

# Gleichwertige Ziele in dynamischen Navigationskarten

JOHANNES OEHRLEIN<sup>1</sup>, THOMAS C. VAN DIJK<sup>2</sup> & JAN-HENRIK HAUNERT<sup>1</sup>

*Zusammenfassung: Die Generierung übersichtlicher Karten erfordert Verfahren der automatischen Generalisierung. Zur Darstellung auf Navigationskarten werden beispielsweise Objekte eines Verkehrsnetzes anhand ihrer Bedeutung für das Netz ausgewählt. Beschränkt man sich auf die Navigation von einem festen Startpunkt aus, verlieren viele Objekte für die Karte an Bedeutung. Durch Verzicht auf deren Darstellung wird die Karte übersichtlicher. Diesen Umstand nutzen van Dijk et al. (2016) für einen Algorithmus zur standortbasierten Generalisierung von Straßennetzen. Dieser trifft – abhängig von einem fest gewählten Standort – durch die Zusammenfassung von als gleichwertig erkannten Zielen eine Auswahl. Hebt man die Fixierung des Standorts auf, ergeben sich neue Möglichkeiten für mobile Geräte. Dieser Beitrag beschäftigt sich mit den Problemen, die mit der Dynamik Einzug in diesen Algorithmus halten, und bietet erste Lösungsansätze.*

## 1 Einleitung

Bei der Erzeugung von Karten bedarf es einer Abwägung zwischen Übersichtlichkeit und Informationsfülle. Dabei müssen Objekte ausgewählt und vereinfacht werden, um die Lesbarkeit einer Karte zu gewährleisten. Soll dieser Prozess automatisch vonstattengehen, benötigt man Algorithmen, die die einzelnen Schritte der kartographischen Generalisierung beherrschen (HAKE et al. 2002)

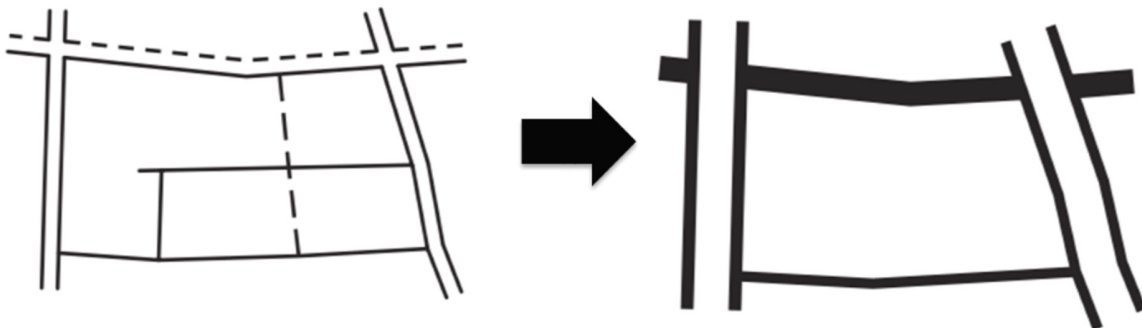


Abb. 1: Generalisierung eines Straßennetzes ( nach HAKE et al. 2002)

Einer dieser Schritte betrifft die Generalisierung von Straßennetzen (Abb. 1). Hierbei müssen zum einen Straßenzüge vereinfacht werden. Zum anderen müssen Straßen anhand ihrer Bedeutung für den auf der Karte dargestellten Inhalt ausgewählt werden. Mit diesem Problem beschäftigen sich VAN DIJK et al. (2016). Dabei konzentrieren sie sich auf die Generalisierung von Straßennetzen,

---

<sup>1</sup> Universität Bonn, Institut für Geodäsie und Geoinformation, Lehrstuhl Geoinformation, E-Mail: [haunert, oehrlein]@igg.uni-bonn.de

<sup>2</sup> Universität Würzburg, Institut für Informatik, Lehrstuhl für Informatik I, E-Mail: thomas.van.dijk@uni-wuerzburg.de

bei der die Bedeutung einzelner Straßen hinsichtlich eines festen Punkts berücksichtigt wird. Dieser Punkt kann beispielsweise als aktueller Standort oder auch als Ziel einer Navigation aufgefasst werden.

Bei der standortbasierten Generalisierung von Straßennetzen wird die Auswahl einer Straße anhand ihrer Bedeutung für den gegebenen Standort getroffen. Dabei kommt einer Straße im Allgemeinen eine größere Bedeutung zu, je näher sie sich zum Standort befindet (HAMPE et al. 2004; KOPF et al. 2010).

