Bachelorarbeit

Kostenrundung von Kanten auf Pfaden

Vivica Wirth

Albert-Ludwigs-Universität Freiburg im Breisgau

Technische Fakultät

Lehrstuhl für Informatik

Betreuer:

Prof. Dr. Hannah Bast

Dr. Sabine Storandt

Abgabetermin: 14.01.2015

ERKLÄRUNG

Hiermit erkläre ich, dass ich diese Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen/Hilfsmittel verwendet habe und alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass diese Abschlussarbeit nicht, auch nicht auszugsweise, bereits für eine andere Prüfung angefertigt wurde.

Freiburg, den 13.01.2015 ……..

Vivica Wirth

Kurzfassung

Diese Arbeit beschäftigt sich damit, Kosten an Kanten in Graphen zu runden. Grundlage dabei ist das mathematische Runden der einzelnen Kanten. Da dieses aber beliebig schlecht werden kann, soll ein besserer Algorithmus gefunden werden. Diese Arbeit befasst sich zunächst mit dem mathematischen Runden, um herauszufinden wo und wie dieses scheitert. Danach werden eigene Ideen entwickelt und präsentiert. Über alternierendes Runden, bei welchem Kanten abwechselnd auf- und abgerundet werden, gelangt man zu gewichtetem Runden. Bei diesem wird sich der Rundungsfehler gemerkt und in Abhängigkeit vom Rundungsfehler gerundet. Die empirische Auswertung zeigt dabei, dass das gewichtete Runden deutlich besser ist, als das mathematische Runden.

Inhaltsverzeichnis

1 Einführung **1**

2 Verwandte Themen **2**

2.1 Summenerhaltendes Runden 2

2.2 Dithering 2

3 Grundlagen **3**

3.1 Code-Struktur 6

4 Basisalgorithmen **8**

4.1 Mathematisches Runden 8

4.1.1 Strukturen mit großem Fehler 10

4.2 Zufallsrunden 12

5 Eigene Ansätze **14**

5.1 Pfadkorrektur 14

5.1.1 Problematik 16

5.2 Alternierendes Runden 17

5.2.1 Zusatz: Superkanten 22

5.2.2 Zusatz: Gewichtung 26

5.3 Gewichtetes Runden 28

6 Auswertung der Algorithmen **32**

6.1 Mathematisches Runden 32

6.2 Zufallsrunden 34

6.3 Alternierendes Runden 37

6.3.1 Grundalgorithmus 37

6.3.2 Zusatz: Superkanten 40

6.3.3 Zusatz: Gewichtung 43

6.4 Gewichtetes Runden 47

7 Erweiterungen **51**

Literaturverzeichnis **52**

1 Einführung

Digitale Wegsuche ist heute nicht mehr wegzudenken. Ein Navigationsgerät erhält Start- und Zieladresse und findet direkt einen Weg. Dabei wird die Distanz ausgegeben, ohne Höhenunterschiede zu beachten. Die Höhenunterschiede könnten aber beispielsweise für einen Radfahrer, der eine Radtour planen möchte, durchaus interessant sein. Er möchte möglicherweise gerne eine Strecke mit wenig Steigung. Die Daten für die Höhen liegen vor, allerdings muss sehr viel Speicher für die Daten aufgewendet werden. Das bedeutet ein Navigationsgerät müsste mit deutlich mehr Speicherkapazität ausgerüstet werden, um das Kartenmaterial abspeichern zu können. Um dies zu vermeiden und auch die spätere Kalkulation mit den Daten effizienter zu machen, können diese Daten gerundet werden. Bei den Daten könnte es sich auch um andere Dinge handeln, wie Durchschnittsgeschwindigkeiten auf Streckenabschnitten.

Nun könnte man hergehen und sagen, es wird einfach jede Kante mathematisch gerundet. Allerdings kann sich auf einer Strecke der Rundungsfehler beim mathematischen Runden stark aufsummieren, beispielsweise bei einer Strecke, deren Kanten alle abgerundet werden. Es würde deutlich weniger Höhe angegeben werden, als tatsächlich überwunden wurde. Es soll daher ein besserer Rundungsalgorithmus für die Problematik der Kostenrundung von Kanten auf Pfaden gefunden werden. Für eine zweite Vergleichsgrundlage für eigene Algorithmen wird ein zufälliges Runden herangezogen, bei dem die Wahrscheinlichkeit aufzurunden durch die Nachkommastellen gegeben ist.

Diese Arbeit befasst sich nun mit neuen Ansätzen zur Kostenrundung von Kanten. Ein erster Ansatz, die Pfadkorrektur, schlägt fehl, bietet aber einen Ansatz für eine Verbesserung des zweiten Ansatzes. Im zweiten Ansatz, dem alternierenden Runden, wird Ab- und Aufrunden abgewechselt. Das sorgt dafür, dass abwechselnd zu viele und zu wenige Höhenmeter gegangen werden. Für diesen Ansatz werden zwei Erweiterungen entwickelt, aus denen sich der dritte Ansatz ergibt. Beim dritten und letzten Ansatz handelt es sich um gewichtetes Runden. Hierbei wird der Rundungsfehler, der bis zu jedem Knoten gemacht wird, in diesem notiert. Werden dann die ausgehenden Kanten gerundet, geschieht dies auf Basis ihres eigenen Rundungsfehlers und des Rundungsfehlers bis zum Knoten. Dieser letzte Ansatz ist mit Abstand der, der in der empirischen Auswertung am wenigsten Fehler aufweist.

2 Verwandte Themen

In den folgenden beiden Abschnitten werden zwei verwandte Themen kurz erläutert. Bei dem ersten handelt es sich um summenerhaltendes Runden. Bei dem zweiten handelt es sich um das Dithering. Dieses entstammt der Bildverarbeitung.

2.1 Summenerhaltendes Runden

Dieser Abschnitt befasst sich mit der Technik des summenerhaltenden Rundens.

Setzt sich ein gerundetes Ergebnis aus mehreren gerundeten Werten zusammen, so besteht die Möglichkeit, dass die Summe der gerundeten Werte nicht dem gerundeten Ergebnis entspricht. In einem solchen Falle werden solange gerundete Werte verändert, bis die Summe der gerundeten Werte der tatsächlichen gerundeten Endsumme entspricht. Dazu werden zunächst alle Summanden abgerundet. Für jede Zahl wird die Differenz zwischen exaktem und abgerundetem Ergebnis ermittelt. Es wird solange auf die Zahl mit der größten Differenz, die noch unverändert ist, 1 addiert, bis die Summe der gerundeten Ergebnisse der gerundeten Endsumme entspricht. (Vgl. [1])

Es wird das Beispiel 0,4 + 0,3 = 0,8 betrachtet. 0,8 wird auf 1 gerundet. Werden allerdings zuerst die beiden Summanden gerundet, ergibt sich: 0 + 0 = 0. 0 entspricht nicht 1. Es werden nun also die Differenzen gebildet. 0,4 – 0 = 0,4 und 0,3 – 0 = 0,3. Die Differenz bei 0,4 ist also größer. Es wird folglich zu 1 + 0 = 1 verändert.

Bei der in [1] vorgestellten Technik gilt allerdings für das Abrunden negativer Zahlen, dass die nächst kleinere Zahl erreicht wird. Daher wird diese Technik nur in abgewandelter Form verwendet.

2.2 Dithering

Dieser Abschnitt bietet eine Kurzbeschreibung des Dithering.

Beim Dithering wird die Anzahl an Farben in einem Bild reduziert. Dabei wird versucht die Farbverfälschung möglichst gering zu halten. Dies geschieht, indem beieinander liegende Pixel unterschiedliche Farben der verkleinerten Palette erhalten, die gemischt für das menschliche Auge eine andere Farbe ergeben. Es soll beispielsweise die Farbe Grau entfernt werden. Schwarz und weiß bleiben in der Farbpalette. Graue Farbtöne können dann dargestellt werden, indem Schwarze und weiße Pixel unterschiedlich angeordnet werden. (Vgl. [2])

3 Grundlagen

In diesem Abschnitt werden einige Grundlagen für die folgende Arbeit erläutert. Zunächst wird erläutert, wie ein Graph vorliegt und welche Daten durch ihn gegeben sind. Es wird kurz erläutert, weshalb die Höhen im Graphen gerundet werden. Danach werden zwei grundsätzliche Möglichkeiten vorgestellt, diese Höhen zu erhalten und eine davon ausgewählt. Aufrunden und Abrunden werden definiert. Es werden zwei Betrachtungsweisen der Höhenunterschiede vorgestellt und erläutert, warum in dieser Arbeit nur eine verwendet wird. Zuletzt wird erklärt, auf welche Art und Weise die einzelnen Rundungsalgorithmen verglichen werden. Es wird kurz das ausführende System beschrieben. Dann folgt ein Abschnitt über die zugrundeliegende Code-Struktur.

Grundsätzlich liegt der Graph als Textdatei vor. In dieser Datei werden zunächst die Anzahl an Knoten n und die Anzahl an Kanten k angegeben. Danach werden alle Knoten aufgeführt mit Breitengrad, Längengrad und Höhe. Es folgen alle Kanten mit Startknoten, Endknoten und Distanz. Entlang einer jeden Kante des Graphen besteht aufgrund der Höhe der Knoten ein Höhenunterschied. Dieser lässt sich durch Subtraktion der Höhe des Startknotens von der Höhe des Zielknotens errechnen.

Die Höhe eines Knotens liegt in Metern und zwölf Nachkommastellen vor. Diese Präzision ist für Anwendungen wie beispielsweise die Angabe von überwundener Höhe entlang einer Strecke nicht notwendig und benötigt viel Speicherplatz. Um eine Kommazahl dieser Genauigkeit zu speichern, muss mindestens der Datentyp double verwendet werden, was im Allgemeinen 64 Bit benötigt. Wird eine solche Zahl jedoch gerundet, kann sie beispielsweise auch im Datentyp short abgespeichert werden und benötigt so nur noch 16 Bit, also nur noch ein Viertel so viel Speicherplatz wie zuvor. Durch den dennoch großen Wertebereich von -32.768 bis 32.767 besteht die Möglichkeit, die Genauigkeit durch Wahl der Einheit zu variieren. Wird die Einheit Meter gewählt, kann der Wert direkt weiterverarbeitet werden; wird beispielsweise Dezimeter gewählt, muss der Wert noch mit 10 multipliziert werden. Im Folgenden wird mit der Einheit Dezimeter gearbeitet.

Es bestehen generell zwei Möglichkeiten die Höhe zu runden. Zum einen kann die Höhe jedes Knotens gerundet werden, zum anderen kann zunächst der Höhenunterschied entlang der Kante zwischen zwei Knoten errechnet werden, bevor dieser gerundet wird. Beide Möglichkeiten bieten Vor- und Nachteile. Wird die Höhe der Knoten gerundet, kann die präzise Darstellung der Höhe durch den gerundeten Wert ersetzt werden, so dass keine zusätzlichen Daten entstehen. Allerdings wirkt sich das Runden eines Knotens auf alle umliegenden Kanten aus, so dass entlang einer Kante die Summe der Rundungsfehler zweier Knoten besteht. Es kann auch zunächst der Höhenunterschied entlang einer Kante errechnet und dann ge-

rundet werden. Durch dieses Vorgehen wird jeder Kante ein Wert, der Höhenunterschied, hinzugefügt. So liegt an jeder Kante nur ein einfacher Rundungsfehler vor.

Es wird eine Kante K betrachtet, die von einem Knoten A zu einem Knoten B geht. Knoten A hat die Höhe 6,49 dm und Knoten B die Höhe 6,51 dm. Werden nun die Höhen beider Knoten mathematisch gerundet, hat Knoten B eine Höhe von 6 dm, Kante B eine Höhe von 7 dm. Entlang K besteht nun ein Höhenunterschied von 1 dm, tatsächlich sind es aber nur 0,02 dm. Der Rundungsfehler von 0,49 dm beider Knoten addiert sich hier zu einem Gesamtfehler von +0,98 dm. Wird zunächst der Höhenunterschied entlang K berechnet, ergibt dieser 0,02 dm. Nach mathematischer Rundung des Wertes ergibt sich ein Höhenunterschied von 0 dm. Der Rundungsfehler beträgt hier nun -0,02 dm, deutlich weniger also, als bei der Rundung der Knoten A und B.

Für Anwendungen, die Höhenunterschiede einbeziehen, ist ein geringerer Rundungsfehler besser. Anwendungen, die keine Höhenunterschiede verwenden, können beim Einlesen der Graph-Datei den zusätzlichen Wert ignorieren. In der Graph-Datei schlägt ein weiterer Wert nicht zu sehr zu Buche. Daher wird im Folgenden die Rundungsvariante verwendet, bei der zunächst der Unterschied entlang einer Kante errechnet wird. Für positive Werte ist offensichtlich, was Abrunden und was Aufrunden bedeutet. Für negative Werte ist es das nicht. Daher wird nun definiert, dass Aufrunden den gerundeten Wert weiter von der 0 entfernt und Abrunden den gerundeten Wert der 0 annähert.

Es können zwei verschiedene Größen betrachtet werden. Zum einen kann die Abweichung der gerundeten Höhenmeter von den tatsächlichen Höhenmetern betrachtet werden. Zum anderen kann die Abweichung der gerundeten Endhöhe, die in Relation zur Anfangshöhe betrachtet wird, von der tatsächlichen Endhöhe betrachtet werden. Da bei der gewählten Art zu Runden die tatsächliche Höhe eines Knotens nicht verloren geht, kann die Endhöhe in Relation zur Anfangshöhe in den Knoten nachgesehen werden. Dafür können die tatsächlichen Höhen schlicht mathematisch gerundet werden, da hier nur ein Vergleich zwischen zwei Knoten geschieht, so dass maximal ein Fehler von 1 dm auftritt. Es wird daher Hauptaugenmerk auf die Höhenmeter gelegt.

Um die einzelnen Methoden beurteilen zu können werden auf vier verschieden großen Graphen Stichproben an Zufallspfaden ausgewertet. Die Größe der Stichproben richtet sich nach der Größe des Graphen. Bei den Graphen der Größe 100k, 500k und 1m Knoten werden jeweils etwa ein Zehntel so viele Pfade wie Anzahl Knoten ausgewertet, für 10k ein Fünftel. Dabei werden zufällige Anfangs- und Zielknoten gewählt, so dass durchaus auch kein Pfad existieren kann für einzelne Paare. Für jeden existierenden Pfad werden exakte überwundene Höhenmeter (im Folgenden nur noch „Höhenmeter“ genannt), Höhenmeter nach Rundung, Distanz und Anzahl der Kanten auf dem Pfad angegeben. Es wird für jeden Pfad die Abweichung der gerundeten Höhenmeter von den exakten Höhenmetern errechnet.

