\mathbf{v}|$ and $||\mathbf{v}||_\infty$ denote the length and infinity-norm of $\mathbf{v}$, respectively, i.e., $|\mathbf{v}| := m$ and $||\mathbf{v}||_\infty := \max\{|\mathbf{v}_1|,\ldots,|\mathbf{v}_m|\}$.

The ϵ-neighborhood of P can be characterized in another way. Let Q be an arc-subpath of P with starting time t. It is then easy to see that $P|_Q(\alpha)$ is contained in $N(P,\epsilon)$ if $|t-\alpha| < \epsilon$. In fact, $N(P,\epsilon)$ contains those paths that can be obtained from P by changing the starting time of some arc-subpath of P at most by ϵ.

Here we introduce the concept of augmenting paths and cycles. Given a (piecewise analytic) flow x with corresponding storage y, the *residual capacity* of a path $P : (i_1,t_1),\ldots,(i_q,t_q)$ is defined as

$$\text{cap}(P) := \min\{\delta_1,\ldots,\delta_{q-1}\},$$

where for $k = 1, \dots, q-1$

$$\delta_k := \begin{cases} a_{i_k,i_{k+1}}(t_k) - x_{i_k,i_{k+1}}(t_k) & \text{if } (i_k, i_{k+1}) \in A, \\ x_{i_{k+1},i_k}(t_{k+1}) & \text{if } (i_k, i_{k+1}) \in \overleftarrow{A}, \\ \min\{b_{i_k}(t) - y_{i_k}(t) \mid t_k \le t \le t_{k+1}\} & \text{if } i_k = i_{k+1},\ t_k < t_{k+1}, \\ \min\{y_{i_k}(t) \mid t_{k+1} \le t \le t_k\} & \text{if } i_k = i_{k+1},\ t_{k+1} < t_k. \end{cases}$$

The value $\text{cap}(P)$ gives the maximum additional flow rate that can be pushed through P without violating the feasibility of x. The path P is called an *augmenting path* under x if for each $\epsilon > 0$, $N(P, \epsilon)$ contains some path P' with positive residual capacity. In other words, a path P is an augmenting path if for each $\epsilon > 0$ we can send an additional flow along a path in $N(P, \epsilon)$. Notice that each path with positive residual capacity is an augmenting path. However, we might have some augmenting path with zero residual capacity (see Example 1). In the same way, we can define an augmenting cycle.

Next we want to define the cost of an augmenting path $P : (i_1, t_1), \dots, (i_q, t_q)$. To do this in a reasonable way, we first observe that for $k = 1, \dots, q-1$ the following holds:

(i) if $(i_k, i_{k+1}) \in A$, then $a_{i_k,i_{k+1}} - x_{i_k,i_{k+1}}$ is not identically zero on any open interval containing t_k,
(ii) if $(i_k, i_{k+1}) \in \overleftarrow{A}$, then x_{i_{k+1},i_k} is not identically zero on any open interval containing t_{k+1},
(iii) if $i_k = i_{k+1}$ and $t_k < t_{k+1}$, then $y_{i_k}(t) < b_{i_k}(t)$ for each $t \in (t_k, t_{k+1})$,
(iv) if $i_k = i_{k+1}$ and $t_{k+1} < t_k$, then $y_{i_k}(t) > 0$ for each $t \in (t_{k+1}, t_k)$.

In particular, if $Q\ :\ (i_\ell, t_\ell), \dots, (i_r, t_r)$ is a maximal arc-subpath of P, then $P|_Q(\alpha)$ is also an augmenting path for each α in $(t_\ell - \epsilon, t_\ell)$ or $(t_\ell, t_\ell + \epsilon)$ for some sufficiently small $\epsilon > 0$. Depending on whether $P|_Q(\alpha)$ is an augmenting path for each α in $(t_\ell - \epsilon, t_\ell)$, $(t_\ell, t_\ell + \epsilon)$, or $(t_\ell - \epsilon, t_\ell + \epsilon)$, we define the cost $c(Q)$ of Q as

$$c(Q) := \begin{cases} \sum_{k=\ell}^{r-1} c_{i_k,i_{k+1}}(t_k-) & \text{if } \alpha \text{ in } (t_\ell - \epsilon, t_\ell), \\ \sum_{k=\ell}^{r-1} c_{i_k,i_{k+1}}(t_k+) & \text{if } \alpha \text{ in } (t_\ell, t_\ell + \epsilon), \\ \sum_{k=\ell}^{r-1} \min\{c_{i_k,i_{k+1}}(t_k-), c_{i_k,i_{k+1}}(t_k+)\} & \text{if } \alpha \text{ in } (t_\ell - \epsilon, t_\ell + \epsilon). \end{cases} \tag{15}$$

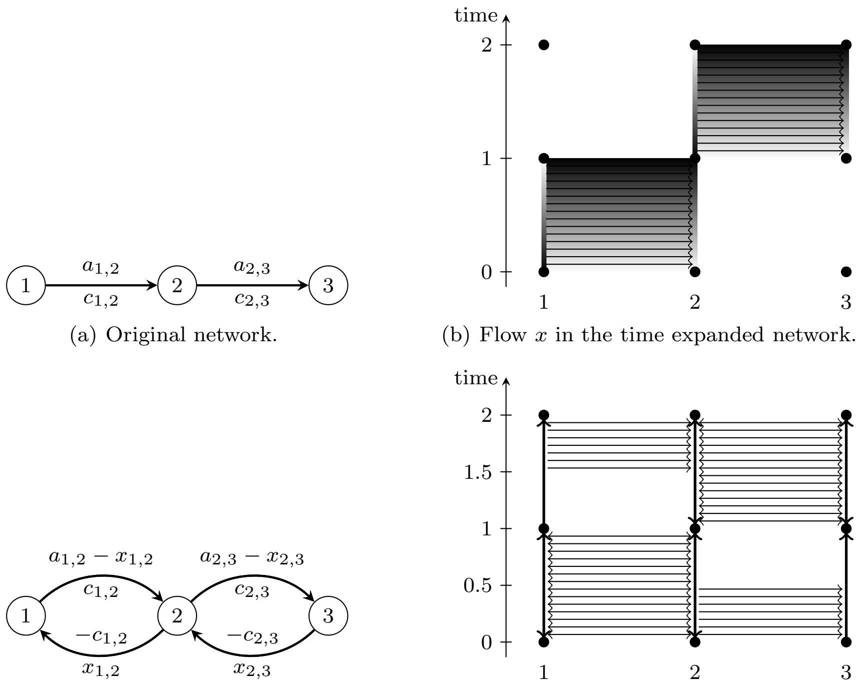
Notice that $c_{i_k,i_{k+1}}(t_k-)$ and $c_{i_k,i_{k+1}}(t_k+)$ denote the limit of $c_{i_k,i_{k+1}}$ at t_k from the left and from the right, respectively, i.e.,

$$c_{i_k,i_{k+1}}(t_k-) := \lim_{t \to t_k^-} c_{i_k,i_{k+1}}(t) \qquad \text{and} \qquad c_{i_k,i_{k+1}}(t_k+) := \lim_{t \to t_k^+} c_{i_k,i_{k+1}}(t).$$

The *cost* $c(P)$ of P is then defined as $c(P) := \sum_Q c(Q)$, where the sum is taken over all maximal arc-subpaths Q of P. An augmenting path P from (i, α) to (j, β) is said to be a *shortest augmenting path* if it has the minimum cost among all augmenting paths from (i, α) to (j, β). Similarly, the cost of an augmenting cycle is defined. An augmenting cycle is called a *negative augmenting cycle* if its cost is negative.