Unser Beitrag ist nun die Untersuchung der Eignung eines so generalisierten Straßennetzes für die Erstellung von dynamischen Karten, beispielsweise zur Navigation. Im Rahmen der Zielführung ist dabei insbesondere von Interesse, wie sich eine solche Karte verhält, wenn der Standort, auf den sich die Generalisierung bezieht, verändert. Insbesondere für mobile Geräte bietet es sich an, hier den aktuellen Standort des Nutzers der Navigation zu verwenden.

In Abschnitt 2 wird zunächst die Arbeit von van Dijk et al. vorgestellt und ihre Eignung zur Zielführung diskutiert. Anschließend werden in Abschnitt 3 die Probleme analysiert, die durch Veränderung des Standortes auftreten. Außerdem wird erörtert, wie sich diese Probleme sinnvoll beheben lassen. Schließlich bietet Abschnitt 4 Raum für ein Fazit.

## 2 Standortbasierte Generalisierung von Straßennetzen

Bei der Reise von A nach B mithilfe eines Navigationsgeräts ist die genaue Adresse des Ziels für den Reisenden erst einmal uninteressant, sobald sie in das Gerät eingegeben wurde. Van Dijk et al. (2016) ziehen als Beispiel eine Reise von Berlin nach Groningen heran: Zu Beginn der Reise interessiert es den Nutzer nicht, wo sich die Zieladresse genau in Groningen befindet. Für den ersten Abschnitt seiner Reise ist dies unerheblich. Wie man Berlin in Richtung der Niederlande verlässt, hängt nicht von der Straße oder des Stadtviertels des Zielorts in Groningen ab. Auf dieser Idee basiert der Algorithmus.

### 2.1 Ausgangssituation

Im zu generalisierenden Straßennetz betrachtet man mögliche Zielorte von einem Standort aus. Zielorte, zu denen die Routen vom gegenwärtigen Standort aus zu großen Teilen übereinstimmen, sollen zu abstrakten Zielen zusammengefasst werden. Hierzu wird das Straßennetz als **Graph**  $G = (V, E)$  aufgefasst: Die Kanten stellen Straßenabschnitte dar, die sich an den Knoten treffen.  $G$  enthält dabei den Standort  $s \in V$ . Die Kanten von  $G$  sind gewichtet. Diese **Gewichtung**  $w: E \rightarrow \mathbb{R}^{\geq 0}$  bestimmt das Kriterium, nach dem die Route geplant werden soll; für die kürzeste Wegstrecke verwendet man beispielsweise als Gewicht einer Kante ihre euklidische Länge. Betrachtet man einen Weg, so lässt sich dieser als eine endliche Folge von Kanten formalisieren. Dabei entspricht das Gewicht des Weges der Summe der Gewichte der einzelnen in ihm enthaltenen Kanten. Unabhängig vom gewählten Kriterium ist man bei der Navigation an einem optimalen, also gewichtsminimalen Weg interessiert. Ermittelt man nun von  $s$  den Weg minimalen Gewichts (bezüglich  $w$ ) zu jedem möglichen Ziel, so ergibt eine Baumstruktur. Im Folgenden beziehen wir uns auf den Fall der euklidischen Abstände. Dann sind diese gewichtsminimalen Wege kürzeste Wege und wir

sprechen vom **Kürzeste-Wege-Baum  $T$** . Auf diesem operiert der Algorithmus. Der Weg von einem beliebigen Knoten  $u \in V$  zu einem Knoten  $v \in V$  in diesem Baum ist eindeutig und wird im Folgenden mit  $P_{uv}$  bezeichnet.

## 2.2 Gleichwertige Zielorte

Ziel ist es, mögliche Bestimmungsorte zusammenzufassen, die über größtenteils übereinstimmende Wege erreicht werden können. Hierzu wird ein Kriterium eingeführt, mit dem entschieden werden kann, ob zwei Knoten  $u, v \in V$  **kompatibel**, also zusammenfassbar, sind. Hierzu wird jeweils der kürzeste Weg  $P_{su}$  bzw.  $P_{sv}$  vom Standort aus betrachtet. Diese beiden Wege stimmen von  $s$  aus bis zu einem gewissen Knoten  $x \in V$  überein.  $u$  und  $v$  werden als kompatibel bezeichnet, wenn für einen bestimmten Schwellenwert  $\alpha \in [0,1]$  folgende Ungleichungen erfüllt sind:

$$\frac{w(P_{sx})}{w(P_{su})} \geq \alpha \quad \text{und} \quad \frac{w(P_{sx})}{w(P_{sv})} \geq \alpha.$$

Das heißt:  $u$  und  $v$  sind kompatibel, wenn der gemeinsame Weg mindestens den Anteil  $\alpha$  am Gesamtweg sowohl nach  $u$  als auch nach  $v$  einnimmt (siehe Abb. 2).