Für jede Stichprobe wird außerdem die korrigierte Stichprobenvarianz dieser Abweichung errechnet. Diese Streuung kann für die verschiedenen Methoden verglichen werden. Dabei gilt: je kleiner die korrigierte Stichprobenvarianz, desto weniger weit vom Optimum, 0 dm Rundungsfehler, streut der Fehler über die Zufallsstrecken bei einer Methode. Die Formel für die korrigierte Stichprobenvarianz lautet:

$$S^2 = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \bar{X})$$

wobei n für die Größe der Stichprobe und $\bar{X}$ für den arithmetischen Mittelwert der Stichprobe steht.

Alle Algorithmen werden auf dem gleichen System ausgeführt. Beim ausführenden System handelt es sich um einen unter Windows 7 laufenden Laptop mit Intel i5-2410M 2.30GHz Prozessor und 6 GB Arbeitsspeicher.

3.1 Code-Struktur

In diesem Abschnitt wird die für die folgende Arbeit verwendete Code-Struktur erläutert. Es werden die verschiedenen Klassen aufgeführt und erklärt wie sie mit einander zusammenhängen.

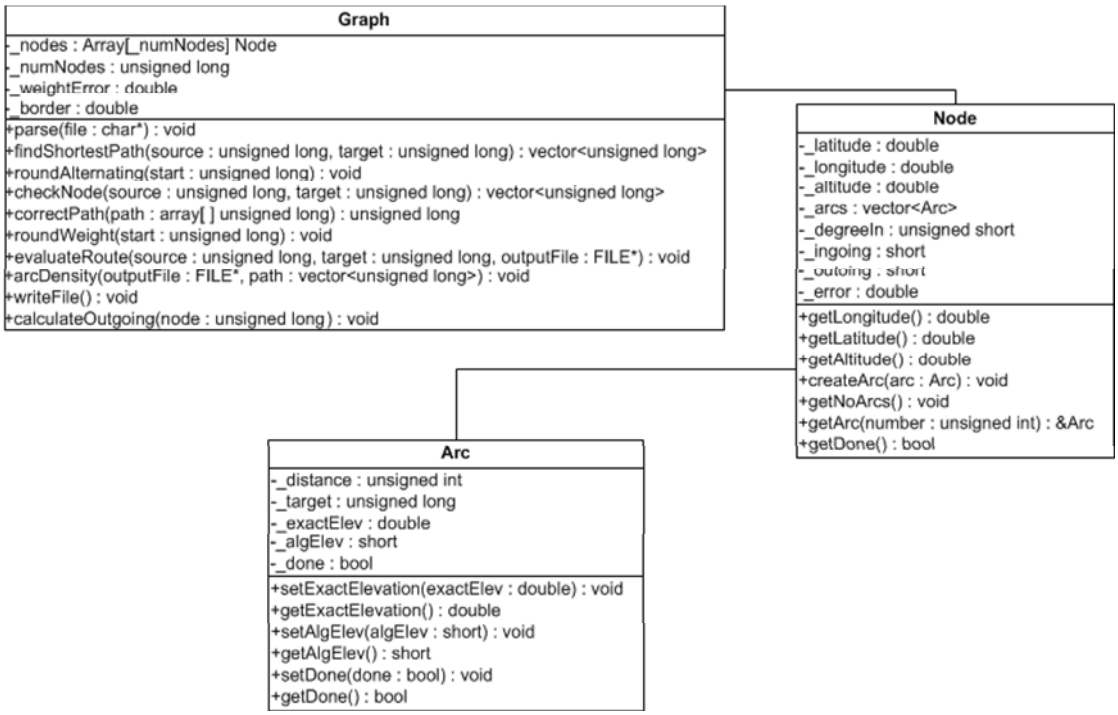

Abbildung 1: Vereinfachte Code-Struktur

In Abbildung 1 ist eine Vereinfachung der zugrundeliegenden Code-Struktur zu sehen.

Die niedrigste Klasse ist Arc. Sie enthält alle Informationen einer Kante, außer den Ausgangsknoten der Kante. Die Distanz und der Zielknoten einer Kante müssen dem Konstruktor übergeben werden. Der Zielknoten wird dabei über die ID des Zielknotens angegeben. Exakter Höhenunterschied und gerundeter Höhenunterschied können später über den jeweiligen Setter gesetzt werden. Die Variable _done wird verwendet, um für bestimmte Rundungsalgorithmen festzulegen, dass die Kante bearbeitet wurde.

Die Klasse Arc wird von der Klasse Node verwendet. Jedes Objekt der Klasse Node enthält alle Informationen eines Knotens, außer seiner ID im Graphen. Breitengrad, Längengrad und Höhe des Knotens müssen dem Konstruktor übergeben werden. Außerdem enthält er einen Vektor aus Kanten. Jede der dort aufgeführten Kanten geht von dem Knoten aus. Eine Kante in diesen Vektor einzufügen geht über createArc. Es können die Anzahl an Kanten und eine bestimmte Kante aus dem Vektor zurückgegeben werden. Die Funktion getDone überprüft für alle Kanten, ob sie bearbeitet wurden. Ist dies der Fall, gilt der Knoten als beendet und die Funktion gibt true zurück. Über die Variable _degreeIn wird der Eingangsgrad, also die

Anzahl an eingehenden Kanten, des Knotens festgehalten. Die Variable _ingoing ist dafür da, beim alternierenden Runden die Richtung, in die die eingehenden Kanten gerundet wurden, abzuspeichern. In _outgoing wird dann festgelegt, in welche Richtung die ausgehenden Kanten zu runden sind.

Die oberste Klasse ist Graph. Sie verwendet die Klasse Node und damit indirekt die Klasse Arc. Dem Konstruktor eines Objektes des Klasse Graph wird der Pfad zu einer Textdatei übergeben. Diese Textdatei hat die Knoten und Kanten des Graphen als Inhalt. Der Konstruktor ruft mithilfe dieses Pfades die parse Funktion auf. Diese liest zunächst die Knoten ein und legt sie in einem Array ab. Die Position im Array wird dabei durch die Zeile, in der die Daten des Knotens stehen, bestimmt. Diese Position ist die ID des Knotens. Die Funktion findShortestPath sucht den kürzesten Pfad zwischen den übergebenen Knoten. Diese Funktion ist mit dem Dijkstra-Algorithmus implementiert. Die Funktion roundAlternating implementiert den im Kapitel 5.2 Alternierendes Runden beschriebenen Algorithmus alternierendes Runden. Die Funktionen checkNode und correctPath werden benötigt, um die im Subkapitel 5.2.1 Zusatz: Superkanten beschriebene Erweiterung zu implementieren. Die Funktion calculateOutgoing errechnet auf Basis des im Knoten abgespeicherten Fehlers die Rundungsrichtung der ausgehenden Kanten. Dies wird für die im Subkapitel 5.2.2 Zusatz: Gewichtung beschriebene Erweiterung für das alternierende Runden verwendet. RoundWeight implementiert den im Kapitel 5.3 Gewichtetes Runden beschriebenen Algorithmus. Die Funktion evaluateRoute evaluiert einen einzelnen Pfad bezüglich dessen exakten und gerundeten Höhenmetern, seiner Distanz und Kantenanzahl. Außerdem werden Anfangs- und Endknoten des Pfades ausgegeben. Mit der Funktion arcDensity wird ein Pfad auf seine Kantendichte innerhalb eines Intervalls hin untersucht. Dabei wird dieses Intervall über die Distanz des Pfades geschoben, bis das Ende erreicht ist. Mit writeFile wird eine neue Textdatei für den nach dem Runden entstandenen Graphen erzeugt.

Außerdem existiert ein namespace Rounding, das in der Abbildung allerdings nicht zu sehen ist. In diesem namespace werden die für das mathematische Runden und das Zufallsrunden benötigten Funktionen definiert.

4 Basisalgorithmen

In diesem Abschnitt werden die beiden Basisalgorithmen mathematisches Runden und Zufallsrunden erläutert. Es wird auf ihren maximal möglichen Fehler eingegangen, sowie auf ihren Mehraufwand im Vergleich zum Einlesen des Graphen ohne Runden.

4.1 Mathematisches Runden

Dieser Abschnitt handelt vom mathematischen Runden. Das mathematische Runden wird zunächst kurz erläutert. Danach wird der maximal mögliche Fehler beschrieben und die Wahrscheinlichkeit für diesen bestimmt. Es folgt der Mehraufwand im Vergleich zum Einlesen ohne Runden. Der folgende Subabschnitt befasst sich mit der empirischen Auswertung. Danach wird untersucht, welche Strukturen für große Rundungsfehler sorgen.

Das mathematisch korrekte Runden ist die Ausgangsmethode zum Runden der Höhenunterschiede. Diese Methode kann direkt beim Einlesen des Graphen vorgenommen werden. Wann immer eine Kante eingelesen wird, wird der Höhenunterschied von Start- & Zielknoten errechnet, gerundet und für die Kante abgespeichert. Das Runden erfolgt dabei mathematisch korrekt. Das bedeutet, im Gegensatz zum kaufmännischen Runden wird bei einer 5 nach der letzten beizubehaltenden Ziffer nicht immer aufgerundet. Ist eine der auf 5 folgenden Zahlen ungleich 0 wird aufgerundet. Ansonsten wird zur nächsten geraden Zahl gerundet.

Die Zahl 2,51 wird aufgrund der 1 nach der 5 auf 3 aufgerundet. Die Zahl 2,50 allerdings wird zur nächsten geraden Zahl gerundet, also zur 2 abgerundet. Die Zahl 3,50 wird ebenfalls zur nächsten geraden Zahl gerundet. Sie wird zur 4 aufgerundet. Im negativen Bereich funktioniert dies genauso. Die Zahl -2,51 wird zur -3 aufgerundet, -2,50 zur -2 abgerundet und -3,50 wieder zur -4 aufgerundet.

Der maximal mögliche Fehler entlang einer Strecke bei dieser Methode liegt bei 0,5 * m dm, wobei m die Anzahl an Kanten der Strecke ist. Die Strecke enthält damit m + 1 Knoten. Dies liegt daran, dass jede Kante um ≤ 0,5 dm auf- bzw. abgerundet werden kann. Enthält eine Strecke von m Kanten nun ausschließlich Kanten, die um 0,5 dm aufgerundet werden, summieren sich die m Rundungsfehler von 0,5 dm zu insgesamt 0,5 * m dm. Während so der maximale Fehler für längere Strecken steigt, sinkt die Wahrscheinlichkeit, diesen zu erhalten. Zur Vereinfachung werden zwei Nachkommastellen betrachtet. Es wird zunächst eine einzelne Kante betrachtet. Für die Nachkommastellen [x],01 bis [x],49 erhält man immer den Wert [x], für die Nachkommastellen [x],51 bis [x],99 immer den Wert [x + 1]. Das bedeutet, von 99 Möglichkeiten sorgen 49 für ein Abrunden und 49 für ein Aufrunden. Für [x],50 gilt, dass auf die nächste gerade Zahl, also in Abhängigkeit von [x] gerundet wird. Da für [x] eine gleich hohe Wahrscheinlichkeit besteht gerade oder ungerade zu sein, wird [x],50 durch-

schnittlich gleich oft ab- und aufgerundet. [x],00 ist bereits eine ganze Zahl und wird nicht gerundet, [x + 1],00 ebenfalls. Damit gilt:

$$\mathrm{P(auf)} = \frac{49{,}5}{99} = \frac{1}{2} \text{ und } \mathrm{P(ab)} = \frac{49{,}5}{99} = \frac{1}{2}$$

Für eine Strecke mit m Kanten folgt also, dass die Wahrscheinlichkeit, alle Kanten aufzurunden, bei $P(\forall auf) = P(auf_1) * P(auf_2) * \dots * P(auf_m) = \frac{1}{2} * \frac{1}{2} * \dots * \frac{1}{2} = \left(\frac{1}{2}\right)^m$ liegt. Wird m = 10 angenommen, liegt die Wahrscheinlichkeit alle Kanten aufzurunden bei 0,09766 %. Die Wahrscheinlichkeit den größtmöglichen Fehler zu erhalten ist noch geringer. Um den maximal möglichen Fehler zu erhalten, muss jede Kante des Weges die Nachkommastelle [x],50 haben. Außerdem muss für jedes [x] gelten, dass es gerade bzw. ungerade ist. Die Wahrscheinlichkeit, dass eine Kante ein gerades [x] hat liegt bei $P(G) = \frac{1}{2}$. Die Wahrscheinlichkeit, die Nachkommastelle [x],50 zu haben, beträgt $P(X) = \frac{1}{100}$. Die 100 kommt an dieser Stelle daher, dass [x],00 als Nachkommastelle möglich ist, auch wenn bei dieser nicht gerundet werden muss. Das kombinierte Ereignis hat eine Wahrscheinlichkeit von

$$P(X_{G50}) = P(X_G) * P(X_{50}) = \frac{1}{2} * \frac{1}{100} = \frac{1}{200}.$$

Für das Ereignis des maximalen Fehlers gilt also:

$$P(F_{max}) = P\left(X_{G50_1}\right) * P\left(X_{G50_2}\right) * \dots * P\left(X_{G50_m}\right) = \frac{1}{200} * \frac{1}{200} * \frac{1}{200} = \left(\frac{1}{200}\right)^m.$$

Wird m wieder mit 10 angesetzt, folgt eine Wahrscheinlichkeit von $9{,}77 * 10^{-22}$ % auf einem solchen Pfad, den maximal möglichen Fehler zu erhalten. Bei der Betrachtung mehrerer Nachkommastellen sinkt die Wahrscheinlichkeit weiter, da statt [x],50 nun [x],50…0 auf allen Kanten liegen muss. Je größer die Anzahl an betrachteten Nachkommastellen, desto geringer die Wahrscheinlichkeit. Bei x betrachteten Nachkommastellen bestehen 10^x mögliche Nachkommazahlen. Für den maximalen Fehler darf auf einer Strecke allerdings nur eine solche Zahl auftauchen, wodurch der Zähler unverändert bleibt. Der nahezu größte Fehler ist allerdings deutlich wahrscheinlicher. Für [x],49 oder [x],51 ist [x] irrelevant. Das bedeutet für den maximalen Fehler ist nur $P(X_{49}) = \frac{1}{100}$ bzw. $P(X_{50}) = \frac{1}{100}$ relevant. Auf einen Pfad mit m Kanten ergibt sich so:

$$P\left(F_{fastMax}\right) = P\left(X_{49_1}\right) * P\left(X_{49_2}\right) * \dots * P\left(X_{49_m}\right) = \frac{1}{100} * \frac{1}{100} * \dots * \frac{1}{100} = \left(\frac{1}{100}\right)^m$$

Wird m wieder mit 10 angesetzt, folgt eine Wahrscheinlichkeit von $1 * 10^{-18}$ %. Dabei ist der Fehler nur um 0,01 * m dm kleiner als der maximale Fehler.

