



22nd International Symposium on Transportation and Traffic Theory

## Optimal Hyperpaths With Non-Additive Link Costs

Saeed Maadi<sup>a</sup> and Jan-Dirk Schmöcker<sup>a\*</sup>

<sup>a</sup>Department of Urban Management, Kyoto University, C1-2-436, Katsura Nishikyo-ku, Kyoto 615-8540, Japan

---

### Abstract

Non-additive link cost functions are common and important for a range of assignment problems. In particular in transit assignment, but also a range of other problems path splits further need to consider node cost uncertainties leading to the notion of hyperpaths. We discuss the problem of finding optimal hyperpaths under non-additive link cost conditions assuming a cost vector with a limited number of marginal decreasing costs depending on the number of links already traversed. We illustrate that these non-additive costs lead to violation of Bellman's optimality principle which in turn means that standard procedures to obtain optimal destination specific hyperpath trees are not feasible. To overcome the problem we introduce the concepts of a "travel history vector" and active critical, passive critical and fixed nodes. The former records the expected number of traversed links until a node, and the latter distinguishes nodes for which the cost can be determined deterministically. With this we develop a 2-stage solution approach. In the first stage we test whether the optimal hyperpath can be obtained by backward search. If this is not the case, we propose a so called "selective hyperpath generation" among hyperpaths to a (small) number of active critical nodes and combine this with standard hyperpath search. We illustrate our approach by applying it to the Sioux Falls network showing that even for link cost functions with large step changes we are able to obtain optimal hyperpaths in a reasonable computational time.

© 2017 The Authors. Elsevier B.V. All rights reserved.

Peer-review under responsibility of the scientific committee of the 22nd International Symposium on Transportation and Traffic Theory.

*Keywords:* Non-additive costs; Bellman principle; Hyperpath; Optimal Strategies; Frequency-based Assignment

---

\* Corresponding author. Tel.: +81-75-383-7491; fax: +81-75-383-3236.

E-mail address: [schmoecker@trans.kuciv.kyoto-u.ac.jp](mailto:schmoecker@trans.kuciv.kyoto-u.ac.jp)

## 1. Introduction

Non-additivity of cost functions along paths in a transport network is an issue for a range of applications. Energy efficiency might depend on driven distance. Regulations such as rest-times for truck drivers or delivery time windows might introduce step functions in the cost function if certain distances/travel times are exceeded. With GNSS road pricing also non-additivities might be introduced to support or discourage long distance trips. Gabriel and Bernstein (1997) provide a general discussion on the need for non-additive costs in traffic assignment and Chen and Nie (2013) extend this. As noted also in Han and Lo (2004), an application where non-linear costs are arguably most common are public transport fares. In most systems the traveler has to pay a fee to enter the system plus additional zone or distance depending fares. In case of distance-depending fares these are usually degressive in that the marginal fare per distance is decreasing. Combinations of flat, zonal and distance-depending fares plus additional features such as peak-hour surcharges are further common practice. To clarify, we are interested in this paper in cases where the fare is not predetermined by the origin and destination but depends on the actual path chosen. In Japan many fare structures comply with this criteria. Furthermore, besides fares, we might argue that congestion aspects, again particular transit congestion, also fulfill our problem description. Travelers on crowded trains often do not mind standing for one or a few stops but would make a significant effort to avoid having to stand over longer distances. In this case, and in opposite to the fare case, we would expect the cost function to be disproportionately ascending with distance though.

In case travelers can be assigned to single paths, non-additivity might be dealt with in an exact manner and network loading is fairly straightforward. However, in case the network is large and one has to consider that travelers en-route react to delays by changing their route, non-additivity forms an important challenge as we will illustrate and discuss in this paper. Considering uncertainty in path choice due to potential delays has led to the concept of hyperpaths where travelers' choices at nodes as to which link to take from an attractive set are determined by the order in which events (such as arrival of buses or trains) are unfolding. Specifically for transit applications this concept has led to a large body of literature on frequency-based assignment with consideration of passenger strategies. Similarly, the idea of hyperpath-based route choice has been applied to road network assignment (Bell, 2009; Bell et al, 2012; Ma and Fukuda, 2016) where hyperpaths might be obtained due to "random" link choices depending on, for example, which exiting link has a green traffic signal at the time a traveler arrives at a junction.

Furthermore, frequency-based transit assignment models are still the main tool to obtain line loads in large scale networks. Especially if one wants to test line load sensitivities to changes in fare distance levels or consider the aforementioned non-additive congestion effects, improved frequency-based assignment methods are required. This motivates this paper, though in the work present here we emphasize the general application of our approach to hyperpath-based traffic or transit assignment without paying attention to transit-specific issues such as walking between platforms, dwell times or capacity constraints.

The remainder of this paper is organized as follows. The next section will review related literature. In Section 3 we introduce our fairly extensive notation before Section 4 defines the problem we address more formally. We then illustrate the resulting assignment problem and classify three types of hyperpath searches depending on the fare structures in Section 5. Section 6 then lays the foundation for our solution approach that is presented in Section 7. Section 8 illustrates the approach with a case study before Section 9 concludes the paper and discusses extensions required for practical applications.

## 2. Literature on (hyperpath-based) assignment with non-linear link costs

A range of literature on non-additive costs has considered obtaining path costs as given and looked at equilibrium assignment approaches focusing on convergence issues given that convexity of costs might not be guaranteed (Wong et al, 2001; Han and Lo, 2004). Recently, within transit assignment, Chin et al (2016) also looked at equilibrium assignment with non-additive costs. Their focus has been further on policy implications by illustrating the impact of non-linear fares on revenue and ridership with a Toronto case study.

Our focus is the step before assignment, that is the determination of the optimal hyperpaths themselves for which there is much less literature. Finding a single shortest path in a network with one cost criteria does not constitute a significant problem as additivity of costs is not a criteria for a range of algorithms (Ahuja et al, 1993). A set of

literature has been looking though at multicriteria optimal paths with non-additive cost (Tsaggouris and Zaroligas, 2004). A fairly recent noteworthy contribution is the work of Chen and Nie (2013). They obtain shortest paths in a network where the path (not hyperpath) costs consist of two elements, one of which might be continuous, non-additive. Path costs are transformed into discrete, additive subproblems. They then develop an algorithm to obtain efficient path set frontiers considering the two objectives. In contrast, in our problem we consider generalized costs combining travel time, waiting time and non-additive, discrete fares into a single cost. Furthermore, as noted, we aim to consider passengers choosing hyperpaths and not being restricted to single paths.

Frequency based transit assignment considering hyperpaths has significantly advanced over the past four decades. Since the seminal work of Spiess and Florian (1989) as well as Nguyen and Pallatino (1988) strategy and hyperpath-based approaches have been widely used. Further, a number of approaches have been proposed to treat issues such as on-board congestion, capacity problems or the effect of real-time information (for a summary see Gentile and Noekel, 2016). Various approaches are used by commercial software to approximate the effect of complex fare structures with frequency-based assignment. For example, base-fares to enter the system and zone-based fares can be modelled by adding costs to particular links in the network (see e.g. VISUM user notes in PTV (2013)).

However, problems to model general non-linear link cost structures remain and contributions to solve such issues are still fairly sparse. Most closely related to our problem are the publications by Lo et al. (2003) and Constantin and Florian (2015). The former paper proposed a “state-augmented” multi-modal (SAM) network description for network assignment. The central idea is to consider as part of the link specific fares which modes have already been taken so that travelers are in different “states”. The approach is taken up by Constantin and Florian (2015) in an extension to strategy-based transit assignment in which user costs can vary along a trip according to a set of pre-defined ‘transit journey levels’. They apply the state-augmented approach to the Puget Sound regional network with scenarios where travelers have to pay reduced fares for a bus if they transfer from a previous bus. They illustrate that the approach can be efficiently implemented into existing software. The limitation of the state-augmented approach is that the fare is independent of the (mode specific) path travelled and that only a fixed, usually low, number of fare stages can be modelled.

We emphasize that the journey level approach has been developed to overcome the issue of non-additive transfer fares whereas the approach proposed here addresses generally non-additivity of distance-based link costs. A principal difference between the two approaches is that the state-augmented network introduces the definition of multiple nodes for each “state”, so that the classic search for optimal strategies becomes feasible again whereas the approach proposed here does not modify the network but requires additional operations to ensure the optimal hyperpath has been found. We note though that the approaches are complementary so that taken together one can model a wide range of link cost structures.

### 3. Notation

Let us define following network and link specific variables.

- $I$ : Set of nodes in the network
- $A$ : Set of links in the network with  $a \in A$
- $a^+, a^-$ : Headnode and tailnode of link  $a$  respectively
- $R, S$ : Sets of origins and destinations with  $r \in R$  and  $s \in S$

To complete the description of the network further following data are required.

