Route Specifications with a Linear Dual Graph

Stephan Winter

Institute for Geoinformation, Vienna University of Technology, Gusshausstr. 27-29, 1040 Vienna, Austria, winter@geoinfo.tuwien.ac.at

Abstract

The objective often explored in Web-based route planners is to find an optimal route in a network for a given mode of transport. Usually this involves searching for the cheapest route corresponding to some cost function. For many types of trips, not all desired route properties can be satisfied in this way. Following is a proposed solution for planning hiking trips. The method can easily be transferred to tourist guides in urban areas, for advice for taking a drive, or can be used in related contexts. It is anticipated that these applications will become increasingly relevant in our mobile leisure society.

Planning a hiking route is based primarily on the intended length of the trip, and then on the decision about whether start and destination points need to coincide. As a result, planning trips represents neither a shortest path nor a travelling salesman problem.

The problem with this approach lies in the fact that hikers require trip proposals that cannot necessarily be provided by a shortest path algorithm: round tours, routes including loops, or routes where a route segment which has to be passed in both directions are more consistent with route planning for hikers. A surprisingly simple solution for these requirements appears to be a linear dual graph, which applies a *k* shortest path algorithm to the linear dual graph. This paper outlines this approach and demonstrates that it achieves realistic and practical route planning.

Keywords: route planning, shortest path, linear dual graph

1 Introduction

Web-based route planners or mobile route guides solve the task of finding an optimal route in the network of a given mode of transport, e.g., by car, public transport, aeroplane, bike, or on foot. They typically search for the cheapest route corresponding to any cost function, like time, fuel, or number of changes. At least for pedestrians (but in principle for all travel modes) not all desired route properties can be satisfied by a simple graph and a shortest path algorithm. Some of the incompatible properties are an intended duration of a trip, or the option of round tours. We are investigating these problems for planning hiking trips (Cziferszky 2002), and propose a linear dual graph approach (not to be confused with the dual graph of a partition). The following discussion can easily be transferred to tourist guides in urban areas, for guidance in taking a drive, or for similar types of contexts. These applications will become increasingly relevant in our mobile leisure society.

Let us consider a person asking advice from a web service concerning her hiking trip next Sunday. Her main criterion may be the intended duration of the trip. It is clear that planning routes for trips may not only be a shortest path problem. The trip planning service needs to be able to search for routes of a predetermined duration. A planning service needs also the flexibility to provide both, routes from a start to a destination and round tours. Finding a round tour can be a boundary condition for instance for hikers who want to come back to their car at the end of the trip. Another required flexibility lies in the fact that hikers accept to visit nodes more than once. Consider for example a loop through a park, added to an otherwise linear trip. Hikers often walk a route segment in both directions, e.g., if this segment is the only one to a mountain's peak, or the only connection between a car park and an attractive hiking area. There are many other route selection criteria, for example, maximal slope, most scenic path, or possible stops at inns. These criteria can likely be discovered, provided that a number of predetermined or preconsidered interests could be established.. However, we discuss neither the introduction of these additional criteria nor the ranking of routes in this paper.

The round tour problem is also not entirely like a travelling salesman problem. The travelling salesman is looking for the shortest round tour that visits a set of places. Hikers on the other hand, who look for a round tour may not necessarily ask for the shortest route, even if they specify points of interests to be visited on that trip.

In principle, the posed problem of route planning for hiking trips requires a combination of all possible route segments in a region, with a set of possible start and destination nodes (hiker parking places, bus stops, and similar). The constructed routes have to be selected that best fit to the specified criteria. This is an exponential combinatorial problem. A known efficient approach is by exploiting algorithms for the *k* shortest paths. These algorithms yield an ordered sequence of the *k* shortest routes (some of them yield even loopless shortest paths). Instead of calculating *k* shortest routes, the algorithms can run recursively, stopping when the intended route duration is exceeded. Consequently, the set of routes satisfying the duration criterion can then be evaluated for other properties. Evaluation could rank the routes to recommend the ‘best’ one first.

The *k* shortest routes show all combinatorial possibilities of extending a shortest path. Many of the routes presented, however, are not appropriate and would not usually be acceptable to hikers. The question to pose is 'how does one specify route properties requested by hikers such that the artificial or inappropriate solutions can be rejected'. A surprisingly simple solution appears to be found through using a linear dual graph, as presented in the following sections of this paper. The hypothesis presented in this paper explores the applicability of the *k* shortest path algorithm on the linear dual graph to determine if it can produce realistic route tips, including round trips, trips with loops, and trips with segments to be walked in both directions – but never twice in the same direction.