We can define the cost of an augmenting path (or cycle) $P : (i_1, t_1), \dots, (i_q, t_q)$ as the sum of the costs of the arcs at the times when they appear along P, i.e.,

$$\text{cost}(P) := \sum_{k:(i_k,i_{k+1}) \in \overleftrightarrow{A}} c_{i_k,i_{k+1}}(t_k).$$



(a) Original network.

(b) Flow x in the time expanded network.

(c) Residual network.

(d) Residual network in time expanded network.

Fig. 1 Network for Example 1.

Here, the index k varies from 1 to $q-1$. We notice that $\text{cost}(P)$ is equal to $c(P)$ for the case that the cost functions are continuous, but in general it is not the case. However, we will show that the network contains a negative augmenting cycle if and only if there is a cycle W with $\text{cap}(W) > 0$ and $\text{cost}(W) < 0$.

The following example illustrates the idea of augmenting paths and cycles.

Example 1 We consider the network shown in Fig. 1(a). The transit costs and transit capacities are as follows:

$$c_{1,2}(t) = \begin{cases} 1 & 0 \le t < 0.5, \\ 2 & 0.5 \le t < 1, \\ 1 & 1 \le t \le 2, \end{cases} \qquad c_{2,3}(t) = \begin{cases} 1 & 0 \le t < 1, \\ 2 & 1 \le t < 1.5, \\ 1 & 1.5 \le t \le 2, \end{cases}$$

$$a_{1,2}(t) = \begin{cases} 4t & 0 \le t < 1, \\ 0 & 1 \le t < 1.5, \\ 1 & 1.5 \le t \le 2, \end{cases} \qquad a_{2,3}(t) = \begin{cases} 1 & 0 \le t < 0.5, \\ 0 & 0.5 \le t < 1, \\ 4t-2 & 1 \le t \le 2. \end{cases}$$

The storage capacities are given as

$$b_1(t) = \infty, \qquad b_2(t) = 1, \qquad b_3(t) = \infty, \qquad t \in [0, 2].$$

The transit times and storage costs are assumed to be zero. The problem is to send an initial storage of one unit from node 1 to node 3 within the time interval $[0, 2]$.

One possible solution x is obtained as follows. We send flow into arc $(1,2)$ with rate $2t$ within the interval $[0,1)$. The flow arriving at node 2 is stored there till time 1. So there will be one unit of flow at node 2 at time 1. We send this amount of flow into arc $(2,3)$ with rate $2t-2$ within the interval $[1,2]$. Fig. 1(b) shows the flow x in the corresponding time-expanded network. Formally, x is given by

$$x_{1,2}(t) = \begin{cases} 2t & t \in [0,1), \\ 0 & t \in [1,2), \end{cases} \qquad x_{2,3}(t) = \begin{cases} 0 & t \in [0,1), \\ 2t-2 & t \in [1,2], \end{cases}$$

with corresponding storage

$$y_1(t) = \begin{cases} 2t & t \in [0,1), \\ 0 & t \in [1,2), \end{cases} \quad y_2(t) = \begin{cases} 0 & t \in [0,1), \\ 2t-2 & t \in [1,2), \end{cases} \quad y_3(t) = \begin{cases} 0 & t \in [0,1), \\ 2t-2 & t \in [1,2). \end{cases}$$

We are now interested in identifying the augmenting paths and augmenting cycles. Fig. 1(c) depicts the network with backward arcs and Fig. 1(d) depicts the paths and cycles with positive residual capacities in the corresponding time-expanded network. However, there are more augmenting paths and cycles in addition to those shown in Fig. 1(d), whose residual capacities are zero. Some of them are given below

$$\begin{aligned}
P_1 :& (1,0),(2,0),(3,0),(3,2), \\
P_2 :& (1,0),(2,0),(2,0.5),(3,0.5),(3,2), \\
P_3 :& (1,0),(1,0.5),(2,0.5),(3,0.5)(3,2), \\
P_4 :& (1,0),(1,1.5),(2,1.5),(2,2),(3,2), \\
P_5 :& (1,0),(1,2),(2,2),(3,2) \\
W_1 :& (1,0),(1,2),(2,2),(3,2),(3,1),(2,1),(2,0),(1,0), \\
W_2 :& (1,1),(1,2),(2,2),(3,2),(3,1.5),(2,1.5),(2,1),(1,1), \\
W_3 :& (1,0.5),(1,2),(2,2),(3,2),(3,1.5),(2,1.5),(2,0.5),(1,0.5),
\end{aligned}$$

with costs

$$\begin{aligned}
c(P_1) &= 2, & c(P_2) &= 2, & c(P_3) &= 2, & c(P_4) &= 2, & c(P_5) &= 2, \\
\operatorname{cost}(P_1) &= 2, & \operatorname{cost}(P_2) &= 1, & \operatorname{cost}(P_3) &= 2, & \operatorname{cost}(P_4) &= 2, & \operatorname{cost}(P_5) &= 2,
\end{aligned}$$

and

$$\begin{aligned}
c(W_1) &= -1, & c(W_2) &= -2, & c(W_3) &= -2, \\
\operatorname{cost}(W_1) &= -1, & \operatorname{cost}(W_2) &= 1, & \operatorname{cost}(W_3) &= -1.
\end{aligned}$$

We observe that the equality $c(P) = \operatorname{cost}(P)$ does not hold for some path or cycle P.

As mentioned already above, an augmenting path (or cycle) must satisfy the conditions (i)-(iv). But the other direction may not hold, that is, a path satisfying these conditions is not necessarily an augmenting path in general. The following paths and cycles show this fact:

$$\begin{aligned}
P_6 :& (1,0),(2,0),(2,1),(3,1),(3,2), \\
P_7 :& (1,0),(1,1),(2,1),(3,1),(3,2), \\
W_4 :& (1,1),(1,2),(2,2),(3,2),(3,1),(2,1),(1,1).
\end{aligned}$$

In what follows, we consider NTP $(1,0)$ as the source and investigate the existence of shortest augmenting paths from NTP $(1,0)$ to every NTP (i,t). For ease of notation, we assume that the network G contains an augmenting path from NTP $(1,0)$ to every other NTP. Notice that this assumption imposes no loss of generality since this is satisfied, if necessary, by introducing an artificial storage node s and adding artificial arcs (s,i) joining node s to node i for each node $i \in N$. The artificial node s has a large initial storage, a large cost and an infinite capacity, and each artificial arc (s,i) has a zero transit time, a large cost, and an infinite capacity. It is clear that no artificial arc would appear in a shortest augmenting path from $(1,0)$ to any NTP (i,t) unless network G contains no augmenting path from $(1,0)$ to (i,t) without artificial arcs.

The problem of determining shortest augmenting paths is closely related to the continuous-time dynamic shortest path problem for which transit times can take negative values. This problem is already studied by Koch and Nasrabadi [22]. They show that the dynamic shortest paths may not exist in general, particularly if transit times are irrational or cost functions have an infinite number of extrema (see [22, Section 4.1] for a detailed discussion on this subject). However, they prove the existence of dynamic shortest paths if the cost functions are piecewise analytic and transit times are rational. In the following, we use the same techniques as in [22] to show that shortest augmenting paths from NTP $(1,0)$ to every NTP (i,t) exist under Assumptions 1 and 2.