- $t_a$ : travel time of link  $a$  with  $a \in A$
- $d_a, f_a$ : maximum delay, frequency of link  $a$ . Assume  $d_a = 1/f_a$  as in Bell (2009).
- $F$ : vector describing the link cost (fare) structure; the first entry describes the base cost for a journey (e.g. terminal charge), the second entry the cost of the 1<sup>st</sup> link, the third of the 2<sup>nd</sup> link and the  $n^{\text{th}}$  entry describes the additional marginal cost of traversing all links including and after the  $n$ -1<sup>th</sup> link.

We assume throughout this paper that travel time and delays are already converted into a common generalised

cost unit.

We require multiple definitions of hyperpaths depending on how these are obtained and whether they are destination and/or origin specific. We utilise accents to distinguish these hyperpaths, in particular arrows for backward search and “selective hyperpath generation”.

- $\bar{H}$ : Set of all destination specific hyperpaths obtained with backward search by considering minimum fare and with  $\bar{h}^s \in \bar{H}$  and  $s \in S$
- $\hat{H}$ : Set of all optimal origin and destination specific hyperpaths with  $\hat{h}^{rs} \in \hat{H}$  and  $r \in R, s \in S$
- $\bar{H}, \vec{H}$ : Sets of origin and destination specific hyperpaths obtained with backward search and selective hyperpath generation with  $\bar{h}^{rs} \in \bar{H}, \vec{h}^{rs} \in \vec{H}$  and  $s \in S, r \in R$

In line with this notation for hyperpath sets we further introduce hyperpath specific node costs.

- $\bar{u}_i^s$ : cost of reaching node  $i$  with backward search by considering the lower bound link cost from specific origins with  $r \in R, s \in S$
- $\hat{u}^{rs}$ : final optimal origin and destination specific cost with  $r \in R, s \in S$
- $u^{rs}$ : optimal origin and destination specific cost with selective hyperpath generation with  $r \in R, s \in S$
- $\vec{u}_i^r, \bar{u}_i^s$ : cost of reaching node  $i$  by “moving forward” and backward search with hyperpath specific costs with  $r \in R, s \in S$

As we will explain we require a combination of backward search and hyperpath generation. In order to limit the need for the latter additional search sets of fixed cost nodes and critical nodes are introduced. Among critical nodes we further distinguish “active” and “passive” ones, but for notational simplicity we avoid distinguishing our node sets further.

- $\bar{\Theta}^r, \vec{\Theta}^r$ : Set of all critical nodes and fixed nodes from origin  $r$  with  $r \in R$  and  $\bar{\theta}^r \in \bar{\Theta}^r, \vec{\theta}^r \in \vec{\Theta}^r$
- $\bar{\Lambda}^r, \vec{\Lambda}^r$ : Set of all critical links and all fixed cost links from origin  $r$  with  $r \in R$  and  $\bar{\lambda}^r \in \bar{\Lambda}^r, \vec{\lambda}^r \in \vec{\Lambda}^r$

A key aspect of our approach is to obtain the “travel history depending” link specific fares. For this we require the transformation of the information of the fare vector into link specific fares and a “shift operator” as we will explain in Section 6.

- $\Gamma$ : shift operator on node fare probabilities
- $\phi_i^{rh}$ : vectors of size  $|F|$  with elements  $\phi_{ki}^{rh}$  denoting the probability of using  $k-1$  fare link stages from origin  $r$  until node  $i$  on hyperpath  $h$
- $c_a^{rh}$ : expected marginal fare of link  $a$  when travelling from origin  $r$  on hyperpath  $h$  with  $r \in R$  and  $a \in A$

The uncertain, potential (or feared if interpreted as a game as in Schmöcker et al, 2009) link delays are causing the hyperpath problem, that is if there is no uncertainty the hyperpath collapses to a single optimal path. To consider the node uncertainties (waiting times) and resulting split probabilities for the different paths within a hyperpath we introduce below additional notation.

- $\bar{q}_i$ : sum of frequencies (inverse maximum link delays) for attractive links exiting node  $i$  with  $i \in I$
- $\bar{q}_i^s$ : sum of frequencies (inverse maximum link delays) for attractive links exiting node  $i$  given destination  $s$  with  $i \in I$
- $\vec{q}_i$ : sum of frequencies (inverse maximum link delays) for attractive links entering node  $i$  with  $i \in I$
- $p_a^{rh}$ : probability of using link  $a$  when travelling from  $r$  on hyperpath  $h$
- $\pi_i^{rh}$ : probability of using node  $i$  when travelling from  $r$  on hyperpath  $h$
- $y_i^{rs}$ : probability of using node  $i$  when travelling from  $r$  to  $s$

The backward arrow above  $\bar{q}$  is added to highlight the fact that we consider delays when waiting at a node to be served by one of the attractive outgoing links from the present node as in backward oriented hyperpath search. In the appendix we introduce in addition  $\bar{q}$  considering delays for incoming links as in Bell (2009) and discuss that the resulting hyperpaths need to be interpreted different.

Above notation is required for obtaining hyperpaths, the focus of this paper. In addition, for completeness, for network loading we define:

- $\mathbf{V}$ : demand matrix of size  $R \times S$  with  $v_{rs} \in \mathbf{V}$
- $\mathbf{v}$ : link flow vector

**4. Problem definition**

We consider a link cost structure that can be described with an  $F$  vector. The first entry denotes the fixed cost for entering the network. The second entry the cost for traversing the first link, the third the cost for traversing the 2<sup>nd</sup> link and so on. The last vector entry describes the cost for traversing link  $n-1$  as well as all subsequent links. E.g.  $F = (10,5,3,2)$  denotes a cost structure where the traveler has to pay a base charge of 10 units, a cost of 5 units for the first link, a cost of 3 units for the second link and a cost of 2 units for all subsequent links. As in this example, and noted in the introduction, we would expect that the marginal link cost is decreasing the longer the journey if we consider fares, but this might not be true if we consider congestion problems. We assume this kind of link number based cost structure to approximate non-additive distance-based cost structures. Note that long links in a network can always be divided into a series of links of equal length so that link numbers can approximate distance better. Further, considering specifically transit fares, we note that in some cities there are tickets that directly depend on the number of stops to be travelled. In Berlin, for example, there are short distance tickets for traveling three stops on Metro and/or Urban rail with transfer or six stops on bus or tram without transfer, that can be represented as a three-element fare structure based on our fare structure assumption.

To illustrate the problem consider below Figure 1 where D is the destination. The figure illustrates that the classic optimal strategy approach where optimal destination specific “hyperpath trees” cannot be used as the costs for Links 3,4 and 5 depend on the “past”, i.e. how many links have been traversed before. Furthermore, even for a single origin-destination specific hyperpath, different costs for the same link might be required as is the case for Link 5 from C to D for travelers from A. Travelers on path  $\{A,C,D\}$  will be charged 3 units on the link (C,D) whereas travelers on path  $\{A,B,C,D\}$  will be charged 2 units on the same link. We therefore require obtaining hyperpath-specific expected link costs for some links. One might consider that origin-specific, i.e. forward-oriented hyperpath searches, as in the “hyperstar” approach discussed in Bell et al (2012) for dynamic traffic assignment could be used as a starting point to overcome this issue, but the non-intuitive interpretation of such forward-oriented hyperpaths means we pursue in the following a different solution approach. We discuss problems with origin-specific hyperpaths further in Appendix A.

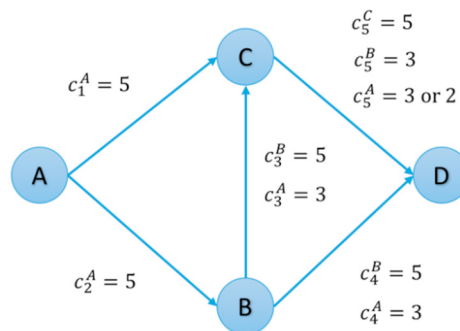


Fig. 1. Problem illustration for Braess network with  $F = (10,5,3,2)$ . Entry links to the network where travelers are charged 10 units are omitted.

## 5. Cost Structure Types

The problem illustrated in Figure 1 can hence lead to different optimal hyperpaths. That is, in cost structures with marginally decreasing fares, travelers can trade-off expected travel time increases with lower link costs. Clearly the problem becomes more important the larger the jumps in discounts for longer journeys are.

We can distinguish three types of scenarios (Table 1). In the first case, we obtain the same hyperpath in the network with considering fixed link costs and hyperpath specific link costs. In the second case, the hyperpath changes after assigning costs but the Bellman principle still holds. In other words, the strategy of taking detours for the sake of saving costs further downstream does not pay off and hence in Figure 2 the optimal (hyper-)path from an origin O to a node A is the optimal hyperpath independent as to whether node A itself is the destination or if a node B is the destination for which the optimal hyperpath includes traversing A. This property might though not always hold so that there are potentially different optimal hyperpaths to an intermediate node depending on what the final destination is. This is the third case in Table 1.

