Time-Expanded Graphs for Flow-Dependent Transit Times

Ekkehard Köhler[1], Katharina Langkau[1], and Martin Skutella[2]

[1] Technische Universität Berlin
Institut für Mathematik, Sekr. MA 6–1
Straße des 17. Juni 136, 10623 Berlin, Germany
{ekoehler,langkau}@math.tu-berlin.de
[2] Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85, 66123 Saarbrücken, Germany
skutella@mpi-sb.mpg.de

Abstract. Motivated by applications in road traffic control, we study flows in networks featuring special characteristics. Firstly, there are transit times on the arcs of the network which specify the amount of time it takes for flow to travel through an arc; in particular, flow values on arcs may change over time. Secondly, the transit time of an arc varies with the current amount of flow using this arc. The latter feature is crucial for various real-life applications; yet, it dramatically increases the degree of difficulty of the resulting optimization problems. While almost all flow problems with constant transit times on the arcs can be solved efficiently by applying classical (static) flow algorithms in a corresponding time-expanded network, no such approach was known for flow-dependent transit times, up to now. One main contribution of this paper is a time-expanded network with flow-dependent transit times to which the whole algorithmic toolbox developed for static flows can be applied. Although this approach does not entirely capture the behavior of flows over time with flow-dependent transit times, we present approximation results which provide evidence of its surprising quality.

Subject classifications: Networks/graphs: flow algorithms. Transportation: network models.

Area of review: Optimization

1 Introduction

In view of the steadily growing car traffic and the limited capacity of our street networks, methods for better traffic management and route guidance are becoming more and more important. In our opinion, network flow theory is currently the most promising approach to model and solve these optimization problems. Other common approaches to study traffic problems are traffic simulation [3, 20] or models based on fluid dynamics [23], control theory [24], and variational inequalities [8]. While simulation is a powerful tool to evaluate traffic scenarios, it does not explicitly allow for optimization. On the other hand, fluid models and other models based on differential equations capture very well the dynamical behavior of traffic as a continuous quantity, but can currently not handle large networks.

Classical network flow theory deals with static (steady-state) situations, that is, every arc of a network is assigned one variable representing the amount of flow on this arc. This model obviously fails to capture important features in network flow problems arising in various applications such as, e.g., road traffic control: First of all, flow values on arcs can obviously change over time. 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. Thus, not only the amount of flow to be transmitted but also the time needed for the transmission plays an essential role. Already Ford and Fulkerson [12, 13] studied such network flows and came up with the model of *flows over time* (or 'dynamic flows'). This model, however, does not capture the third and, by far, most complicating aspect of real-life flow problems: There is usually a highly complex dependency of transit times on the flow situation in the network. Various interesting examples can be found in the survey articles of Aronson [2] and Powell, Jaillet, and Odoni [22].

A fully realistic model of flow-dependent transit times on arcs must take density, speed, and flow rate evolving along an arc into consideration; see, e.g., the book of Sheffi [26] and the report by Gartner, Messer, and Rathi [14] for details in the case of traffic flows. It is a highly nontrivial and open problem to find an appropriate and, above all, tractable mathematical model. There are hardly any algorithmic techniques known which are capable of providing reasonable solutions even for networks of rather modest size. For problem instances of realistic size (as those occurring in real-life applications), already the solution of mathematical programs relying on simplifying assumptions is in general still beyond the means of state-of-the-art computers. We refer to [2, 16, 21, 22, 25] for more details and examples. In this paper we present a new model for flows over time with flow-dependent transit times which, for the first time, allows to attack these problems by very efficient network flow algorithms.

Flows over time with fixed transit times. As mentioned above, flows over time were introduced more than forty years ago by Ford and Fulkerson [12, 13]. They consider networks with capacities and transit times on the arcs. The transit time of an arc specifies the amount of time it takes for flow to travel from the tail to the head of that arc. In contrast to classical static flows, a flow over time 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.

Time-expanded networks. Ford and Fulkerson also introduce the notion of *time-expanded networks.* A time-expanded network contains one copy of the node set of the underlying 'static' network for each discrete time step (building a *time layer*). Moreover, for each arc with transit time τ in the static network, there is a copy between each pair of time layers of distance τ in the time-expanded network. A flow over time in the static network can be interpreted as a static flow in this time-expanded network and vice versa. In particular, the concept of time-expanded networks allows to solve a variety of flow-over-time problems by applying algorithmic techniques developed for static network flows; see, e.g., Fleischer and Tardos [11]. On the other hand, one has to pay for this simplification of the considered flow problem in terms of an enormous increase in the size of the underlying network. In particular, the size of the time-expanded network is only pseudo-polynomial in the input size. Nevertheless, Fleischer and Skutella [9, 10] show that the time-expanded network can be reduced to polynomial size at the cost of a slightly degraded solution space.

Models for flow-dependent transit times. In the following we mention some models developed for flows over time with flow-dependent transit times. More pointers to the literature can for example be found in the survey articles [2, 16, 22, 25]. A non-linear and non-convex program with discretized time steps is proposed by Merchant and Nemhauser [18]. In their model, the outflow out of an arc in each time period solely depends on the amount of flow on that arc at the beginning of the time period. Unfortunately, the non-convexity of their model causes analytical and computational problems. Merchant and Nemhauser [19] and Carey [5] describe special constraint qualifications which are necessary to ensure optimality of a solution in this model. Moreover, Carey [6] introduces a slight revision of the model of Merchant and Nemhauser yielding a convex problem instead of a non-convex one.

Recently, Köhler and Skutella [15] came up with a different model of flow-dependent transit times. In their model, the pace of flow on an arc depends on its current load, i.e., the entire amount of flow which is currently traveling along that arc. For the problem of sending a given amount of flow within minimal time through a network, a near-optimal solution can be computed in this model by a static convex-cost flow computation. Unfortunately, it is not clear whether and how this approach can be generalized to more realistic problems.

Carey and Subrahmanian [7] introduce a generalized time-expanded network for flow-dependent transit times. For each time period, there are several copies of an arc of the underlying 'static' network corresponding to different transit times. However, in order to enforce flow-dependent transit times, special capacity constraints are introduced which give rise to a dependency between the flow on all copies of an arc corresponding to one time step. As a consequence, the resulting static problem on the generalized time-expanded graph can no longer be solved by standard network flow techniques but requires a general linear programming solver. This constitutes a serious drawback with regard to the practical efficiency and applicability of this model.

A new model. We propose a closely related approach which overcomes the latter drawback. As in the model of Carey and Subrahmanian, we consider a generalized time-expanded network with multiple copies of each arc for each time step. However, in contrast to their model, we introduce additional 'regulating' arcs which enable us to enforce flow-dependent transit times without using generalized capacity constraints. As a result, we can apply the whole algorithmic toolbox developed for static network flows to the generalized time-expanded network. In the following, we refer to this time-expanded network as *fan graph*.

The underlying assumption for our approach (as for the related approach of Carey and Subrahmanian) is that at any moment of time the transit time on an arc solely depends on the current rate of inflow into that arc. In the following, we will therefore speak of *flows over time with inflow-dependent transit times*, emphasizing that transit times are considered as functions of the rate of inflow. Thus, in contrast to the load-dependent model developed in [15], the flow units traveling on the same arc at the same time do not necessarily experience the same pace, as the transit time and thus the pace of every unit of flow is determined when entering the arc and remains fixed throughout. Of course, this model is only a rough approximation of the actual behavior of flow in real-life applications. For example, flows over time with inflow-dependent transit times do in general not obey the *first-in-first-out* (*FIFO*) property (i.e., flow units are in general not entering and leaving an arc in the same order). A violation of the FIFO property can occur if the inflow rate of an arc drops down quickly. This causes a strong decrease in the transit time of that arc such that some flow units might pass other flow units which are ahead on the arc traveling at a slower pace. However, in the special setting that we discuss later, this phenomenon does not occur.