Throughout the rest of this section, we fix a feasible flow x with corresponding storage y and suppose that $P : (i_1,t_1),(i_2,t_2),\dots,(i_q,t_q)$ is an augmenting path from NTP $(1,0)$ to NTP (n,T). Due to Theorem 2, the flow x is assumed to be piecewise analytic. The path P is said to be a *local shortest augmenting path* if there exists an $\epsilon > 0$ such that $c(P) \le c(P')$ for all augmenting paths P' in the ϵ-neighborhood of P. A local shortest augmenting path can be characterized in another way in terms of its arc-subpaths. To this end, we need some definitions and lemmas.

Let f be a real-valued function defined on $[0,T]$. The *support* of f, denoted by supp(f), is defined as the set of all points $t \in [0,T]$ for which f is not identically zero on any open interval containing t. For the case that f is a piecewise analytic function, supp(f) can be expressed as a finite union of disjoint closed intervals.

Lemma 4 *Let $f : [0,T] \to \mathbb{R}$ be a piecewise analytic function. If f is not identically zero, then supp(f) is a finite union of disjoint closed intervals.*

Proof We know, by the definition of piecewise analytic functions, that there exist a partition $\{t_0,t_1,\dots,t_m\}$ of $[0,T]$, a real value $\epsilon > 0$, and analytic functions g_k on $(t_{k_1} - \epsilon, t_k + \epsilon)$ with $g_k(t) = f(t)$ for $t \in [t_{k_1}, t_k)$, $k = 1,\dots,m$. Moreover, it is well known that the Lebesgue-measure of zero set of a nonzero analytic function is zero. Thus, for each $k = 1,\dots,m$, f is either identically zero or has a zero set of Lebesgue-measure zero on the interval $[t_{k_1}, t_k)$, implying $(t_{k_1}, t_k) \subseteq [0,T] \setminus \text{supp}(f)$ or $[t_{k_1}, t_k] \subseteq \text{supp}(f)$, respectively. This establishes the desired result. □

Now let $Q : (i_\ell,t_\ell),\dots,(i_r,t_r)$ be an arc-subpath of P. It is not difficult to see that a new augmenting path under x can be constructed from P by slightly changing the starting time t_ℓ of Q. More precisely, there exists an (inclusion-wise) maximal closed interval $[u,v]$, say, containing t_α so that the path $P|_Q(\alpha)$, given by (14), is an augmenting path for every $\alpha \in [u,v]$. Here the term "maximal closed interval" means that there is no closed interval $[u',v']$ strictly containing $[u,v]$ so that $P|_Q(\alpha)$ is an augmenting path for each $\alpha \in [u',v']$. The interval $[u,v]$ can be found in the following way.

We define a function $f : [0,T] \to \mathbb{R}_{\geq 0}$ as

$$f(t) := \min_{\ell \leq k \leq r-1} \left\{ f_k \left(t + \sum_{p=\ell}^{k-1} \lambda_{i_p, i_{p+1}} \right) \right\},$$

where for $k = \ell, \ldots, r-1$, f_k is a real-valued function on $[0,T]$ given by

$$f_k(t) := \begin{cases} a_{i_k,i_{k+1}}(t) - x_{i_k,i_{k+1}}(t) & \text{if } (i_k, i_{k+1}) \in A, \\ x_{i_{k+1},i_k}(t) & \text{if } (i_k, i_{k+1}) \in \overleftarrow{A}, \end{cases}$$

Notice that for each $t \in [0,T]$, the value $f(t)$ represents the maximum additional flow rate that can be sent from node i_ℓ to node i_r along path Q at time t. Further, we define two more functions $g_\ell, g_r : [0,T] \to \mathbb{R}_{\geq 0}$ as

$$g_\ell(t) := \begin{cases} b_{i_\ell}(t) - y_{i_\ell}(t) & \text{if } t \geq t_\ell, \\ y_{i_\ell}(t) & \text{if } t \leq t_\ell, \end{cases}$$

$$g_r(t) := \begin{cases} y_{i_r}(t + \lambda_Q) & \text{if } t \geq t_\ell, \\ b_{i_r}(t + \lambda_Q) - y_{i_r}(t + \lambda_Q) & \text{if } t \leq t_\ell. \end{cases}$$

where λ_Q is the transit time of arc-subpath Q, that is,

$$\lambda_Q := \sum_{\ell \leq k \leq r-1} \lambda_{i_k, i_{k+1}}.$$

For each $t \in [0,T]$ the value $g_\ell(t)$ gives an upper bound on the amount of flow that can be increased or decreased from the stored flow at node i_ℓ at time t, depending on whether $t > t_\ell$ or $t \leq t_\ell$, respectively. A similar interpretation holds for the value $g_\ell(t)$ for each $t \in [0,T]$.

The function f is not identically zero and in particular, t_ℓ is a member of supp(f) since P is an augmenting path. However, t_ℓ may not be a member of supp(g_ℓ) or supp(g_r). If it is the case, then setting $[u,v] := [t_\ell, t_\ell]$ leads to the desired interval. Now we consider the case that $t_\ell \in \text{supp}(g_\ell) \cap \text{supp}(g_r)$. In this case, there exists a maximal closed interval $[u^*, v^*]$ containing t_ℓ such that $g_\ell(t) > 0$ and $g_r(t) > 0$ for each $t \in (u^*, v^*)$. On the other hand, it follows from Lemma 4 that supp(f) can be expressed as $\bigcup_{k \in I} [u_k, v_k]$, where I is a finite set of indices and $[u_k, v_k] \cap [u_{k'}, v_{k'}] = \emptyset$ for each $k, k' \in I$ with $k \neq k'$. The fact that P is an augmenting path implies $t_\ell \in \text{supp}(f)$ and consequently $t_\ell \in [u_k, v_k]$ for some $k \in I$. We now define $[u,v] := [u_k, v_k] \cap [u^*, v^*]$. It is now easy to see that $P|_Q(\alpha)$ is an augmenting path for every $\alpha \in [u,v]$ and moreover, there is no closed interval $[u', v']$ strictly containing $[u,v]$ so that $P|_Q(\alpha)$ is an augmenting path for each $\alpha \in [u', v']$.

So far, we have proved the existence of a maximal interval $[u,v]$ for which the path $P|_Q(\alpha)$ is an augmenting path for all $\alpha \in [u,v]$. We now define a cost function $c_Q : [u,v] \to \mathbb{R}$ with respect to Q as

$$c_Q(\alpha) := \begin{cases} \sum_{k:(i_k,i_{k+1}) \in A} c_{i_k,i_{k+1}}(\alpha_k -) & \alpha = v, \\ \sum_{k:(i_k,i_{k+1}) \in A} \min\{c_{i_k,i_{k+1}}(\alpha_k -), c_{i_k,i_{k+1}}(\alpha_k +)\} & \alpha \in (u,v), \\ \sum_{k:(i_k,i_{k+1}) \in A} c_{i_k,i_{k+1}}(\alpha_k +) & \alpha = u, \end{cases} \tag{16}$$

where the index k varies from ℓ to $r-1$. We recall that $\alpha_\ell = \alpha$ and $\alpha_{k+1}(\alpha) = \alpha_k + \lambda_{i_k,i_{k+1}}$ for $k = \ell, \ldots, r-1$. For the case that $u = v = t_\ell$, we define $c_Q(\alpha) := c(Q)$, where $c(Q)$ is given by (15). The function c_Q is lower semi-continuous at any point $\alpha \in [u,v]$ and such a function attains its local minimum on a closed interval. We shall use this fact later on to prove the existence of shortest augmenting paths.