Da diese Methode direkt beim Einlesen angewandt werden kann, steigert sich der Aufwand nur um eine Konstante c für jede einzelne Kante. Das bedeutet insgesamt einen Mehraufwand von c*k im Vergleich zu einem Einlesen ohne Rundungsverfahren. Auf dem ausführenden System benötigt ein Aufruf der Funktion ungefähr 0,06 µs. Insgesamt ergibt sich so für den 1m Graphen ein Mehraufwand von 137 ms.

4.1.1 Strukturen mit großem Fehler

In diesem Abschnitt wird beschrieben bei welchen Strukturen große kumulierte Rundungsfehler auftreten.

Es besteht für jede einzelne Kante die gleiche Wahrscheinlichkeit einen großen Rundungsfehler zu erzielen. Wird allerdings dieselbe Distanz einmal durch wenige Kanten und einmal durch viele Kanten überbrückt, ist der mögliche Gesamtrundungsfehler für die Strecke mit vielen Kanten höher. Dies weist darauf hin, dass Abschnitte mit hoher Kantendichte für einen höheren kumulierten Rundungsfehler sorgen könnten. Um dies zu Überprüfen wird der kumulierte Rundungsfehler auf Zufallspfaden untersucht. Dazu wird über jeden Pfad in kleinen Schritten ein Bereich geschoben. Innerhalb von diesem Bereich werden alle vollständigen Kanten gezählt. Außerdem wird die Summe des Rundungsfehlers über diesen vollständigen Kanten berechnet. Die Schrittgröße wird willkürlich auf 50 m festgelegt. Die Bereichsgröße wird auf 1800 m festgelegt. Dies hat den Sinn die die größte Kante auf den vier Graphen mit 1.725 m Distanz mitzählen zu können. Bei einem Bereich von unter 1.725 m könnte diese Kante in keinem Bereich mitgezählt werden, da nur vollständige Kanten gezählt werden. Für jeden Zufallspfad wird ausgegeben: Start- und Zielknoten, durchschnittliche Kantendichte innerhalb eines Bereiches, die maximale Kantendichte, den kumulierten Rundungsfehler bei maximaler Kantendichte, den maximalen Rundungsfehler und die Kantendichte bei maximalem Rundungsfehler auf dem Pfad. Für jeden Pfad wird dann aufgeschlüsselt, bei wie viel Prozent seiner maximalen Dichte die maximale Summe der Rundungsfehler liegt. Dies wird in 10 % Schritten gemacht. Es wird noch separat betrachtet, wie viele Pfade bei exakt 100 % ihrer maximalen Kantendichte ihren maximalen kumulierten Rundungsfehler haben.

Tabelle 1: Anzahl an Pfaden mit größtem Fehler bei Prozent von maximaler Kantendichte

% der max. Dichte	10 - 20	20 - 30	30 - 40	40 - 50	50 - 60	60 - 70	70 - 80	80 - 90	90 - 100	100
10k	1	23	69	229	326	270	323	300	424	173
100k	123	177	784	1.161	1.355	1.682	2.841	922	912	437
500k	80	1.392	4.989	8.650	8.901	7.672	7.613	5.883	4.833	1.021
1m	316	2.507	7.805	16.684	19.071	18.237	13.270	12.756	8.799	1.861

Für die in Tabelle 1 aufgeführten Zahlen gilt, dass die untere Grenze ausschließlich und die obere Grenze einschließlich ist. 0 – 10 % wird nicht dargestellt, da auf keinem der vier Graphen ein Pfad in diesen Bereich fällt. Die Spalte 100 % ist separat von der restlichen Tabelle zu betrachten. Dies ist jeweils die Anzahl an Pfaden, die bei 100 % maximaler Kantendichte ihren maximalen Rundungsfehler aufweisen. Im Bereich bis 50 % der maximalen Kantendichte treten minimal 16,54 % und maximal 30,37 % der auf einem Graphen betrachteten Pfade auf. In den Bereichen von 70 – 100 % treten insgesamt minimal 35,02 % und maximal 52,85 % der betrachteten Pfade auf. Das bedeutet, in den obersten drei Bereichen treten für jeden Graphen mehr Pfade auf als in den untersten fünf. Für den 10k Graphen gilt, dass bei 100 % der maximalen Dichte 8,89 % der Pfade auftreten. In allen Bereichen bis 40 % sind es nur 4,78 %. Auf dem 100k Graphen sind es bei 100 % 4,43 %, in allen Bereichen bis 30 % sind es nur 3,04 %. Auf dem 500k Graphen sind es bei 100 % 2,05 %, in allen Bereichen bis 30 % sind es 2,96 %. Auf dem 1m Graphen sind es bei 100 % 1,87 %, in allen Bereichen bis 30 % sind es 2,84 %. Das bedeutet, dass die untersten drei Bereiche maximal knapp 1 % mehr Pfade enthalten, als exakt 100 % der maximalen Kantendichte. Dies lässt den Schluss zu, dass eine höhere Kantendichte für einen größeren summierten Rundungsfehler verantwortlich sein kann.

Strukturen mit höherer Kantendichte sorgen also auch für höhere Rundungsfehler. Bei solchen Strukturen handelt es sich auf einem Straßengraphen, der auch Höhendaten enthält, um beispielsweise Kurven oder Gelände mit unterschiedlicher Steigung. Das bedeutet, eine Straße, die auf flachem Gelände geradeaus führt, hat wahrscheinlich eine geringere Summe von Rundungsfehlern als eine Straße, die in Kurven über hügeliges Gelände führt.

4.2 Zufallsrunden

Um eine zweite Vergleichsgrundlage zu haben, werden die Höhenunterschiede über die Kanten zufällig gerundet. Dabei werden für die Wahrscheinlichkeit aufzurunden jeweils die Nachkommastellen des exakten Höhenunterschiedes gewählt. Die Wahrscheinlichkeit wird als Zahl zwischen [0, 1) dargestellt und mit einer Zufallszahl zwischen [0, 1) verglichen. Ist diese Zufallszahl kleiner als die Wahrscheinlichkeit, wird aufgerundet. Ein Höhenunterschied von [x],000 dm kann so niemals aufgerundet werden. Hier liegt bereits eine ganze Zahl vor, so dass ein Runden auch nicht notwendig ist. Das bedeutet:

Besteht entlang einer Kante ein Höhenunterschied von 3,999 dm, hat diese eine Wahrscheinlichkeit von 99,9 % aufgerundet zu werden, da von den bei 3 Nachkommastellen möglichen 1000 Zahlen 999 zum Aufrunden führen: Liegt die Zufallszahl im Bereich [0,000, 0,998] wird aufgerundet. Im Fall von 0,999 hingegen kann auch 3,999 dm auf 3 dm abgerundet werden. Besteht ein Höhenunterschied von 3,001 dm besteht eine Wahrscheinlichkeit von 0,1 % aufzurunden. Hat die Zufallszahl einen Wert von 0,000 wird auf 4 dm aufgerundet. Bei einem Höhenunterschied von 3,000 dm besteht eine Wahrscheinlichkeit von 0,0 % aufzurunden, es wird also in jedem Fall „abgerundet“ auf 3 dm.

Im Vergleich zum mathematischen Runden erhöht sich der maximal mögliche Fehler auf das Doppelte. Da für jede Kante eine Wahrscheinlichkeit aufzurunden besteht, kann über eine Strecke jede Kante aufgerundet werden. Wird nun eine Strecke von m Kanten (bzw. m + 1 Knoten) betrachtet, deren Höhenunterschiede alle einen Nachkommawert von [x],001 haben und aufgerundet wurden, ergibt sich für die gesamte Strecke ein Fehler von 0,999 * m dm. Der Fehler lässt sich also mit O(m) abschätzen.

Die Wahrscheinlichkeit den maximalen Fehler zu erhalten ist allerdings geringer, als beim mathematischen Runden. Es muss auf einem Pfad nicht nur immerzu die gleiche Kommazahl auftreten, sondern auch die Zufallszahl muss für jede Kante gleich gering oder hoch ausfallen. Wie bereits beim mathematischen Runden wird zunächst zur Vereinfachung nur eine Nachkommastelle betrachtet. Weiterhin wird zur Vereinfachung angenommen, dass die Zufallszahlen alle mit gleicher Wahrscheinlichkeit auftreten. Die Wahrscheinlichkeit auf einer Kante die Nachkommastelle [x],1 zu haben, beträgt $P(X_1) = \frac{1}{10}$. Um den Wert dieser Kante aufzurunden, muss eine Zufallszahl von 0,0 erhalten werden; dies ist mit $P(Z_{0,0}) = \frac{1}{10}$ der Fall. Das bedeutet, die Wahrscheinlichkeit eine Kante mit Nachkommastelle [x],1 aufzurunden, liegt bei

$$P(X_{1auf}) = P(X_1) * P(Z_{0,0}) = \frac{1}{10} * \frac{1}{10} = \frac{1}{100}.$$

Dies müsste für jede Kante eines Pfades eintreten um den maximal möglichen Fehler zu erhalten. Die Wahrscheinlichkeit hierfür liegt bei einem m Kanten langen Pfad bei:

$$P(F_{max}) = P(X_{1auf1}) * P(X_{1auf2}) * \dots * P(X_{1aufm}) = \frac{1}{100} * \frac{1}{100} * \dots * \frac{1}{100} = \left(\frac{1}{100}\right)^m$$

Diese Rundungsmethode lässt sich direkt beim Einlesen des Graphen durchführen. Der Aufwand erhöht sich dabei für jede Kante um eine Konstante c. Insgesamt erhöht sich der Aufwand also um c * k. Auf dem ausführenden System benötigt ein Aufruf der Funktion ungefähr 0,11 µs. Insgesamt ergibt sich so für den 1m Graphen ein Mehraufwand von 240 ms.

5 Eigene Ansätze

In diesem Kapitel werden eigene Ansätze zum Runden auf Graphen vorgestellt. Bei dem ersten Ansatz handelt es sich um die Idee der Pfadkorrektur. Der zweite Ansatz basiert auf dem Dithering, es handelt sich um alternierendes Runden. Dieser Ansatz wird um zwei Zusätze erweitert. Der dritte Ansatz entwickelt sich aus dem zweiten, es handelt sich um gewichtetes Runden.

5.1 Pfadkorrektur

Dieser Abschnitt befasst sich mit der Idee der Pfadkorrektur. Zunächst wird die Idee erläutert. Danach wird erklärt, warum die Pfadkorrektur nicht zielführend ist.

Alle Kanten des Graphen sollen gerundet werden, indem solange zufällige Pfade korrigiert werden, bis alle Kanten gesehen wurden. Die Korrektur erfolgt dabei mit Hilfe einer Abwandlung des summenerhaltenden Rundens. Das Abrunden wird wie in Kapitel 3 Grundlagen definiert durchgeführt. Da positive und negative Höhenmeter separat betrachtet werden, wird bei den negativen Höhenmetern für die Zahlen mit größter Differenz 1 subtrahiert anstatt addiert. Die Korrektur erfolgt in zwei Schritten:

Im ersten Schritt werden die gesamten Höhenmeter, die gesamten positiven Höhenmeter und die gesamten negativen Höhenmeter errechnet. Dabei werden exakte, abgerundete und mathematisch gerundete Werte verwendet. Für die gesamten Höhenmeter werden die exakten und die mathematisch gerundeten Werte verwendet. Für die positiven und negativen Höhenmeter werden die exakten Werte und die abgerundeten Werte verwendet, da diese nur benötigt werden, wenn die exakten Höhenmeter nach Rundung nicht mit den gerundeten Höhenmetern übereinstimmen. Stimmen sie überein, passen die Höhenmeter über den Pfad bereits und es muss nichts getan werden. Andernfalls wird überprüft, ob nach mathematischer Rundung die Summe aus exakten positiven und absoluten negativen Höhenmetern den gerundeten, exakten gesamten Höhenmetern entspricht. Ist dies der Fall, muss hier nichts weiter geschehen. Ist dies nicht der Fall, wird überprüft, ob die Summe größer oder kleiner ist. Im Fall von größer wird je nach Differenz von den positiven Höhenmetern 1 abgezogen oder auf die negativen Höhenmeter 1 addiert. Im Fall von kleiner wird je nach Differenz auf die positiven Höhenmeter 1 addiert oder von den negativen Höhenmetern 1 subtrahiert. Mit diesen Werten erfolgt danach Schritt 2.

An einem Beispiel erklärt:

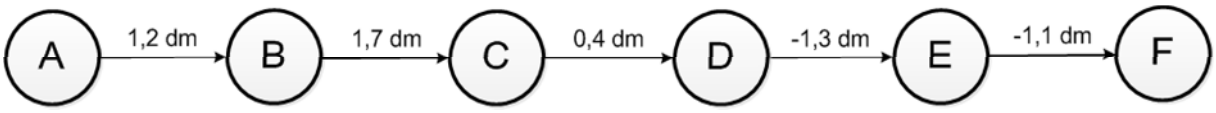

Abbildung 2: Beispielpfad zur Pfadkorrektur Schritt 1

Es wird der Pfad aus Abbildung 2 betrachtet. Die exakten Höhenmeter betragen 5,7 dm, die mathematisch gerundeten Höhenmeter betragen 5 dm. Die exakten positiven Höhenmeter betragen 3,3 dm, die abgerundeten positiven Höhenmeter betragen 2 dm. Die exakten negativen Höhenmeter betragen -2,4 dm, die abgerundeten positiven Höhenmeter betragen -2 dm. Der Vergleich zwischen gerundeten exakten Höhenmetern und mathematisch gerundeten Höhenmetern zeigt, dass die mathematisch gerundeten Höhenmeter zu gering sind, 5 dm sind kleiner als 6 dm. Es werden die exakten positiven und negativen Höhenmeter mit den exakten gesamten Höhenmetern nach mathematischer Rundung verglichen: 3 dm + |-2 dm| ≠ 6 dm. Folglich muss eine Korrektur erfolgen. Die beschriebene Differenz für die positiven Höhenmeter beträgt 0,3 dm und für die negativen Höhenmeter 0,4 dm. Es wird also von den negativen Höhenmetern 1 abgezogen. Sie ergeben sich nun zu -3,4 dm, gerundet -3 dm. Wird nun erneut mit den gesamten Höhenmetern verglichen, zeigt sich 3 dm + |-3 dm| = 6 dm. An diese Werte wird im nächsten Schritt angenähert.