Table 1. Cost structure types

|               | Hyperpath       | Bellman principle | Solution method (Section 8) |
|---------------|-----------------|-------------------|-----------------------------|
| <b>Type 1</b> | Does not change | Holds             | Stage 1 only                |
| <b>Type 2</b> | Changes         | Holds             | Stages 1 and 2              |
| <b>Type 3</b> | Changes         | Does not hold     |                             |

The following example on a four link network where  $t_a$  denotes an additional fixed link cost (travel time) and  $d_a$  the headway (or potential delay on a road link) illustrates the three cases. O is the origin and B and C the two destinations. Table 2 shows that for the cost structure  $F = (0,5,3,2)$  we will find the same hyperpaths for both destinations compared to the case without F, so that this cost structure and OD pairs can be classified as Type 1. In case of  $F = (0,20,12,10)$  instead the hyperpath changes compared to the without F. It is important though that the hyperpath changes for both (all) destinations so that the Bellman principle still holds.

This is not the case for  $F = (0,50,30,2)$  so that this cost structure has to be classified as Type 3. The optimal hyperpath to B consists of the link (O,B) only, whereas the optimal hyperpath to C includes both paths to B. In words, the detour via A pays off for travelers to C as it reduces the costs from B to C more than the additional costs for possibly travelling via A. The larger the marginal differences between costs on the  $n-1^{\text{th}}$  and  $n^{\text{th}}$  link, the more likely the cost structure is of Type 3.

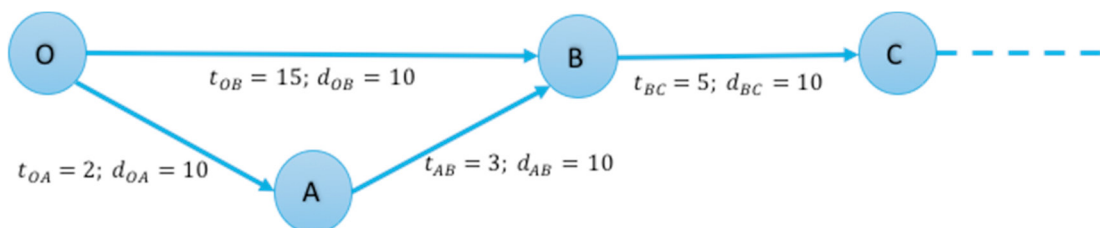


Fig.2 Four link network example

Table 2. Illustration of different optimal hyperpath types

|               |                  | O --> B   |            |             | O --> C    |              |            |
|---------------|------------------|-----------|------------|-------------|------------|--------------|------------|
|               |                  | O-B Only  | O-A-B only | both paths  | O-B-C only | O-A-B-C only | both paths |
| <b>No F</b>   | F = (0,0,0,0)    | 25        | 25         | <b>20</b>   | 40         | 40           | <b>35</b>  |
| <b>Type 1</b> | F = (0,5,3,2)    | 30        | 33         | <b>26.5</b> | 48         | 50           | <b>44</b>  |
| <b>Type 2</b> | F = (0,20,12,10) | <b>45</b> | 57         | 46          | <b>70</b>  | 82           | 71         |
| <b>Type 3</b> | F = (0,50,30,2)  | <b>75</b> | 105        | 85          | 120        | 122          | <b>116</b> |

### 6. Obtaining Hyperpath Costs

In order to calculate the hyperpath depending expected link costs, we have to find origin specific link and node choice probabilities. This section will explain how we derive these utilising our afore introduced concept of the three hyperpath types.

#### 6.1. Link and node choice probabilities

In line with previous literature on hyperpath search, but now origin depending, for every  $r \in \mathbf{R}$ , we obtain link choice probabilities as in (1), where  $f_a$  is the frequency of link  $a$  if we consider a transit context.

$$p_a^{rh} = \frac{f_a}{\sum_{a' \in A_i^- \cap h} f_{a'}}, \forall a \in h \tag{1}$$

We also utilise  $p_a^r$  which is defined with the difference that the denominator sums over all  $a' \in A_i^-$ . Using the hyperpath specific link probabilities  $p_a^{rh}$  or the probabilities  $p_a^r$  we can then also recursively obtain hyperpath node choice probabilities as in (2) or node choice probabilities  $\pi_j^r$  if we consider all links attractive.

$$\pi_j^{rh} = \sum_{a \in A_j^+ \cap h} p_a^{rh} \pi_i^{rh}, \forall j \in I, \forall a \in h \tag{2}$$

#### 6.2. Expected link costs

The vector  $\varphi_j^{rh}$  is used as a record of the travel history. Our definition as in (3) obtains the expected cost stage of the traveler considering the cost stage of the upstream nodes weighted by the probability of traversing those. The equation utilises the shift operator as in (4) which “shifts” the link cost one stage further.

$$\varphi_j^{rh} = \frac{\sum_{a \in A_j^+ \cap h} p_a^{rh} \pi_i^{rh} \Gamma(\varphi_i^{rh})}{\pi_j^{rh}}, \forall j \in I \tag{3}$$

$$\Gamma(\varphi_i^{rh}) \equiv \begin{pmatrix} 0 \\ \varphi_{1,i}^{rh} \\ \varphi_{2,i}^{rh} \\ \vdots \\ \varphi_{n-2,i}^{rh} \\ \min(1, \varphi_{n-1,i}^{rh} + \varphi_{n,i}^{rh}) \end{pmatrix} \tag{4}$$

We emphasise the difference of our  $\varphi$  vector definition to journey stages as in the SAM network. In the SAM approach journey stages consider that a traveler has already used specific modes (or possibly links), whereas the history vector considers generally the number of links traversed. This is required for our problem of considering (mode-independent) non-additive costs.

The shift operator is used to describe the link cost of the link exiting a node  $i$ . For example if  $\varphi_i^{rh} = (0, 0.5, 0.2, 0.3)$  this denotes that the traveler has reached node  $i$  with a probability of 0.5 directly from the origin, with a probability of 0.2 by using exactly two links and that there is a probability of 0.3 that the traveler has used three or more links to reach node  $i$ . With  $\Gamma(\varphi_i^{rh}) = (0, 0, 0.5, 0.5)$  the next link downstream from  $i$  will then be charged with a 0.5 probability the link cost associated with the 2<sup>nd</sup> link after the origin and with a 0.5 chance the link cost associated for links after the 2<sup>nd</sup> link.

Having defined  $\Gamma$  and  $\varphi_i^{rh}$  we can now obtain the expected link cost with (5) for all links and all origin depending hyperpaths. Similar to (1) and (2) we will also utilise  $c_a^r$  as the link cost when all links are considered attractive.

$$c_a^{rh} = \mathbf{F} \times \Gamma(\varphi_i^{rh}), \forall a \in h \quad (5)$$

### 6.3. Critical nodes and links

Following our definitions,  $\varphi_i^r = (0, 0, 1, 0)$  indicates that a traveler originating from  $r$  will certainly take a path to node  $i$  consisting of two links and not consider (or not have the option to take) shorter or longer paths. Therefore, there is no uncertainty regarding the cost of downstream links from  $i$  for travelers from  $r$ . For some nodes and links in the network we can further define certain link costs prior to determining any hyperpath. In Figure 2, for example,  $\varphi_A^r = (0, 1, 0, 0)$  is given by the network structure. We define all such nodes with deterministic, pre-determinable costs as fixed nodes  $\bar{\theta}^r \in \bar{\Theta}^r$  and their outgoing links as fixed links  $\bar{\lambda}^r \in \bar{\Lambda}^r$ . In Figure 2, all downstream links and nodes from Node C, the furthest critical node from the origin, are fixed and costs can be assigned before finding the optimal hyperpath.

In contrast, Node B in Figure 2 is not a fixed node since some travelers can reach it directly from the origin and some of them can reach Node B by using exactly two links. Let us define these kind of nodes as critical nodes  $\check{\theta}^r \in \check{\Theta}^r$  and links with critical nodes as tail nodes we define accordingly as critical links  $\check{\lambda}^r \in \check{\Lambda}^r$ . To improve the runtime of the algorithm presented subsequently, we further distinguish critical nodes as to whether they are “passive critical” or “active critical” for a specific origin. The former means that though the node is critical, no further search is required as the optimal hyperpath from the origin to the node has already been found. We explain the idea of active versus passive critical nodes in more detail in Section 7.3 after introducing the main structure of our solution algorithm.

### 6.4. Selective hyperpath generation

In case the cost structure might be of Type 3, a selective enumeration between origins and its critical nodes cannot be avoided. This computational expensive search is to be avoided as much as possible so that distinguishing active and passive critical nodes is important. Only for active critical nodes enumeration of hyperpaths has to be performed and, as our case study will show, only a small proportion of nodes are active critical. Furthermore, we limit our search to reasonable hyperpaths that include elementary paths with a length of maximal the cost structure size. For our test networks, we found that this heuristics always finds the optimal hyperpaths. Combining these ideas, we refer to this search that finds the optimal hyperpath by comparing the costs of a limited number of hyperpaths as “selective hyperpath generation”.