The paper is structured as follows. In section 2, the problem is discussed in relation to the state of the art. section 3 introduces the conceptual model of a dual linear graph and a dual path. In section 4, the application of the linear dual graph to trip planning is presented. Section 5 discusses the further exploitation of the conceptual model in the calculation of trip proposals. We close with a discussion and outlook (Section 6).

2 State of the Art

The requirements for creating hiking trip proposals is based on a user-centred approach to model the route planning (Cziferszky 2002), and is supported by general research in human wayfinding (Golledge 1995; Golledge and Stimson 1997) and navigation (Denis et al. 1999; Frank 2001). According to a survey of Cziferszky most people planning an excursion first specify the duration of the trip.

Shortest path algorithms, among them the most prominent (Bellman 1958; Dijkstra 1959) yield minimum spanning trees from a start node which through execution navigates to all other nodes in a network (Chvatal 1983; Cormen et al. 1990), i.e., exactly one path from a start to any destination node. Different trip duration can only be found by varying the start and destination points. Common for all shortest path algorithms is a graph search strategy for local optima such that a node, registered once, will never be visited again. This property excludes loops.

The *k* shortest path algorithms (Bellman and Kalaba 1960; Azevedo et al. 1990; Eppstein 1994; Ruppert 2000) vary the shortest path, finding not only the shortest, but also the second shortest, and so on. The variations are produced either by route labelling methods, or by route deletion. There are loopless and non-loopless *k* shortest path algorithms. Loops extend the length of a route but then continue with already known parts of a path. Loops also include edges that can be travelled in both directions.

The travelling salesman algorithms find shortest round tours to a set of nodes in the graph. Some variation has been applied to shortest round tours through development of the sub-tour problem presented by (Stille 2001). In this case, a subset of the available points of interest (inns, views, and peaks) is combined to (shortest) round tours. However, this option is still not flexible enough to find tours of a predetermined duration.

The linear dual graph was originally proposed to model graphs with edge *and* node costs (Caldwell 1961; Knödel 1969). It can be used to express relations between consecutive edges (Winter 2001). Although the idea is appealing for its elegance, it did not find much attention in the graph theory literature nor in applications. The only implementation of which we are aware is reported by

(Anez et al. 1996). Current network analysis tools deal with relations differently (Zeiler 1999).

Due to the complexity of the trip planning problem current services use mostly a set of predefined routes (Davies et al. 2001) or, alternatively, have to restrict the problem to a very limited set of route segments.

3 The Linear Dual Graph

In the following sections we use undirected and directed weighted graphs, paths, and the linear dual graph. These concepts are introduced in the following sections.

3.1 Graphs

A graph G consists of a set of nodes N and a set of edges E. The edges represent binary relations between nodes, $e: N \rightarrow N$. We distinguish the start node f, $f: E \rightarrow N$, and the end node l, $l: E \rightarrow N$. We treat an undirected edge u as a short form of a pair of bi-directional edges: $u = \{e_1\ (f(u), l(u)),\ e_2\ (l(u), f(u))\}$. Typically edges have some costs attached, $w: E \rightarrow \Re^+$; default value is 1. Then a graph is given by the tuple $G(N,E)$ with f_E, l_E, and w_E.

The total number of edges ending in a node n is called the *indegree* of n, $\text{indegree}(n) = \sum_{i=1}^{|E|} e_i \mid l(e_i) = n$, and the number of edges starting in n is its *outdegree*, $\text{outdegree}(n) = \sum_{i=1}^{|E|} e_i \mid f(e_i) = n$.

A path p from node $f(e_1)$ to $l(e_k)$ is the sequence of edges $p=(e_1,\ldots,e_k)$ such that $l(e_i)=f(e_{i+1})$, $i=1,\ldots,k-1$, and $e_i \neq e_j$ for $i \neq j$. The path include loops if there exist any pairs $f(e_i)=l(e_j)$ for $i < j$. The length of a path is k; the total cost of a path is $W(p) = \sum_{i=1}^{k} w(e_i)$.