Der zweite Schritt ist zweigeteilt. Positive und negative Höhenmeter werden separat betrachtet und korrigiert. Grundsätzlich werden die gleichen Schritte ausgeführt. Bei den negativen Höhenmetern ist es allerdings leichter, die Differenzen der exakten von den abgerundeten Zahlen im Betrag zu vergleichen. Außerdem wird zum betragsmäßigen Erhöhen subtrahiert und zum betragsmäßigen Verringern addiert. Bei Betrachtung der positiven Höhenmeter werden alle Kanten, deren exakter Höhenunterschied < 0 ist, ignoriert, bei Betrachtung der negativen Höhenmeter die > 0. Kanten, deren exakte Höhenmeter = 0 sind, werden generell ignoriert. Der Einfachheit halber wird das Vorgehen bei den positiven Höhenmetern beschrieben. Solange die gerundeten Höhenmeter von den exakten Höhenmetern abweichen, muss korrigiert werden. Dazu werden alle Kanten mit positivem Höhenunterschied, die noch nicht abgeändert wurden, betrachtet. Aus ihnen wird die Kante mit der größten und die mit der kleinsten Differenz zwischen exaktem und abgerundetem Wert ausgewählt. Bei mehreren Kanten mit gleicher Differenz wird aus programmiertechnischen Gründen die letzte ausgewählt. Dies kann dazu führen, dass die veränderten Kanten, die meist einen höheren Rundungsfehler aufweisen, sich am Ende des Pfades sammeln. Überwinden die gerundeten Kanten zu viel Höhe, wird von der kleinsten Kante 1 abgezogen. Bei zu wenig Höhe wird auf die größte 1 addiert. Schritt 2 wird so oft ausgeführt, bis der Fehler korrigiert ist.

Am Beispiel erklärt:

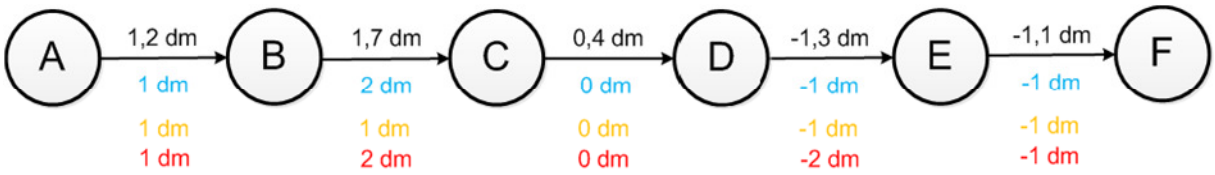

Abbildung 3: Beispielpfad zur Pfadkorrektur Schritt 2

In Abbildung 3 ist erneut der Beispielpfad zu sehen. Das Beispiel wird an dieser Stelle fortgesetzt. Nach Rundung sollten die Kanten mit positivem Höhenunterschied insgesamt 3 dm überwinden. Die Kanten mit negativem Höhenunterschied sollen -3 dm überwinden. In der Abbildung sind in Blau die mathematisch gerundeten Höhenunterschiede abgebildet. In Orange stehen die Ausgangszahlen für die Durchführung des zweiten Schritts. Für die positiven Höhenmeter sieht man nun, dass die Summe der abgerundeten Werte nicht mit dem zu erzielenden Wert übereinstimmt. Es wird also nach der Kante mit größter Differenz zwischen exaktem Wert (Schwarz) und abgerundetem Wert (Orange) gesucht. Es handelt sich um die Kante zwischen B und C. Auf diese Kante wird nun 1 addiert. Für die negativen Werte stimmt die Summe der abgerundeten Werte ebenfalls nicht mit dem zu erzielenden Wert überein. Auch hier wird nach der Kante mit größter Differenz gesucht. Es handelt sich um die Kante zwischen C und E. Von dem Wert der Kante wird 1 abgezogen. In der Abbildung sind die endgültigen Werte in Rot zu sehen. Diese werden nach nur einer Ausführung des zweiten Schritts erreicht.

5.1.1 Problematik

Dieser Abschnitt befasst sich mit der Problematik des Rundens durch Pfadkorrektur.

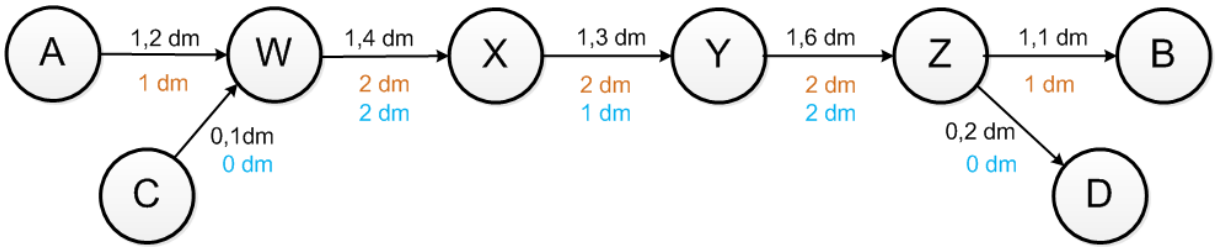

Abbildung 4: Pfadüberlagerung bei der Pfadkorrektur

In Abbildung 4 sind zwei Pfade zu sehen, die sich drei Kanten teilen. Ein Pfad verläuft von A zu B, ein Pfad von C zu D. In Schwarz ist an jeder Kante der exakte Höhenunterschied angegeben. In Orange sind die Werte, die die Pfadkorrektur für den Pfad von A nach B ermittelt, angegeben. In Blau sind die Werte angegeben, die die Pfadkorrektur für den Pfad von C nach D angegeben. Es wird dabei offensichtlich, dass für die Kante zwischen X und Y zwei verschiedene Werte auftreten. Es wird angenommen, dass zuerst der Pfad von A nach B gefunden wurde. Es ergeben sich zwei Möglichkeiten: Entweder einmal gesehene Kanten können nicht mehr verändert werden oder sie können verändert werden. Können die Kanten nicht mehr verändert werden, besteht keine Möglichkeit den Pfad von C nach D zu korrigie-

ren. Kanten, die positive exakte Höhenunterschiede haben, können nicht negativ werden. Es müsste abgebrochen werden. Können bereits gesehene Kanten erneut verändert werden, würde die Korrektur für den Pfad von A nach B zerstört. Es könnte keine Aussage mehr über den gesamten Rundungsfehler über einen Pfad getroffen werden. Aus diesen Gründen wird dieser Ansatz verworfen.

5.2 Alternierendes Runden

Dieser Abschnitt befasst sich mit dem alternierenden Runden. Es wird erklärt, wie das alternierende Runden funktioniert, welcher maximale Fehler auftritt und um wie viel sich der Aufwand erhöht. Es folgen Abschnitte über Erweiterungen für den Grundalgorithmus des alternierenden Rundens: Bei der ersten Erweiterung handelt es sich um Superkanten, bei der zweiten um Gewichtung beim Runden. Die Erweiterungen werden nicht gleichzeitig verwendet, sondern nur separat voneinander. Aus den beiden Erweiterungen des alternierenden Rundens ergibt sich die dritte Idee, das gewichtete Runden.

Die Idee des alternierenden Rundens basiert auf dem Dithering. Beim Dithering werden durch Verwendung bestimmter Farben andere Farben dargestellt. Es kann beispielsweise durch Abwechseln von Schwarz und Weiß Grau dargestellt werden. Hier soll durch alternieren von Auf- und Abrunden, also das Abwechseln von zu viel und zu wenig überwundener Höhe, ein geringer gesamter Rundungsfehler erzielt werden. Dabei wird nicht auf einzelnen Kanten alterniert, sondern es gilt für alle von einem Knoten ausgehenden Kanten, dass sie einheitlich auf- oder abgerundet werden.

Zunächst wird ein Zufallsknoten aus dem gesamten Graphen ausgewählt. Dann wird für diesen Knoten bestimmt, ob ein Auf- oder Abrunden aller von ihm ausgehenden Kanten einen geringeren Fehler hat. Dieses wird dann ausgeführt. Jede Kante erreicht in eine der beiden Richtungen gerundet einen Knoten, für diesen wird nun festgelegt, dass alle seine Kanten in die andere Richtung gerundet werden. Außerdem wird festgehalten, in welche Richtung die eingehenden Kanten gerundet wurden, da andere in diesen Knoten eingehende Kanten erst zu einem späteren Zeitpunkt erreicht werden können. Stimmt die Rundungsrichtung einer solchen Kante mit der Eingangsrichtung des Zielknotens überein, wird der Wert der Kante einfach in die vorgegebene Richtung gerundet. Stimmt sie nicht überein, wird der Wert der Kante mathematisch gerundet. Das Muster breitet sich dabei ähnlich einer Breitensuche aus. Das bedeutet Folgendes: Jeder Knoten, der vom Startknoten aus mit x Kanten erreicht werden kann, wird abgearbeitet, bevor ein Knoten, der x + 1 Kante vom Startknoten entfernt ist, abgearbeitet wird.

In Pseudo-Code:

```
1  void function roundAlternating()
2  done = false;
3  start = get random node;
4  set outgoing of starting node;
5  list seenNodes;
6  seenNodes add start;
7  while (done is false) do
8      while (seenNodes is not empty) do
9          activeNode = first of seenNodes;
10         delete first of seenNodes;
11         for (all arcs of activeNode) {
12             if (targetNode was never seen) {
13                 round arc according to outgoing of active;
14                 seenNodes add targetNode;
15                 set ingoing and outgoing of target node;
16                 set arc as done;
17             }else {
18                 if (outgoing of active matches ingoing of target) {
19                     round arc according to outgoing of active;
20                 } else {
21                     round mathematically;
22                 }
23             }
24         }
25     }
26     done = true;
27     for (all nodes) {
28         if (node not done) {
29             done = false
30             start = get new starting node;
31             seenNodes add start;
32         }
33     }
34 end
```

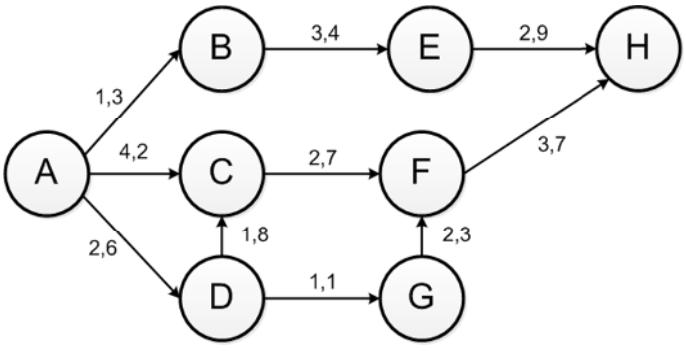

Abbildung 5: Beispielgraph für alternierendes Runden

In Abbildung 5 ist ein Beispielgraph zu sehen, anhand dessen das alternierende Runden erläutert wird. Für alle Kanten wird als Einheit Dezimeter verwendet. Als Startknoten wird Knoten A ausgewählt, allgemein wird dies zufällig bestimmt, siehe im Pseudo-Code Zeile 3. Wie in Zeile 4 wird nun bestimmt in welche Richtung die von A ausgehenden Kanten gerundet werden: Werden alle Kanten aufgerundet, ergibt sich ein Fehler von +1,9 dm. Werden alle Kanten abgerundet, ergibt sich ein Fehler

von -1,1 dm, daher wird abgerundet. A wird nun in die Liste der gesehenen Knoten eingefügt und die while-Schleife in Zeile 7 beginnt. Der aktive Knoten ist nun Knoten A, nacheinander werden die Kanten zu B, C und D gefunden. Wie in Zeile 12 wird überprüft, ob die Knoten bereits gesehen wurden. Da dies nicht der Fall ist, werden wie in Zeile 13 bis 16 die Kanten jeweils in die festgelegte Richtung gerundet und als beendet markiert. Die Knoten werden in die Liste der gesehenen Knoten eingefügt, die Eingangsrichtung gemerkt und die Ausgangsrichtung festgelegt. Sind alle ausgehenden Kanten eines Knoten fertig, ist der Knoten fertig.

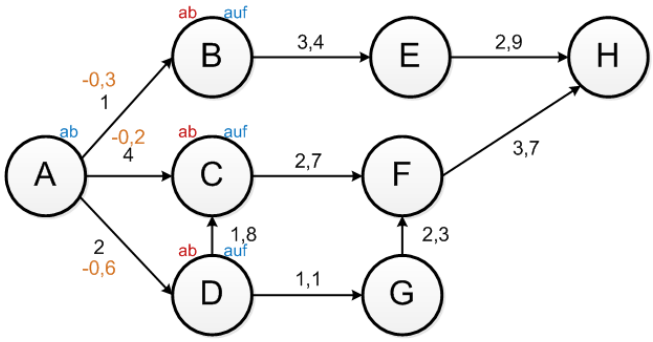

Abbildung 6: Beispielgraph nach Schritt 1

In Abbildung 6 ist der Beispielgraph nach Schritt 1 zu sehen. In Rot wird an den bereits gesehenen Knoten die Richtung festgehalten, in die ihre zuerst gesehene Eingangskante gerundet wurde. In Blau wird die Rundungsrichtung angegeben, in die ihre ausgehenden Kanten zu runden sind. Die Zahlen in Orange geben den auf der jeweiligen Kante gemachten Fehler an. In Schritt 2 gilt für die Kanten E, F und G, dass sie noch nicht gesehen wurden. Es gilt also, dass sie wie bereits erklärt gerundet werden. Für Knoten C gilt, dass er bereits gesehen wurde, also wird wie in Zeile 18 überprüft, ob Eingangsrichtung und Ausgangsrichtung übereinstimmen. Da dies nicht der Fall ist wird nach Zeile 21 die Kante mathematisch auf 2 dm gerundet.