Evaluation of the new model. The generalized time-expanded network presented in this paper, as the one considered in [7], does not entirely capture the behavior of flows over time with inflow-dependent transit times. A static flow in the fan graph can use several adjacent arcs corresponding to the same arc of the underlying static network but representing different transit times. Thus, in the corresponding flow over time, several units of flow entering the same arc at the same time might experience different transit times. In fact, the solution space defined by static flows on the fan graph can be seen as a relaxation of the space of flows over time with inflow-dependent transit times. However, for the special case of fixed transit times, the fan graph equals the standard time-expanded network such that the two solution spaces coincide.

Based on this observation, we present approximation results providing strong evidence of the quality of this relaxation and thus justifying its consideration for practical purposes. We show that an optimal solution to a special problem, the 'quickest s-t-flow problem', in the fan graph can be turned into a feasible flow over time with inflow-dependent transit times while only losing a factor of $2+\epsilon$ in the value of the objective function (in the worst case). The parameter $\epsilon > 0$ can be chosen arbitrarily. This result holds for non-decreasing transit time functions. Furthermore, if the transit time functions are concave, the performance ratio can be improved to $3/2+\epsilon$.

Finally, we would like to stress the fact that the fan graph can be applied to solve a large variety of time-dependent flow problems with inflow-dependent transit times, such as multi-commodity flows, flows with costs on the arcs, and so forth. The quickest s-t-flow problem is used as an example to demonstrate the quality of the fan graph and thus the strength of this approach.

The paper is organized as follows. After defining necessary notation and reviewing Ford and Fulkerson's algorithm in Section 2, we introduce the fan graph and the related static bow graph in Section 3. Then, in Sections 4, 5, and 6 we demonstrate the strength of this new concept by presenting approximation algorithms which are based thereupon. Finally, concluding remarks are given in Section 7.

2 Preliminaries

We start with a description of the concept of flows over time. For a detailed discussion of basic network flow theory, we refer to the standard textbook [1].

Basic definitions and notation. Let $G = (V, E)$ be a directed graph with node set V and arc set E. Each arc $e \in E$ has associated with it a positive capacity u_e, which is interpreted as an upper bound on the rate of flow entering e at any moment of time, and a non-negative transit time function τ_e, which determines the time it takes for flow to traverse arc e. Finally, there is a source node $s \in V$ and a sink node $t \in V$. An *s-t-flow over time* (also known as *dynamic s-t-flow* or *time-dependent s-t-flow* in the literature) with time horizon T is given by functions $f_e : [0, T) \to \mathbb{R}^+$, for $e \in E$, where $f_e(\theta)$ defines the rate of flow (per unit time) entering arc e at time θ. This flow arrives at the head node of e at time $\theta + \tau_e(f_e(\theta))$.

In the setting of flows over time, flow conservation in a node $v \in V \setminus \{s\}$ at time $\theta \in [0, T)$ requires that the total inflow into node v until time θ is an upper bound on the total outflow out of node v until time θ. In particular, flow can be stored at intermediate nodes, i.e., it is possible to hold inventory at a node which is neither a source nor a sink before sending it onward. If storage of flow at intermediate nodes is not allowed, equality must hold for all $v \in V \setminus \{s,t\}$ and all $\theta \in [0,T)$. In the following, we mostly assume that storage of flow at nodes is allowed. Finally, all arcs must be empty at time T, i.e., $\tau_e(f_e(\theta)) < T - \theta$ for all $e \in E$ and $\theta \in [0,T)$, and flow must not remain in any node other than the sink at time T.

The value of a flow over time f is the total outflow out of node s until time T minus the total inflow into s until time T. Notice that, due to flow conservation, this amount, which is denoted by $|f|$, is equal to the total inflow into node t until time T minus the total outflow out of t until time T.

Ford and Fulkerson's Algorithm Ford and Fulkerson [12, 13] consider the problem of computing a maximal s-t-flow over time with fixed time horizon T. For the case of constant transit times τ_e, $e \in E$, they show how this problem can be solved by only one (static) minimum cost flow computation in G. Since we will employ some of the underlying basic insights of this algorithm, we give a brief description of it here.

The algorithm is based on the concept of *temporally repeated flows*. Let $(y_e)_{e\in E}$ be a feasible static s-t-flow in G with path decomposition $(y_P)_{P\in\mathcal{P}}$, where $\mathcal{P}$ is a set of s-t-paths. If the transit time $\tau_P := \sum_{e\in P}\tau_e$ of every path $P \in \mathcal{P}$ is bounded from above by T, the static s-t-flow y can be turned into a *temporally repeated* s-t-flow f as follows. Starting at time zero, the flow over time f sends flow at constant rate y_P into path $P \in \mathcal{P}$ until time $T - \tau_P$, thus ensuring that the last unit of flow arrives at the sink before time T. Feasibility of f with respect to capacity constraints immediately follows from the feasibility of the underlying static flow y. Notice that the value of f is given by

$$|f| \;=\; \sum_{P\in\mathcal{P}} (T-\tau_P)\, y_P \;=\; T\,|y| - \sum_{e\in E} \tau_e\, y_e \ . \tag{1}$$

Here, $|y|$ denotes the value of the static s-t-flow y.

The algorithm of Ford and Fulkerson computes a static s-t-flow y maximizing the right hand side of (1), then determines a path decomposition of y, and finally returns the corresponding temporally repeated flow f. Ford and Fulkerson show that this temporally repeated flow is in fact a maximal s-t-flow over time.

The following observation, which will turn out to be useful later, follows immediately from the optimality of y with respect to the objective function (1).

Observation 1. *Any path decomposition $(y_P)_{P\in\mathcal{P}}$ of the optimal static s-t-flow y computed by the algorithm of Ford and Fulkerson sends flow along paths of transit time at most T.*

A problem closely related to the one considered by Ford and Fulkerson is the *quickest s-t-flow problem*. Here, instead of fixing the time horizon T and asking for a flow over time of maximal value, the value of the flow (demand) is fixed and T is to be minimized. This problem can be solved in polynomial time by incorporating the algorithm of Ford and Fulkerson into a binary search framework. Using Megiddo's method of parametric search [17], Burkard, Dlaska, and Klinz [4] present a faster algorithm which solves the quickest s-t-flow problem in strongly polynomial time.

3 A Model for Flows Over Time with Flow-Dependent Transit Times

In this section we define a generalized time-expanded graph—the *fan graph* G^F—in which transit times indirectly depend on the flow rate. As a result, classical network algorithms for static flow problems can be applied to G^F in order to compute flows over time with inflow-dependent

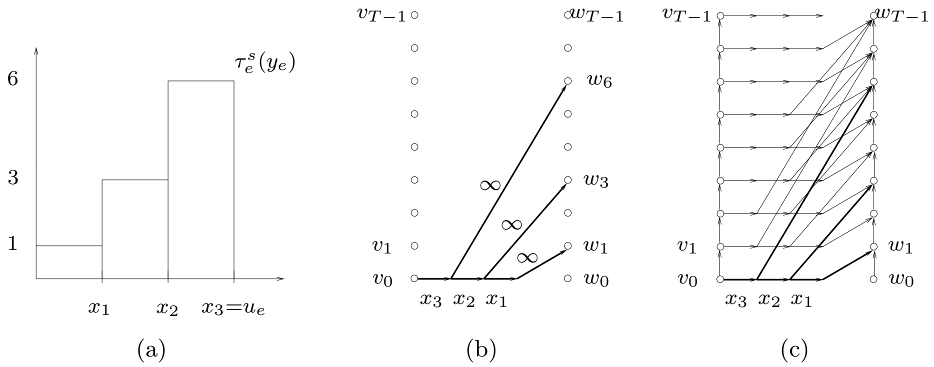

Fig. 1. Definition of the fan graph; expansion of a single arc e.

transit times in G. In the following sections, we demonstrate the strength of this approach by computing approximate quickest s-t-flows with inflow-dependent transit times in G.