## 7. Two-Stage Solution Approach

We are now ready to define a 2-stage algorithm for finding optimal hyperpaths. In Section 7.1 we provide an overview to the algorithm and in Sections 7.2 and 7.3 we provide a detailed explanation and justification. In the



overview, we include information on the maximum complexity of each step.

### 7.1. Overview

Initialisation of node costs,  $\varphi_r^{rh}$ ,  $\pi_r^{rh}$  and working sets  $M \leftarrow \emptyset$ ,  $M^o \leftarrow \emptyset$ .

Stage 1: Backward search and extracting ODs not of Type 1

- Step 1-1( $O(|I| |A| \log|A|)$ ): Obtain hyperpath  $\bar{h}^s$ , its costs  $\bar{u}^s$  and their subsets  $\bar{h}^{rs}$  and  $\bar{u}_r^s$ , considering for all links their lower cost limit,  $\forall s \in S$
- Step 1-2( $O(|I| |A|)$ ): Obtain  $c_a^{rh} = \mathbf{F} \times \Gamma(\varphi_a^{rh})$  following Section 7.2  $\forall a \in \bar{h}^s$ ; and set  $c_a^{rh} = \min(\mathbf{F}), \forall a \in A - \bar{h}^s, \forall r \in R, \forall s \in S$
- Step 1-3( $O(|I| |A| \log|A|)$ ): Obtain optimal hyperpath ( $\tilde{h}^{rs}$ ) and its cost  $\tilde{u}_r^s$  considering link costs  $c^{rh}$ , if  $\bar{h}^{rs} = \tilde{h}^{rs}$  then  $\hat{u}^{rs} \leftarrow \tilde{u}_r^s$ ,  $\hat{h}^{rs} \leftarrow \tilde{h}^{rs}$ , else  $M \leftarrow M + \{(r, s)\}, \forall r \in R, \forall s \in S$
- Step 1-4( $O(|I| |A|)$ ): If  $\exists \hat{u}^{rs} = 0, \forall s \in S$ , then  $M^o \leftarrow M^o + \{r\}, \forall r \in R$

Stage 2: Selective hyperpath generation and backward search for Type 2 and 3 ODs

- Step 2-1( $O(|I| |A|)$ ): Obtain  $\varphi^r$  by moving forward,  $\forall r \in M^o$
- Step 2-2( $O(|I| |A|)$ ): Classify nodes into fixed, passive and active critical based on  $\varphi^r$  as well as results of optimal hyperpaths obtained in Stage 1,  $\forall r \in M^o$
- Step 2-3(see note below): Obtain optimal hyperpaths  $\bar{h}^{r\theta}$  and its costs  $u^{r\theta}$  for critical nodes  $i \in \theta^r$  from origin  $r$ . For active critical nodes use selective hyperpath generation. For passive critical nodes use optimal hyperpaths obtained in Stage 1. For all critical nodes  $\hat{u}^{r\theta} \leftarrow u^{r\theta}, \hat{h}^{r\theta} \leftarrow \bar{h}^{r\theta}, \forall r \in M^o$  and  $(r, \theta) \in M$ .
- Step 2-4 ( $O(|A|)$ ): For fixed links, obtain  $c_a^r$  and  $c_a^{rh} \leftarrow c_a^r$ , and for critical links, obtain  $\varphi^{rh}$  based on  $\bar{h}^{r\theta}$  in increasing order of  $\bar{u}_\theta^s$ , then set  $c_a^{rh}$ .
- Step 2-5( $O(|I| |A| \log|A|)$ ): Obtain OD specific hyperpath  $\tilde{h}^{rs}$  by backward search,  $\forall (r, s) \in M$ , If a link with tailnode  $r$  is found as part of the optimal hyperpath then finish backward search.

Stage 3: Loading

- Stage 3-1( $O(|I| |A|)$ ): Assign demand according to optimal hyperpath  $\hat{h}^{rs}, \forall r \in R$  and  $\forall s \in S$

For all steps with the exception of 2-3 we can provide in above the maximum runtime complexity. The runtime of Step 2-3 depends on the number of active critical nodes, the size of  $M^o$ , as well as number of selective hyperpath generations to be performed. These aspects in turn will depend on the network structure and the cost structure  $\mathbf{F}$ , so that we can not provide a general upper limit.

### 7.2. Discussion and details of Initialization and Stage 1

In the initialization step at first the node costs are initialized for the backward search to  $\bar{u}_i^s = \bar{u}_i^s \leftarrow \infty \forall i \in I - \{s\}; \bar{u}_s^s = \bar{u}_s^s \leftarrow 0$ . For all origin nodes we further initialise  $\pi_r^{rh} = 1$  and the travel history to “zero links” as in  $\varphi_r^{rh} \leftarrow (1, 0, \dots, 0)^T$ . To track OD pairs and origins for which the hyperpath is changing due to the cost structure we further introduce the sets  $M$  and  $M^o$  which are initially empty,  $M \leftarrow \emptyset, M^o \leftarrow \emptyset. S \leftarrow I, R \leftarrow I$ .

In Step 1-1 we then apply the well-known “optimal strategies” algorithm for obtaining hyperpaths that fit the assumption of (1) and assuming fixed hyperpath-independent link costs. For brevity, we omit details and refer the reader to Spiess and Florian (1989). It can be shown that the problem in fact can be formulated also as a linear program and hence be solved fast. We apply the algorithm considering the minimum cost for each link as a lower bound.

We then want to check if optimal hyperpaths  $\bar{H}$  differ from  $\tilde{H}$  when applying the true link costs. For this we firstly obtain the hyperpath specific costs in Step 1-2. The cost is obtained by moving forward using the hyperpaths

obtained in Step 1-1 ( $\bar{H}$ ). The details of Step 1-2 are below.

- Step 1-2-1: Set node  $r$  as an origin and node  $s$  as a destination,  $\forall r \in R, \forall s \in S$   
 Step 1-2-2: Obtain  $p_a^{rh}$  by Eq. 1,  $\forall a \in \bar{h}^{rs}$  and  $N \leftarrow \bar{h}^{rs}$   
 Step 1-2-3: Select  $a \in N$  with minimum  $\bar{u}_{a^+}^{s*} = \bar{u}_{a^+}^s + t_a$ , Set  $N \leftarrow N - \{a\}$   
 Step 1-2-4: Update cost of head node  $a^+$   
 If  $\bar{u}_{a^+}^s = \infty$  then  $\beta \leftarrow 1$  else  $\leftarrow \bar{q}_{a^+}^s \bar{u}_{a^+}^s, \bar{u}_{a^+}^s \leftarrow \frac{\beta + f_a(\bar{u}_{a^+}^s + t_a)}{\bar{q}_{a^+}^s + f_a}, \bar{q}_{a^+}^s \leftarrow \bar{q}_{a^+}^s + f_a$   
 Step 1-2-5: Update  $\pi_{a^+}^{rh}$  and  $\varphi_{a^+}^{rh}$  by Eq. 2 and 3 for  $a^+ \in \bar{h}^{rs}$   
 Step 1-2-6: If  $N = \emptyset$  then go to next step else go to Step 1-2-2  
 Step 1-2-7: Set  $c_a^{rh} = F \times \Gamma(\varphi_{a^+}^{rh}), \forall a \in \bar{h}^s$ , and  $c_a^{rh} = \min(F), \forall a \in A - \bar{h}^s$

In Step 1-3 the backward search with the updated link costs is then repeated. The algorithm resembles the one in Spiess and Florian (1989) with the addition of 1-3-6 which tests if hyperpaths have changed and collects the ones for which it has:

- Step 1-3-0: Set  $L \leftarrow A$   
 Step 1-3-1: Set node  $r$  as an origin and node  $s$  as a destination,  $\forall r \in R, \forall s \in S$   
 Step 1-3-2: Select  $a \in L$  with minimum  $\bar{u}_{a^-}^{s*} = \bar{u}_{a^-}^s + t_a + c_a^{rh}$ , Set  $L \leftarrow L - \{a\}$   
 Step 1-3-3: Check if  $a$  should be added to  $\bar{h}^s$ ; if  $\bar{u}_{a^-}^s \geq \bar{u}_{a^-}^{s*}$  then  $\bar{h}^{rs} \leftarrow \bar{h}^{rs} + \{a\}$   
 Step 1-3-4: Update cost of tail node  $a^-$   
 If  $\bar{u}_{a^-}^s = \infty$  then  $\beta \leftarrow 1$  else  $\leftarrow \bar{q}_{a^-}^s \bar{u}_{a^-}^s, \bar{u}_{a^-}^s \leftarrow \frac{\beta + f_a(\bar{u}_{a^-}^s + t_a + c_a^{rh})}{\bar{q}_{a^-}^s + f_a}, \bar{q}_{a^-}^s \leftarrow \bar{q}_{a^-}^s + f_a$   
 Step 1-3-5: If  $L = \emptyset$  then go to next step else go to step 1-3-1  
 Step 1-3-6: If  $\bar{h}^{rs} = \bar{h}^{rs}$  then  $\hat{u}^{rs} \leftarrow \bar{u}_r^s, \hat{h}^{rs} \leftarrow \bar{h}^{rs}$ , else  $M \leftarrow M + \{(r, s)\}, \forall r \in R, \forall s \in S$

We can prove that for hyperpaths not in  $M$  the optimal hyperpath has been found and these OD pairs do not need to be revisited.