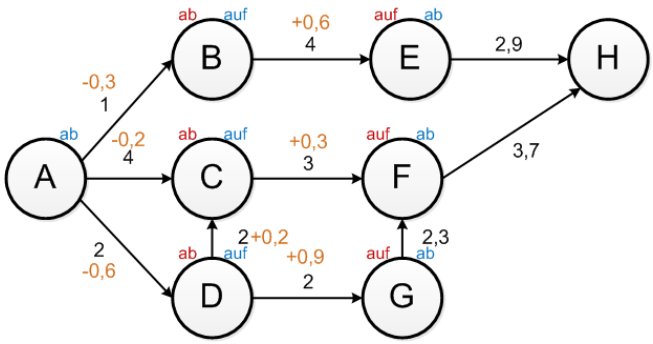

Abbildung 7: Beispielgraph nach Schritt 2

In Abbildung 7 ist der Beispielgraph nach Schritt 2 zu sehen. Für die Kante von G nach F besteht in Schritt 3 wieder ein Unterschied zwischen Ausgangsrichtung von G und Eingangsrichtung von F. Auch hier wird wieder mathematisch gerundet. Im Fall von Knoten H wird zunächst die Kante von E nach H gefunden, in diesem Moment ist Knoten H noch unbekannt. Die Eingangsrichtung von H wird auf Abrunden festgelegt. Danach findet F ebenfalls den Knoten H, es wird also nach Zeile 18 überprüft, ob

Eingangsrichtung des Knotens H und Ausgangsrichtung des Knotens F übereinstimmen. Da dies der Fall ist, wird nach Zeile 19 der Wert der Kante in die vorgegebene Richtung gerundet.

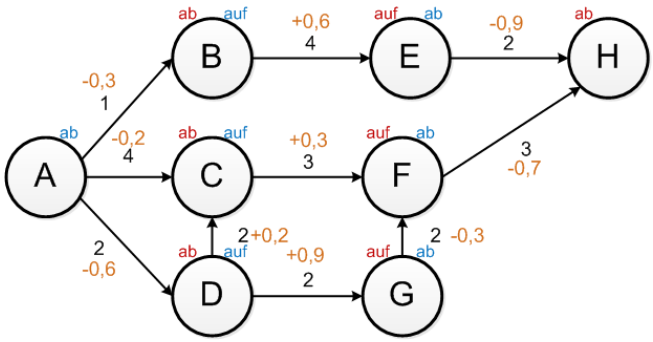

Abbildung 8: Beispielgraph nach Schritt 3

In Abbildung 8 ist der Beispielgraph nach Schritt 3 zu sehen. Da H keine ausgehenden Knoten hat, befinden sich nun keine Knoten mehr in der Liste der gesehenen Knoten. Es wird nun nach Zeile 27 für alle Knoten überprüft, ob sie fertig sind. Da dies der Fall ist, wird kein neuer Startknoten ausgewählt und der Algorithmus ist fertig.

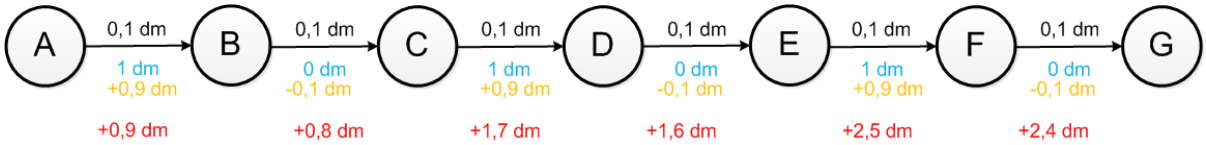

Abbildung 9: Veranschaulichung des Fehlers beim alternierenden Runden

In Abbildung 9 ist ein Pfad abgebildet, der zur Veranschaulichung für den maximalen Fehler beim alternierenden Runden dienen soll. Dieser beachtet allerdings noch keine Kanten, die mathematisch zu runden sind. In Schwarz ist der Höhenunterschied entlang der jeweiligen Kante angegeben, in Blau der Wert, auf den gerundet wird. In Orange ist der Fehler auf der jeweiligen Kante angegeben und darunter in Rot der gesamte Fehler ab Knoten A.

Für eine Kante gilt, bei Vereinfachung auf eine Nachkommastelle, dass sie um maximal 0,9 dm vom exakten Wert abweichen kann. Im alternierenden Muster zieht die folgende Kante den Gesamtfehler allerdings wieder in Richtung 0 dm. Bei einer Nachkommastelle ist dabei das Minimum, um das der Fehler verringert werden kann 0,1 dm. Das bedeutet bei 2 Kanten einen Fehler von 0,8 dm, siehe in Abbildung 9 den Pfad von A nach C. Dies deutet einen Fehler von 0,4 * m dm für gerade m an. Bringt jede zweite Kante, beginnend mit der ersten, ein Fehlerplus von 0,9 dm und erneut jede zweite Kante, beginnend mit der zweiten, ein Fehlerminus von 0,1 dm, vgl. hierzu den Pfad aus Abbildung 9. Dies ergibt sich zu:

$$0{,}9 * \frac{m}{2}\ dm - 0{,}1 * \frac{m}{2}\ dm = \left(\frac{0{,}9}{2} - \frac{0{,}1}{2}\right) * m\ dm = \frac{0{,}8}{2} * m\ dm = 0{,}4 * m\ dm.$$

Für gerade m ergibt sich also ein maximaler Fehler von 0,4 * m dm. Für ungerade m ist der

maximale Fehler zu erzielen, indem die Strecke mit einer Kante mit einem Fehler von +0,9 dm beginnt und endet, vgl. in Abbildung 9 beispielsweise den Pfad von A bis F. Das bedeutet $\frac{m-1}{2}$ Kanten erzielen ein Fehlerminus von 0,1 dm und $\frac{m-1}{2}+1$ Kanten bringen ein Fehlerplus von 0,9 dm, dies ergibt sich zu:

$$0{,}9 * \left(\frac{m-1}{2}+1\right) dm - 0{,}1 * \frac{m-1}{2} dm = 0{,}9\ dm + \left(\frac{0{,}9}{2} - \frac{0{,}1}{2}\right) * (m-1) dm = 0{,}9\ dm + 0{,}4 * (m-1)\ dm$$

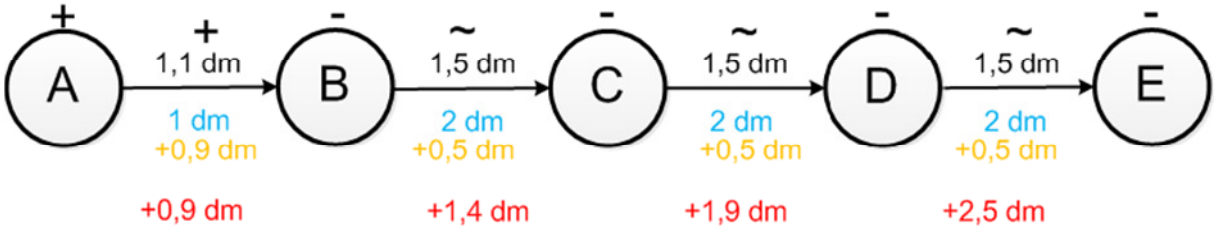

Abbildung 10: Veranschaulichung des Fehlers beim alternierenden Runden mit mathematischem Runden

Werden auch mathematisch zu rundende Kanten betrachtet, liegt der Fall anders. In Abbildung 10 ist ein solcher Pfad abgebildet. Wie bereits in Abbildung 9 sind der Höhenunterschied in Schwarz, der gerundete Wert in Blau, der Fehler entlang der Kante in Orange und die Summe des Fehlers in Rot abgebildet. Jedoch kommen nun noch die Rundungsrichtung für die Ausgangskanten oberhalb der Kanten und die tatsächliche Rundungsrichtung der ausgehenden Kanten oberhalb des Höhenunterschieds hinzu. Dabei steht „+“ für Aufrunden, „-“ für Abrunden und „~“ für mathematisches Runden. Mathematisches Runden kann nur dann eintreten, wenn die Ausgangsrichtung des Startknotens nicht mit der Eingangsrichtung des Zielknotens übereinstimmt. Das bedeutet für den Pfad in Abbildung 10, dass, wenn Kanten aus Knoten B abgerundet werden, Kanten, die in Knoten C hineingehen, aufgerundet werden müssten, um die Kante mathematisch zu runden. Eingehendes Aufrunden bedeutet für Knoten C wiederum, dass ausgehende Kanten abgerundet werden, so dass, um den Gesamtfehler nicht zu verringern, wieder mathematisch gerundet werden muss. Wird mathematisch gerundet, muss die Kante um den Gesamtfehler maximal zu erhöhen, eine Kommazahl von [x],5 haben. Insgesamt bedeutet dies für den maximalen Fehler für einen Pfad von m Kanten, dass für m – 1 Kanten der Fehler des mathematischen Rundens angesetzt wird, für eine Kante ein Fehler von 0,9 dm. Insgesamt also: 0,9 + 0,5 * (m – 1) dm.
Dies ist um 0,1 * (m – 1) dm größer als der zuvor angesetzte maximale Fehler.

Auch bei dieser Methode ist für den maximalen Fehler immer eine bestimmte Kommazahl für jede Kante notwendig. Daher ergibt sich die gleiche Wahrscheinlichkeit für den maximalen Fehler wie beim mathematischen Runden: $\mathrm{P}(\mathrm{F}_{\mathrm{max}}) = \left(\frac{1}{200}\right)^{\mathrm{m}}$, für den nahezu maximalen Fehler $\mathrm{P}(\mathrm{F}_{\mathrm{fastMax}}) = \left(\frac{1}{10}\right)^{\mathrm{m}}$.

Für unterschiedliche Startknoten ergeben sich unterschiedliche Rundungsergebnisse. Das bedeutet, dass auch der gesamte Rundungsfehler über einen Pfad für verschiedene Startknoten unterschiedlich ist.

Diese Methode des Rundens kann erst nach dem vollständigen Einlesen des Graphen durchgeführt werden. Dies ist der Fall, da für jede Kante der Zielknoten zugreifbar sein muss, um an diesem die Eingangs- und Ausgangsrundungsrichtung festlegen zu können. Es muss dabei auf jede Kante einmal zugegriffen werden. Es muss allerdings vor dem Runden der Kante auch auf den Zielknoten zugegriffen und ein Vergleich durchgeführt werden. Das führt dazu, dass zwar nur jede Kante betrachtet werden muss, der konstante Mehraufwand c pro Kante jedoch größer als beim mathematischen oder zufälligen Runden ist. Er setzt sich aus den Kosten für die Überprüfung der Ausgangsrundungsrichtung des Startknotens und der Eingangsrundungsrichtung des Zielknotens und den eigentlichen Rundungskosten zusammen. Außerdem muss vor dem Ende des Algorithmus noch für maximal alle n Knoten, und damit all k Kanten überprüft werden, ob sie fertig sind. Ist ein Knoten nicht fertig, wird mit diesem weitergemacht und die Überprüfung kann abgebrochen werden. Dies muss so oft durchgeführt werden, bis bei unzusammenhängenden Graphen alle Teilgraphen g bearbeitet wurden. Das bedeutet, unter Annahme $k > n$, einen Mehraufwand von $O(c * k)$ für das eigentliche Runden und einen Zusatz von $O(k * g)$. Insgesamt also $O(c * k + g * k) = O((c + g) * k)$. Insgesamt benötigt das ausführende System im Schnitt über 1.000 Durchläufe 36,31 Sekunden, um auf dem 1m Graphen das alternierende Runden durchzuführen.

5.2.1 Zusatz: Superkanten

Dieser Abschnitt erklärt den Zusatz der Superkanten. Zunächst wird erläutert, was eine solche Superkante ist. Danach wird die für eine solche Superkante benötigte Struktur gezeigt und dargelegt, wie algorithmisch eine solche Struktur zu finden ist.

In einem Graph können Pfade existieren, die ohne Abzweigungen oder zusätzliche Zugänge von Start bis Ziel gehen. Das bedeutet für jeden Knoten des Pfades, außer Anfangs- und Endknoten, einen maximalen Eingangsgrad von 2 und einen maximalen Ausgangsgrad von 2. Für diese Pfade kann zunächst der Höhenunterschied über alle Kanten des Pfades zu einer Superkante summiert werden. Es folgt ein Runden des Gesamtwertes im Sinne des alternierenden Musters. Danach wird der Pfad durch summenerhaltendes Runden korrigiert, bis die Summe der gerundeten Kanten der gerundeten Summe der Kanten entspricht. Bei diesem Vorgehen wird jedoch keine tatsächliche Kante in den Graphen eingefügt. Es wird lediglich der Pfad wie eine einzelne Kante betrachtet, bevor die Kanten des Pfades optimiert werden.