Let us, for the moment, turn back to the case of fixed transit times. Assume that we are given a positive, integral transit time τ_e for every arc $e \in E$. Given an integral time horizon T, the time-expanded graph of G introduced by Ford and Fulkerson [12, 13] contains one copy of every node in V for every integral time step, i.e., its node set is $\{v_\theta : v \in V, \theta = 0, 1, \dots, T-1\}$. For every arc $e = (v, w) \in E$, and for every $0 \le \theta < T - \tau_e$, it contains an arc e_θ from v_θ to $w_{\theta+\tau_e}$ of the same capacity as e. Additionally, there are arcs with infinite capacity from v_θ to $v_{\theta+1}$, for all $v \in V$ and $\theta = 0, \dots, T-2$, which model the possibility to hold flow at a node.

A static flow in the time-expanded graph can be interpreted as a flow over time in G. Take the flow on arc e_θ as the flow rate into arc e during the time interval $[\theta, \theta+1)$. Similarly, any flow over time f with time horizon T corresponds to a flow in the time-expanded graph; for every arc e_θ, take the average flow rate of f on e over the time interval $[\theta, \theta+1)$ as the flow value on e_θ. The relationship between flows over time in G and static flows in the time-expanded graph of G allows to solve many time-dependent flow problems in G by transforming them to static flow problems in the time-expanded graph; see, e.g., [11].

3.1 The Fan Graph

In order to model flow-dependent transit times, we take a similar approach, i.e., we expand the graph according to the transit time functions in such a way that transit times indirectly depend on the current flow rate. For the moment, we will assume that the transit time function τ_e of an arc e is given as a piecewise constant, non-decreasing, and left-continuous function with only integral values. Of course, the latter assumption can easily be relaxed to allow arbitrary rational values if time is scaled by an appropriate factor. To stress the step function character of this transit time function, we will denote it by τ_e^s. Later we will use the fact that general transit time functions can be approximated by step functions within arbitrary precision.

The *fan graph* G^F is defined on the same set of nodes $\{v_\theta : v \in V, \theta = 0, 1, \dots, T-1\}$ as the time-expanded graph. An illustration of the definition is given in Figure 1, where a single arc $e = (v, w)$ is expanded according to its transit time function. For every $\theta \in \{0, 1, \dots, T-1\}$, we define a 'fan' of arcs leaving v_θ, which represent all possible transit times of arc e. In the example, there are three possible transit times on arc e, namely transit times 1, 3, and 6; see Figure 1 (a). The fan leaving v_0 is shown in Figure 1 (b). The fan consists of capacitated horizontal arcs and uncapacitated arcs pointing upwards. While the latter arcs model the different possible transit times, the capacities of the horizontal arcs try to control the distribution of flow according to the transit time function. More precisely, if flow is being sent into the fan at rate y_e, the transit time assigned to the slowest portion of this flow is forced to be at least $\tau_e^s(y_e)$. In the example, the capacities are chosen in such a way that at most x_1 units of flow can travel with transit time $\tau_1 = 1$, at most x_2 units of flow can travel with transit time $\tau_2 = 3$ or faster, and so on.

Figure 1 (c) shows the fan graph of a single arc e. As for the time-expanded graph, there is an arc with infinite capacity from v_θ to $v_{\theta+1}$, for all $v \in V$ and $\theta = 0, \dots, T-2$, which models the possibility to hold flow at node v.

Clearly, by the choice of capacities, transit times in the fan graph indirectly depend on the inflow. Consider a static network flow in the fan graph. As in the time-expanded graph, the flow on an arc from v_θ to $w_{\theta+\Delta}$ can be interpreted as the flow through arc $e = (v, w)$ that starts at node v in the time interval $[\theta, \theta + 1)$ and needs Δ time units to arrive in w. The fan graph defines a simple discrete model of flow-dependent transit times. It allows to formulate and solve various network flow problems in the fan graph using algorithms known from static network flow theory.

Eventually, we are interested in finding good network flow solutions for the model of inflow-dependent transit times defined above. Although, in the fan graph, we have chosen transit times and capacities according to the given transit time functions, the transit times experienced in the fan graph do not completely reflect inflow-dependent transit times. Consider the example in Figure 1. Assume that flow is entering arc e at constant rate x_3. According to the transit time function, every flow unit should traverse e with transit time $\tau_e^s(x_3) = 6$. In the fan graph, however, it is feasible to let x_1 units of flow traverse e with transit time 1, another $x_2 - x_1$ units with transit time 3, and the rest with transit time 6. Thus, only a small portion of the flow is experiencing the correct transit time 6.

Nevertheless, expanding transit times like in the fan graph provides a relaxation of the original setting of flows over time with inflow-dependent transit times. Based on this relaxation we develop approximation algorithms for the quickest flow problem with inflow-dependent transit times in the next section. On the one hand, these approximations lead to provably good solutions. On the other hand, they provide results on the quality of the relaxation constituted by the fan graph.

3.2 The Bow Graph

A major drawback of the fan graph is that it may become very large, and is thus not practical to work with for many real-world applications. Even if all transit times are constant, implying that the fan graph and the time-expanded graph essentially coincide, this graph is only of pseudo-polynomial size. Ford and Fulkerson [12, 13] prove that, in order to compute a maximal flow over time with constant transit times, it is not necessary to explicitly work with the time-expanded graph. Instead, as described earlier, they present a polynomial time algorithm which solely works on the original graph. Motivated by this result, we give a slightly different view of the fan graph, as the time-expansion of a smaller graph, the so-called *bow graph*. We will explain how flows over time with inflow-dependent transit times in G relate to a certain class of flows over time in the bow graph.

Again, let us assume that all transit time functions are given as piecewise constant, non-decreasing, and left-continuous functions τ_e^s, $e \in E$. The bow graph, denoted $G^B = (V^B, E^B)$, arises from the original graph by expanding each arc $e \in E$ according to its transit time function. In G^B, every arc $a \in E^B$ has a capacity u_a and a constant transit time $\tau_a \in \mathbb{R}^+$.

For the definition, let us consider a particular arc $e \in E$. For arc e, we are given breakpoints $0 = x_0 < x_1 < \cdots < x_k = u_e$ and corresponding transit times $\tau_1 < \cdots < \tau_k$. Flow entering at rate $x \in (x_{i-1}, x_i]$ needs τ_i time units to traverse arc e. Consider the example in Figure 2, where an arc is expanded according to the step function in Figure 1 (a). Arc e is replaced by arcs of two types; *bow arcs* denoted $b_1, \dots, b_k$ and *regulating arcs* denoted $r_1, \dots, r_k$. The bow

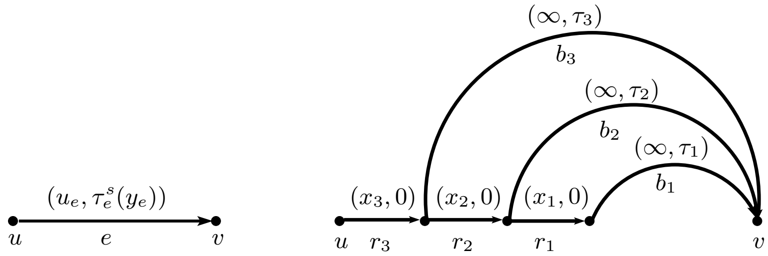

Fig. 2. Definition of the bow graph; expansion of a single arc e.

arcs are uncapacitated, they represent all possible transit times of arc e. The transit time of arc b_i is given by τ_i, $i = 1, \dots, k$; in the example we have $\tau_1 = 1, \tau_2 = 3$, and $\tau_3 = 6$. The regulating arcs have zero transit time, they limit the amount of flow entering the bow arcs. Their capacities are chosen according to the breakpoints of transit time function $\tau_e^s(y_e)$; more precisely, the capacity of arc r_i is set to x_i, $i = 1, \dots, k$. We will denote the set of bow arcs and regulating arcs of an arc e by E_e^B. Note that the size of this *expansion* E_e^B of arc e is linear in the number of breakpoints of $\tau_e^s(y_e)$. Also note that the fan graph (see Figure 1 (c)) is essentially the time-expanded graph of the bow graph (see Figure 2).