3.2 Linear Dual Graphs

A *linear dual graph D* is defined by the following definition (see also Fig. 1):

Definition: Given a (primal) graph $G(N_G,E_G)$ with f_G, l_G, w_G, the graph $D(N_D, E_D)$ with f_D, l_D, w_D with the following properties is called its *linear dual graph*:

- For each edge e_i in G there is a node ν in D, assigned by a function d: $\nu_i=d(e_i)$. d is a bijective function so that $d^{-1}(\nu_i)=e_i$. Thus, $N_D=d(N_G)$.
- For each pair of edges (e_i, e_j) in G with $l(e_i)=f(e_j)$ there is an edge ε in D between the corresponding nodes $\nu_i=d(e_i)$, $\nu_j=d(e_j)$, such that $f(\varepsilon)= \nu_i$ and $l(\varepsilon)=\nu_j$. $E_D = \bigcup_i \varepsilon_i$.

- A cost function w_D: $E_D \rightarrow \Re^+$.

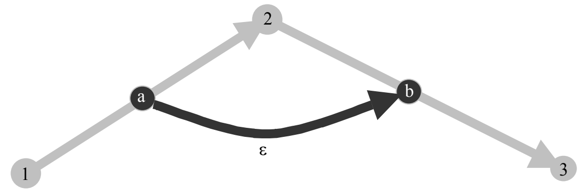

Fig. 1. A directed primal graph *G* (gray) and its linear dual *D(G)* (dark)

For simplicity we will speak of a *dual graph* in the following, always referring to the linear dual graph. Note that the more common notion of a dual graph to a partition is completely different.

The size of the dual graph *D* is given by the following equations:

- The number of nodes in *D* equals the number of edges in *G*: $|N_D| = |E_G|$.
- The number of edges in *D* equals the number of paths of length 2 in *G*. For each $n_i \in G$ there are $p(n_i)=indegree(n_i) \times outdegree(n_i)$ such paths. With route graphs being sparse, we can estimate the upper limit of such paths assuming a maximal number of ingoing (*inmax*) and outgoing edges (*outmax*) for all nodes. Then the number of edges in *D* is $\leq |E_G|$ $min\ (inmax, outmax)$.

Theorem: For each primal path with a length $k \geq 2$ there exists a corresponding path in the dual graph.

Proof. For each neighboured pair of edges in the primal path there exists per definition a dual edge. Thus, a sequence $(e_1,..., e_k)$ corresponds to a sequence $(\varepsilon_1, ..., \varepsilon_{k-1})$ such that $f(\varepsilon_1)=d(e_1)$, $l(\varepsilon_1)=d(e_2)$, and so forth till $l(\varepsilon_{k-1})=d(e_k)$. q.e.d.

Corollary: For a primal path of length *k* the length of the corresponding path in the dual graph is *k-1*.

3.3 Dual Paths

The properties of paths in the dual graph are not promising for the tasks at hand: in the context of trips paths with only one primal edge are valid paths; but they cannot be represented as paths in the dual graph. For that reason we propose to change to a *dual path*:

Definition: A dual path π to a given path $p=(e_1, ..., e_k)$ is the sequence of nodes $\nu_1=d(e_1), ..., \nu_k=d(e_k)$.

For each path in the primal graph exists a dual path in the dual graph. The length of the dual path is *k*, which is equal to the length of the path. The costs of the primal edges can be transferred to the corresponding dual nodes by the relation

$v(v_i)=w(e_i)$. In the context of this paper[1] we limit the total costs of a dual path to $V(\pi) = \sum_{i=1}^{k} v(v_i)$.

The dual path consists of a subset of elements of the dual graph, of a primal path. Only if the primal path contains no loops, the dual path consists of all elements of the dual graph.

4 Route Planning with the Linear Dual Graph

4.1 Start and Destination in the Dual Path

The dual graph is introduced in this section and discussed in the context of specifying routes for trip planning.

In the primal graph, a node means a state and an edge a transition from its start to its end node. For hiking trips we deal with embedded graphs, i.e., the state described by a node is its (or a visiting agent's) position in a plane. With the restriction that trips start and end at graph node positions, a path in the primal graph can directly be translated into a trip route, and constructed by a sequence of route segments.

In the dual graph a node shall mean a movement along the corresponding primal edge from its start to end node. An edge shall mean a reorientation in the corresponding primal node with continuation of hiking along another primal edge. Hence, a dual path $\pi=(v_1, \ldots, v_k)$ in the dual graph needs to be translated carefully into a trip route.