Es werden zunächst die nötigen Strukturen betrachtet:

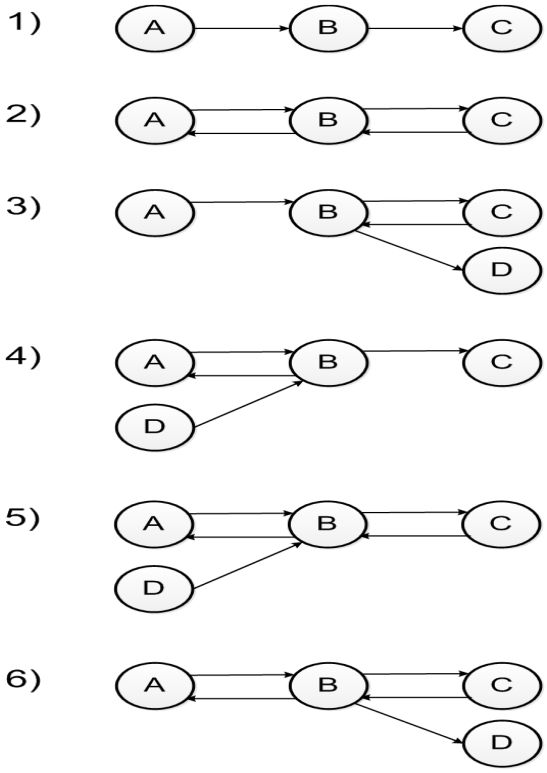

Abbildung 11: Anschauung für die Superkanten

In Abbildung 11 sind Pfade von A bis C zu sehen, bei denen B unterschiedliche Ein- und Ausgangsgrade hat. Die Ein- & Ausgangsgrade von A und C sind dabei irrelevant. Eine eingehende Kante muss zwingend vom Vorgängerknoten kommen, eine ausgehende Kante zwingend zum Nachfolgerknoten gehen. Daher sind ein Eingangsgrad von 1 und ein Ausgangsgrad von 1 für einen Pfad immer gegeben. In 1) ist genau dies der Fall, damit ist es möglich eine Superkante von A nach C zu berechnen und die Kanten entsprechend zu korrigieren. In 2) hat B einen Eingangsgrad von 2 und einen Ausgangsgrad von 2. In diesem Fall muss die zusätzliche Eingangskante vom Nachfolger kommen und die zusätzliche Ausgangskante zum Vorgänger gehen, damit eine Superkante möglich ist. In 3) ist ein Knoten B abgebildet, der zwar einen Ausgangsgrad von 2 hat, aber dessen zusätzliche Ausgangskante nicht zum Vorgänger geht. In einem solchen Fall gibt es eine Abzweigung auf dem Pfad, so dass eine Superkante nicht verwendet werden kann. In 4) ist es ein Eingangsgrad von 2, wobei die zusätzliche Eingangskante nicht vom Nachfolgeknoten kommt, so dass hier aus D ein Zugang auf den Pfad besteht. Dies sorgt dafür, dass die Superkante hier nicht verwendet werden kann. In 5) ist der Eingangsgrad von B zu hoch. In 6) ist der Ausgangsgrad von B zu hoch. In beiden Fällen kann keine Superkante gebildet und korrigiert werden. Die Superkante kann in den Fällen 3) – 6) nicht verwendet werden, da bei der Pfadkorrektur einzelne Kanten verfälscht werden. Würde in den Fällen 4) und 5) bei einer Pfadkorrektur die Kante zwi-

schen B und C verfälscht, so wäre der Pfad von D bis C verfälscht. Auf einem Pfad können dabei beliebig viele Knoten wie B aus 1) und 2) auftreten. Um solch einen Pfad zu finden, wird bei Finden eines neuen Knotens überprüft, ob der neue Knoten Struktur 1) oder 2) entspricht. In Abbildung 11 wäre es die Kante von A nach B, die den neuen Knoten B findet. Für B wird nun der Ein- und Ausgangsgrad überprüft. In 3) bis 6) würde abgebrochen und eine Liste zurückgegeben, die nur A und B beinhaltet. Damit wäre der Pfad nur eine Kante lang. Dies ist keine Superkante. In 1) und 2) würde B als gültige Struktur gesehen. Da der Pfad nicht in B endet, würde als nächstes C betrachtet. Dies wird fortgesetzt bis ein Knoten erreicht wird, der weder Struktur 1) noch 2) entspricht. Es wird in einem weiteren Fall abgebrochen, nämlich wenn der Anfangsknoten wieder erreicht würde, d.h. ein Kreis aus Knoten und Kanten vorhanden wäre. Bis zum Abbruch werden die IDs aller Knoten, inklusive Anfangs- und Endknoten des Pfades, in eine Liste geschrieben. In Pseudo-Code:

```
1  list nodeIds function findPath (startNode, endNode)
2  list nodeIds add startNodeId;
3  currentNode = endNode
4
5  while (true) {
6      if (currentNode is startNode) return nodeIds;
7      nodeIds add currentNodeId;
8      if (currentNode degreeIngoing <= 2 and degreeOutgoing <= 2) {
9          if (currentNode degreeIngoing == 1 and degreeOutgoing == 1) {
10             // degree ingoing and outgoing 1, fits structure
11             currentNode = get next node;
12         } else {
13             // degreeIngoing and/or degreeOutgoing is 2
14             if (currentNode degree ingoing == 2) {
15                 if (ingoing arc comes from next node) {
16                     currentNode = get next node;
17                 } else {
18                     return nodeIds; // done
19                 }
20             } else {
21                 // degreeOutgoing is 2
22                 if (outgoing arc goes to former node) {
23                     currentNode = get next node;
24                 } else {
25                     return nodeIds; //done
26                 }
27             }
28         }
29     } else {
30         return nodeIds;
31     }
32 }
```

Das Korrigieren erfolgt wie im Kapitel 5.1 Pfadkorrektur erläutert. Statt auf den gesamten Zufallspfad wird die Pfadkorrektur nur auf den Bereich der Superkante angewandt. Dieser Bereich wird von der eben beschriebenen Funktion gefunden.

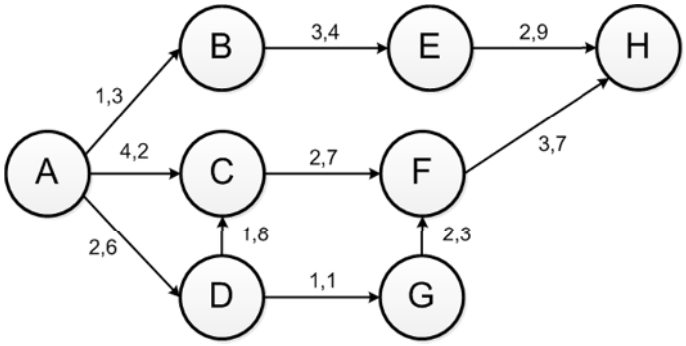

Abbildung 12: Beispielgraph für Superkanten

In Abbildung 12 ist ein Beispielgraph zu sehen, anhand dessen die Superkanten erklärt werden. Es handelt sich um den gleichen Graphen wie beim Grundalgorithmus. Als Anfangsknoten wird wieder A verwendet. Die Kanten ausgehend von A werden wieder abgerundet. Allerdings wird nun beim Auffinden eines neuen Knotens überprüft, ob dieser zu einer Superkante gehört. Die erste Kante ausgehend von A trifft auf den Knoten B. B hat Eingangsgrad 1 und Ausgangsgrad 1, ist damit Teil einer Superkante. Von B ausgehend wird E erreicht. E hat ebenfalls Eingangsgrad 1 und Ausgangsgrad 1 und ist damit auch Teil der Superkante. Von E aus wird H erreicht. H besitzt Eingangsgrad 2 und Ausgangsgrad 0. Mit Ausgangsgrad 0 gibt es keinen Nachfolgeknoten. Die Superkante endet hier. Die Superkante geht also von A über B und E nach H.

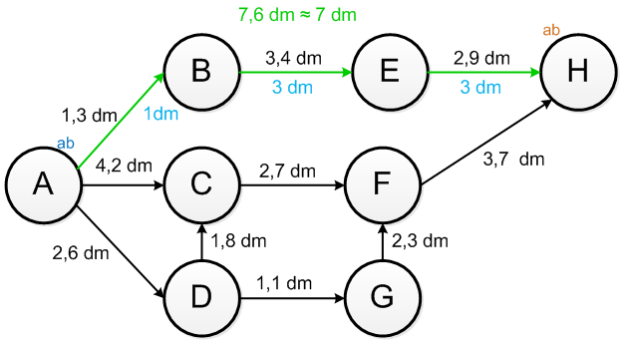

Abbildung 13: Beispielgraph; gefundene Superkante

In Abbildung 13 ist die Superkante Grün eingezeichnet. Diese wird nun auf ihre Höhenmeter überprüft und gegebenenfalls korrigiert. Die Gesamtsumme der exakten Höhenunterschiede auf der Superkante ergibt 7,6 dm. Da die von A ausgehenden Kanten abgerundet werden, wird der exakte Höhenunterschied über die Superkante abgerundet. Er beträgt damit 7 dm. Die mathematisch gerundeten Höhenunterschiede sind in Abbildung 13 in Blau eingetragen. Die Summe dieser Höhenunterschiede beträgt 7 dm. Es muss daher keine Korrektur vorgenommen werden und die Superkante ist beendet. Für den Rest des Graphen wird gleich vorgegangen, es existiert aber keine weitere Superkante.

Für den maximalen Fehler werden zwei Fälle betrachtet. Im ersten Fall sind alle Kanten einer Superkante auf dem Pfad. Die Berechnung des maximalen Fehlers bleibt gleich im Vergleich zum alternierenden Runden ohne Superkanten. Allerdings verringert sich die Anzahl an Kan-

ten. Enthält eine Superkante s Kanten, so wird dies auf 1 Kante reduziert. Das bedeutet, statt m Kanten auf dem Pfad sind es nur noch m – s + 1 Kanten. Im zweiten Fall wird nur ein Teil der Kanten einer Superkante auf einem Pfad verwendet. Für veränderte Kanten besteht ein höherer Rundungsfehler. Diese Kanten treten vermehrt gegen Ende einer Superkante auf. Verläuft ein Pfad nun nicht über die gesamte Superkante, sondern nur über beispielsweise das letzte Drittel, kann jede Kante einen Fehler von bis zu 1 dm haben. Das heißt der maximale gesamte Rundungsfehler beträgt O(m).

Im Vergleich zum Grundalgorithmus ergibt sich ein Mehraufwand. Für jeden neuen Knoten wird überprüft, ob er Teil einer Superkante ist. Das bedeutet, die Funktion zum Finden einer Superkante wird maximal n mal aufgerufen. Der Aufwand für einen Aufruf ist abhängig von der Zahl der gefundenen Knoten auf einer Superkante. Maximal können dies n – 1 sein, da der erste Knoten der Startknoten ist und für diesen nicht überprüft wird, ob er Teil einer Superkante ist. Für jeden erreichten Knoten wird seine Struktur überprüft. Diese Überprüfung braucht für jeden Knoten eine konstante Zeit. Der Aufwand kann also mit O(c * n) abgeschätzt werden. Für maximal jeden der maximal n Aufrufe wird auch die Korrekturfunktion aufgerufen. Werden der Korrekturfunktion g + 1 Knoten übergeben, können maximal g Kanten korrigiert werden. Dabei wird jedes Mal jede Kante auf ihre Differenz hin überprüft. Das bedeutet, die Korrekturfunktion hat einen Aufwand von $O(g^2)$. In bestimmten Graphen kann g = n – 1 sein. Es ergibt sich also ein Aufwand von $O(n^2)$. Insgesamt folgt ein Mehraufwand von $O(c * n^3)$. Insgesamt benötigt das ausführende System im Schnitt über 1.000 Durchläufe 43,25 Sekunden, um auf dem 1m Graphen das alternierende Runden mit Superkanten durchzuführen.

5.2.2 Zusatz: Gewichtung

In diesem Abschnitt wird der Zusatz der Gewichtung beschrieben. Die Gewichtung wird zunächst erklärt und dann am Beispiel verdeutlicht. Es folgt eine Analyse des Mehraufwandes.

Um die Ausgangsrundungsrichtung eines Knotens zu bestimmen, soll nicht mehr nur die Eingangsrundungsrichtung betrachtet werden. Es sollen mit bestimmter Gewichtung auch der Fehler über die eingehenden Kanten, die eingehenden Kanten der Vorgängerknoten und der Fehler auf den ausgehenden Kanten eines Knotens beachtet werden. Die Summe der Rundungsfehler über die ausgehenden Kanten eines Knotens wird als Eigenfehler bezeichnet. Die Summe der Rundungsfehler über die eingehenden Kanten eines Knotens wird als Eingangsfehler bezeichnet. Die Summe der Rundungsfehler über die in den Vorgängerknoten eingehenden Kanten wird als Vorgängerfehler bezeichnet. Die Gewichtungen sind entsprechend der Bezeichnungen für die Fehlersummen benannt, also Vorgängergewichtung, Eingangsgewichtung und Eigengewichtung. Da der Eigenfehler einmal mit den aufgerunde-

ten Werten und einmal mit den abgerundeten Werten berechnet werden kann, existiert er zweimal. Es existieren also auch zwei Gesamtsummen. Diese werden verglichen. Die Gesamtsumme, die mit der nach Muster vorgegebenen Rundungsrichtung gebildet wird, wird grundsätzlich vorgezogen. Ist aber die andere Gesamtsumme um eine gewisse Grenze näher an 0 dm, wird sie verwendet. Am Beispiel erläutert:

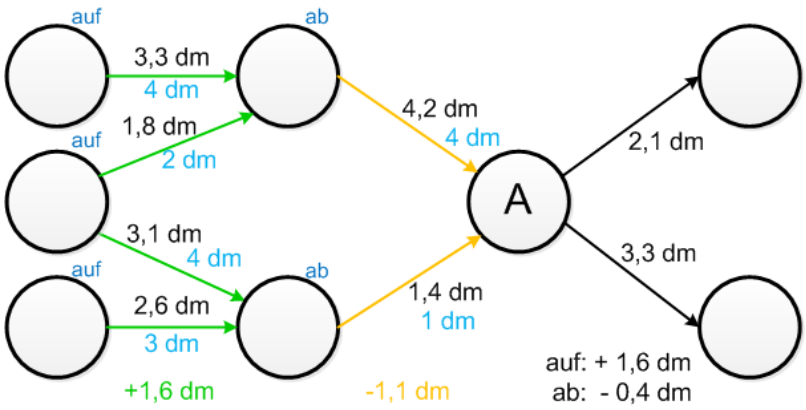

Abbildung 14: Beispiel für Gewichtung

In Abbildung 14 ist ein Beispiel für die Gewichtung gezeigt. An jeder Kante ist in Schwarz der exakte Höhenunterschied angegeben. An den bereits gerundeten Kanten steht das Rundungsergebnis in Blau. Außerdem stehen die Ausgangsrundungsrichtungen oben rechts an den Knoten. Die Kanten, über welche der Vorgängerfehler gebildet wird, sind in Grün abgebildet. Der Vorgängerfehler beträgt +1,6 dm. Die Kanten, über welche der Eingangsfehler gebildet wird, sind in Orange abgebildet. Der Eingangsfehler beträgt -1,1 dm. Die Kanten, über welche der Eigenfehler gebildet wird, sind in Schwarz abgebildet. Der Eigenfehler beim Aufrunden beträgt +1,6 dm, beim Abrunden -0,4 dm. In diesem Beispiel werden die Gewichtungen gleichwertig mit 1 gewählt. Die Grenze für die Wahl der Rundungsrichtung wird auf 0,5 dm festgesetzt. Für die beiden Möglichkeiten, die aus A ausgehenden Kanten zu runden, ergibt sich:

Aufrunden: 1 * 1,6 dm + 1 * (-1,1 dm) + 1 * 1,6 dm = +2,1 dm
Abrunden : 1 * 1,6 dm + 1 * (-1,1 dm) + 1 * (-0,4 dm) = +0,1 dm

Die Gesamtsumme beim Abrunden ist um 2,0 dm näher an 0 dm. 2,0 dm ist größer als 0,5 dm. Hier darf also das alternierende Muster durchbrochen werden. Statt die aus A ausgehenden Kanten aufzurunden, werden sie abgerundet. Es ergibt sich Abbildung 15.