An arbitrary flow over time with inflow-dependent transit time functions τ_e^s in G can be interpreted as a flow over time with constant transit times in the bow graph G^B (and thus as a static flow in the fan graph G^F). The idea is the following. Flow entering arc e at time θ and traversing e with transit time τ_i in the original graph G also enters the expansion of arc e in G^B at time θ and traverses it in time τ_i, using the corresponding bow arc b_i. To be more precise, let f be a flow over time in G with inflow-dependent transit time functions τ_e^s and time horizon T. Then, f is given by flow rates $f_e : [0, T) \to \mathbb{R}^+$ for all $e \in E$. According to the transit time functions, flow entering arc e at time θ will need $\tau_e^s(f_e(\theta))$ time to traverse arc e. Let $i \in \{1, \dots, k\}$ be chosen such that $f_e(\theta) \in (x_{i-1}, x_i]$ and consider the corresponding transit time $\tau_i = \tau_e^s(f_e(\theta))$. We define a flow over time f^B on the expansion of arc e by setting $f_a^B(\theta) := f_e(\theta)$ if a is either the bow arc b_i or a regulating arc r_j with $j \geq i$. For all other arcs we set $f_a^B(\theta) = 0$. Notice that f^B obeys capacity constraints and flow conservation constraints at all intermediate nodes.

Observation 2. *Let f be a flow over time with inflow-dependent transit time functions τ_e^s in G, which sends D units of flow from s to t within time T. Then, there exists a flow over time with constant transit times in G^B which sends D units of flow from s to t within time T.*

Hence, every flow over time in G can be regarded as a flow over time in G^B; however, the converse is not true. By our definition of flow-dependent transit times, flow units entering arc $e \in E$ at the same time, simultaneously arrive at the head node of e. In the bow graph, however, flow units entering the expansion of an arc simultaneously, do not necessarily travel through the expansion at the same pace. The flow is allowed to split up and use bow arcs representing different transit times. Only a portion of the flow is traversing the expansion of arc e at the speed prescribed by the transit time function or possibly slower. The rest might travel at a faster speed. Therefore, the bow graph is a relaxation of our original model of inflow-dependent transit times. The advantage of this relaxation is that it relies on a graph having constant transit times on the arcs. This implies that one can apply algorithms that are known for this much simpler model of flows over time. In particular, this provides the opportunity to use standard network flow algorithms on the time-expanded fan graph G^F of G^B.

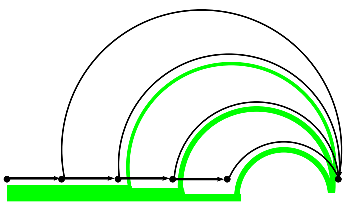

Fig. 3. Static flow y^B.

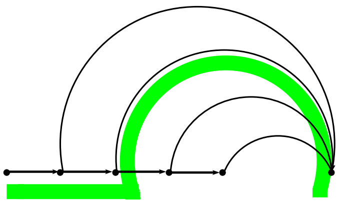

Fig. 4. Static flow $\tilde{y}^B$.

4 Approximate Solutions to the Quickest Flow Problem

The purpose of this section is to demonstrate the strength and the usefulness of the bow and the fan graph presented in the last section for solving time-dependent flow problems with inflow-dependent transit times. This is done by presenting approximation algorithms for the quickest flow problem with inflow-dependent transit times that are based on optimal solutions to the relaxation given by G^B and G^F.

4.1 An Approximation Algorithm for the Quickest Flow Problem

In this section an approximation algorithm is presented for the case of piecewise constant, non-decreasing, and left-continuous transit time functions. Using the algorithm of Burkard et al. [4], one can determine a quickest flow over time in the bow graph G^B in strongly polynomial time. Like the algorithm by Ford and Fulkerson, the computation yields the minimal time horizon $\overline{T}^B$ and a static flow y^B in G^B such that the value of the resulting temporally repeated flow f^B in G^B is

$$|f^B| \;=\; \overline{T}^B\,|y^B| - \sum_{a\in E^B} \tau_a\, y_a^B \;=\; D\ . \tag{2}$$

Since the quickest flow problem on G^B can be seen as a relaxation of the quickest flow problem with inflow-dependent transit times on G, the value $\overline{T}^B$ is a lower bound on the optimal time horizon $\overline{T}$ in G.

For the following lemma, we consider the bow graph expansion of a particular arc $e \in E$ to bow arcs $b_1, \ldots, b_k$ and regulating arcs $r_1, \ldots, r_k$, where the capacity of r_j is given by x_j.

Lemma 1. *If y^B sends flow along the i^{th} bow arc b_i, then for all bow arcs b_j with index $j \le i$ we have $y^B_{b_j} = x_j - x_{j-1}$. Less formally speaking, y^B fills the bows from bottom to top.*

Proof. For $j \le i$, the transit time of arc b_j is smaller than the transit time of arc b_i. Therefore, shifting flow from b_i to b_j improves the value of the objective function (2) of y^B. The result thus follows from the optimality of y^B and the choice of capacities on the regulating arcs. □

Eventually, we are interested in a flow over time in G^B which yields a feasible flow over time in G. Unfortunately, as discussed above, f^B does in general not satisfy this requirement. One reason is that two flow units entering the expansion of an arc $e \in E$ simultaneously might in fact experience different transit times in f^B due to different transit times on bow arcs in E_e^B.

Therefore, the static flow y^B is rerouted to make sure that it does not split among bows representing different transit times of the same arc $e \in E$. This is achieved by pushing flow from 'fast' bow arcs up to the 'slowest' flow-carrying bow arcs in y^B. An illustration is given in Figures 3 and 4. More formally, for the expansion of every arc $e \in E$, the modified static flow

$\tilde{y}^B$ is defined by setting

$$\tilde{y}^B_{b_i} := \begin{cases} y^B_{r_i} & \text{if } y^B_{b_i} > 0 \text{ and } y^B_{b_j} = 0 \text{ for all } j > i, \\ 0 & \text{else,} \end{cases}$$

$$\tilde{y}^B_{r_i} := \begin{cases} y^B_{r_i} & \text{if } y^B_{b_j} = 0 \text{ for all } j > i, \\ 0 & \text{else.} \end{cases}$$

Notice that the value of the flow remains unchanged, i.e., $|\tilde{y}^B| = |y^B|$. Later, we will need the following observation which follows by construction of $\tilde{y}^B$ and Lemma 1.

Observation 3. *For every arc $e \in E$, the flow $\tilde{y}^B$ is routed through a bow arc b^e with transit time $\tau_{b^e} = \tau^s_e(\tilde{y}^B_{b^e})$ in the expansion of e.*

We show that the modified flow $\tilde{y}^B$ yields a flow over time in G with value D and time horizon at most $2\overline{T}^B$. Decompose $\tilde{y}^B$ into flows on s-t-paths $P \in \tilde{\mathcal{P}}$ in G^B with flow values $\tilde{y}^B_P$.

Observation 4. *The transit time τ_P of every path $P \in \tilde{\mathcal{P}}$ is bounded from above by $\overline{T}^B$.*

Proof. Since all flow-carrying arcs in $\tilde{y}^B$ carry flow in y^B as well, the result follows from Observation 1. □

As a consequence, the path-decomposition $(\tilde{y}^B_P)_{P\in\tilde{\mathcal{P}}}$ of $\tilde{y}^B$ induces a temporally repeated flow $\tilde{f}^B$ in G^B for any time horizon $T \geq \overline{T}^B$. It is easy to determine T such that $|\tilde{f}^B| = T\,|\tilde{y}^B| - \sum_{a\in E^B} \tau_a\, \tilde{y}^B_a = D$.