It is straightforward that the cost function c_Q has a local minimum on $[u,v]$ at the point t_ℓ if P is a local shortest augmenting path. Conversely, if for each arc-subpath $Q : (i_\ell, t_\ell), \ldots, (i_r, t_r)$ of P, then c_Q attains a local minimum at the point t_ℓ within the interval $[u,v]$, then P is a local shortest augmenting path. Thus we have established the following lemma, which gives an alternative characterization of local shortest augmenting paths.

Lemma 5 *The path P is a local shortest augmenting path if and only if for each arc-subpath Q of P with starting time t_ℓ the cost function c_Q, given by* (16)*, has a local minimum at the point t_ℓ.*

In what follows, let $\mathcal{P}_{\text{loc}}$ be the set of all augmenting paths P from NTP $(1,0)$ to NTP (n,T) such that for each maximal arc-subpath Q of P with starting time t the function c_Q, given by (16), has a local minimum at t and is not constant on any open neighborhood containing t. Further, we assume that two paths P_1 and P_2 are identified if they differ only in the starting time t_1 and t_2 $(t_1 < t_2)$, respectively, of one common arc-subpath Q and c_Q is constant over $[t_1,t_2]$. Note that in this case $P1$ and P_2 have the same cost, i.e., $c(P_1) = c(P_2)$. Then, for each local shortest augmenting path, one augmenting path with the same cost is contained in $\mathcal{P}_{\text{loc}}$. Hence, the following lemma shows that the set of local shortest augmenting paths from NTP $(1,0)$ to NTP (n,T) is finite.

Lemma 6 *The set $\mathcal{P}_{loc}$ is finite.*

Proof Let $\hat\lambda$ be the greatest common factor of the transit times and so

$$\hat\lambda = \min_{S>0} \left\{ S \text{ is a finite sum of transit times } \lambda_{i,j}, (i,j) \in \overleftrightarrow{A} \right\}.$$

Note that $\hat\tau$ exists and is greater than zero because of Assumption 1. Thus each arc $(i,j) \in \overleftrightarrow{A}$ can appear at most $\left\lceil \frac{T}{\hat\lambda} \right\rceil$ times in any arc-suppath of an arbitrary path. In other words, every arc-subpath of any augmenting path contains at most $\left\lceil \frac{T}{\hat\lambda} \right\rceil \cdot |A|$ arcs. Consequently, the number of possible maximal arc-subpaths is bounded by a constant where two arc-subpaths that differ by the starting time are identified. We now assume by contradiction that the cardinality of $\mathcal{P}_{\text{loc}}$ is infinite. Hence there exists an infinite number of paths in $\mathcal{P}_{\text{loc}}$ all containing the same maximal arc-subpath Q, say, but with different starting times. It then follows from Lemma 5 that the cost function c_Q, given by (16), has an infinite number of local minimum points. This is a contradiction because c_Q is a piecewise analytic function and has only a finite number of local extrema. This establishes the lemma.

□

The next lemma shows that $\mathcal{P}_{\text{loc}}$ contains the shortest augmenting path from NTP $(1,0)$ to NTP (n,T).

Lemma 7 *Let P be an augmenting path from NTP $(1,0)$ to NTP (n,T). Then there exists an augmenting path $P' \in \mathcal{P}_{loc}$ with $c(P') \leq c(P)$.*

Proof If $P \in \mathcal{P}_{\text{loc}}$, then we are done. So we consider the case where P is not in $\mathcal{P}_{\text{loc}}$. In this case we iteratively apply the following procedure to construct an augmenting path $P' \in \mathcal{P}_{\text{loc}}$:

(i) Let $Q : (i_k, t_k), \ldots, (i_r, t_r)$ be a maximal arc-subpath of P such that the cost function c_Q does not have a local minimum at t_k or is constant on an open interval containing t. Notice that such a arc-subpath exists because of the definition of $\mathcal{P}_{\text{loc}}$ and the fact that P is not in $\mathcal{P}_{\text{loc}}$. Further, choose P' such that it contains a minimal number of arcs.
(ii) The function c_Q is also lower semi-continuous. Thus it takes its minimum over $[u, v]$ at some point t. If it has several local minimum, then choose t to be the one with maximum value.
(iii) Let $P|_Q(t)$ be the augmenting path from NTP $(0, 1)$ to NTP (n, T) obtained from P by shifting the arc-subpath Q by $t_k - t$ time units. Since $P|_Q(t)$ may contain continuous-time dynamic cycles, we delete all of them in $P|_Q(t)$.
(iv) Set $P := P|_Q(t)$. If P is not in $\mathcal{P}_{\text{loc}}$, then go to (i).

The above procedure terminates after a finite number of iterations and the resulting augmenting path P is contained in $\mathcal{P}_{\text{loc}}$. Further, in each iteration the cost of P does not increase which proves the lemma. □

As a consequent of Lemmas 6 and 7, we conclude that a shortest augmenting path from NTP $(1, 0)$ to NTP (n, T) exists, that is the one in $\mathcal{P}^K$ with minimum cost. Further, Lemma 6 as well as Lemma 7 remain true if NTP (n, T) is replaced by every other NTP (i, t). This leads to the main result of this section.

Theorem 3 *Suppose that x is a piecewise analytic solution for CDNFP. For each NTP (i, t) let $d_i(t)$ be the cost of a shortest augmenting path from $(0, 1)$ to (i, t). Then, for each node $i \in N$, the label $\tau_i(t)$ exists for all $t \in [0, T]$ and the function $\tau_i : [0, T] \to \mathbb{R}$ is piecewise analytic.*

Proof The existence of $\tau_i(t)$ follows from 6 and 7 for each NTP (i, t). It thus remains to show that τ_i is piecewise analytic on $[0, T]$ for each $i \in N$. In the following we fix a node $i \in N$. Similar to the definition of $\mathcal{P}_{\text{loc}}$ define $\mathcal{P}_{\text{loc}}(t)$ as the set of augmenting paths P from $(1, 0)$ to (i, t) such that for each maximal arc-subpath Q of P with starting time $\bar{t}$ the function c_Q has a local minimum at $\bar{t}$ and is not constant on any open neighborhood containing $\bar{t}$. Then $\mathcal{P}_v := \cup_{t \in [0,T]} \mathcal{P}_{\text{loc}}(t)$ contains (nearly) all shortest augmenting paths for any point in time $\theta \in [0, T]$.

Next we define an equivalence relation $\sim$ on $\mathcal{P}_v$. let $P : (i_1, t_1), \ldots, (i_q, t_q)$ and $P' : (i'_1, t'_1), \ldots, (i'_{q'}, t'_{q'})$ be two members of $\mathcal{P}_v$. We define $\sim$ on $\mathcal{P}_v$ by $P \sim P'$ if and only if $q = q'$ and there is some $r \in \{1, \ldots, q-1\}$ such that

(i) $(i'_k, t'_k) = (i'_k, t'_k)$ for each $k \leq r$, $i_r = i_{r+1} = i'_r = i'_{r+1}$ and $t_{r+1} \neq t'_{k+1}$,
(ii) $i_k = i'_k$ for each $k \geq r + 1$ and the NTP sequences $(i_{r+1}, t_{r+1}), \ldots, (i_q, t_q)$ and $(i'_{r+1}, t'_{r+1}), \ldots, (i'_{q'}, t'_{q'})$ are arc-subpaths of P and P', respectively.