- The start position s of a route is (with the above restriction) $f(d^{-1}(v_1))$.
- The destination t is accordingly $l(d^{-1}(v_k))$.
  This leads to a slightly different planning problem.

4.2 Routes and the Dual Path

We transfer the route planning problem to the dual graph.

A trip-planning problem identifies among the primal nodes a start and a destination of the route. Then its goal is to find a path with costs close to the intended trip duration. In case of round trips, start and destination are identical.

- A hiker always leaves the (primal) start node taking an outgoing edge. We call all the dual nodes v_i for which $f(d^{-1}(v_i))=s$ the *start gateways* of the route planning problem.

[1] In other contexts it can make sense to consider the costs of the dual edges also, see for example Winter, S., 2001: Weighting the Path Continuation in Route Planning. In: Aref, W.G. (Ed.), 9th ACM International Symposium on Advances in Geographic Information Systems. ACM Press, Atlanta, GA, pp. 173-176..

- A hiker always reaches the (primal) end node along an incoming edge. We call all the dual nodes v_i for which $l(d^{-1}(v_i))=t$ the *destination gateways* of the route planning problem.

For any path from node *s* to node *t* the dual path contains at least one element of the start gateways and one element of the destination gateways. Both sets can overlap (if a path is of length 1), hence, the element from the start gateways and the element of the destination gateways may coincide.

The route planning problem in the dual graph is then to find a dual path between any node from the start gateways and any node of the destination gateways such that its costs are close to the intended duration of the trip.

4.3 Extended Modelling Capabilities with the Dual Path

As indicated previously, a shortest path algorithm shall be applied for the calculation of the route planning problem.. Shortest path algorithms cannot include loops (by definition). This is certain limitation for planning trips, and can be solved by the extended expression of the dual path.

- A round trip is a popular trip type. Paths may have loops, but no shortest path algorithm allows visiting a node twice. The algorithm will stop after initialisation.
  For the dual path the problem is formulated differently: for a round trip start and destination gateways are disjoint (a loop of length 1 is excluded in the graph definition), and thus, the length of a round trip is at least 2

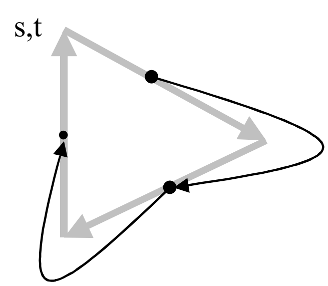

Fig. 2. A round trip does not require visiting any dual path node more than once

- Hiking routes may include loops in between. Shortest path algorithms will never visit a node more than once. Again, in the dual path no node is visited more than once (Fig. 3).

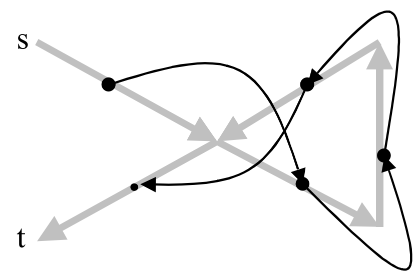

Fig. 3. Loops do not require visiting any dual path node more than once

- Hiking routes may include segments that have to be hiked in both directions. Again, in the dual path no node is visited more than once (Fig. 4).

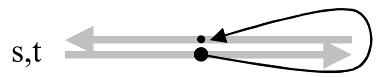

Fig. 4. Segments travelled in both directions do not require visiting any dual path node more than once

- The argument is even valid for combined cases (Fig. 5).

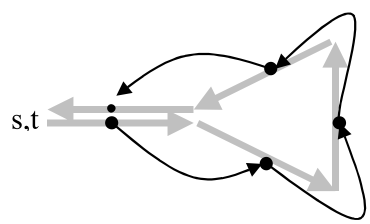

Fig. 5. A combination of previous cases

That means, common shortest path algorithms work in some central cases only on the dual graph. The specific properties of the dual graph would avoid hiking a segment a second time in the same direction. We expect that this property is conforming to the expectations of hikers on trip proposals, although we have not checked that.

5 Routes of Intended Duration

A shortest path algorithm between start and destination always yields the shortest path, however, we are looking for paths close to a predefined cost. One possibility to consider is the variation of start and destination. A route planning service will make use of this possibility, however, it is limited in most areas to a few numbers of hiker parking places, bus stops or similar points of interests.