Lemma 2. *The value of T is bounded from above by $2\overline{T}^B$.*

Proof. Notice that $|\tilde{f}^B|$ is an increasing function of T. Therefore, it suffices to show that $|\tilde{f}^B| \geq D$ for $T = 2\overline{T}^B$:

$$\begin{aligned} |\tilde{f}^B| &= 2\overline{T}^B\,|\tilde{y}^B| - \sum_{a\in E^B} \tau_a\,\tilde{y}^B_a = 2\overline{T}^B\,|\tilde{y}^B| - \sum_{P\in\tilde{\mathcal{P}}} \tau_P\,\tilde{y}^B_P \\ &= \overline{T}^B\,|\tilde{y}^B| + \sum_{P\in\tilde{\mathcal{P}}} \big(\overline{T}^B - \tau_P\big)\,\tilde{y}^B_P \geq \overline{T}^B\,|\tilde{y}^B| = \overline{T}^B\,|y^B| \geq |f^B| = D\ . \end{aligned}$$

The first inequality follows from Observation 4 and the second inequality follows from (2). This concludes the proof. □

Finally, we show that $\tilde{f}^B$ can be interpreted as a flow over time with inflow-dependent transit times in G. For any arc $e \in E$, the static flow $\tilde{y}^B$ uses at most one bow arc $b^e \in E^B_e$. Thus, $\tilde{f}^B$ naturally induces a flow over time f in G by setting $f_e(\theta) := \tilde{f}^B_{b^e}(\theta)$ for any time θ.

Lemma 3. *The flow over time f in G is feasible, that is, it fulfills the flow conservation constraints.*

Proof. The flow entering arc e at time θ in f arrives at the head-node v of e exactly $\tau^s_e(f_e(\theta))$ time units later. Consider the corresponding flow in $\tilde{f}^B$ that enters arc b^e at time θ. This flow arrives in node v exactly τ_{b^e} time units later. It therefore suffices to show that $\tau^s_e(f_e(\theta)) \leq \tau_{b^e}$, for all θ, that is, the flow in f travels faster than in $\tilde{f}^B$. The result then follows from flow conservation for the temporally repeated flow $\tilde{f}^B$ in G^B and the possibility to store flow at intermediate nodes. Since $f_e(\theta) := \tilde{f}^B_{b^e}(\theta) \leq \tilde{y}^B_{b^e}$, we get $\tau^s_e(f_e(\theta)) \leq \tau^s_e(\tilde{y}^B_{b^e}) = \tau_{b^e}$, by Observation 3 and since τ^s_e is non-decreasing. This concludes the proof. □

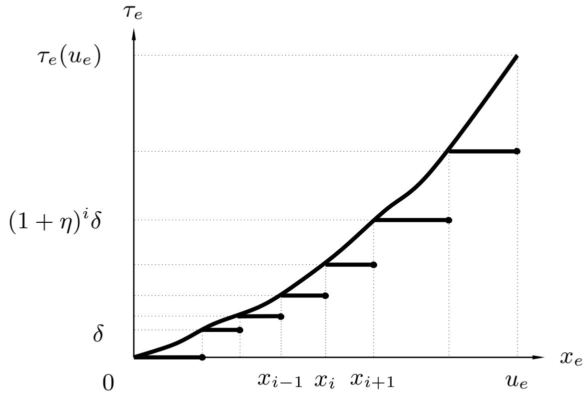

Fig. 5. Definition of step function.

Notice that, in contrast to the temporally repeated flow $\tilde{f}^B$ in G^B, the flow over time f in G uses storage of flow at intermediate nodes. More precisely, the flow arriving via arc e at node $v \in V$ at time $\theta + \tau_e^s(f_e(\theta))$ in f waits there for $\tau_{b^e} - \tau_e^s(f_e(\theta))$ time units until the corresponding flow in $\tilde{f}^B$ has also arrived. We call f a *temporally repeated flow (with inflow-dependent transit times)*. The underlying static flow y in G is given by $y_e := \tilde{y}^B_{b^e}$. Putting everything together, we can state the following result.

Theorem 1. *Consider an instance of the quickest flow problem with inflow-dependent transit times where all transit time functions are non-decreasing step functions. If there is a flow over time with inflow-dependent transit times sending D units of flow from s to t within time $\overline{T}$, then there exists a temporally repeated flow with inflow-dependent transit times satisfying demand D within time horizon at most $2\overline{T}$. Moreover, such a flow can be computed in strongly polynomial time.*

Proof. It follows from the description above that the temporally repeated flow f can be computed in strongly polynomial time. Moreover, $|f| = |\tilde{f}^B| = D$ and the time horizon of f is equal to T, the time horizon of $\tilde{f}^B$. Finally, by Lemma 2, $T \leq 2\overline{T}^B \leq 2\overline{T}$. □

Notice that in our analysis we have compared the value T of the computed solution f to the lower bound $\overline{T}^B$ given by an optimal solution f^B to the relaxation of the problem defined by the bow graph G^B. This yields the following result on the quality of this relaxation and thus on the quality of the time-expanded fan graph G^F.

Corollary 1. *The relaxation of the quickest flow problem on the bow graph G^B is a 2-relaxation, that is, the value of an optimal solution to the quickest flow problem with inflow-dependent transit times is within a factor of 2 of the value of an optimal solution to this relaxation.*

5 An Approximation for General Transit Time Functions

So far we have derived a 2-approximation for the case that all transit time functions are non-decreasing step functions. In this section, the approach is generalized to arbitrary non-decreasing transit time functions. The idea is to approximate them by step functions. In order to do this, we need the technical requirement that the transit time functions are left-continuous. See Figure 5 for a visualization of a possible step function as claimed in the following observation.

Observation 5. *Let $\delta, \eta > 0$. For every non-negative, non-decreasing, and left-continuous function $\tau : [0, u] \to \mathbb{R}^+$, there exists a step function $\tau^s : [0, u] \to \mathbb{R}^+$, with*

(i) $\tau^s(x) \le \tau(x) \le (1+\eta)\,\tau^s(x) + \delta$ *for every* $x \in [0,u]$,
(ii) *the number of breakpoints of* τ^s *is bounded by* $\lceil \log_{1+\eta}(\tau(u)/\delta) \rceil + 1$.

In the following we consider transit time functions τ_e^s for the arcs e of graph G, that fulfill the requirements stated in Observation 5. Then, an instance of the quickest flow problem on G with transit time functions τ_e^s, $e \in E$, is a relaxation of the corresponding instance with transit time functions τ, since $\tau_e^s(x) \le \tau_e(x)$ for all $e \in E$ and $x \in [0,u_e]$. In particular, the optimal time horizon $\overline{T}^s$ for transit times τ_e^s is a lower bound on the optimal time horizon $\overline{T}$ for transit times τ_e, $e \in E$. Moreover, we can state the following lemma.

Lemma 4. *Let* $\delta, \eta > 0$. *A temporally repeated flow* f^s *for transit time functions* τ_e^s *in* G *with time horizon* T^s *naturally induces a temporally repeated flow* f *for transit time functions* τ_e *in* G *with* $|f| \ge |f^s|$ *and time horizon* $(1+\eta)\,T^s + \delta\,|E|$.