Der Parameter  $\alpha$  ist hierbei die einzige nicht-geometrische Information, die dem Algorithmus übergeben werden muss. Dieser Wert entscheidet über die Dichte des dargestellten Straßennetzes. Kleine Werte (0,5 und kleiner) sorgen für eine starke Ausdünnung des kürzeste-Wege-Baums.

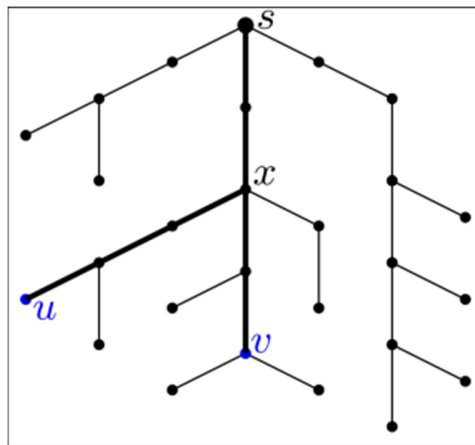


Abb. 2:  $u$  und  $v$  mit gemeinsamen Vorgänger  $x$ . Nimmt man gleiche Kantengewichte an, so beträgt der gemeinsame Weg  $2/5$  des Weges nach  $u$  und  $1/2$  des Weges nach  $v$ . (Abb.: VAN DIJK et al., 2016)

Große Werte (0,9 und größer) erhalten viele Details und geben auch in größerer Entfernung zum gewählten Standort noch viele Informationen über den Eingabegraphen wieder. Das Ziel, dass das Gebiet nahe dem Standort detaillierter dargestellt wird als die weitere Umgebung, wird allerdings unabhängig von der Wahl von  $\alpha$  erreicht.

### 2.3 Algorithmus

Der Algorithmus fasst nun ausgehend von den Blättern des Baumes möglichst viele kompatible Knoten zusammen. Dabei wird jeder Knoten genau einer **Zelle** zugewiesen, die nur Knoten enthält, die paarweise kompatibel sind (siehe Abb. 3, links). Das Resultat ist ein von  $T$  induzierter Baum, der für jede Zelle einen Repräsentanten enthält und diese untereinander gemäß den in  $T$  enthaltenen Kanten verknüpft. Ein sinnvoller Repräsentant ist der (eindeutige) Knoten, der dem Standort  $s$  am nächsten ist (siehe Abb. 3, rechts).

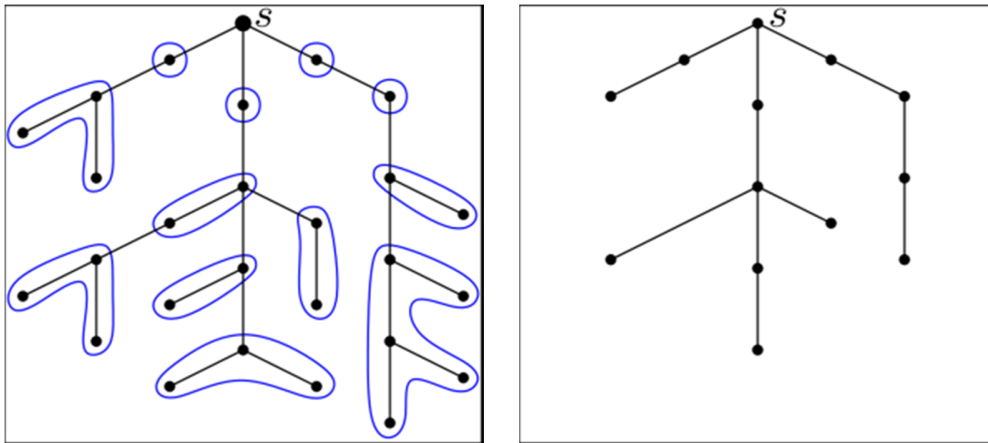


Abb. 3: Ergebnis des Algorithmus für  $\alpha = 2/3$ . Links: vom Algorithmus ermittelte Zellen. Rechts: Ausgabe nach Zusammenfassung. (Abb. nach VAN DIJK et al., 2016)

Van Dijk et al. (2016) zeigen, dass ihr **Greedy-Algorithmus** ein optimales Ergebnis zurückliefert in dem Sinne, dass es keine Möglichkeit gibt, die kompatible Knoten zu einer geringeren Anzahl an Repräsentanten zusammenzufassen. Dazu benötigt der Algorithmus asymptotisch Linearzeit (bezüglich der Anzahl  $|V|$  an Knoten in  $T$ ).