Roughly speaking, P and P' are equivalent if they differ only in the starting time of the last maximal arc-subpath. For an equivalence class $[P]$ we denote by P_1 the path consisting of the first r NTPs of P and by P_2 the arc-path consisting of the last $q - r + 1$ NTPs of P. Note that P_1 and P_2 can be the empty path. Further, P_1 and P_2 are well-defined in the sense that they are coincide for any member of $[P]$. On the other hand, any augmenting path in $[P]$ is obtained by concatenating P_1 and P_2 and changing the starting time of P_2 (if P is an arc-path we put it in the equivalence class $P_1 = \emptyset$ and $P_2 = P$).

We now consider the quotient set $\mathcal{P}_v/\sim$ and an equivalence class $[P] \in \mathcal{P}_v/\sim$. Then each maximal arc-subpath Q of P_1 and the maximal arc-subpath P_2 locally minimizes $c_{P'}$. Hence, along the same lines as in the proof of Lemma 6 we obtain that there exists only a finite number of possibilities for P_1 and P_2. Hence, $\mathcal{P}_v/\sim$ is a finite set. In order to get an expression for τ_i we define a cost function $c_{[P]} : [0,T] \to \mathbb{R}$ by

$$c_{[P]}(t) := c(P_1) + \begin{cases} c_{P_2}(t - \lambda_{P_2}) & \text{if } t > \lambda_{P_1} + \lambda_{P_2}, \\ \infty & \text{if } t \leq \lambda_{P_1} + \lambda_{P_2}, \end{cases}$$

Then, for every $P \in \mathcal{P}_v$ we have $c(P) = c_{[P]}(t)$ where t is the last time that we reach node i along P, i.e., (i,t) is the last NTP of P. Thus we obtain $\tau_i = \min\{c_{[P]}\}$. Therefore τ_i is piecewise analytic since it is the minimum of a finite number of piecewise analytic functions. □

4 Optimality conditions and strong duality

In this section we return to the optimality conditions for CDNFP. In particular we show that not only conditions (CS1)-(CS4) are sufficient for optimality, but also are necessary under Assumptions 1 and 2. Furthermore, we develop more necessary and sufficient conditions for optimality and derive a strong duality result between CDNFP and CDNFP*.

We consider a feasible flow x and suppose that $W : (i_1,t_1),\ldots,(i_q,t_q)$ is an augmenting cycle. We have defined the cost of W in two different ways: $c(W)$ in terms of the cost of arc-subpaths of W and $\mathrm{cost}(W)$ as the sum of the costs of the arcs at the times they appear around the cycle W. Further, we have observed that these two values are not equal in general. However, we have the following result.

Lemma 8 *Let x be a piecewise analytic flow. The network G contains a negative augmenting cycle if and only if there is a cycle W with cap$(W) > 0$ and cost$(W) < 0$.*

Proof Suppose first that W is a cycle with $\mathrm{cap}(W) > 0$ and $\mathrm{cost}(W) < 0$. Clearly W is an augmenting cycle since $\mathrm{cap}(W) > 0$. So we need to show that $c(W) < 0$. Recall that $c(W) = \sum_Q c(Q)$ where sum is taken over all maximal arc-subpaths Q of W and $c(Q)$ is computed by (15). For each maximal arc-subpath $Q : (i_\ell,t_\ell),\ldots,(i_r,t_r)$ of W, we define $\mathrm{cost}(Q) := \sum_{k:(i_k,i_{k+1})\in \overleftrightarrow{A}} c_{i_k,i_{k+1}}(t_k)$ where the index k varies from ℓ to $r-1$. The fact that $\mathrm{cap}(W) > 0$ implies that there exists some $\epsilon > 0$ so that $W|_Q(\alpha)$ is an augmenting cycle for each $\alpha \in (t_\ell, t_\ell+\epsilon)$. Due to the definition of $c(Q)$ and $\mathrm{cost}(Q)$ and the fact that cost functions are right-continuous, we can conclude $c(Q) \leq \mathrm{cost}(Q)$. Therefore, $c(W) \leq \mathrm{cost}(W)$ which gives the result in one direction.

To prove the other direction, suppose that W is a negative augmenting cycle. Let $Q : (i_\ell,t_\ell),\ldots,(i_r,t_r)$ be a maximal arc-subpath of W. Then we know that there is some $\epsilon > 0$ such that $W|_Q(\alpha)$ is also an augmenting cycle for each α in $(t_\ell-\epsilon,t_\ell)$ or $(t_\ell,t_\ell+\epsilon)$. We assume without loss of generality that $W|_Q(\alpha)$ is an augmenting cycle for each α in $(t_\ell-\epsilon,t_\ell)$. Then for some $\alpha \in (t_\ell-\epsilon,t_\ell)$, we have $|\mathrm{cost}(Q(\alpha)) - c(Q)| < \epsilon$. Here $Q(\alpha)$ denotes the arc path $(i_\ell,i_{\ell+1}),\ldots,(i_{r-1},i_r)$ with starting time α. More precisely, we have $Q(\alpha) : (i_\ell, \alpha_\ell), \ldots, (i_r, \alpha_r)$ where $\alpha_\ell = \alpha$ and $\alpha_{k+1}(\alpha) = \alpha_k + \lambda_{i_k, i_{k+1}}$ for $k = \ell, \ldots, r-1$. Further, α can be chosen in such a way that $\text{cap}(Q(\alpha)) > 0$. Now we consider the cycle $W|_Q(\alpha)$ and repeat the above procedure for all remaining maximal arc-subpaths of W. Let W' be the resulting cycle. It is easy to see that for sufficiently small $\epsilon > 0$, we get $\text{cap}(W') > 0$ and $\text{cost}(W') < 0$. □

Lemma 8 provides another characterization of negative augmenting cycles. According to this we can conclude the following result.

Lemma 9 *Let x be a piecewise analytic flow. Then x is not optimal if the network G has a negative augmenting cycle.*

Proof Suppose that the network G contains some negative augmenting cycle. Then, we conclude from Lemma 8 that the network G contains a cycle $W : (i_1, t_1), \ldots, (i_q, t_q)$ with $\text{cap}(W) > 0$ and $\text{cost}(W) < 0$. Therefore, for every $k = 2, \ldots, q$ with $i_{k-1} \neq i_k$, there exist δ_k and γ_k such that

$$x_{i_{k-1}, i_k}(t) \leq a_{i_{k-1}, i_k}(t) - \delta_k, \quad t \in [t_{k-1}, t_{k-1} + \gamma_k),$$

if $(i_{k-1}, i_k) \in A$, and

$$\delta_k \leq x_{i_k, i_{k-1}}(t), \quad t \in [t_k, t_k + \gamma_k),$$

otherwise. Let δ and γ be the minimum of δ_k and γ_k, respectively, and define $\epsilon_k = 2\delta\gamma$. Also for every $k = 2, \ldots, q$ with $i_{k-1} = i_k$, there exist δ_k and γ_k such that

$$y_{i_k}(t) \leq b_{i_k}(t) - \delta_k, \quad t \in (t_{k-1} - \gamma_k, t_k + \gamma_k),$$

if $t_{k-1} < t_k$, and

$$\delta_k \leq y_{i_k}(t), \quad t \in (t_k - \gamma_k, t_{k-1} + \gamma_k),$$

otherwise. Let δ and γ be the minimum of δ_k and γ_k, respectively. We then define

$$\epsilon_k := \begin{cases} 2\delta\gamma, & \text{if } i_{k-1} \neq i_k, \\ \delta, & \text{if } i_{k-1} = i_k, \end{cases}$$

for $k = 2, \ldots,$ and let

$$z^* := \frac{1}{2\gamma} \min\{\epsilon_2, \epsilon_3, \ldots, \epsilon_q\}.$$

We now define

$$z_{i,j}(t) := \begin{cases} z^* & \text{if } i = i_k, j = i_{k+1}, t \in [t_k, t_k + \gamma) \text{ and } k = 1, \ldots, q-1, \\ -z^* & \text{if } j = i_{k+1}, i = i_k, t \in [t_{k+1}, t_{k+1} + \gamma), \text{ and } k = 1, \ldots, q-1, \\ 0 & \text{otherwise.} \end{cases}$$

We can easily see that $x + z$ is a feasible flow.

