



Earliest Arrival Flows in Networks with Multiple Sinks

Melanie Schmidt²

*Lehrstuhl II, Fakultät für Informatik
Technische Universität Dortmund
Dortmund, Germany*

Martin Skutella^{1,3}

*Fakultät II, Institut für Mathematik
Technische Universität Berlin
Berlin, Germany*

Abstract

Earliest arrival flows model a central aspect of evacuation planning: In a dangerous situation, as many individuals as possible should be rescued *at any point in time*. Unfortunately, given a network with multiple sinks, flows over time satisfying this condition do not always exist. We analyze the special case of flows over time with zero transit times and characterize which networks always allow for earliest arrival flows.

Keywords: earliest arrival flow, flow over time, evacuation problems

¹ Supported by DFG Research Center MATHEON in Berlin.

² Email: melanie.schmidt@cs.tu-dortmund.de

³ Email: martin.skutella@tu-berlin.de

1 Introduction

Flows over time provide a valuable tool to model evacuation situations. While *quickest* flows over time capture the idea that all individuals shall leave a dangerous area as quickly as possible, *earliest arrival* flows additionally require that the number of individuals already saved shall be maximal at every point in time. This ensures that an earliest arrival flow is optimal regardless of the time actually available for the evacuation. However, the goal to compute a flow satisfying the earliest arrival property is somewhat ambitious, and indeed not even the existence of such flows is assured in general. It turns out that the number of sinks (modelling exits/escapes) is crucial in this context: Minieka [6] shows that earliest arrival flows do always exist if there is only one sink. On the other hand, very simple examples of networks are known that do not allow for an earliest arrival flow [1,2]. Consequentially, a lot of research has been devoted to single-sink networks and algorithms to compute earliest arrival flows in this special situation, both for single-source-single-sink networks [5,6,9] and for multiple-sources-single-sink networks [1,2,3].

But how do we proceed if we encounter a situation where multiple sinks are inevitable (e. g., a ship with several life-boats or widespread pick-up bus stations in an urban evacuation)? Although in this setting earliest arrival flows may not exist in general, it might very well be the case that they do exist in the special case at hand. The question which networks with multiple sinks still allow for earliest arrival flows was stated as an open problem in [1] and has, to our knowledge, not been studied before. In this paper, we analyze flows over time in networks where all edge transit times are zero. Surprisingly, even in this reduced scenario, earliest arrival flows do not necessarily exist. However, we give a classification of the class of all networks that always allow for earliest arrival flows, regardless of the number of individuals, the capacity of passages and the capacity of the sinks.

Due to space restrictions, many details are omitted in this extended abstract. A complete description of all results and proofs can be found in [7] and will appear in the full version of the paper.

2 Preliminaries

We consider directed networks N consisting of a directed graph $G = (V, E)$, capacities $u : E \rightarrow \mathbb{Z}_{\geq 0}$ and balances $\nu : V \rightarrow \mathbb{Z}$. The capacity of an edge restricts the amount of flow that can travel through the edge in one time step. We use the balance function ν to denote how much flow a node wants to send

into the network. Nodes $s \in V$ with positive value $\nu(s) > 0$ are called *sources*. A node $t \in V$ with negative balance $\nu(t) < 0$ wishes to receive $-\nu(s)$ units of flow and is called *sink*, and by S^+ and S^- we denote the set of all sources and sinks, respectively. We demand that $\sum_{v \in V} \nu(v) = 0$.

A *flow over time*⁴ with zero transit times $f : E \times \mathbb{Z}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is a function that assigns a flow value to each edge at each point in time. As we have no transit times, flow sent into an edge will arrive at the end of the edge in the same time step. We demand that a flow f respects the capacity constraint $f(e, t) \leq u(e)$ on each edge $e \in E$ at any point in time $t \in \mathbb{Z}_{\geq 0}$. Additionally, f has to satisfy flow conservation at all nodes except the sources and sinks: At any point in time, the amount of flow entering the node has to be equal to the amount of flow leaving the node, i. e., for all t in $\mathbb{Z}_{\geq 0}$ and for all nodes v in $V \setminus (S^+ \cup S^-)$ it holds $\text{value}(v, t) = 0$, where

$$\text{value}(v, t) = \sum_{e=(v,u) \in E} f(e, t) - \sum_{e=(u,v) \in E} f(e, t)$$

is the amount of flow sent out of v at time t . A maximum flow over time with zero transit times with time horizon T has the additional property that $\sum_{t=1}^T \sum_{s \in S^+} \text{value}(s, t)$ is maximal. In the following, we will omit the term ‘with zero transit times’, since all variants of flows over time discussed here are flows with zero transit times.