Proof. Let $\mathcal{P}$ denote the set of s-t-paths used by f^s and y_P^s the flow rate on path $P \in \mathcal{P}$. Moreover, for every arc $e \in E$, let $y_e^s := \sum_{P \in \mathcal{P}:\, e \in P} y_P^s$. Then, flow in f^s traveling along path $P \in \mathcal{P}$ with transit time functions τ_e^s needs $\tau_P^s := \sum_{e \in P} \tau_e^s(y_e^s) \le T^s$ time to reach the sink. If we return to the original transit time functions τ_e, the transit time of path P increases to $\tau_P := \sum_{e \in P} \tau_e(y_e^s) \le (1+\eta)\tau_P^s + \delta\,|E|$ (see Observation 5 (i)). Therefore, the value of the corresponding temporally repeated flow f for transit time functions τ and time horizon $T' := (1+\eta)\,T^s + \delta\,|E|$ is

$$
\begin{aligned}
|f| \;&=\; T'\,|y^s| - \sum_{P \in \mathcal{P}} \tau_P\, y_P^s \ge\; T'\,|y^s| - \sum_{P \in \mathcal{P}} \big((1+\eta)\tau_P^s + \delta\,|E|\big)\, y_P^s \\
&\ge\; T'\,|y^s| - (\eta\, T^s + \delta\,|E|)\,|y^s| - \sum_{P \in \mathcal{P}} \tau_P^s\, y_P^s =\; T^s\,|y^s| - \sum_{P \in \mathcal{P}} \tau_P^s\, y_P^s \;=\; |f^s| \;.
\end{aligned}
$$

This concludes the proof. □

If the optimal time horizon $\overline{T}$, or at least a good estimate is known in advance, a provably good solution to the quickest flow problem can be computed in strongly polynomial time.

Lemma 5. *Given a lower bound* L *on* $\overline{T}$ *with* $L \le \overline{T} \le p(|E|)\,L$, *for some polynomial* p, *a temporally repeated flow satisfying demand* D *within time horizon at most* $(2+\epsilon)\,\overline{T}$ *can be computed in strongly polynomial time.*

Proof. Since any flow over time with time horizon $\overline{T}$ can send flow with rate at most $u_e' := \max\{x \in [0,u_e] \mid \tau_e(x) \le p(|E|)\,L\}$ into arc e at any moment in time, we can set $u_e := u_e'$, for all $e \in E$, without changing the optimal time horizon $\overline{T}$. Set $\delta := \epsilon\,L/(2\,|E|)$ and $\eta := \epsilon/4$. Then, by Observation 5 (ii), the number of breakpoints of τ_e^s is in $O(\log(|E|/\epsilon)/\epsilon)$ and thus strongly polynomially bounded in the input size.

Hence, by Theorem 1, for transit time functions τ_e^s a temporally repeated flow f^s in G with time horizon at most $2\overline{T}^s$ and value D can be computed in strongly polynomial time. By Lemma 4, f^s induces a solution f to the quickest flow problem for transit time functions τ_e with time horizon at most $(1+\eta)\,2\overline{T}^s + \delta\,|E| \le (2+\epsilon)\,\overline{T}$. □

In view of Lemma 5, it remains to show that a lower bound L with $L \le \overline{T} \le p(|E|)\,L$, for some polynomial p, can be found in strongly polynomial time.

Lemma 6. *There exists an algorithm with strongly polynomial running time which computes a lower bound* L *such that* $L \le \overline{T} \le p(|E|)\,L$, *for some polynomial* p.

1) Compute the lower bound L on $\overline{T}$ (see proof of Lemma 6) and set η and δ accordingly.
2) Construct a relaxed instance by replacing all transit time functions τ_e by step functions τ_e^s (see Observation 5).
3) Further relax the problem by considering the corresponding bow graph G^B with constant transit times on the arcs and compute a quickest flow there. Let y^B denote the underlying optimal static flow in G^B.
4) Turn y^B into a static flow y in the original graph G by setting y_e to the amount of flow sent through the expansion of arc e in y^B, for all $e \in E$.
5) Determine the time horizon T such that $T\,|y| - \sum_{e\in E} \tau_e(y_e)\, y_e \;=\; D$ and output the corresponding temporally repeated flow with inflow-dependent transit times in G.

Fig. 6. A high-level description of the algorithm claimed in Theorem 2.

Proof. For an arc $e \in E$, let $\ell_e := \min\{\tau_e(x) + D/(|E|\,x) \mid x \in (0, u_e]\} > 0$. Notice that ℓ_e is the minimal amount of time that is needed to send $D/|E|$ units of flow through arc e in a temporally repeated fashion. Furthermore, for an arbitrary s-t-path P in G, let $\ell_P := \max_{e\in P} \ell_e$. Then, ℓ_P is a lower bound on the amount of time that is needed to send $D/|E|$ units of flow through path P in a temporally repeated flow. Finally, set

$$L \;:=\; \min\{\ell_P/2 \mid P \text{ is an } s\text{-}t\text{-path in } G\}\ . \tag{3}$$

Notice that L can be computed by a Dijkstra-type algorithm in polynomial time.

We claim that L is a lower bound on $\overline{T}$. It follows from the proof of Lemma 5 that there exists a temporally repeated flow in G that sends D units of flow within time $2\overline{T}$. Since the s-t-paths used by this temporally repeated flow result from a path decomposition of the underlying static flow in G, their number can be bounded by $|E|$. In particular, there must exist an s-t-path P which carries at least $D/|E|$ units of flow in the temporally repeated solution. This yields $\ell_P \le 2\overline{T}$ and therefore $L \le \ell_P/2 \le \overline{T}$.

Finally, we show that $\overline{T} \le 2\,|E|^2\,L$. Let P be an s-t-path for which the minimum in (3) is attained. By definition of ℓ_P, and since P consists of at most $|E|$ edges, we can send $D/|E|$ units of flow from s to t on P within time $2\,|E|\,L$. By repeating this flow $|E|$ times, we can send D units of flow within time horizon $2\,|E|^2\,L$. □

We can now state the main result of this section, which follows from Lemmas 5 and 6. A compact description of the final algorithm is given in Figure 6.

Theorem 2. *There exists a $(2+\epsilon)$-approximation algorithm with strongly polynomial running time for the quickest flow problem with inflow-dependent transit times (given by left-continuous functions).*

6 An Improved Result for Concave Transit Time Functions

In this section we show that, for the special case of non-negative, non-decreasing, and concave transit time functions the algorithm described in Figure 6 achieves a better performance ratio.

Theorem 3. *Consider an instance of the quickest flow problem with inflow-dependent transit times where the transit time function of every arc is non-negative, non-decreasing, and concave. Then, a variant of the algorithm in Figure 6 achieves performance ratio $3/2+\epsilon$.*

In fact, we slightly modify the algorithm by defining $\eta := \epsilon/6$ (instead of $\epsilon/4$ as in the proof of Lemma 5). The parameter $\delta := \epsilon\, L/(2\,|E|)$ remains unchanged. Let y^B denote the static flow in G^B computed in Step 3 of the algorithm and y the corresponding static flow in G computed in Step 4. We first bound the total transit time $\sum_{e\in E} \tau_e(y_e)\, y_e$ of y.

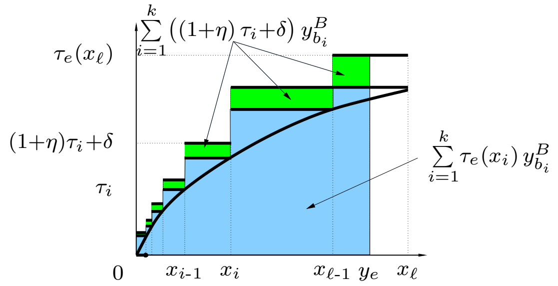

Fig. 7.

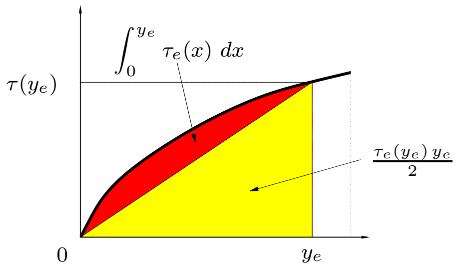

Fig. 8.