Thus far we have seen that another feasible flow $\bar{x} = x + z$ can be obtained by augmenting the constant flow rate z^*, given by (17), along the arcs involved in the cycle W. The *cost of augmenting*, that is, the change in the objective function value in moving from x to $\bar{x}$, is computed by $z^* \sum_{k=2}^{q} \zeta_k$, where

$$\zeta_k := \begin{cases} \int_{t_k}^{t_k+\gamma} c_{i_k,i_{k+1}}(t)\,dt & \text{if } i = i_k, j = i_{k+1}, \\ \int_{t_{k+1}}^{t_{k+1}+\gamma} -c_{i_{k+1},i_k}(t)\,dt & \text{if } i = i_{k+1}, j = i_k, \\ 0 & \text{otherwise.} \end{cases}$$

for $k = 2, \ldots, q$. Since $z^* > 0$, $\bar{x}$ will be a strictly improved feasible solution than $x(t)$ if $\sum_{k=2}^{q} \zeta_k < 0$. We know that the cost functions c are piecewise analytic and right-continuous. This implies $\sum_{k=2}^{q} \zeta_k < 0$ for γ small enough since we have $\text{cost}(W) < 0$. This establishes the lemma. □

In the following we show that the converse of Lemmas 3 and 9 is also true and then develop a strong duality result between CDNFP and CDNFP*. To do so, we need the following lemma.

Lemma 10 *Suppose that x is a piecewise analytic flow. The network G contains no negative augmenting cycle if and only if there exist piecewise analytic functions $\tau_i, i \in N$ defined on $[0, T]$ which satisfy the following conditions:*

(SP1) if $x_{i,j}(t) > 0$, then $c_{i,j}(t) + \tau_i(t) - \tau_j(t + \lambda_{i,j}) \leq 0$;
(SP2) if $x_{i,j}(t) < a_{i,j}(t)$, then $c_{i,j}(t) + \tau_i(t) - \tau_j(t + \lambda_{i,j}) \geq 0$;
(SP3) if $y_i(t) > 0$ on (u, v), then τ_i is monotonic increasing on (u, v);
(SP4) if $y_i(t) < b_i(t)$ on (u, v), then τ_i is monotonic decreasing on (u, v).

Proof We first suppose that there exist piecewise analytic functions $(\tau_i)_{i\in N}$ on $[0, T]$ satisfying conditions (1)-(4). Now let $W : (i_1, t_1), (i_2, t_2), \ldots, (i_q, t_q)$ be a cycle with $\text{cap}(W) > 0$. We can write down the cost of W as

$$\begin{aligned} \text{cost}(W) &= \sum_{k:(i_k,i_{k+1})\in A} c_{i_k,i_{k+1}}(t_k) + \sum_{k:(i_{k+1},i_k)\in A} -c_{i_{k+1},i_k}(t_{k+1}) \\ &= \sum_{k:(i_k,i_{k+1})\in A} c_{i_k,i_{k+1}}(t_k) + \tau_{i_k}(t_k) - \tau_{i_{k+1}}(t_{k+1}) \\ &\quad + \sum_{k:(i_{k+1},i_k)\in A} -c_{i_{k+1},i_k}(t_{k+1}) + \tau_{i_k}(t_{k+1}) - \tau_{i_k}(t_k) \\ &\quad + \sum_{k:i_k=i_{k+1},t_k<t_{k+1}} \tau_{i_k}(t_k) - \tau_{i_k}(t_{k+1}) \\ &\quad + \sum_{k:i_k=i_{k+1},t_k<t_{k+1}} \tau_{i_k}(t_{k+1}) - \tau_{i_k}(t_k), \end{aligned}$$

where the index k varies from 1 to $q-1$. It is not difficult to see that each of the above four summation terms on the right-hand side of the second equal sign is nonnegative since $\text{cap}(W) > 0$ and the pair x and τ satisfies (SP1)-(SP4). Hence $\text{Cost}[W] \geq 0$ and it now follows from Lemma 8 that there are no negative augmenting cycles under x.

Let us now consider the other direction, that is, there exists no negative augmenting cycle under x. For each node $i \in N$, we consider the function $\tau_i : [0, T] \to \mathbb{R}$ for which $\tau_i(t)$ is defined to be the cost of a shortest augmenting path from $(1, 0)$ to (i, t). We know from Theorem 3 that τ_i exists and is piecewise analytic on $[0, T]$.

It remains to show that the functions $\tau_i, i \in N$ satisfy conditions (1)-(4). Suppose by contradiction that the condition (1) does not hold, that is, there are some arc $(i, j) \in A$ and some point in time $t \in [0, T]$ such that $x_{i,j}(t) > 0$, but $c_{i,j}(t) + \tau_i(t) - \tau_j(t + \lambda_{i,j}) < 0$. Since x, c, and τ are piecewise analytic and thus right-continuous, there is some $\epsilon > 0$ for which $x_{i,j}(s) > 0$ and $c_{i,j}(s) + \tau_i(s) - \tau_j(s + \lambda_{i,j}) < 0$ for each $s \in [t, t + \epsilon)$. Let us fix a point $s \in (t, t + \epsilon)$ and let $P : (i_1, t_1), \dots, (i_q, t_q)$ be the shortest augmenting path from NTP $(0, 1)$ to NTP (i, s). We now consider the augmenting walk $P' : (i_1, t_1), \dots, (i_q, t_q), (j, s + \lambda_{i,j})$ from $(0, 1)$ to $(j, s + \lambda_{i,j})$ with augmenting cost $c(P') = \tau_i(s) + c_{i,j}(s)$. Since $\tau_j(s + \lambda_{i,j})$ is the cost of the shortest augmenting path from $(1, 0)$ to $(j, s + \lambda_{i,j})$ and there are no negative augmenting cycles under x, we get $\tau_j(s + \lambda_{i,j}) \leq \tau_i(s) + c_{i,j}(s)$. This is a contradiction and so the condition (1) must hold. In a similar way, we can show that the conditions (2)-(4) are fullfiled. This completes the proof of the theorem. □

We are now in a position to prove the main results of this paper.

Theorem 4 (Reduced Cost Optimality Conditions) *Suppose that x is a piecewise analytic flow. Then x is optimal if and only if there are piecewise analytic functions $\pi_i, i \in N$ defined on $[0, T]$ so that satisfy the following* reduced cost optimality conditions*:*

(RC1) if $x_{i,j}(t) > 0$, then $c_{i,j}(t) - \pi_i(t) + \pi_j(t + \lambda_{i,j}) \leq 0$;
(RC2) if $x_{i,j}(t) < a_{i,j}(t)$, then $c_{i,j}(t) - \pi_i(t) + \pi_j(t + \lambda_{i,j}) \geq 0$;
(RC3) if $y_i(t) > 0$ on (u, v), then π_i is monotonic decreasing on (u, v);
(RC4) if $y_i(t) < b_i(t)$ on (u, v), then π_i is monotonic increasing on (u, v).