A *transshipment over time* is a flow over time that sends exactly $\nu(v)$ units of flow out of node v for all nodes $v \in V$. For sinks, this implies that flow is actually sent *into* the sink, for intermediate nodes $v \notin (S^+ \cup S^-)$ this only reformulates the flow conservation. We are interested in special transshipments: A *quickest transshipment over time* is a transshipment over time where all balances are cancelled out as quickly as possible, i. e. the first point in time t^* where $\sum_{t=1}^{t^*} \text{value}(v, t) = \nu(v)$ holds for all nodes $v \in V$, is minimized. *Earliest arrival transshipments over time* are special quickest transshipments over time that simultaneously maximize the flow already sent for all points in time, i. e., $\sum_{t=1}^{t'} \sum_{s \in S^+} \text{value}(s, t)$ equals the value of a maximum flow over time with time horizon t' for all $t' \in \mathbb{N}_{\geq 0}$.

Why are several sinks so problematic for the existence of earliest arrival flows? Fleischer [2] states the main difficulty as follows: ‘*The problem with multiple sinks is that, in the rush to send flow, some source may send flow into the wrong sink. This is not a problem when there is only one sink.*’ Surprisingly, this can already be seen in very small examples.

⁴ For a detailed introduction to flows over time we refer to [8].

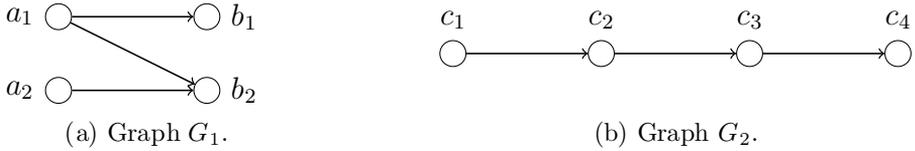


Figure 1. Two graphs that do not always allow for earliest arrival transshipments.

Example 2.1 We define two graphs (see Figure 1) where the existence of earliest arrival transshipments is not guaranteed:

$$G_1 = (V_1, E_1) \text{ with } V_1 = \{a_1, a_2, b_1, b_2\}, E_1 = \{(a_1, b_1), (a_1, b_2), (a_2, b_2)\} \text{ and}$$

$$G_2 = (V_2, E_2) \text{ with } V_2 = \{c_1, c_2, c_3, c_4\}, E_2 = \{(c_1, c_2), (c_2, c_3), (c_3, c_4)\}.$$

By assigning capacities $u_1 \equiv 1$ and balances $\nu(a_1) = \nu(a_2) = 2, \nu(b_1) = \nu(b_2) = -2$, we get a network N_1 where no earliest arrival transshipment exists: In the first time step, the best we can do is to send one unit of flow on each edge, achieving a flow value of three. By this, the supply of a_1 and the demand of b_2 are reduced to zero. This implies that no more flow can be sent, i. e., the amount of flow in time step two is also three. On the other hand, leaving the edge (a_1, b_2) empty in the first step allows us to send two units of flow again in the second step, so that we can send all flow in two steps. Thus we can either send three units in the first time step or four units in the first and second time step, but not both, in other words there is no earliest arrival transshipment.

Likewise, assigning $u_2(c_1, c_2) = u_2(c_3, c_4) = 2, u_2(c_2, c_3) = 1$ and $\nu_2(c_1) = 4, \nu_2(c_2) = -2, \nu_2(c_3) = 2$ and $\nu_2(c_4) = -4$ leads to a network N_2 without an earliest arrival transshipment: The flow in the first time step is maximized by sending two units of flow from c_1 to c_2 and, in parallel, sending two units of flow from c_3 to c_4 , but this will completely balance c_2 and c_3 , such that the remaining two units can only be send in different time steps, leading to a total of three time steps. Then again, sending only three units of flow in the first time step helps: By sending one unit from c_1 to c_2 , one from c_1 to c_4 and one from c_3 to c_4 we achieve less in the first step, but can balance all nodes in the second one by repeating the same flow.

As we will see later on, these two examples are crucial for deciding whether earliest arrival transshipments exist.

3 Networks allowing for earliest arrival flows

As we already mentioned, earliest arrival flows do always exist in networks where $|S^-| = 1$, even in the much more general setting with arbitrary transit times [1,4,6]. In the special case of zero transit times, the following lemma implies that they also exist if $|S^+| = 1$.