**Lemma 1:** If  $\bar{h}^{rs} = \hat{h}^{rs}$  then the optimal hyperpath has been obtained.

**Proof of Lemma 1.** In Step 1-1, we consider the least cost for each link and obtain  $\bar{u}_r^s \leq \bar{u}_r^s$ . Then, in Step 1-2, we assign costs based on  $\bar{h}^{rs}$ . If then in Step 1-3 we reach the same hyperpath, this means that we have  $\bar{u}_r^s + c^{rh} \leq \bar{u}_r^s$ . Thus, the final optimal hyperpath has been obtained since adding costs to the (optimal) hyperpath but not to other hyperpaths has not altered the optimal hyperpath. Adding costs to alternative hyperpaths will only widen the gap between the optimal one and alternative ones.

Step 1-4 then further creates a set of origins for which the hyperpath to at least one destination has changed. We require this set in order to understand for which origins selective hyperpath generation to active critical nodes cannot be avoided.

### 7.3. Discussion and details of stage 2

In order to find possible critical nodes (active and passive), Step 2-1 obtains the vector  $\varphi^r$  for the largest possible hyperpath, that is assuming all links might be attractive. The details of this step are as follows:

- Step 2-1-0: Set  $E \leftarrow A; \bar{u}_i^r \leftarrow \infty \forall i \in I - \{r\}, \bar{u}_r^r \leftarrow 0, \forall r \in R$   
 Step 2-1-1: Set node  $r$  as an origin,  $\forall r \in M^o$   
 Step 2-1-2: obtain  $p_a^r$  by Eq. 1,  $\forall a \in A$   
 Step 2-1-3: Select  $a \in E$  with minimum  $\bar{u}_{a^+}^{s*} = \bar{u}_{a^+}^s + t_a$ , Set  $E \leftarrow E - \{a\}$

- Step 2-1-4: Update cost of head node  $a^+$   
 If  $\bar{u}_{a^+}^s = \infty$  then  $\beta \leftarrow 1$  else  $\leftarrow \bar{q}_{a^+}^s \bar{u}_{a^+}^s, \bar{u}_{a^+}^s \leftarrow \frac{\beta + f_a(\bar{u}_{a^+}^s + t_a)}{\bar{q}_{a^+}^s + f_a}, \bar{q}_{a^+}^s \leftarrow \bar{q}_{a^+}^s + f_a$
- Step 2-1-5: Update  $\pi_{a^+}^r$  and  $\varphi_{a^+}^r$  by Eqs. 2 and 3 for  $a^+ \in I$
- Step 2-1-6: If  $E = \emptyset$  then go to next step else go to Step 2-1-3

Having obtained  $\varphi^r$  we can now obtain fixed and critical nodes in Step 2-2:

- Step 2-2-1: Define the set of critical nodes  $i \in \check{\Theta}^r$  as those with more than one non-zero entry in the  $\varphi_i^r$  vector. Further distinguish active and passive critical nodes considering whether  $i \in M^o$ . Define fixed nodes  $\bar{\Theta}^r$  as all others so that  $\check{\Theta}^r + \bar{\Theta}^r = I, \forall r \in M^o$
- Step 2-2-2: Define possible critical links  $a \in \check{\Lambda}^r$  with tail node  $i$  as those where with more than one non-zero entry in vector  $\Gamma(\varphi_i^r)$  and fixed links  $\bar{\Lambda}^r$  that  $\check{\Lambda}^r + \bar{\Lambda}^r = A, \forall r \in M^o$

Following our distinction of fixed, active and passive critical nodes selective hyperpath generation is only required between origins and active critical nodes. Active and passive critical nodes are distinguished in 2-2-1 utilising the results of Stage 1: A node  $i$  might be critical en-route from  $r$  to another, further away destination  $j$ , but if node  $i$  is the destination this OD pair  $r$  to  $i$  can be of Type 1, so that in fact the optimal hyperpath  $r$  to  $i$  has been obtained in Stage 1. Such a node  $i$  is called a passive critical node, whereas critical nodes en-route from  $r$  to  $j$  for which the optimal hyperpath has not been found in Stage 1 are referred to as active critical.

We note that according to our definition of the cost vector we tend to have critical nodes only within a limited range from the origin. In other words, the size of the network to be searched does not depend on the network size but on the size of structure vector  $F$ . This feature is important to allow applicability to large scale networks. We will illustrate this in the subsequent case study. For the origin depending active critical nodes we now perform selective hyperpath generation to obtain the optimal hyperpath. To the best of our knowledge no alternative formulations appears to provide us successfully with our desired destination oriented hyperpaths with origin depending costs. Hyperpaths for passive critical nodes are assigned based on optimal hyperpaths from Stage 1.

- Step 2-3-1: Check hyperpaths obtained by “selective hyperpath generation” between origin  $r$  and active critical nodes  $i \in \check{\Theta}^r$  and select optimal one  $\bar{h}^{r\check{\theta}}$  with minimum  $u^{r\check{\theta}}$ .  $\forall r \in M^o$
- Step 2-3-2: Store hyperpath from selective hyperpath generation  $\bar{h}^{r\check{\theta}}$  and its cost  $u^{r\check{\theta}}$  as the final OD hyperpath  $\hat{h}^{r\check{\theta}}$  and cost  $\hat{u}^{r\check{\theta}}$  for  $r \in \check{\Theta}^r, \forall r \in M^o$ .

The remaining two steps are now combining the link costs obtained from critical nodes and the ones from the fixed nodes. In Step 2-4 we firstly obtain the corrected (final) link costs for all links and optimal hyperpaths. Below steps are repeated  $\forall (r, s) \in M$ .

- Step 2-4-0: Set  $V^r \leftarrow \emptyset$
- Step 2-4-1: Set node  $r$  as an origin and node  $s$  as destination
- Step 2-4-2: If  $a \in \bar{\Lambda}^r$ , then set  $c_a^r = F \times \Gamma(\varphi_i^r)$  and  $c_a^{rh} \leftarrow c_a^r, \forall a \in \bar{\Lambda}^r$
- Step 2-4-3: Obtain  $\varphi_{\check{\theta}}^{rh}$  by using the iterative process of Steps 1-2-2 to 1-2-6 but based on  $\bar{h}^{r\check{\theta}}$  between origin  $r$  and all possible critical nodes  $\check{\Theta}^r$  and  $V^r \leftarrow \check{\Theta}^r, \forall r \in M^o$
- Step 2-4-4: Select  $\check{\theta}^r \in V^r$  with minimum  $u^{r\check{\theta}}$ , Set  $V^r \leftarrow V^r - \{\check{\theta}^r\}$ ,
- Step 2-4-5: Update  $c_a^{rh} = F \times \Gamma(\varphi_a^{r\check{\theta}})$ ,  $i \in \check{\Theta}^r, \forall a \in \check{\Lambda}^r$
- Step 2-4-6: If  $V^r = \emptyset$  then go to Step 2-5 else return to Step 2-4-4

We can now obtain the OD specific hyperpaths  $\hat{h}^{rs}$  by backward search,  $\forall (r, s) \in M$ , so that we have obtained all optimal hyperpaths.

- Step 2-5-1: Set node  $r$  as an origin and node  $s$  as destination  
 Step 2-5-2: Obtain  $\tilde{h}^{rs}$  and its cost  $\tilde{u}_r^s$  by iterating Steps 1-3-2 to 1-3-5 for origin  $r$  and destination  $s$  by backward search and if  $a = (i, r)$  is found as next link in Step 1-3-2 then backward search can be terminated.  
 Step 2-5-3: Set  $\hat{u}^{rs} \leftarrow \tilde{u}_r^s$ ,  $\hat{h}^{rs} \leftarrow \tilde{h}^{rs}$  and consider next OD pair.

For completeness, we further add details of the network loading step. This step is the same as the network loading in Spiess and Florian (1989). Below two steps are repeated for each OD pair with non zero demand.

- Step 3-1: Set node  $r$  as an origin and node  $s$  as destination  
 Step 3-2: For every link  $a \in A$  in decreasing order of  $(\tilde{u}_r^s + t_a + c_a^{rh})$ , if  $a \in A$  then  $v_a = (f_a / \tilde{q}_a^s -)$   
 $V_{a^+}$  and  $V_{a^-} \leftarrow V_{a^-} + v_a$  else  $V_{a^-} \leftarrow 0$ .