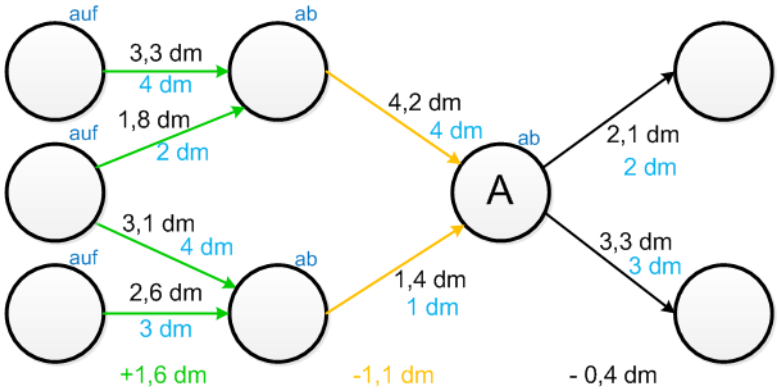

Abbildung 15: Beispielgraph nach Gewichtung

Der Mehraufwand für diesen Zusatz lässt sich linear halten. Dafür benötigt jeder Knoten eine Variable für den Eingangsfehler und eine Variable für den Vorgängerfehler. Wird nun ein Knoten über eine Kante erreicht, werden zwei Dinge getan: Zum einen wird der Rundungsfehler der Kante auf den Eingangsfehler des Zielknotens aufgeschlagen. Zum anderen wird der Eingangsfehler des Ausgangsknotens auf den Vorgängerfehler des Zielknotens addiert. So müssen für das Berechnen der Ausgangsrundungsrichtung eines Knotens nur zwei Summen gebildet werden. Die Differenz der Summen wird mit dem Grenzwert verglichen. Addition, Subtraktion und der Vergleich von Differenz und Grenzwert brauchen konstanten Aufwand. Der Mehraufwand beläuft sich also auf O(n). Insgesamt benötigt das ausführende System im Schnitt über 1.000 Durchläufe 38,42 Sekunden, um auf dem 1m Graphen das alternierende Runden mit Gewichtung durchzuführen.

5.3 Gewichtetes Runden

In diesem Abschnitt wird das gewichtete Runden erläutert. Die Idee basiert auf den beiden Erweiterungen für den Algorithmus alternierendes Runden.

Beim gewichteten Runden wird kein Rundungsmuster verwendet. Der Algorithmus beginnt an einem zufälligen Startknoten. Es wird für jede Kante einzeln bestimmt, ob sie auf- oder abgerundet wird. Dafür wird der Rundungsfehler, der bis zu einem Knoten gemacht wurde, in diesem abgespeichert. Das Ziel jeder ausgehenden Kante wird auf eine Superkante hin untersucht. Die Länge des zurückgegebenen Pfades ist irrelevant. Ist der Pfad nur zwei Knoten lang, ist keine Superkante vorhanden, ist der Pfad länger, ist eine Superkante vorhanden. Im Fall einer Superkante wird zunächst die Summe der exakten Höhenmeter gebildet. Ansonsten werden die exakten Höhenmeter der einzelnen Kante betrachtet. Es werden zwei Summen gebildet. Eine Summe wird aus dem Rundungsfehler beim Aufrunden der Höhenmeter und dem bisherigen Rundungsfehler gebildet. Die andere Summe wird aus dem Rundungsfehler beim Abrunden der Höhenmeter und dem bisherigen Rundungsfehler gebildet. Es wird

in die Richtung gerundet, deren Summe betragsmäßig kleiner ist. Das bedeutet, eine Kante oder Superkante wird immer so gerundet, dass ihr Rundungsfehler den bisherigen Rundungsfehler in Richtung 0 dm zieht. Die Superkanten werden wie im Kapitel 5.1 Pfadkorrektur korrigiert. Der bisherige Rundungsfehler wird vom Startknoten aus berechnet. Im Startknoten besteht ein Rundungsfehler von 0 dm. Es wird eine Summe aus bisherigem Rundungsfehler eines Knotens und Rundungsfehler der aus ihm hinausführenden Kante gebildet. Diese Summe wird auf den bisherigen Rundungsfehler des Zielknotens addiert. Dabei werden erst alle Knoten gesehen, die mit x Kanten vom Startknoten aus erreichbar sind, bevor die Knoten die x + 1 Kanten benötigen. Dabei wird eine Superkante als eine Kante gezählt. Jeder erreichte Knoten mit unbeendeten ausgehenden Kanten gilt als aktiv. Gibt es keine aktiven Knoten mehr, wird überprüft, ob alle Knoten beendet wurden. So werden auch auf unzusammenhängenden Graphen alle Knoten erreicht. Am Beispiel:

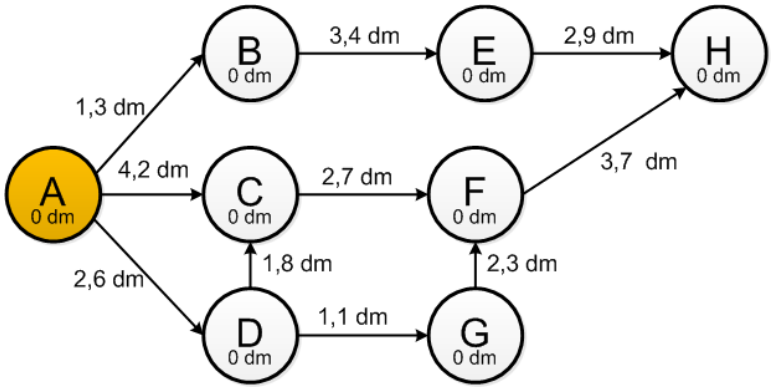

Abbildung 16: Beispielgraph gewichtetes Runden vor Rundung

In Abbildung 16 ist ein Beispielgraph zu sehen, anhand dessen das gewichtete Runden erläutert wird. Die Zahlen unter den Knotenbezeichnungen stellen den bis zu dem Knoten entstandenen Rundungsfehler dar. Zu Beginn sind alle diese Rundungsfehler 0 dm. Als Startknoten wird A gewählt, dieser Knoten ist zu Beginn aktiv. Von A aus wird als erstes Knoten B erreicht. Für Knoten B wird überprüft, ob er Teil einer Superkante ist. Das ist der Fall, von A bis H besteht eine Superkante. Über die Superkante besteht ein Höhenunterschied von 7,6 dm. Die Summe aus bisherigem Rundungsfehler und Rundungsfehler bei Aufrunden beträgt 0 dm + 0,4 dm = +0,4 dm. Die Summe aus bisherigem Rundungsfehler und Rundungsfehler bei Abrunden beträgt 0 dm – 0,6 dm = -0,6 dm. Betragsmäßig ist +0,4 dm kleiner als -0,6 dm. Die Superkante wird also aufgerundet. Die Korrektur für die Superkante erfolgt somit auf 8 dm. Da H keine ausgehenden Kanten hat, wird H nicht zu den aktiven Knoten gezählt. Diese Kante ist damit abgeschlossen. Die Kante von A nach C ist nicht Teil einer Superkante. Für 4,2 dm Höhenmeter und 0 dm bisheriger Rundungsfehler ist Abrunden besser. Diese Kante ist abgeschlossen und C gehört zu den aktiven Knoten. Die Kante von A nach D ist nicht Teil einer Superkante. Für 2,6 dm Höhenmeter und 0 dm bisheriger Rundungsfehler ist Aufrunden besser. Die Kante ist abgeschlossen und D gehört zu den aktiven Knoten.

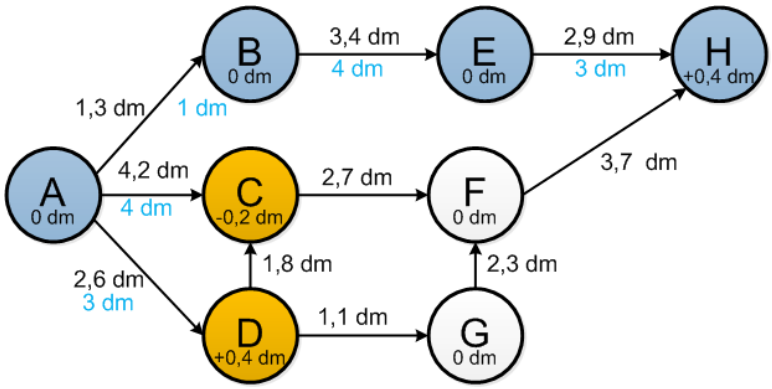

Abbildung 17: Beispielgraph gewichtetes Runden nach Abschluss Ebene 1

Abbildung 17 stellt den Beispielgraphen nach Abschluss von Ebene 1 dar. Die blau hinterlegten Knoten sind beendet. C ist der erste aktive Knoten. Er hat eine Kante zu F mit 2,7 dm Höhenmeter. Beim Aufrunden der Kante entsteht ein Rundungsfehler von +0,3 dm. Mit dem bisherigen Fehler von C addiert ergibt sich +0,1 dm Rundungsfehler. Beim Abrunden entsteht ein Rundungsfehler von -0,7 dm. Mit dem bisherigen Fehler von C ergibt sich ein Rundungsfehler von -0,9 dm. Aufrunden ist also besser. Die Kante wird auf 3 dm aufgerundet. Die Kante ist beendet und F ist aktiv. Der nächste aktive Knoten ist D. Es besteht eine Kante zu C. Durch Aufrunden und bisherigen Fehler ergibt sich ein Rundungsfehler von +0,6 dm. Durch Abrunden und bisherigen Fehler ergibt sich ein Rundungsfehler von -0,4 dm. Folglich wird abgerundet auf 1 dm. Der neue bisherige Fehler in C beträgt nun -0,2 dm + (0,4 dm – 0,8 dm) = -0,6 dm. Die Kante ist beendet. Da C bereits aktiv war, gehört C nicht zu den aktiven Knoten. Es besteht außerdem eine Kante zu G. Durch Abrunden und bisherigen Fehler ergibt sich ein Rundungsfehler von +0,3 dm. Durch Aufrunden und bisherigen Fehler ergibt sich ein Rundungsfehler von +1,3 dm. Abrunden ist folglich besser. Die Kante ist beendet und G gehört zu den aktiven Knoten.

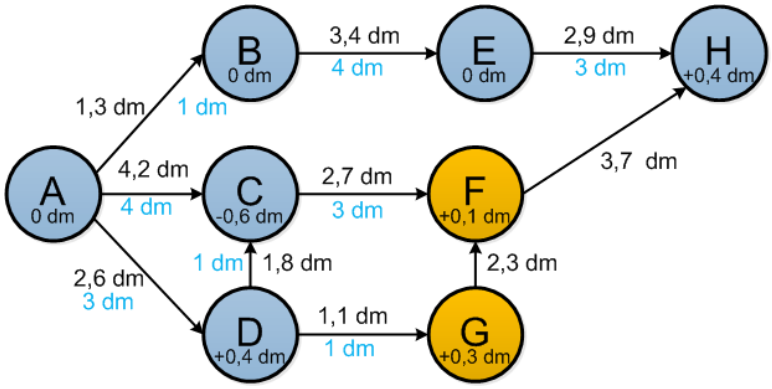

Abbildung 18: Beispielgraph gewichtetes Runden nach Abschluss Ebene 2

Abbildung 18 stellt den Beispielgraphen nach Abschluss von Ebene 2 dar. Knoten F ist der erste der verbliebenen aktiven Knoten. Durch Abrunden seiner Kante zu H und seinen bisherigen Fehler ergibt sich ein Rundungsfehler von -0,6 dm. Durch Aufrunden der Kante und seinen bisherigen Fehler ergibt sich ein Rundungsfehler von +0,4 dm. Aufrunden wir also vorgezogen. Die Kante ist beendet und H gehört nicht zu den aktiven Knoten, da er keine ausgehenden Kanten hat. Der letzte aktive Knoten ist

G. Wird seine Kante aufgerundet, ergibt sich insgesamt ein Rundungsfehler von +0,6 dm, wird seine Kante abgerundet, ergibt sich ein Fehler von 0,0 dm. Das Abrunden ist also vorzuziehen. Die Kante ist beendet und F wird kein aktiver Knoten, da F bereits aktiv war. Der Graph ist damit beendet. Es ergibt sich der Graph aus Abbildung 19.

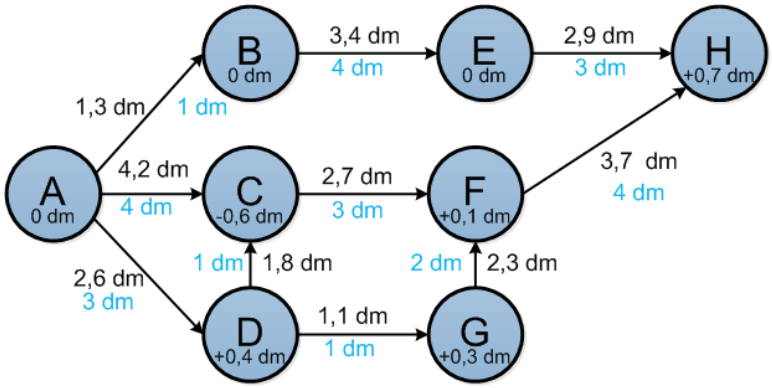

Abbildung 19: Beispielgraph nach Ende des Algorithmus

In Pseudo Code:

```
1  void function roundWeight (unsigned long start)
2  PriorityQueue active add (0, start); // 0 is the level of the node
3  vector <unsigned int > path;
4  while (active is not empty) do
5      current = first of active;
6      for (all arcs of current) {
7          path = checkNode(current, target of arc);
8          calculate rounding direction for path;
9          correct path;
10         if (target of arc was never seen before) {
11             active add (current level + 1, target of arc);
12         }
13     }
35     if (active is empty) {
36         for (all nodes) {
37             if (node not done) {
38                 start = get new starting node;
39                 active add (0, start);
40             }
41         }
42     }
43 end
```

Um die Rundungsrichtung zu bestimmen, wird zunächst überprüft ob eine Superkante vorliegt. Dies benötigt O(c * n), wie im Kapitel 5.2.1 Zusatz Superkanten beschrieben. Danach wird bestimmt, ob Auf- oder Abrunden der gefundenen Kante besser ist. Es wird dazu über alle Knoten des Pfades summiert. Dies benötigt maximal O(n), da maximal alle Knoten auf einem Pfad liegen können. Danach wird die Superkante korrigiert, was maximal $O(n^2)$ benötigt, wie aus Kapitel 5.2.1 bekannt. Insgesamt kommt der Algorithmus also auf einen Aufwand von $O(n + n + n^2)$. Auf dem ausführenden System benötigt der Algorithmus für den 1m Graphen im Durchschnitt über 1.000 Durchführungen etwa 2,14 Sekunden.