### 2.4 Querverbindungen

Bei der Reduktion eines Graphen (Eingabe-Sträßennetz) auf einen Baum (Kürzeste-Wege-Baum) gehen viele Informationen über den Zusammenhang im Graphen verloren. Insbesondere gibt es Orte im Sträßennetz, die über zwei völlig verschiedene, gleichlange Wege vom gegenwärtigen Standort aus erreichbar sind. In einem gewöhnlichen kürzeste-Wege-Baum ist dieser Umstand nicht mehr erkennbar. Daher wird die folgende Modifikation am Kürzeste-Wege-Baum vorgenommen.

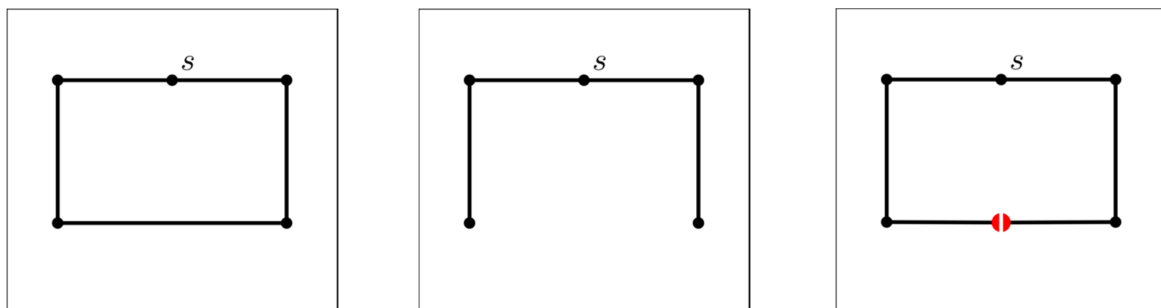


Abb. 4: Vom Graph (links) über den kürzeste-Wege-Baum (Mitte) zum modifizierten kürzeste-Wege-Baum (rechts) mit virtuellen Knoten (rot)

Es gibt Kanten  $e = (u, w)$  in  $G$ , die nicht in  $T$  enthalten sind. Die Ursache hierfür ist, dass der kürzeste Weg von  $s$  nach  $w$  nicht über  $u$  führt und umgekehrt. Unter der Annahme, dass das ge-