## 8. Case study

Numerical examples on a medium-size network are carried out to show the application of our proposed model and solution algorithm. To illustrate the general applicability of our approach we test it with the well-known Sioux falls network that is publically available from <https://github.com/bstabler/TransportationNetworks> and shown in Fig. 3. To generate link travel time uncertainty (or waiting time in a transit context) and resulting hyperpaths we consider initially  $\alpha = 6$  times the free flow time as potential delay for each link. The larger the potential delay, in general the more complex the hyperpaths become as illustrated in Bell (2009).

We firstly consider two different cost structures and obtain the hyperpaths for two specific OD pairs in the network. Numerical results are summarized in Tables 3 and 4. In the first column of Table 3 we set Node 20 as origin and Node 1 as destination. For the cost structure  $\mathbf{F} = (0,20,12,10)$  we obtain the same hyperpath in Steps 1-1 and 1-3 so that we can be certain that this is an OD pair of Type 1 where the hyperpath does not change and hence Stage 2 is not required.

The rightmost column regarding OD pair 15-18 shows that the cost structure can change the type classification of as well as the hyperpath itself. This can be seen by considering that the optimal hyperpath from the origin to critical node 16 consists of just one path when considering that the traveler has as final destination Node 18 though it consists of two paths if Node 16 itself would be the destination (see Fig. 3). We specifically list the hyperpath to critical Node 16 in the table and the optimal hyperpath to show that Bellman's principle does not hold. (Hyperpaths to other critical nodes not element of the final optimal hyperpath are for brevity excluded in the table in row 'Step 2-3'). We can therefore classify this OD pair for this cost structure as Type 3.

Table 3. Hyperpaths for selected OD pairs with  $F = (0,20,12,10)$

|                |                                     | OD pair                                   |   |
|----------------|-------------------------------------|---|---|
|                |                                     | 20-1                                      | 15-18   |
| Stage 1        | Step 1-1 $\bar{H}$                  | (20,18,7,8,6,2,1)<br>(20,21,24,13,12,3,1) | (15,19,20,18)<br>(15,22,20,18)<br>(15,19,17,16,18)                  |
|                | Step 1-3 $\tilde{H}$                | (20,18,7,8,6,2,1)<br>(20,21,24,13,12,3,1) | (15,19,20,18)<br>(15,22,20,18)<br>(15,10,16,18)<br>(15,19,17,16,18) |
|                | Comparing $\bar{H}$ and $\tilde{H}$ | No change                                 | Changes   |
| Stage 2        | Step 2-3 $\bar{h}^{\bar{\theta}}$   | -   | 15-16:<br>(15,10,16)<br>(15,19,17,16)                               |
|                | Step 2-5 $\tilde{H}$                | -   | (15,19,20,18)<br>(15,22,20,18)<br><b>(15,19,17,16,18)</b>           |
|                | Bellman principle                   | Holds                                     | Holds   |
| Classification |                                     | Type 1                                    | Type 3  |

In Table 4 we analyse the same OD pairs but with a different cost structure. In this revised cost structure travelers are charged more on the first two links they travel but only 2 units for any links after the second one. For OD pair 20-1 we now obtain different hyperpaths in Steps 1-1 (initial backward hyperpath search) and 1-3 (backward search with updated cost) so that Stage 2 is required. In Stage 2-5 we find that the optimal hyperpath remains identical to the one found in Table 3 as it collapses again to the two optimal paths (20,18,7,8,6,2,1) and (20,21,24,13,12,3,1). For the second OD pair, 15 to 18, we also obtain different hyperpaths after applying Stage 1 so that we need to continue by applying Stage 2. Selective hyperpath generation is applied between Node 15 and its active critical nodes. The OD pair remains of Type 3 since Bellman’s principle once again does not hold.

In Fig. 4 we provide one more example for a different OD pair. We add this example to show that for some OD pairs the hyperpath can change significantly depending on the cost structure. In this example with a cost structure that includes a large reduction in marginal costs as in  $F = (0,50,30,2)$  we obtain a hyperpath consisting of numerous elementary paths whereas for a cost structure  $F = (0,20,12,10)$  the hyperpath consists of a single path.

Table 4. Hyperpaths for selected OD pairs with  $F = (0,50,30,2)$

|                |                                     | OD pair  |   |
|----------------|-------------------------------------|--|---|
|                |                                     | 20-1   | 15-18   |
| Stage 1        | Step 1-1 $\bar{H}$                  | (20,18,7,8,6,2,1)<br>(20,21,24,13,12,3,1)                                  | (15,19,20,18)<br>(15,22,20,18)                            |
|                | Step 1-3 $\tilde{H}$                | (20,22,23,24,13,12,3,1)<br>(20,21,24,13,12,3,1)<br>(20,22,21,24,13,12,3,1) | (15,19,20,18)<br>(15,10,16,18)                            |
|                | Comparing $\bar{H}$ and $\tilde{H}$ | Changes  | Changes   |
| Stage 2        | Step 2-3 $\bar{h}^{\bar{\theta}}$   | 20-7:<br>(20,18,7)   | 15-16:<br>(15,10,16)<br>(15,10,17,16)<br>(15,19,17,16)    |
|                | Step 2-5 $\tilde{H}$                | <b>(20,18,7,8,6,2,1)</b><br>(20,21,24,13,12,3,1)                           | (15,19,20,18)<br>(15,22,20,18)<br><b>(15,19,17,16,18)</b> |
|                | Bellman principle                   | Holds  | Does not hold   |
| Classification |                                     | Type 2   | Type 3  |

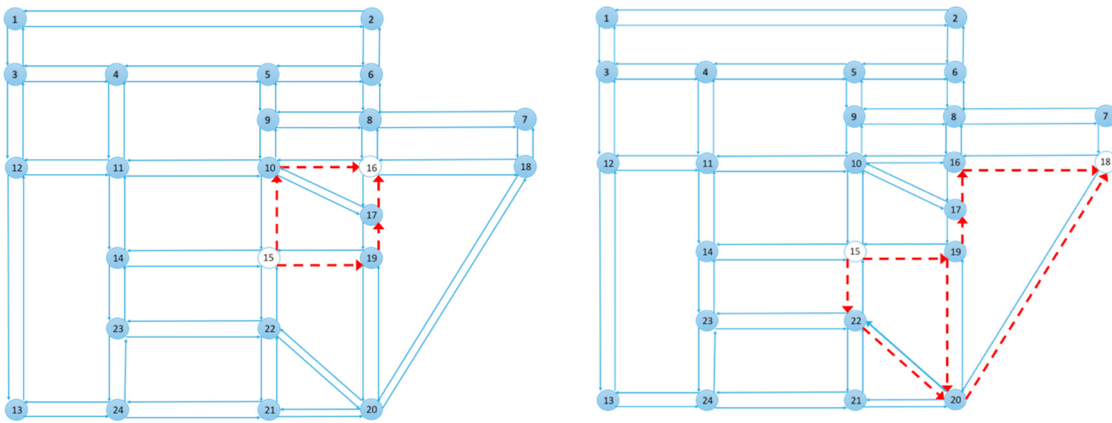


Fig. 3. Illustration of changing hyperpath from origin 15 to node 16 depending on whether it is destination (left) or an intermediate node en-route to final destination 18 (right)

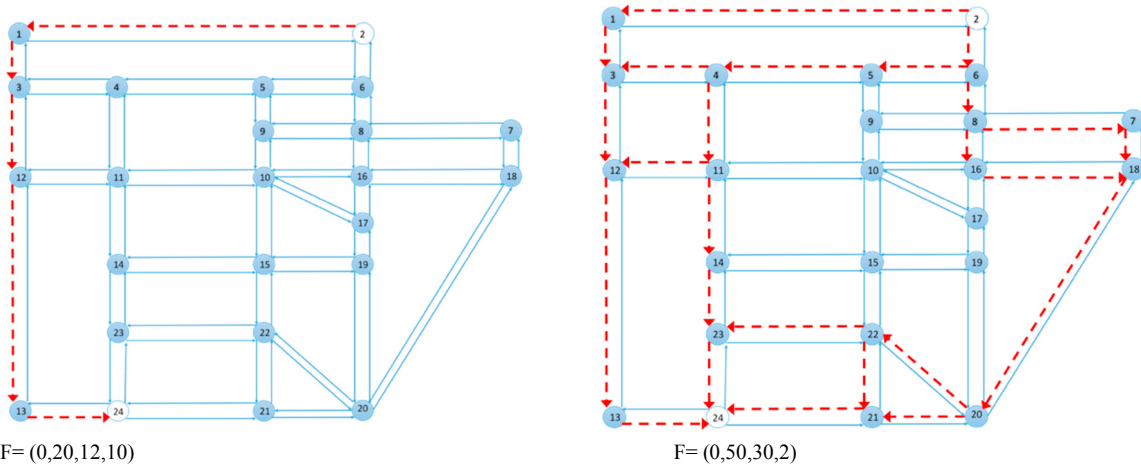


Fig. 4. Illustration of different hyperpaths from origin 2 to destination 24 due to changes in the cost structure.