The idea is to therefore systematically vary the shortest path, as done in *k* shortest path algorithms. The result is an ordered list of *k* routes from start to destination with monotonic increasing costs. Setting *k* large and storing the found routes in a database allows off-line calculation. The route candidates have only to be updated if the network changes. A request for a trip with intended duration x can access the database selecting all route candidates with a duration close to x, let us say, $0.85x \ldots 1.15x$.

6 Discussion and Outlook

This paper has introduced the linear dual graph and a dual path as a conceptual model for route specifications in trip planning. It has been demonstrated that this model can express critical planning cases, which were not achievable in primal graphs using shortest path algorithms.

Currently a *k* shortest path algorithm has been implemented to prove the concept. A subsequent paper will report on the algorithm, experiences with test cases, and computing performance of the calculations. In Winter, 2001, the linear dual graph for modelling turning costs and turn restrictions has been demonstrated.. In the current paper the specific needs of trip planning, can be accomplished, as well. Hence, one can expect that the potential of the linear dual graph has not been exhausted.

Acknowledgements

The idea for this paper came from discussions with Andreas Cziferszky, Technical University Vienna, who is working on multi-criteria optimisation for hiking routes.

References

Anez J, de la Barra T, Perez B (1996) Dual Graph Representation of Transport Networks. Transportation Research 30(3):209-216

Azevedo JA, Silvestre Madeira JJER, Martins EQV, Pires FMA (1990) A shortest paths ranking algorithm. In: Proc. of the Annual Conference of the Operations Research Society of Italy (AIRO '90). Associazione Italiana Ricerca Operativa, pp 1001-1011

Bellman RE (1958) On a Routing Problem. Quart. Appl. Math 16:87-90

Bellman RE, Kalaba R (1960) On *k*th best policies. Journal of SIAM 8:582-588

Caldwell T (1961) On finding minimum routes in a network with turn penalties. Communications of the ACM 4(2): 107-108

Chvatal V (1983) Linear Programming. W. H. Freeman & Co.

Cormen TH, Leiserson CE, Rivest RL (1990) Introduction to Algorithms. MIT Electrical Engineering and Computer Science Series. MIT Press, Cambridge

Cziferszky A (2002) Routenplanung anspruchsvoller Wanderungen. Diploma thesis in progress, Institute for Geoinformation, Technical University Vienna, Austria

Davies N, Cheverst K, Mitchell K, Efrat A (2001) Using and Determining Location in a Context Sensitive Tour Guide. IEEE Computer Journal 34(8):35-41

Denis M, Pazzaglia F, Cornoldi C, Bertolo L (1999) Spatial Discourse and Navigation: An Analysis of Route Directions in the City of Venice. Applied Cognitive Psychology 13:145-174.

Dijkstra EW (1959) A note on two problems in connexion with graphs. Numerische Mathematik 1:269-271

Eppstein D (1994) Finding the *k* Shortest Paths. In: IEEE Symposium on Foundations of Computer Science. pp 154-165

Frank AU (2001) Pragmatic Information Content - How to Measure the Information in a Route Description. GeoInformatica (accepted)

Golledge RG (1995) Path Selection and Route Preference in Human Navigation: A Progress Report. In: Frank AU, Kuhn W (eds) Spatial Information Theory - A Theoretical Basis for GIS. Lecture Notes in Computer Science, 988. Springer, Berlin, pp 207-222

Golledge RG, Stimson RJ (1997) Spatial Behavior: A Geographic Perspective. The Guildford Press, New York

Knödel W (1969) Graphentheoretische Methoden und ihre Anwendungen. Ökonometrie und Unternehmensforschung, 13. Springer, Berlin

Ruppert E (2000) Finding the *k* Shortest Paths in Parallel. Algorithmica 28:242-254

Stille W (2001) Lösungsverfahren für Prize-Collecting Traveling Salesman Subtour Probleme und ihre Anwendung auf die Tourenplanung im Touristeninformationssystem Deep Map. Diploma thesis, Fakultät für Mathematik und Informatik, Ruprecht-Karls-Universität Heidelberg, Germany

Winter S (2001) Weighting the Path Continuation in Route Planning. In: Aref WG (ed) 9th ACM International Symposium on Advances in Geographic Information Systems. ACM Press, Atlanta, GA, pp 173-176

Zeiler M (1999) Modeling Our World. ESRI Press, Redlands