Lemma 7. *Let $\overline{T}$ be the optimal time horizon of a quickest flow in G. Then,*

$$\frac{1}{2}\sum_{e\in E}\tau_e(y_e)\, y_e \;\leq\; (1+\epsilon/6)\sum_{a\in E^B}\tau_a\, y^B_a + \epsilon\,|y|\,\overline{T}/2 \;. \tag{4}$$

Proof. Consider the expansion E^B_e of a fixed arc $e \in E$ consisting of bow arcs $b_1,\dots,b_k$ with transit times $\tau_1,\dots,\tau_k$ and of regulating arcs $r_1,\dots,r_k$ with capacities $x_1,\dots,x_k$. Since all regulating arcs have zero transit time, they do not contribute to the sum on the right hand side of (4). Moreover, by definition of y, we have $\sum_{i=1}^{k} y^B_{b_i} = y_e \leq |y|$. Therefore, it suffices to show that

$$\tfrac{1}{2}\,\tau_e(y_e)\, y_e \;\leq\; \sum_{i=1}^{k}\big((1+\eta)\,\tau_i+\delta\big)\, y^B_{b_i} \;, \tag{5}$$

since summing up these inequalities over all arcs $e \in E$ yields the statement of the lemma. Notice that the value of the step function approximating τ_e when evaluated at point x_i is τ_i. Therefore, the first property stated in Observation 5 yields

$$(1+\eta)\,\tau_i+\delta \;\geq\; \tau_e(x_i) \tag{6}$$

(see also Figure 7). Assume now that y^B is sending flow along bow arcs b_1 to b_ℓ. Then, the right hand side of (5) can be bounded as follows

$$\begin{aligned}
\sum_{i=1}^{k}\big((1+\eta)\,\tau_i+\delta\big)\, y^B_{b_i} \;&\geq\; \sum_{i=1}^{\ell}\tau_e(x_i)\, y^B_{b_i} && \text{by (6),}\\
&=\; \sum_{i=1}^{\ell-1}\tau_e(x_i)\,(x_i-x_{i-1}) \;+\; \tau_e(x_\ell)\,(y_e-x_{\ell-1}) && \text{by Lemma 1,}\\
&\geq\; \int_0^{y_e}\tau_e(x)\; dx \;\geq\; \tau_e(y_e)\, y_e/2 \;.
\end{aligned}$$

An illustration of the first two inequalities is given in Figure 7. The last inequality holds since the function τ_e is non-negative, non-decreasing, and concave; see Figure 8. □

Proof (of Theorem 3). Let $(y_P)_{P\in\mathcal{P}}$ be a path decomposition of y. The first property stated in Observation 5 and Observation 1 yield

$$\tau_P(y) \;:=\; \sum_{e\in P}\tau_e(y_e) \;\leq\; \sum_{e\in P}\big((1+\eta)\,\tau^s_e(y_e)+\delta\big) \;\leq\; (1+2\epsilon/3)\,\overline{T} \;. \tag{7}$$

As a consequence we get

$$\begin{aligned}
&(3/2+\epsilon)\,\overline{T}\,|y| \;-\; \sum_{e\in E}\tau_e(y_e)\,y_e \\
&\quad=\; (1+2\epsilon/3)\,\overline{T}\,|y| \;+\; \frac{1}{2}\sum_{P\in\mathcal{P}}\Big((1+2\epsilon/3)\overline{T}-\tau_P(y)\Big)\,y_P \;-\; \frac{1}{2}\sum_{e\in E}\tau_e(y_e)\,y_e \\
&\quad\geq\; (1+2\epsilon/3)\,\overline{T}\,|y| \;-\; \frac{1}{2}\sum_{e\in E}\tau_e(y_e)\,y_e && \text{by (7)}, \\
&\quad\geq\; (1+\epsilon/6)\,\overline{T}\,|y| \;-\; (1+\epsilon/6)\sum_{a\in E^B}\tau_a\,y_a^B && \text{by Lemma 7}, \\
&\quad=\; (1+\epsilon/6)\Big(\overline{T}\,|y^B| \;-\; \sum_{a\in E^B}\tau_a\,y_a^B\Big) \;=\; (1+\epsilon/6)\,D \;\geq\; D\;.
\end{aligned}$$

This concludes the proof. □

Lower bounds. The following example of a single arc shows that already in the case of linear transit time functions the algorithm can be off the optimal time horizon $\overline{T}$ by a factor of $\sqrt{2}$. Consider an arc $a=(u,v)$ with transit time function $\tau_a(x)=x$ on $[0,2]$. We want to send two units of flow from s to t as quickly as possible. The optimal flow is defined by $f_a(\theta)=2-\theta$, $\theta\in[0,2)$, it completes at time $\overline{T}=2$. For every point in time, it is sending flow at the highest possible rate such that every flow unit arrives not later than time 2. A temporally repeated flow can only send 1 unit of flow from s to t within the same time horizon by choosing a constant flow rate $y_a=1$. How much larger do we need to choose the time horizon to ensure that 2 units of flow can be sent? If we choose a constant flow rate y_a and increase the time horizon $\overline{T}=2$ by a factor $\alpha>1$, we manage to send $(\alpha\cdot\overline{T}-\tau_a(y_a))y_a=(\alpha\cdot 2-y_a)y_a$ units of flow from s to t. To guaranty that two units of flow arrive in t, one can easily check that α has to be chosen to be at least $\sqrt{2}$.

A similar analysis applied to the same instance with transit time function chosen to be $\tau_a(x)=x^{e-1}$ gives a slightly better lower bound of $e^{1/e}\geq 1.444$.

Generalizing Ford and Fulkerson's algorithm. We have shown that there always exists a temporally repeated flow solution to the quickest flow problem whose time horizon is not larger than twice the optimal time horizon $\overline{T}$. More precisely, we have described an algorithm which, for every $\epsilon>0$, constructs a temporally repeated flow solution that finishes not later than $(2+\epsilon)\overline{T}$. All computations are performed in a related network, the bow graph.

The question arises if there is a direct way to compute temporally repeated solutions without having the overhead of constructing the bow graph. Indeed, we can consider a Ford & Fulkerson-type formulation for this problem: find a static flow y in G maximizing $T|y|-\sum_{e\in E}\tau_e(y_e)y_e$. A solution to this maximization problem can be turned into a temporally repeated flow with time horizon T and value equal to the objective value. In particular, for $T\geq 2\overline{T}$, this value is at least D. Moreover, if all transit time functions are convex, this problem can be transformed into a minimum convex cost flow problem with transit times interpreted as costs.

This leads to the following algorithm: search for an estimate $\tilde{T}$ of $\overline{T}$ satisfying $\overline{T}\leq\tilde{T}\leq(1+\epsilon/2)\overline{T}$ (recall that Lemmma 6 yields a sufficiently small search interval $[L,p(|E|)L]$ containing $\overline{T}$ to which we can apply a binary search). Solve the maximization problem above with T set to $2\tilde{T}$. This produces a temporally repeated solution satisfying demand D in at most time $2\tilde{T}\;\leq\;(2+\epsilon)\overline{T}$.

The approximation algorithm achieves performance ratio $2+\epsilon$. Again, the example discussed above shows that its performance ratio is not better than $e^{1/e}$. A major drawback of this approach

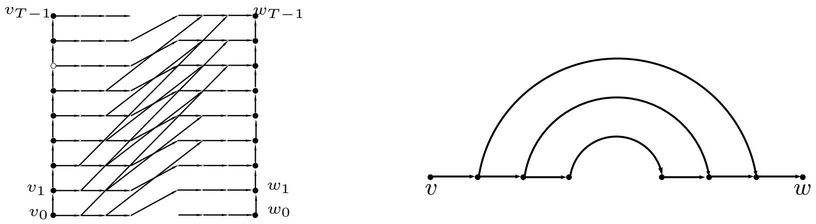

Fig. 9. Generalized fan and bow graph.

is that all transit time functions have to be convex, while the algorithm presented in this paper works for general transit time functions.

7 Concluding Remarks

We have defined the fan graph as a generalization of the time-expanded graph in order to be able to attack flows over time with flow-dependent transit times by standard network flow techniques. While Ford and Fulkerson's time-expanded graph completely captures the notion of flows over time with constant transit times, the fan graph is only a relaxation for the setting of inflow-dependent transit times; flow units entering an arc simultaneously, might travel through the arc at different paces. Nevertheless, we were able to show that an optimal solution to the relaxed quickest flow problem can be turned into a provably good feasible flow over time. In particular, this result suggests that static flows in the fan graph provide a reasonable approximation for the actual setting of flows over time with flow-dependent transit times.