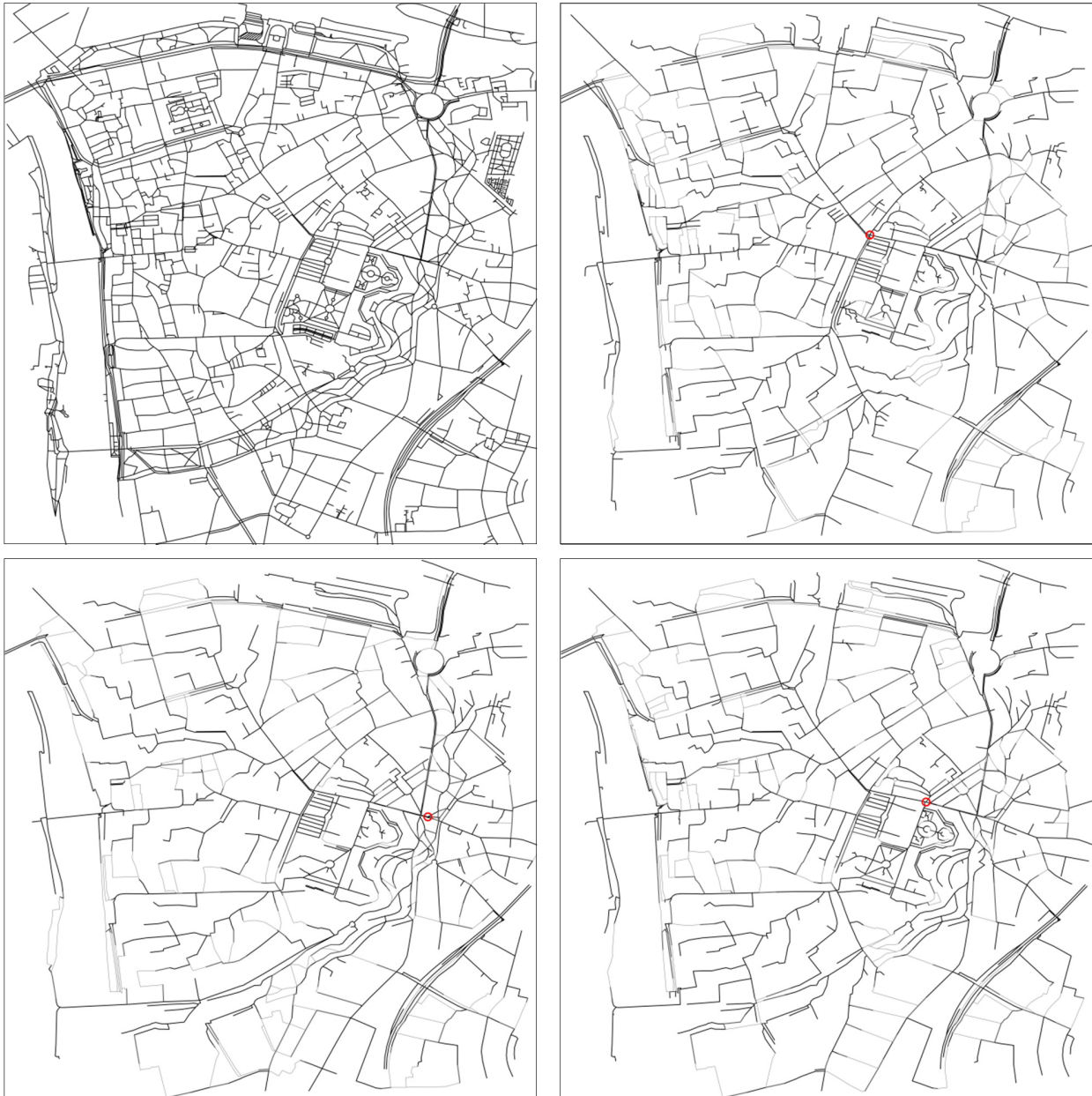


Abb. 5: Beispiel an einem Ausschnitt des Würzburger Straßennetzes: Die Eingabedaten werden oben links dargestellt. Alle weiteren Abbildungen zeigen das Ergebnis des Algorithmus für verschiedene Standorte (●) mit demselben Wert  $\alpha = 0,9$ . Die schwarzen Kanten bilden den erzeugten Baum, die grauen Kanten die Querverbindungen (siehe Abschnitt 2.4). Verschiedene Standorte erzeugen verschiedene Ausgaben. Betrachtet man die Bilder von oben rechts nach unten links im Uhrzeigersinn (oder umgekehrt), so kann man die Auswirkungen einer Standortveränderung entlang eines Straßenzugs nachvollziehen.

wählte Kantengewicht sich **stetig entlang der Kante verteilt**, findet sich auf diesen Kanten jeweils ein Punkt (kein Knoten!), der auf zwei verschiedenen Wegen gleichen Gewichts erreicht werden kann. Im Fall euklidischer Abstände ist diese Annahme erfüllt, das Gewicht einer Kante ist sogar gleichmäßig verteilt. Für solche Punkte werden jeweils zwei **virtuelle Knoten** eingefügt, die entsprechend der ursprünglichen, in  $T$  fehlenden Kante über zwei eingefügte Kanten mit  $T$  verbunden werden. So entsteht der **modifizierte kürzeste-Wege-Baum  $T'$**  (Abb. 4).

Um derlei Information im Ergebnis wiederzugeben, wird für jedes Paar Repräsentanten, das von mindestens einem Paar virtueller Knoten jeweils einen Knoten repräsentiert, eine **Querverbindung** eingeführt. Diese verbindet die Repräsentanten über das Paar virtueller Knoten, welches dem aktuellen Standort  $s$  am nächsten ist.

## 2.5 Diskussion

Der vorgestellte Algorithmus liefert generalisierte Informationen über ein Straßennetz, die in der Nähe des Standorts detailreicher sind als in der weitläufigen Umgebung. Dieses Problem wird nach einem exakt definierten Kriterium optimal und effizient gelöst. Damit bildet der Algorithmus eine gute Grundlage für standortbasierte Karten (Hampe et al. 2004). Da für die Generalisierung von Straßennetzen allerdings weitere Kriterien berücksichtigt werden müssen, bedarf der Algorithmus in jedem Fall einer Ausarbeitung für die Anwendung. Beispiel hierfür ist die Erhaltung von Straßenzügen, die der Algorithmus in der vorgestellten Form nicht gewährleistet (THOMSON & RICHARDSON 1999).