Lemma 3.1 *Let N be a directed network on a directed graph $G = (V, E)$ with capacities u and balances ν . Define $G_r = (V, E_r)$ by $E_r = \{(w, v) \mid (v, w) \in E\}$. The directed network N_r is the network consisting of G_r , capacities $u_r := u$ and balances $\nu_r : V \rightarrow \mathbb{Z}$ with $\nu_r(v) = -\nu(v)$ for all $v \in V$.*

N allows for an earliest arrival transshipment iff N_r allows for an earliest arrival transshipment.

We note that this implies that a network without an earliest arrival transshipment has to have at least two sources and two sinks, and thus the examples given above are in this sense minimal. Additionally, the second example shows that the class of *trees* is not a valid candidate as a graph class always allowing for earliest arrival transshipments. The graph G_2 is even an in-tree as well as an out-tree of depth three. An in-tree with root r is a directed graph satisfying that there is exactly one directed path from v to r for every node v . Similar, an out-tree with root r satisfies that there is exactly one directed path from r to every v for every node v . The depth of an in- or out-tree is the number of edges on the longest path contained in it. To ensure the existence of earliest arrival flows, we have to restrict the graph class even more.

Lemma 3.2 *Let B be an in- or out-tree with depth $d \leq 2$. Then B allows for an earliest arrival transshipment for all choices of capacities and balances.*

We have a closer look on another class of graphs allowing for earliest arrival transshipments. By G_u we denote the undirected graph induced by a directed graph G , and $G - \{v\}$ is the graph that results from deleting v and all adjacent edges from G .

Lemma 3.3 *Let N_v be a network on a graph $G = (V, E)$ that satisfies the following property: There exists a node $v \in V$ such that $G_u - \{v\}$ consists of two connected components $G_u^L = (V_u^L, E_u^L)$ and $G_u^R = (V_u^R, E_u^R)$ with $S^+ \subseteq V_u^L$ and $S^- \subseteq V_u^R$. Then N_v allows for an earliest arrival transshipment for all choices of capacities and balances that respect S^+ and S^- .*

Proof (idea). The definition of N_v already suggests how to proceed: We take advantage of the fact that v 'looks like' a sink to one part of N_v and like a

source to the other part. Each of the smaller networks allows for an earliest arrival transshipment, and combining the two flows in a clever way leads to an earliest arrival transshipment in the entire network. \square

With the above in mind, we can now proceed to characterize the class of graphs that *always* allow for earliest arrival transshipments, i.e. such a transshipment exists regardless of the capacities and balances we choose.

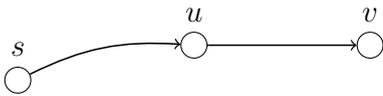
Theorem 3.4 *Let N be a directed network on graph G . Network N allows for an earliest arrival transshipment for all choices of capacities and balances iff G contains no subdivision of the graphs G_1 and G_2 .*

Proof (sketch) It is easy to see that subdivisions G_1 and G_2 may not be included in G if earliest arrival transshipments are supposed to exist regardless of capacities and balances: By setting the capacity of all edges outside the subdivision to zero and then adjusting the remaining capacities and the balances according to Example 2.1, we can create a network where no earliest arrival transshipment exists.

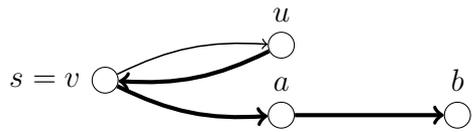
We show that graphs not containing these subdivisions are either very small or have a nice structure that supports earliest arrival transshipments. We consider connected graphs, otherwise the connected components can be checked independently. Let $G = (V, E)$ be a connected graph that does not contain any subdivision of G_1 and G_2 . In particular, G cannot contain any path of length three or more (by the length we denote the number of edges).

Case 1: The first case that we investigate is that G does not even contain a path of length two. Let $v \in V$ be a node with at least two outgoing edges. We notice that v cannot have any incoming edge (because that would create a path of length two), and the endpoints of outgoing edges of v cannot have any outgoing edges (by the same argument). Additionally, these endpoints cannot have incoming edges because that would complete a G_1 subgraph. We see that v is the only node in G that has outgoing edges, therefore G allows for an earliest arrival transshipment because there is only one possible source. Analogously the existence of a node with two incoming edges leads to the conclusion that G has only one possible sink. The only remaining possibility is that G is a single edge.