These first results for the fan graph model emphasize its suitability for being used also in practical applications. Here, plausible models which allow efficient computations also for very large scale networks are badly needed. From the perspective of practical applications, also interesting variations of the presented model seem to be worth studying. One very natural such modification is the generalization to a model that captures in- and outflow dependent transit times. Similarly to the definition of travel time expansions of the arcs according to the flow entering an arc (see Sect. 3), one could refine the model by introducing a similar expansion also at the head of each arc (see Figure 9). This latter model better reflects the behavior of flow in practical applications, as for example traffic in street networks.

The approximation results presented in this paper are partly inspired by the work of Fleischer and Skutella [9] and the subsequent results of Köhler and Skutella [15]. The latter paper presents a $(2+\epsilon)$-approximation algorithm for the quickest s-t-flow problem in the setting of load-dependent transit times. More precisely, this result holds for the case of convex, non-decreasing transit time functions $\tau_e(y_e)$ in this setting. Interestingly, the analysis in [15] can easily be carried over to the model of flows over time with inflow-dependent transit times. In fact, one can check that the generated solutions are $(2+\epsilon)$-approximations for both models, the load-dependent model and the inflow-dependent model. This implies a surprisingly close relationship between these rather different flow models. If all transit time functions $\tau_e(y_e)$ are convex and non-decreasing, then the time horizon of a quickest s-t-flow in the load-dependent model and the time horizon of a quickest s-t-flow in the inflow-dependent model cannot differ by more than a factor of 2.

Notice, however, that the algorithm in [15] can only be applied if all transit time functions are convex, while we allow a considerably richer family of transit time functions in this paper. Also, the stronger performance ratio for the case of concave transit time functions does not seem to translate to the model of flows over time with load-dependent transit times. In fact, the solution of the algorithm in [15] and the solution of the algorithm presented in this paper can differ by a factor arbitrarily close to 2, even for the case of constant transit times. Consider the following example consisting of only one arc e from s to t with fixed transit time $\tau_e \geq 1$ and unit capacity. A quickest s-t-flow for demand $D := 1$ sends flow at rate 1 through the arc and therefore needs $1 + \tau_e$ time units to complete. Since the fan graph precisely captures the case of fixed transit times, this is also the solution which is computed by the algorithm presented in this paper. However, the algorithm presented in [15] outputs a flow over time with flow rate $1/\tau_e$ and time horizon $2\tau_e$. Thus, for large τ_e, the ratio of the two solution values approaches 2.

Acknowledgments. The authors are much indebted to Lisa Fleischer, Rolf Möhring, and Günter Rote for helpful comments and interesting discussions on the topic of this paper. This work was supported in part by the EU Thematic Networks APPOL I & II, Approximation and Online Algorithms, IST-1999-14084 and IST-2001-30012, by the European graduate program 'Combinatorics, Geometry, and Computation', Deutsche Forschungsgemeinschaft, grant GRK 588/1, and by the Bundesministerium für Bildung und Forschung (bmb+f), grant no. 03-MOM4B1.

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. J. E. Aronson. A survey of dynamic network flows. *Annals of Operations Research*, 20:1–66, 1989.
3. M. Ben-Akiva, H. Koutsopoulos, R. Mishalani, and Q. Yang. Simulation laboratory for evaluating dynamic-traffic management systems. *ASCE Journal of Transportation Engineering*, 123:283–289, 1997.
4. R. E. Burkard, K. Dlaska, and B. Klinz. The quickest flow problem. *ZOR — Methods and Models of Operations Research*, 37:31–58, 1993.
5. M. Carey. A constraint qualification for a dynamic traffic assignment model. *Transportation Science*, 20:55–58, 1986.
6. M. Carey. Optimal time-varying flows on congested networks. *Operations Research*, 35:58–69, 1987.
7. M. Carey and E. Subrahmanian. An approach to modelling time-varying flows on congested networks. *Transportation Research B*, 34:157–183, 2000.
8. S. P. Dirkse and M. C. Ferris. Traffic modeling and variational inequalities using GAMS. In Ph. L. Toint, M. Labbe, K. Tanczos, and G. Laporte, editors, *Operations Research and Decision Aid Methodologies in Traffic and Transportation Management*, volume 166 of *NATO ASI Series F*, pages 136–163. Springer, Berlin, 1998.
9. L. Fleischer and M. Skutella. The quickest multicommodity flow problem. In W. J. Cook and A. S. Schulz, editors, *Integer Programming and Combinatorial Optimization*, volume 2337 of *Lecture Notes in Computer Science*, pages 36–53. Springer, Berlin, 2002.
10. L. Fleischer and M. Skutella. Minimum cost flows over time without intermediate storage. In *Proceedings of the 14th Annual ACM–SIAM Symposium on Discrete Algorithms*, pages 66–75, Baltimore, MD, 2003.
11. L. K. Fleischer and É. Tardos. Efficient continuous-time dynamic network flow algorithms. *Operations Research Letters*, 23:71–80, 1998.
12. L. R. Ford and D. R. Fulkerson. Constructing maximal dynamic flows from static flows. *Operations Research*, 6:419–433, 1958.
13. L. R. Ford and D. R. Fulkerson. *Flows in Networks*. Princeton University Press, Princeton, NJ, 1962.
14. N. Gartner, C. J. Messer, and A. K. Rathi. Traffic flow theory: A state of the art report. http://www-cta.ornl.gov/cta/research/trb/tft.html, 1997.
15. E. Köhler and M. Skutella. Flows over time with load-dependent transit times. In *Proceedings of the 13th Annual ACM–SIAM Symposium on Discrete Algorithms*, pages 174–183, San Francisco, CA, 2002.
16. H. S. Mahmassani and S. Peeta. System optimal dynamic assignment for electronic route guidance in a congested traffic network. In N. H. Gartner and G. Improta, editors, *Urban Traffic Networks. Dynamic Flow Modelling and Control*, pages 3–37. Springer, Berlin, 1995.
17. N. Megiddo. Combinatorial optimization with rational objective functions. *Mathematics of Operations Research*, 4:414–424, 1979.
18. D. K. Merchant and G. L. Nemhauser. A model and an algorithm for the dynamic traffic assignment problems. *Transportation Science*, 12:183–199, 1978.
19. D. K. Merchant and G. L. Nemhauser. Optimality conditions for a dynamic traffic assignment model. *Transportation Science*, 12:200–207, 1978.

20. K. Nagel, J. Esser, and M. Rickert. Large-scale traffic simulations for transportation planning. In D. Stauffer, editor, *Annual Review of Computational Physics VII*, pages 151–202. World Scientific Publishing Company, 2000.
21. K. Nagel, P. Wagner, and R. Woesler. Still flowing: Approaches to traffic flow and traffic jam modeling. *Operations Research*, 2003. To appear.
22. W. B. Powell, P. Jaillet, and A. Odoni. Stochastic and dynamic networks and routing. In M. O. Ball, T. L. Magnanti, C. L. Monma, and G. L. Nemhauser, editors, *Network Routing*, volume 8 of *Handbooks in Operations Research and Management Science*, chapter 3, pages 141–295. North–Holland, Amsterdam, The Netherlands, 1995.
23. I. Prigogine and R. Herman. *Kinetic Theory of Vehicular Traffic*. Elsevier Science B. V., Amsterdam, 1971.
24. B. Ran and D. E. Boyce. *Dynamic Urban Transportation Network Models: Theory and Implications for Intelligent Vehicle-Highway Systems*, volume 417 of *Lecture Notes in Economics and Mathematical Systems*. Springer, Berlin, 1994.
25. B. Ran and D. E. Boyce. *Modelling Dynamic Transportation Networks*. Springer, Berlin, 1996.
26. Y. Sheffi. *Urban Transportation Networks*. Prentice-Hall, New Jersey, 1985.