Figure 5 is added to illustrate the concept of fixed, passive critical and active critical nodes. It shows the critical nodes of Node 15 for a cost structure consisting of four elements  $F = (0,50,30,2)$ . The left figure illustrates that the eleven critical nodes form a “critical area” around the origin. For any links outside this area we can assign the cost independent of the hyperpath chosen.

The right figure illustrates that among the 11 critical nodes, we have found for eight of these OD pairs (from 15 to its orange color critical nodes) the optimal hyperpath already in Stage 1. We classify these as passive critical so that only three active critical nodes remain for which we need to conduct our selective hyperpath generation. As all three nodes are fairly close to the origin there are only a limited number of reasonable hyperpaths so that our selective hyperpath generation is not computational expensive. Subsequently, costs can be assigned to the critical links in increasing expected cost order from the origin. Finally, a simple backward search can be applied to obtain the optimal hyperpath.

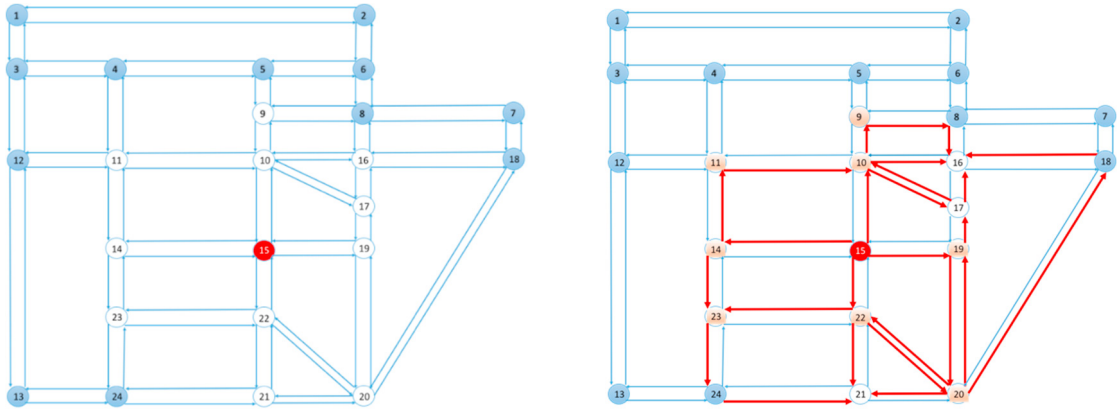


Fig. 5. Left: Critical (no color) and fixed (blue) nodes Right: Active critical (orange), passive critical (no color) nodes and critical links (red)

Table 5 shows a summary of the network in terms of OD pair classification and run time for the above two cost structures as well as a more mildly nonlinear cost structure with  $\alpha = 2$ . The code has been written in a standard Matlab R2011 application. In the case of  $\mathbf{F} = (0,5,3,2)$  we find that for 98.4% of the OD pairs the optimal OD pair can be found without the need for Stage 2. This percentage does not decrease significantly if the overall cost per distance increases but if the marginal differences in cost stages stays fairly constant as in  $\mathbf{F} = (0,20,12,10)$ . Only if there are big jumps in the cost stages as in  $\mathbf{F} = (0,50,30,2)$  we observe a large reduction in Type 1 OD pairs and a corresponding increase in the need for selective hyperpath generation and hence an increase in run time.

Table 5. Run time for different cost structures

|                             | Stage 1         |                | Stage 2           |                | Network Loading | Total Run Time (sec) |
|-----------------------------|-----------------|----------------|-------------------|----------------|-----------------|----------------------|
|                             | Type 1 OD pairs | Run Time (sec) | Type 2&3 OD pairs | Run Time (sec) | Run Time (sec)  |                      |
| $\mathbf{F} = (0,5,3,2)$    | 98.4            | 13             | 1.7               | 2              | <1              | 15                   |
| $\mathbf{F} = (0,20,12,10)$ | 95.3            | 12             | 4.7               | 6              | <1              | 18                   |
| $\mathbf{F} = (0,50,30,2)$  | 41.3            | 13             | 58.7              | 62             | <1              | 75                   |

Tables 6 and 7 illustrate results of additional tests. Table 6 shows that the runtime of our approach depends on the size of the cost structure vector with  $\alpha = 2$ . The more cost levels the more potential critical nodes one might obtain and the more selective hyperpath generations have to be performed. Finally, Table 7 illustrates the potential delays as  $\alpha$  times the free flow time and run time. More potential delays lead to more uncertainty resulting in more complex hyperpaths. Interestingly though more complex hyperpaths do not mean longer runtime, in fact we observe the contrary. We can explain this as follows: If  $\alpha$  is larger more complex hyperpaths may be found in Step 1-3 and as a result more often the same hyperpaths are obtained in Step 1-4 when costs are modified. This is because costs become of relatively reduced importance for path choice with more delays.

Table 6. Run time for different number of cost distance levels

| Cost structure             | % Type 1 OD pairs | Run time (Sec) |
|----------------------------|-------------------|----------------|
| <b>F = (0,5,3)</b>         | 100               | 12             |
| <b>F = (0,5,3,2)</b>       | 98.4              | 13             |
| <b>F = (0,5,3,2,1)</b>     | 91.5              | 25             |
| <b>F = (0,5,3,2,1,0.5)</b> | 82.6              | 165            |

Table 7. Run time for different potential delays with  $F=(0,50,30,2)$ 

| Potential delays $\alpha$ times the free flow time | % Type 1 OD pairs | Run time (Sec) |
|--|-------------------|----------------|
| <b><math>\alpha = 1</math></b>                     | 34.9              | 83             |
| <b><math>\alpha = 3</math></b>                     | 46.7              | 77             |
| <b><math>\alpha = 5</math></b>                     | 58.3              | 52             |
| <b><math>\alpha = 7</math></b>                     | 68.7              | 44             |

## 9. Discussion and Conclusions

In this paper we discussed the problem of finding optimal hyperpaths considering non-additivity of link costs. We discretize the non-additive part of the link cost function in that we consider that the link cost is based on the number of traversed links and develop a methodology that is based on “link cost stages” where a “history vector” describing the expected number of links traversed is defined that allows us to obtain the expected, hyperpath depending link costs. Such a link cost structure is similar to distance-based fare structures where fares depend not (only) on the origin and destination but the actual routes chosen. Such fare structures are currently applied in some cities and we suggest their spread might increase in future if pricing and GPS tracking of travelers will be more connected. As discussed in the introduction, we suggest that the problem this paper deals with has further applications also in logistics and other assignment problems.

We illustrate that such cost structures can lead to violation of Bellman’s optimality principle which means that standard procedures to obtain optimal destination specific hyperpath trees are not feasible. Our solution approach therefore firstly derives a test to distinguish OD pairs for which this principle is not fulfilled. Our case study suggests that these are in general few. For these OD pairs we then develop a methodology that minimizes the need for expensive hyperpath enumeration.

In order to do so, we firstly utilize our assumption that the cost structure has a limited number of cost stages. This appears to us as reasonable given that maximum charges in cities are often limited which can be achieved if the last entry in  $F$  is set to zero. Even if maximum costs are not limited there might be only a relative low number of important jumps in the cost structure. Our analysis suggests that omitting small changes in the marginal costs is not likely to lead to significant errors in the hyperpath estimation. Secondly, for a (fairly large) number of nodes there are no (or no reasonable) multi-path hyperpaths from given origins so that resulting link costs can be uniquely determined before searching for the optimal hyperpath.

Both facts lead us to distinguish “fixed” nodes and links from “critical” ones. Among critical nodes we further distinguish active from passive critical ones and only for the former ones a selective enumeration is required. We then develop an algorithm that combines backward search and a (very) selective generation of hyperpaths to active critical nodes to find the optimal hyperpaths.

The runtime of our approach is almost independent of network size but instead depends on the size of the cost structure vector. This is a direct consequence of the cost vector being assumed to have a limited number of entries.



We further show that if costs are less important compared to other network costs the run time of our approach reduces.

We illustrate with a simple case study how the cost structure can influence the hyperpath of OD pairs. In general, with larger step changes in the cost structure violation of Bellman's principle will become more frequent. Also, in general we expect with higher cost per distance levels (compared to travel time and waiting times) hyperpaths to be less complex and shorter paths to be preferred. We can therefore use the cost structure as a policy variable to shift travelers to avoid congested areas, e.g. in the network the long link 20-18 in the Sioux Falls network as it is only charged once.

Clearly the link-number based cost structure considered in this paper is for many applications a simplification. We noted already that one might split long links into several links to generate networks with links of fairly equal length to avoid this issue. Alternatively, future work will consider advances where the elements of the cost vector will become a function of the link length. We also recognize that in general marginal link costs are often not defined identical for all origins (see for example the Hong Kong fare structure on p. 474 in Gentile and Noekel, 2016). We can cover these cases with our approach by defining several cost vectors  $\mathbf{F}$ . Our approach would only fail if marginal costs per distance on different paths for the same OD pair differ.