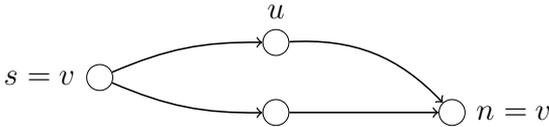
Case 2: In this case we assume that G contains a path of length two that is either parallel to another path of length two or to one of length one (i.e., an edge). A case distinction shows that this implies that G cannot contain additional nodes. What remains to show is in a thus restricted graph, earliest arrival transshipments always exist. This can be done by constructing the



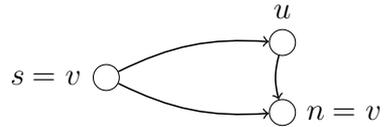
(a) If $v \notin V_B$, adding v to B does not interfere with the out-tree property, and the depth is still bounded by two.



(b) If $w = v$, then $(u, s), (s, a)$ and (a, b) or $(u, w), (w, x)$ and (x, y) form a G_2 .



(c) If $v = n$ for $n \in V_B$, but $v \neq s$, we get case 1. Nodes s and n are either connected directly ...



(d) ... or indirectly.

Figure 2. An illustration of the first subcase of case three (proof of Theorem 3.4).

transshipment explicitly.

Case 3: Now we cover the situation that G contains two paths starting in the same node s , but continuing disjointly. Let (s, a) and (a, b) be the edges of the first path and $(s, x), (x, y)$ be the edges of the second path. We will see that G is an out-tree whose depth is bounded by two. The root of the out-tree is node s , as we can already guess by observing that s cannot have any incoming edges. We consider the subgraph $B = (V_B, E_B)$ of G that is defined in the following way: B is an out-tree with root s , and adding any edge $e = (u, v) \in E \setminus E_B$ with $u \in V_B$ or $v \in V_B$ to B destroys this property. We show that there can be no edge in $E \setminus E_B$ and thus B equals G .

Assume $e = (u, v) \in E \setminus E_B$ and $u \in V_B$. Because B is an out-tree with root s , B contains a path from s to u . Assume this path only consists of the edge (s, u) . We observe that there is no place where u could be in G that would not violate an assumption by systematically iterating through all possibilities as shown in Figure 2. Alternatively, the path from s to u could contain an intermediate node. This can be disproved in the same way. Combined, we gathered that $u \in V_B$ is not possible. To see that $v \in V_B$ (and $u \notin V_B$) is not possible either, we observe the following: As B is an out-tree with root s , there is a path p from s to v . Additionally, there has to be at least one other outgoing edge e from s that has an endpoint not contained in p . But as p, e and (u, v) form the graph G_1 , this is not possible.

As an out-tree of depth at least two, G allows for earliest arrival transshipments because of Lemma 3.2.

Case 4: If G contains two paths of length two that start in different nodes, but end in the same node, we can use Lemma 3.1 and case 3 to see that G always contains an earliest arrival transshipment.

Case 5: It remains to see how G can look like if neither of the first four cases occurs. In particular this means that G contains at least one path of length two. Starting with this path, we investigate which additional nodes and edges can be part of G . A case distinction shows that G is either a star-shaped graph with one central node that allows for earliest arrival transshipments due to Lemma 3.3 or it has a structure very similar to ordered trees such that the existence of earliest arrival transshipments can be shown in an analogous way as Lemma 3.2. \square

References

- [1] Baumann, N. and M. Skutella, *Solving evacuation problems efficiently: Earliest arrival flows with multiple sources*, *Mathematics of Operations Research* **34** (2009), pp. 499–512.
- [2] Fleischer, L. K., *Faster algorithms for the quickest transshipment problem*, *SIAM Journal on Optimization* **12** (2001), pp. 18–35.
- [3] Fleischer, L. K. and M. Skutella, *Quickest flows over time*, *SIAM Journal on Computing* **36** (2007), pp. 1600–1630.
- [4] Gale, D., *Transient flows in networks*, *Michigan Mathematical Journal* **6** (1959), pp. 59–63.
- [5] Hoppe, B. and É. Tardos, *The quickest transshipment problem*, *Mathematics of Operations Research* **25** (2000), pp. 36–62.
- [6] Minieka, E., *Maximal, lexicographic, and dynamic network flows*, *Operations Research* **21** (1973), pp. 517–527.
- [7] Schmidt, M., “Earliest Arrival Flows mit mehreren Senken,” Master’s thesis, Technische Universität Dortmund (2009), in German.
- [8] Skutella, M., *An introduction to network flows over time*, in: W. Cook, L. Lovász and J. Vygen, editors, *Research Trends in Combinatorial Optimization*, Springer, 2009 pp. 451–482.
- [9] Wilkinson, W. L., *An algorithm for universal maximal dynamic flows in a network*, *Operations Research* **19** (1971), pp. 1602–1612.