## 3 Auswirkungen eines beweglichen Standorts

Neben statischen Anwendungen wie beispielsweise Anfahrtsbeschreibungen bieten sich standortbasierte Karten insbesondere für **dynamische Karten** an. Hier wird die Karte regelmäßig an einen aktualisierten Standort angepasst. Dieser ergibt sich durch Simulation oder durch die Verwendung eines mobilen Geräts. Dabei gilt, dass verschiedene Standorte im Allgemeinen verschiedene Ausgaben erzeugen. Eine Möglichkeit, daraus eine dynamische Karte zu erzeugen, besteht darin, bei einer Standortaktualisierung eine neue Ausgabe zu erzeugen und anzuzeigen. In Abb. 5 wird eine Folge solcher Darstellungen präsentiert. Um die Ansicht für einen Standort in die Ansicht für den nächsten Standort sinnvoll zu überführen, müssen die Auswirkungen einer Standortveränderung auf das generalisierte Straßennetz untersucht werden. Diese Auswirkungen werden im Folgenden erläutert.

### 3.1 Veränderungen im modifizierten kürzeste-Wege-Baum

Durch die Variation des Standorts muss der Kürzeste-Wege-Baum aktualisiert werden. Dabei ist es nicht ausreichend, die Abstände der Knoten zum (veränderten) Standort zu aktualisieren. Auch die Struktur des Baums kann sich dadurch verändern.

Untersucht man die Auswirkungen einer Standortveränderung auf den kürzeste-Wege-Baum, so stellt man fest, dass es vom Standort  $s$  abhängt, welche Kanten des Graphen  $G$  in  $T$  nicht enthalten sind. Diese Veränderung lässt sich am modifizierten Kürzeste-Wege-Baum  $T'$  nachvollziehen. Hierzu betrachten wir ein Paar virtueller Knoten  $(v_1, v_2)$  auf einer Kante  $(u, w)$ . Der Weg  $P_{sv_1}$  erreicht  $v_1$  über den Knoten  $u$ , der Weg  $P_{sv_2}$  führt von  $s$  nach  $v_2$  über  $w$ . Die Wege  $P_{sv_1}$  und  $P_{sv_2}$

haben hierbei dieselbe Länge. Nun betrachtet man die Veränderung des Standorts als stetigen Prozess. Verschiebt man  $s$ , so müssen auch die virtuellen Punkte  $v_1$  und  $v_2$  hinsichtlich Lage und Abstand zu  $s$  aktualisiert werden.

Da  $T'$  ein Baum ist, existiert ein eindeutiger Weg von  $v_1$  nach  $v_2$ . In  $G$  beschreibt dieser Weg einen Kreis  $K$ , da das „virtuelle Hindernis“  $(v_1, v_2)$  in  $G$  nicht existiert. Unterteilt man die Verschiebung von  $s$  in infinitesimal kleine Etappen, so muss für diese zwischen folgenden Fällen unterschieden werden (siehe Abb. 6):

- a)  $s \in K$ . Liegt der Standort auf diesem Kreis, so bewegen sich  $s$  und das Punktepaar  $(v_1, v_2)$  gleichförmig auf diesem Kreis. Der Abstand von  $s$  bleibt unverändert.
- b)  $s \notin K$ . Liegt der Standort nicht auf diesem Kreis, so hat eine Variation von  $s$  keinen Einfluss auf die Position von  $(v_1, v_2)$ . Der Abstand von  $s$  muss aktualisiert werden.

Somit ist die Position der virtuellen Punkte stetig abhängig vom Standort (Abb. 6).

Alle übrigen (also nicht-virtuellen) Knoten in  $T'$  besitzen eine feste Position. Bei einer Variation des Standorts, verändert sich bei diesen Knoten lediglich der Abstand zum Standort.

### 3.2 Auswirkungen auf den Baum der Ausgabe und Stabilisierungsversuche

Durch Veränderung des Eingabe-Baums liefert der Algorithmus mit großer Wahrscheinlichkeit ein anderes Ergebnis. Hierzu betrachten wir eine Zelle im Ausgabe-Baum. Insbesondere gilt unser Interesse dem Knoten  $x$  dieser Zelle, der dem Standort  $s$  am nächsten ist.