Proof First suppose that x is optimal. Hence, by Lemma 9, there are no negative augmenting cycles with respect to x. It next follows from Lemma 10 that there are piecewise analytic functions $\tau_i, i \in N$ satisfying the conditions (SP1)-(SP4). Now we let $\pi_i := -\tau_i$ for each node $i \in N$, which gives the result in one direction. To prove the other direction, it is easy to see that the conditions (RC1) and (RC2) are equivalent to the conditions (CS2) and (CS2). On the other hand, by means of Lemma 1, we can show that the conditions (RC3) and (RC4) are equivalent to conditions (CS3) and (CS4). Now Lemma 3 implies that x optimal, which concludes the proof. □

Theorem 5 (Negative Cycle Optimality Condition) *A piecewise analytic flow x is optimal if and only if the network G contains no negative augmenting cycle under x.*

Proof Because of Lemma 9, it is sufficient to show that x is optimal if there are no negative augmenting cycles under x. So we assume that the network G contains no negative augmenting cycle. Then, by a similar argument as in the proof of Theorem 4, we can deduce that there are piecewise analytic functions $\pi_i, i \in N$ such that the pair x and $\pi = (\pi_i)_{i \in N}$ satisfies the optimality conditions (CS1)-(CS4). It now follows from Lemma 3 that x is optimal for CDNFP and π is optimal for CDNFP*. □

Theorem 6 (Strong Duality) *There exist piecewise analytic solutions x and π for CDNFP and CDNFP*, respectively, so that $V[CDNFP, x] = V[CDNFP^*, \pi]$.*

Proof We know from Theorem 2 that CDNFP has a piecewise analytic optimal solution, say x, due to the fact that Assumptions 1 and 2 hold. Then, by means of Theorem

Algorithm 1 NEGATIVE CYCLE-CANCELING ALGORITHM

establish an initial solution x
while G contains a negative augmenting cycle under x **do**
  identify a negative augmenting cycle W under x
  augment maximum flow rate along W
  update x
end while

4, we conclude that there exists piecewise analytic functions $\pi_i, i \in N$ such that the pair x and $\pi := (\pi_i)_{i \in N}$ satisfy the optimality conditions (RC1)-(RC4). These conditions are equivalent to conditions (CS1)-(CS4). The assertion of the theorem now follows from Lemma 3. □

5 Conclusions and further work

In this paper we have studied the continuous-time dynamic network flow problem (CDNFP) to include time-varying features encountered in many practical situations. In this problem, arc and node costs, arc and node capacities, and supplies and demands are functions of time and the passage of time is continuous. Several network-related optimality conditions as well as a stung duality result have been developed for CDNFP under the assumption that the input functions are piecewise analytic and the transit times are rational. These results can be used to develop algorithms for solving CDNFP in a similar way as in static network flows. For example, Theorem 5 lays the ground for an algorithmic approach which we call the NEGATIVE CYCLE-CANCELING ALGORITHM. Here we discuss the essential steps of a generic version of this algorithm. Further details are beyond the scope of this paper and are left for further work.

Like the negative cycle-canceling algorithm for the static minimum cost flow problem, the algorithm maintains a feasible solution at each iteration and successively improves the solution towards optimality. More specifically, the algorithm first establishes a feasible solution x. It then proceeds by identifying negative augmenting cycles under x and sending flow rate in these cycles, while preserving feasibility. The algorithm terminates when the network contains no negative cycle with respect to x. Theorem 5 implies that when the algorithm terminates it has found an optimal solution. Alg. 1 specifies the generic version of this procedure.

In what follows, we investigate in more detail how we can implement the NEGATIVE CYCLE-CANCELING ALGORITHM and discuss further research directions.

5.1 Obtaining an initial feasible solution

The problem of finding an initial feasible solution for CDNFP is not a difficult task. In fact, for the case that b is piecewise linear, and a and r are piecewise constant, we can construct a feasible flow by a static minimum cost flow computation in a so-called *time-expanded network* (see [27, Section 3.2] for more details). However, for the more general setting where the input functions are piecewise analytic, but not piecewise constant/linear, we could construct an initial solution, which is also an extreme point of the feasible region F, in a similar manner as in static network flows. Specifically, this can be done by introducing an artificial storage node s and artificial arcs of zero transit time and infinite capacity and cost joining node i to s. We note that no such arc would appear in an optimal solution unless the problem contains no feasible solution without artificial arcs.

5.2 Identifying a negative-cycle

The most important task in the NEGATIVE CYCLE-CANCELING ALGORITHM is how to check whether or not there exists a negative augmenting cycle with respect to a given solution. Moreover, if such a cycle exists, then how to detect it. In the context of static network flows, the problem of detecting negative cycles also plays an important role in negative cycle-canceling algorithms for solving the minimum cost flow problem. Hence several algorithms have been developed for detecting the presence of a negative cycle if one exists (see, e.g., [14]). Most of them combine a shortest path algorithm and a negative cycle detection strategy. Thus a natural approach to detect dynamic cycles with negative cost could be developing algorithms analogous to those that are available in the static case and would be an important topic for further investigation.

Another possible approach for detecting augmenting cycles with negative cycle is to maintain an extreme point solution x and a potential function π which is complementary slack with x at each iteration. It is worth to mention that extreme points of the feasible region for the static minimum cost flow problem correspond to the flows which do not admit augmenting cycles. A similar characterization of the extreme points for CDNFP has been derived by Anderson [3] (see also [27, Section 4.2]). We now consider a feasible solution x for CDNFP which is an extreme point of the feasible region[5]. The problem here is how to compute a potential function π which is complementary slack with x. It can be done in a similar manner as described in Anderson and Philpott [7] (see also [7, Section 3] for details) by having complementary slackness conditions (CS1)-(CS4) and the fact that x is an extreme point solution. Then we check whether complementary slackness (or reduced costs) optimality conditions hold or not. If optimality conditions hold for the pair x and π, then Lemma 3 (or Theorem 4) implies that x is optimal for CDNFP and π is optimal CDNFP*. Otherwise, there is a negative augmenting cycle with respect to x, and the pair x and π enable us to identify a negative augmenting cycle. We should mention that due to *degeneracy*, this version of the algorithm cannot necessarily send a positive flow rate along this cycle. The problem that how to overcome degeneracy and develop a network simplex version of the NEGATIVE CYCLE-CANCELING ALGORITHM is very interesting and certainly deserves further study.

5.3 Augmenting flow around a cycle

It remains to discuss how to augment flow rate around a negative dynamic cycle so that the largest decrease in the objective function value is obtained. Suppose that $W : (i_1, t_1), (i_2, t_2), \ldots, (i_q, t_q)$ is a negative augmenting cycle with respect to a given feasible solution x. Because of Lemma 8 we can assume that $\text{cap}(W) > 0$ and $\text{cost}(W) < 0$. We now consider two cases: that W is an *arc-cycle*, i.e., $(i_k, i_{k+1}) \in \overleftrightarrow{A}$

[5] We note that any feasible flow can be converted into an extreme point solution without increasing the objective function value using the purification algorithm described in [27]

for $k = 1, \ldots, q-1$ and that $W : (i_1, t_1), (i_2, t_2), \ldots, (i_q, t_q)$ is not an arc-cycle, i.e., for some k, we have $i_k = i_{k+1}$. The former case is simple since augmenting flow along an arc-cycle does not effect the storage at nodes. But the latter case requires a complicated argument and further investigation. The main reason for this is that it effects the storage at node i_k for which $i_k = i_{k+1}$ during the time interval $[\min\{t_k, t_{k+1}\}, \max\{t_k, t_{k+1}\}]$.