We further suggest that our idea of "critical nodes" as introduced here is a general one. We can also use this to model combinations of different link cost structures within different parts of the network. As an example in some transit networks there are flat fares within a certain area and a distance-based fare outside this area. In such cases critical nodes might be seen as the boarder nodes between the two areas. In line with our methodology the total fare within the flat fare area are known before obtaining the path cost for the remaining part of the journey which might be charged with a different fare structure (zonal or distance-based.)

Finally, the idea of our "history vector" can be modified to account for different stages such as having traversed not a general number of links but specific links or modes a traveler has already travelled. This would allow also covering problems addressed in previous literature with the state augmented approach (Lo et al, 2003, Florian and Constantin, 2015) such as having to pay surcharges for e.g. express services or specific modes.

## Acknowledgements

This research has been support by the JSPS Kaken Project 26289174 "Complex Fare Structures: Modelling and Potential Impacts".

## Appendix A. Note on Forward vs Backward Hyperpath search

We note that an obvious solution to the history-vector problem would be forward-oriented hyperpath search as in Bell et al (2012). They propose an origin depending hyperpath in order to account for time-dependent link travel times. Similarly, forward search is also required in the problem at hand here as we need to consider the origin depending cost property. However, for route choice probabilities as well as for (transit) assignment purposes we prefer the backward hyperpath solution as it is easier to interpret. To explain the difference, we introduce  $\bar{q}$  and  $\tilde{q}$  for forward and backward search respectively. In the former  $\bar{q}_i = \sum_{a \in A_i^+} d_a$ , i.e. the sum of inverse maximum delays entering the nodes, whereas in the latter,  $\tilde{q}_i = \sum_{a \in A_i^-} d_a$ , i.e. the sum of inverse maximum delays exiting the nodes.

Consider Figure 6 example with  $d_a = 10$  for all links where the hyperpath search and assignment on the left is carried out as in Spiess and Florian (1989) and as in Bell (2012) on the right. Node A is the origin and D the destination and let's suppose link travel times so that all links are found to be attractive. The resulting hyperpaths must be interpreted differently according to the definition of  $q$ . Especially within a frequency based transit concept we consider the left hand side more intuitive though. For example, given that both services from the origin are attractive and that both services have the same frequency, it appears more reasonable to assume that the traveler will choose both services with the same likelihood, i.e.  $p_{AB} = p_{AC} = 0.5$ .

Whereas Fig. 6 illustrates that for the same hyperpath we obtain different link choice probabilities, Fig. 7 shows

that in some cases we obtain different sets of attractive links with forward-oriented hyperpath searches even without consideration of costs. The reader can verify that applying the Spiess and Florian approach to this network one obtains an optimal hyperpath including all links, whereas applying “hyperstar” leads to a hyperpath without link BC.

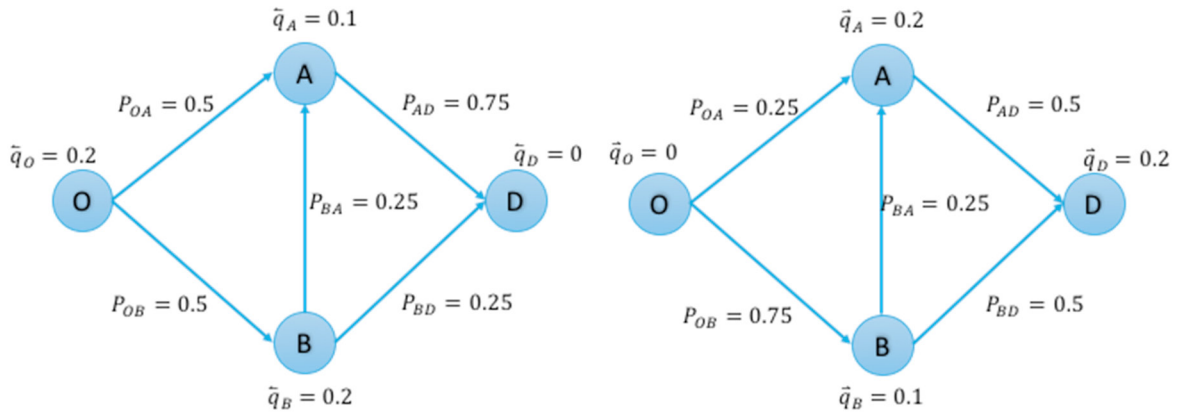


Fig. 6. Example of resulting hyperpaths with the “classic” Spiess and Florian approach (left) and Hyperstar (right) for Braess network assuming all links have a low frequency/large delay (here  $d_a = 10$  for all links) and that all links are attractive.

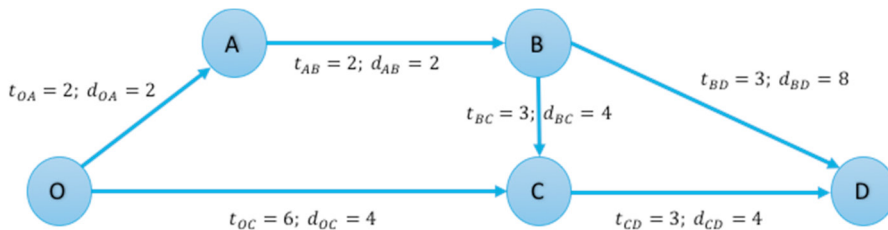


Fig. 7. Illustration of changing hyperpath depending on forward and backward search

**References**

Ahuja, R.K., Magnanti, T.L., Orlin, J.B. 1993. Network Flows: Theory, Algorithms, and Applications. John Wiley & Sons.  
 Bell, M.G.H., 2009. Hyperstar: A multi-path Astar algorithm for risk averse vehicle navigation. Transportation Research Part B 43(1), 97–107.  
 Bell, M.G.H., Trozzi, V., Hosseinloo, S.H., Gentile, G. and A. Fonzone. 2012. Time-dependent Hyper-star algorithm for robust vehicle navigation. Transportation Research Part A, 46(5), 790-800.  
 Chen, P. and Nie, Y. 2013. Bicriterion shortest path problem with a general nonadditive cost. Transportation Research Part B 57, 419-435 and ISTTT20 proceedings.  
 Chin, A.; Lai, A. and Chow, J. Y. 2016. Non-additive public transit fare pricing under congestion with policy lessons from Toronto case study. Transportation Research Board Annual Meeting; Paper #16-1710.  
 Constantin, I. and D. Florian. 2015. Integrated fare modelling with strategy-based transit assignment, 13th Conference on Advanced Systems in Public Transport, CASPT2015, Rotterdam, Netherlands.  
 Fukuda, D., Ma, J., Yamada, K. and Shinkai, N. 2016. Tokyo: Simulating Hyperpath-Based Vehicle Navigations and its Impact on Travel Time Reliability. In: Horni, A., Nagel, K. and Axhausen, K.W. (eds.) The Multi-Agent Transport Simulation MATSim. London: Ubiquity Press, 517-522.

- Gabriel, S. and Bernstein, D. 1997. The traffic equilibrium problem with non-additive path costs, *Transportation Science* 31, 337–348.
- Gentile, G. and K. Noekel. 2016. (Editors). *Modelling Public Transport Passenger Flows in the Era of Intelligent Transport Systems*. Final Report of COST Action TU1004 (TransITS). Springer.
- Han, D. and Lo, H.K. 2004. Solving non-additive traffic assignment problems: A descent method for co-coercive variational inequalities. *European Journal of Operational Research* 159, 529–544.
- Lo, H. K., Yip, C. W. and K. H. Wan. 2003. Modeling transfer and non-linear fare structure in multi-modal network, *Transportation Research Part B*, 37(2), 149–170.
- Nguyen, S. and S. Pallottino. 1988. Equilibrium Traffic Assignment for Large Scale Transit Networks. *European Journal of Operational Research*, 37(2), 176-186.
- PTV 2013. *Using Fares in Headway-Based Public Transport Assignment*. Memo to VISUM users. Unpublished.
- Spiess, H. and M. Florian. 1989. Optimal strategies: a new assignment model for transit networks, *Transportation Research Part B*, 23(2), 83-102.
- Schmöcker, J.D., Bell, M.G.H., Kurauchi, F. and Shimamoto, H. 2009. A Game Theoretic Approach to the Determination of Hyperpaths in Transportation Networks. *Selected Proceedings of the 18th International Symposium on Transportation and Traffic Theory (ISTTT)*. Hong Kong, July 2009.
- Tsaggouris, G., Zaroliagis, C., 2004. Non-additive shortest paths. In: Albers, S. and Radzik, T. (Eds.), *Algorithms - ESA 2004*, *Lecture Notes in Computer Science*, 3221, 824-832.
- Wong, S.C., Yang, C. and Lo, H.K. 2001. A path-based traffic assignment algorithm using the TRANSYT traffic model, *Transportation Research Part B* 35, 163–181.