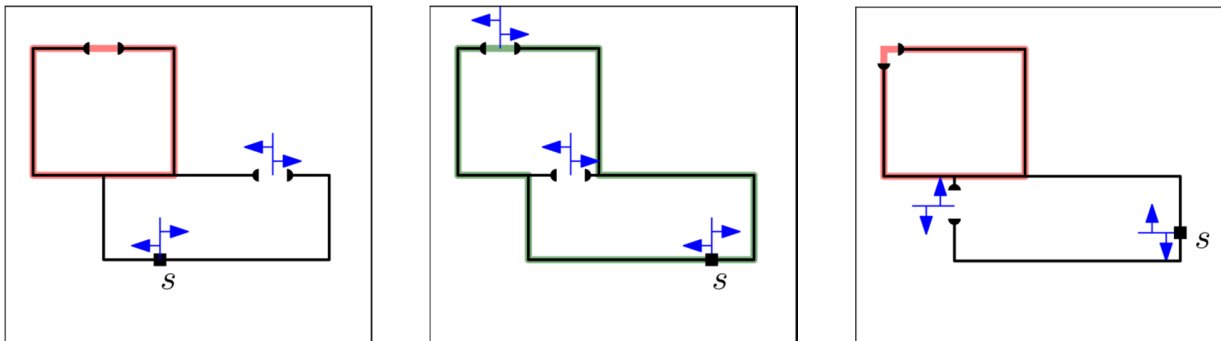


Abb. 6: Für das obere virtuelle Punktepaare ist  $K$  farbig markiert, Lageveränderungen sind in blau dargestellt. Eine Variation des Standorts  $s$  sorgt nur dann für eine Veränderung der Lage eines virtuellen Punktepaars in  $T'$ , falls  $s$  sich auf dem oben beschriebenen Kreis befindet.

Für die Bildung einer solchen Zelle ist der Parameter  $\alpha$  von großer Bedeutung. Nur die Nachfahren  $v$  eines Knotens  $x$  werden der Zelle zugewiesen, für die

$$\frac{w(P_{sx})}{w(P_{sv})} \geq \alpha$$

gilt. Im allgemeinen Fall besitzt  $x$  Nachfahren in  $T'$  und ist somit kein virtueller Knoten. Somit ist  $x$  statisch und  $w(P_{sx})$  verändert sich um den Betrag der Variation von  $s$ . Bei  $v$  kann es sich um einen virtuellen Knoten handeln. Somit besteht nach Fallunterscheidung aus Abschnitt 3.1 die Möglichkeit, dass sich die Länge des Weges  $P_{sv}$  nicht ändert. Das heißt, dass für eine Standortänderung um einen kleinen Wert  $\varepsilon > 0$  folgende Szenarien möglich sind:

$$\frac{w(P_{sx}) \pm \varepsilon}{w(P_{sv}) \pm \varepsilon} \geq \alpha \quad \text{oder} \quad \frac{w(P_{sx}) \pm \varepsilon}{w(P_{sv})} \geq \alpha.$$

Unabhängig vom Nenner der linken Seite wird der Wert des Bruchs also größer, wenn sich der Standort von der untersuchten Zelle entfernt, und kleiner, wenn er sich auf sie zu bewegt. Insbesondere bleibt die Zuweisung eines Knotens  $v$  zu einer Zelle mit Repräsentant  $x$  legitim, solange sich  $s$  von der Zelle wegbewegt. Diese Beobachtung ist im Einklang mit der Grundidee der standortbasierten Generalisierung, dass ein Gebiet detailreicher dargestellt wird, je näher es sich zum Standort befindet.

Durch die Auffassung als Graph wird das Straßennetz diskretisiert. Diese Diskretisierung ist im Resultat verstärkt sichtbar. Da lediglich Knoten im modifizierten Kürzeste-Wege-Baum auf Kompatibilität überprüft werden, wird die mögliche Größe einer gebildeten Zelle im Allgemeinen nicht voll ausgenutzt. Die folgende Rechnung verdeutlicht diesen Umstand, wobei ausgenutzt wird, dass man ausgehend vom Standort  $s$  einen Knoten  $v$  in der Zelle von  $x$  in  $T'$  nur über  $x$  erreichen kann.

$$\frac{w(P_{sx})}{w(P_{sv})} = \frac{w(P_{sx})}{w(P_{sx}) + w(P_{xv})} \geq \alpha \quad \Leftrightarrow \quad \frac{1 - \alpha}{\alpha} w(P_{sx}) \geq w(P_{xv})$$

Da der Algorithmus jeweils diskrete Punktmengen zu solchen Zellen zusammenfasst, herrscht hier im Allgemeinen keine Gleichheit. Somit besteht für die einzelnen Zellen ein gewisser Rahmen, in dem sich  $s$  bewegen darf, ohne dass dies Konsequenzen für die Stabilität der einzelnen Zellen hat. Eine Möglichkeit, die Stabilität der Ausgabe zu gewährleisten, besteht darin, die Toleranz für die Kompatibilität der Knoten zu erhöhen. Das heißt, die Zellen werden grundsätzlich für einen gegebenen Wert  $\alpha$  gebildet. Verändert sich der Standort, so wird die Kompatibilität der Zellen jedoch für einen Wert  $\alpha^* < \alpha$  überprüft. Erst wenn Knoten einer Zelle bezüglich  $\alpha^*$  inkompatibel sind, wird die Zelle neu berechnet.