We conclude the paper by noting that the termination of the NEGATIVE CYCLE-CANCELING ALGORITHM after a finite number of iteration is still an open problem and deserves attention. Hence it is of great interest to investigate the convergence properties of the algorithm, even for the special case that r and a are piecewise constant and c and b are piecewise linear.

References

1. R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows. Theory, Algorithms, and Applications.* Prentice Hall, Englewood Cliffs, NJ, 1993.
2. E. J. Anderson. *A Continuous Model for Job-Shop Scheduling.* PhD thesis, University of Cambridge,, 1978.
3. E. J. Anderson. Extreme-points for continuous network programs with arc delays. *Journal of Information and Optimization Sciences*, 10:45–52, 1989.
4. E. J. Anderson and P. Nash. *Linear Programming in Infinite-Dimensional Spaces.* Wiley, New York, 1987.
5. E. J. Anderson, P. Nash, and A. F. Perold. Some properties of a class of continuous linear programs. *SIAM Journal on Control and Optimization*, 21:258–265, 1983.
6. E. J. Anderson, P. Nash, and A. B. Philpott. A class of continuous network flow problems. *Mathematics of Operations Research*, 7:501–514, 1982.
7. E. J. Anderson and A. B. Philpott. A continuous-time network simplex algorithm. *Networks*, 19:395–425, 1989.
8. E. J. Anderson and A. B. Philpott. Optimisation of flows in networks over time. In F. P. Kelly, editor, *Probability, Statistics and Optimisation*, chapter 27, pages 369–382. Wiley, New York, 1994.
9. K. M. Anstreicher. Generation of feasible descent directions in continuous time linear programming. Technical report, SOL 83–18, Department of Operations Research, Stanford University, Stanford, CA, 1983.
10. T. M. Apostol. *Mathematical Analysis.* 2th Edition, Addison-Wesley, 1974.
11. R. E. Bellman. Bottleneck problem and dynamic programming. *Proceedings of the National Academy of Sciences*, 39:947–951, 1953.
12. R. E. Bellman. *Dynamic Programming.* Princeton University Press, Princeton, NJ, 1957.
13. R. N. Buie and J. Abrham. Numerical solutions to continuous linear programming problems. *Zeitschrif für Operations Research*, 17:107–117, 1973.
14. B. V. Cherkassky and A. V. Goldberg. Negative-cycle detection algorithms. *Mathematical Programming*, 85:277–311, 1999.
15. J. B. Conway. *A Course in Functional Analysis.* 2nd Edition, New York, Springer-Verlag, 1990.
16. W. P. Drews. A simplex-like algorithm for continuous-time linear optimal control problems. In R. W. Cottle and J. Krarup, editors, *Optimization Methods for Resource Allocation*, pages 309–322. Crane Russak and Co. Inc., New York, 1974.
17. L. Fleischer and J. Sethuraman. Efficient algorithms for separated continuous linear programs: The multi-commodity fow problem with holding costs and extensions. *Mathematics of Operations Research*, 30:916–938, 2005.
18. L. R. Ford and D. R. Fulkerson. Constructing maximal dynamic flows from static flows. *Operations Research*, 6:419–433, 1958.
19. L. R. Ford and D. R. Fulkerson. *Flows in Networks.* Princeton University Press, 1962.
20. R. J. Hartberger. Representation extended to continuous time. In R. W. Cottle and J. Krarup, editors, *Optimization Methods for Resource Allocation*, pages 309–322. Crane Russak and Co. Inc., New York, 1974.

21. S. M. Hashemi, E. Nasrabadi, and M. Skutella. On solving continuous-time dynamic network flows. Technical Report 13-2008, Technische Universität Berlin, 2008. Submitted to Networks.
22. R. Koch and E. Nasrabadi. Continuous-time dynamic shortest paths with negative transit times. Technical Report 6-2010, Technische Universität Berlin, 2010. Submitted to Mathematical Programming.
23. R. Koch, E. Nasrabadi, and M. Skutella. Continuous and discrete flows over time: A general model based on measure theory. Technical Report 16-2009, Technische Universität Berlin, 2009. Submitted to Mathematical Methods of Operations Research.
24. R. S. Lehman. On the continuous simplex method. Technical report, RM-1386, Rand Corporation, Santa Monica, CA, 1954.
25. X. Luo and D. Bertsimas. A new algorithm for state-constrained separated continuous linear programs. *SIAM Journal on Control and Optimization*, 37:177–210, 1998.
26. X. D. Luo. *Continuous Linear Programming: Theory, Algorithms and Applications.* PhD thesis, MIT Operations Research Center, Cambridge MA, 1995.
27. E. Nasrabadi. *Dynamic Flows in Time-varying Networks.* PhD thesis, Technische Universität Berlin, Berlin, 2009.
28. A. F. Perold. Fundamentals of a continuous time simplex method. Technical report, SOL 78–26, Department of Operations Research, Stanford University, Stanford, CA, 1978.
29. A. F. Perold. Extreme points and basic feasible solutions in continuous time linear programming. *SIAM Joural on Control and Optimization*, 19:52–63, 1981.
30. A. B. Philpott. *Algorithms For Continuous Network Flow Problems.* PhD thesis, University of Cambridge, UK, 1982.
31. A. B. Philpott. Network programming in continuous time with node storage. In E. J. Anderson and A. B. Philpott, editors, *Infinite Programming: Proceedings of an International Symposium on Infinite Dimensional Linear Programming*, volume 259 of *Lecture notes in economics and mathematical systems*, pages 136–153. Springer, Berlin, 1985.
32. A. B. Philpott. Continuous-time flows in networks. *Mathematics of Operations Research*, 15:640–661, 1990.
33. A. B. Philpott and M. Craddock. An adaptive discretization algorithm for a class of continuous network programs. *Networks*, 26:1–11, 1995.
34. M. C. Pullan. An algorithm for a class of continuous linear programs. *SIAM Journal on Control and Optimization*, 31:1558–1577, 1993.
35. M. C. Pullan. Forms of optimal solutions for separated continuous linear programs. *SIAM Journal on Control and Optimization*, 33:1952–1977, 1995.
36. M. C. Pullan. A duality theory for separated continuous linear programs. *SIAM Journal on Control and Optimization*, 34:931–965, 1996.
37. M. C. Pullan. Existence and duality theory for separated continuous linear programs. *Mathematical Modeling of Systems*, 3:219–245, 1997.
38. M. C. Pullan. A study of general dynamic network programs with arc time-delays. *SIAM Journal on Optimization*, 7:889–912, 1997.
39. M. C. Pullan. Convergence of a general class of algorithms for separated continuous linear programs. *SIAM Journal on Optimization*, 10:722–731, 2000.
40. M. C. Pullan. An extended algorithm for separated continuous linear programs. *Mathematical Programming*, 93:415–451, 2002.
41. R. G. Segers. A generalised function setting for dynamic optimal control problems. In R. W. Cottle and J. Krarup, editors, *Optimization Methods for Resource Allocation*, pages 279–296. Crane Russak and Co. Inc., New York, 1974.
42. M. Skutella. An introduction to network flows over time. In W. Cook, L. Lovász, and J. Vygen, editors, *Research Trends in Combinatorial Optimization*, pages 451–482. Springer, Berlin, 2009.
43. G. Weiss. A simplex based algorithm to solve separated continuous linear programs. *Mathematical Programming*, 115:151–198, 2008.