Eine andere Möglichkeit, die Ausgabe zu stabilisieren, besteht darin, die dynamische Karte möglichst in Echtzeit zu aktualisieren. Die stetige Abhängigkeit der virtuellen Punkte des modifizierten Kürzeste-Wege-Baums verleitet zu diesem Ansatz. Eine Echtzeit-Realisierung ist nur möglich, wenn die Prozessierung nicht zu aufwändig wird. Diesbezüglich ist ein Blick auf die mögliche Anzahl der virtuellen Knoten interessant. Da je Kante maximal zwei virtuelle Knoten hinzukommen, liegt deren Anzahl in  $\mathcal{O}(|E|) \subseteq \mathcal{O}(|V|^2)$ . Nehmen wir an, dass an einer Kreuzung stets höchstens  $k \in \mathbb{N}$  Straßen beteiligt sind. Dann ergibt sich die Abschätzung  $|E| \leq k \cdot |V|$ . Die Annahme ist sinnvoll, da in einem realen Straßennetz im Allgemeinen höchstens eine geringe zweistellige Anzahl an Straßen eine Kreuzung bildet. Für planare Eingabegraphen lässt sich diese Abschätzung zu  $|E| < 3 \cdot |V|$  verschärfen. Allerdings ist die Planarität eines Straßennetzes durch Über- bzw. Unterführungen verletzt. Gemessen an der Anzahl Über- bzw. Unterführungen ist diese Verletzung jedoch geringfügig.

Für eine kontinuierliche Dynamik in der zu erzeugenden Karte müssen die Auswirkungen einer Standortveränderung auf die gebildeten Zellen näher untersucht werden. Der stetige Zusammenhang zwischen der Lage des Standort und der Lage der virtuellen Knoten des modifizierten Kürzeste-Wege-Baums ist nicht hinreichend.

## 4 Zusammenfassung und Ausblick

Der diskrete Ansatz klingt auf den ersten Blick vielversprechend. Durch eine gewisse Fehlertoleranz bezüglich der Kompatibilität soll die Stabilität der Ausgabe erhöht werden. Allerdings führt



eine stete Bewegung des Standorts in eine Richtung zum Aus- und Überreizen dieser Toleranz. Zu einem bestimmten Zeitpunkt muss die Ausgabe aktualisiert werden. Durch die erhöhte Fehlertoleranz ist dann der Unterschied zur korrekten Lösung allerdings vermutlich auffälliger. Es muss überprüft werden, ob diese selteneren, womöglich heftigeren Veränderungen der Ausgabe die Informationsvermittlung in stärkerem Maße beeinflussen, als dies eine sofortige Behebung auftretender Kompatibilitätsprobleme nach sich zöge.

Die Echtzeit-Lösung stimmt hier zuversichtlicher, insbesondere da der effiziente Algorithmus und eine Topologie nahe der Planarität diesen Ansatz begünstigen. In Abschnitt 3.2 wurde bereits festgestellt, dass die Auswirkung einer Standortveränderung noch stärker untersucht werden muss. Davon abgesehen ist das Verhalten der Querverbindungen bei Veränderung des Standorts von großem Interesse. Diese sind unverzichtbar, wenn dem Leser der Karte Informationen zum Zusammenhang des Graphen nicht gänzlich verwehrt werden sollen. Da die einzelnen Querverbindungen entweder dargestellt oder nicht dargestellt werden, liegt kein stetiger Zusammenhang mit der Lage des Standorts vor. Daher muss untersucht werden, wie sprunghaft diese Auswahl ist und wie diese möglichst stabilisiert werden kann.

## 5 Literaturverzeichnis

- VAN DIJK, T. C., HAUNERT, J.-H. & OEHRLEIN, J., 2016: Location-dependent generalization of road networks based on equivalent destinations. *Computer Graphics Forum* **35**(3), 451-460.
- HAKE, G., GRÜNREICH, D. & MENG, L., 2002: *Kartographie: Visualisierung raum-zeitlicher Informationen*. 8<sup>th</sup> ed., Berlin: De Gruyter Lehrbuch.
- HAMPE, M., SESTER, M. & HARRIE, L., 2004: Multiple Representation Databases to Support Visualization on Mobile Devices. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **35**(B4), 135-140.
- KOPF, J., AGRAWALA, M., BARGERON, D., SALESIN, D. & COHEN, M., 2010: Automatic generation of destination maps. *ACM Transactions on Graphics* **29**(6), 158.
- THOMSON, R. C. & RICHARDSON, D. E., 1999: The “Good Continuation” Principle of Perceptual Organization applied to the Generalization of Road Networks. *Proceedings of the ICA 19<sup>th</sup> International Cartographic Conference*, 14-21.