6 Auswertung der Algorithmen

Dieses Kapitel befasst sich mit der empirischen Auswertung der einzelnen Algorithmen. Die eigenen Ansätze werden mit den Basisalgorithmen verglichen. Die betrachtete Abweichung ist dabei die Abweichung der gerundeten Höhenmeter von den exakten Höhenmetern.

6.1 Mathematisches Runden

Das mathematische Runden wird auf den unterschiedlichen Graphen angewandt, anschließend wird der arithmetische Mittelwert der Abweichungen auf der Stichprobe und die korrigierte Stichprobenvarianz errechnet. Diese Auswertung bildet die Vergleichsgrundlage für die weiteren Rundungsmethoden.

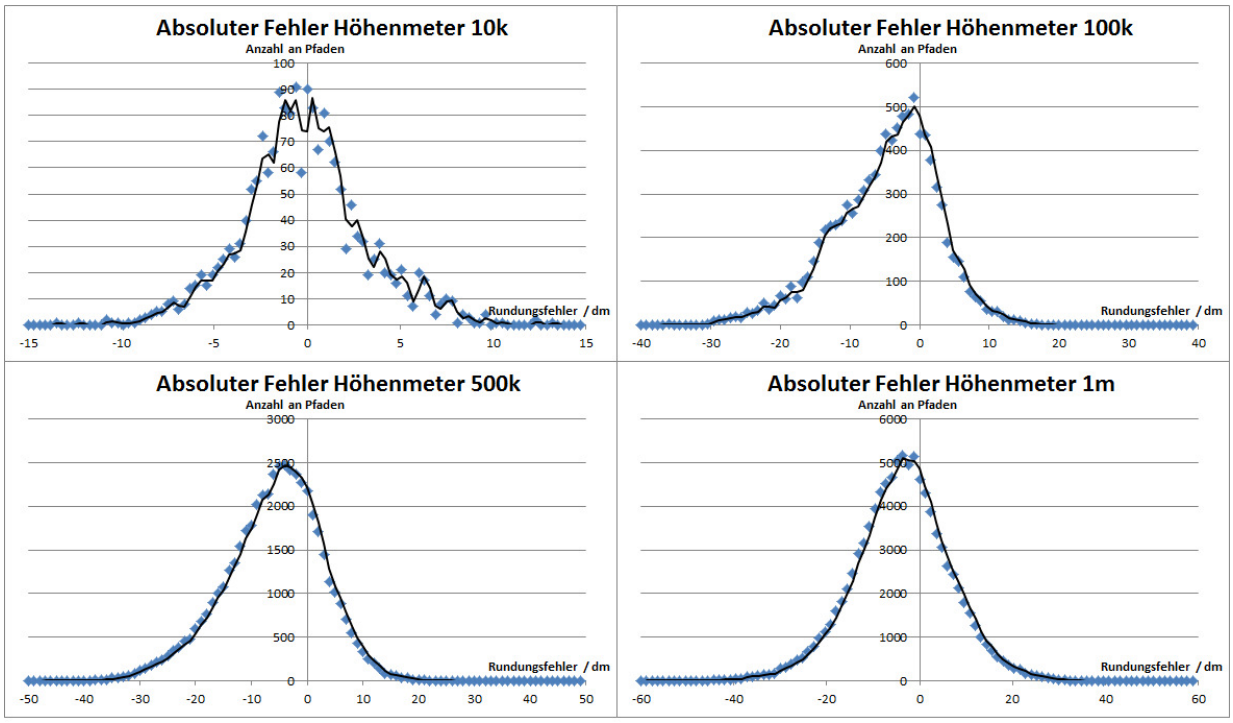

Abbildung 20: Verteilung des absoluten Fehlers der Höhenmeter

In den Graphiken in Abbildung 20 wird die Anzahl an Pfaden mit einem bestimmten gesamten Rundungsfehler der Höhenmeter aufgetragen. Die x-Achse bildet dabei den gesamten Rundungsfehler in Dezimetern ab, die y-Achse die Anzahl der Pfade. Die vier Graphiken bilden die Verteilung auf den vier Graphen ab. Blau aufgetragen sind die Datenpunkte der jeweiligen Bereiche, bei der schwarzen Linie handelt es sich um eine Trendlinie, die den gleitenden Durchschnitt abbildet. Welcher Graph gezeigt wird, kann dem jeweiligen Titel entnommen werden.

Tabelle 2: Daten zur empirischen Auswertung des mathematischen Rundens

Graph	10k	100k	500k	1m
Stichprobengröße	2.000 Paare	10.000 Paare	50.000 Paare	100.000 Paare
kein Pfad	69 Paare	126 Paare	232 Paare	563 Paare
durchschn. Anzahl Kanten pro Pfad	113,36	534,14	908,53	1.323,89
durchschn. max. mögl. Fehler	56,68 dm	267,07 dm	454,26 dm	661,94 dm
mittlere Abweichung	-0,08 dm	-0,02 dm	-5,60 dm	-3,42 dm
korr. Stichprobenvarianz	10,70 dm	54,89 dm	76,88 dm	109,34 dm

In Tabelle 2 werden Daten von der Auswertung des mathematischen Rundens auf den vier Graphen dargestellt. Mit Paare werden Paare aus Start- und Zielknoten bezeichnet. Mit steigender Anzahl an Knoten im Graph steigt auch die durchschnittliche Anzahl an Kanten pro Pfad. Dies wiederum erhöht den durchschnittlichen maximal möglichen Fehler. Je größer dieser ist, desto größer auch die tatsächlichen Fehler. Die mittlere Abweichung bezeichnet die durchschnittliche Abweichung zwischen gerundeten Höhenmetern und exakten Höhenmetern auf einer Stichprobe.

Diese Daten bilden nun die Grundlage, um die anderen Algorithmen auszuwerten.

6.2 Zufallsrunden

Das Zufallsrunden wird auf den unterschiedlichen Graphen angewandt, anschließend wird der arithmetische Mittelwert der Abweichungen auf der Stichprobe und die korrigierte Stichprobenvarianz errechnet. Diese Auswertung wird mit der Auswertung des mathematischen Rundens verglichen. Gemeinsam bilden sie die Grundlage für die Auswertung der eigenen Algorithmen.

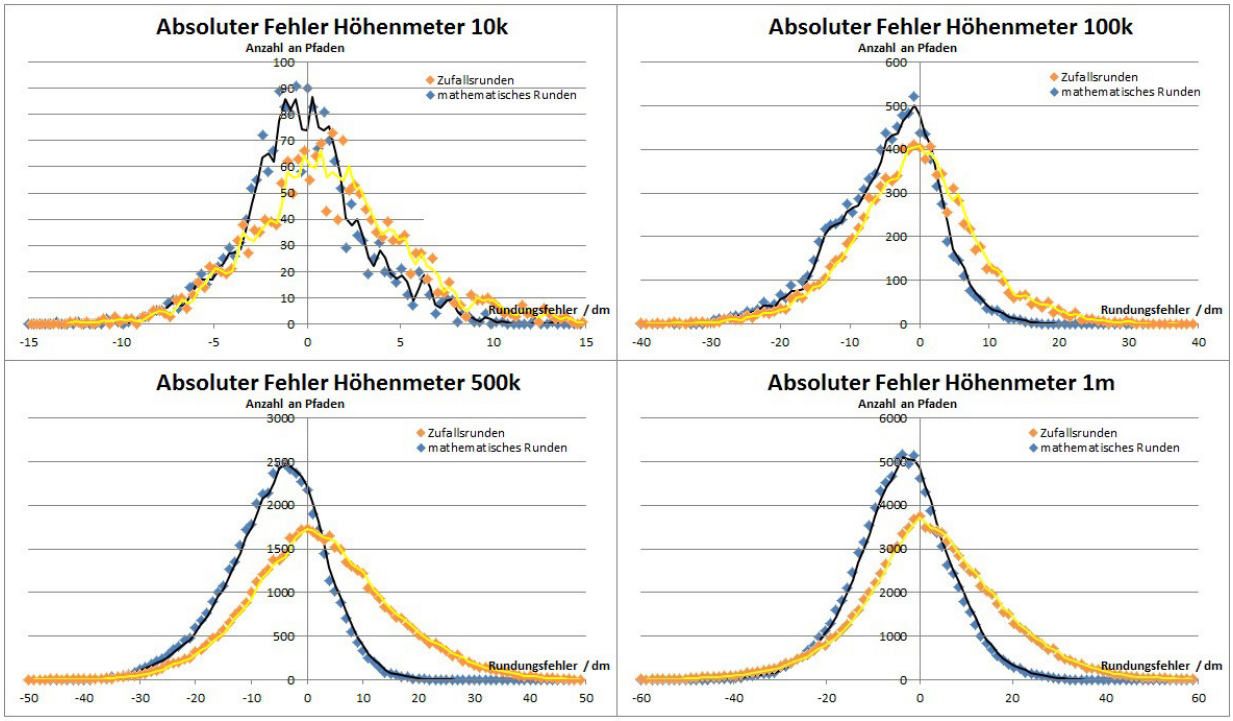

Abbildung 21: Verteilung des absoluten Fehlers der Höhenmeter

In den Graphiken in Abbildung 21 wird die Anzahl an Pfaden mit einem bestimmten gesamten Rundungsfehler der Höhenmeter aufgetragen. Die x-Achse bildet dabei den gesamten Rundungsfehler in Dezimetern ab, die y-Achse die Anzahl der Pfade. Die vier Graphiken bilden die Verteilung auf den vier Graphen ab. Welcher Graph gezeigt wird, kann dem jeweiligen Titel entnommen werden. In Blau mit schwarzer Trendlinie ist die Verteilung beim mathematischen Runden aufgezeichnet. In Orange mit gelber Trendlinie ist die Verteilung beim Zufallsrunden aufgezeichnet. Es ist zu sehen, dass das Zufallsrunden wie erwartet schlechter abschneidet als das mathematische Runden.

Tabelle 3: Daten zur empirischen Auswertung des Zufallsrundens

Graph	10k	100k	500k	1m
Stichprobengröße	2.000 Paare	10.000 Paare	50.000 Paare	100.000 Paare
kein Pfad	69 Paare	126 Paare	232 Paare	563 Paare
durchschn. Anzahl Kanten pro Pfad	113,36	534,14	908,53	1.323,89
durchschn. max. mögl. Fehler	113,36 dm	534,14 dm	908,53 dm	1.323,89 dm
mittlere Abweichung	1,12 dm	-0,66 dm	2,66 dm	2,78 dm
korr. Stichprobenvarianz	18,81 dm	83,93 dm	168,60 dm	244,41 dm

In Tabelle 3 werden Daten von der Auswertung des Zufallsrundens auf den vier Graphen dargestellt. Mit Paare werden Paare aus Start- und Zielknoten bezeichnet. Mit steigender Anzahl an Knoten im Graph steigt auch die durchschnittliche Anzahl an Kanten pro Pfad. Dies wiederum erhöht den durchschnittlichen maximal möglichen Fehler. Je größer dieser ist, desto größer auch die tatsächlichen Fehler. Die mittlere Abweichung bezeichnet die durchschnittliche Abweichung zwischen gerundeten Höhenmetern und exakten Höhenmetern auf einer Stichprobe.

Tabelle 4: Vergleich korrigierte Stichprobenvarianz

Graph	10k	100k	500k	1m
mathematisches Runden	10,70 dm	54,89 dm	76,88 dm	109,34 dm
Zufallsrunden	18,81 dm	83,93 dm	168,60 dm	244,41 dm
Verhältnis Zufalls-. zu math. Runden*	1,78	1,53	2,19	2,24

* Verhältnis zwischen Zufallsrunden und mathematischem Runden

Tabelle 4 ist der Vergleich zwischen der Stichprobenvarianz des mathematischen Rundens und des Zufallsrundens zu entnehmen. Dabei fällt auf, dass in zwei Stichproben die korrigierte Stichprobenvarianz des Zufallsrundens mehr als doppelt so groß ist wie beim mathematischen Runden. Dies kann überraschend erscheinen, da der maximal mögliche Fehler des Zufallsrundens lediglich doppelt so groß ist wie der des mathematischen Rundens. Dies ist dennoch möglich, da die korrigierte Stichprobenvarianz nicht der maximal mögliche Fehler ist. Außerdem richtet sich die korrigierte Stichprobenvarianz nach den Abweichungen der Zufallspfade der Stichprobe. Die Abweichungen treten mit gewissen Wahrscheinlichkeiten auf. Befinden sich zufälligerweise in der Stichprobe des Zufallsrundens mehr Pfade mit großer Abweichung, ist auch die korrigierte Stichprobenvarianz größer.

6.3 Alternierendes Runden

In diesem Abschnitt werden die einzelnen Varianten des alternierenden Rundens ausgewertet. Für die Zusätze gilt dabei, dass sie einzeln verwendet werden. Das heißt, für den Unterabschnitt 6.3.2 Zusatz: Superkanten wird der Grundalgorithmus mit Superkanten erweitert. Für den Unterabschnitt 6.3.3 Zusatz: Gewichtung wird der Grundalgorithmus mit Gewichtung erweitert.

6.3.1 Grundalgorithmus

Das alternierende Runden wird auf den unterschiedlichen Graphen angewandt, anschließend wird der arithmetische Mittelwert der Abweichungen auf der Stichprobe und die korrigierte Stichprobenvarianz errechnet. Diese Auswertung wird mit den Auswertungen des mathematischen und des Zufallsrundens verglichen.

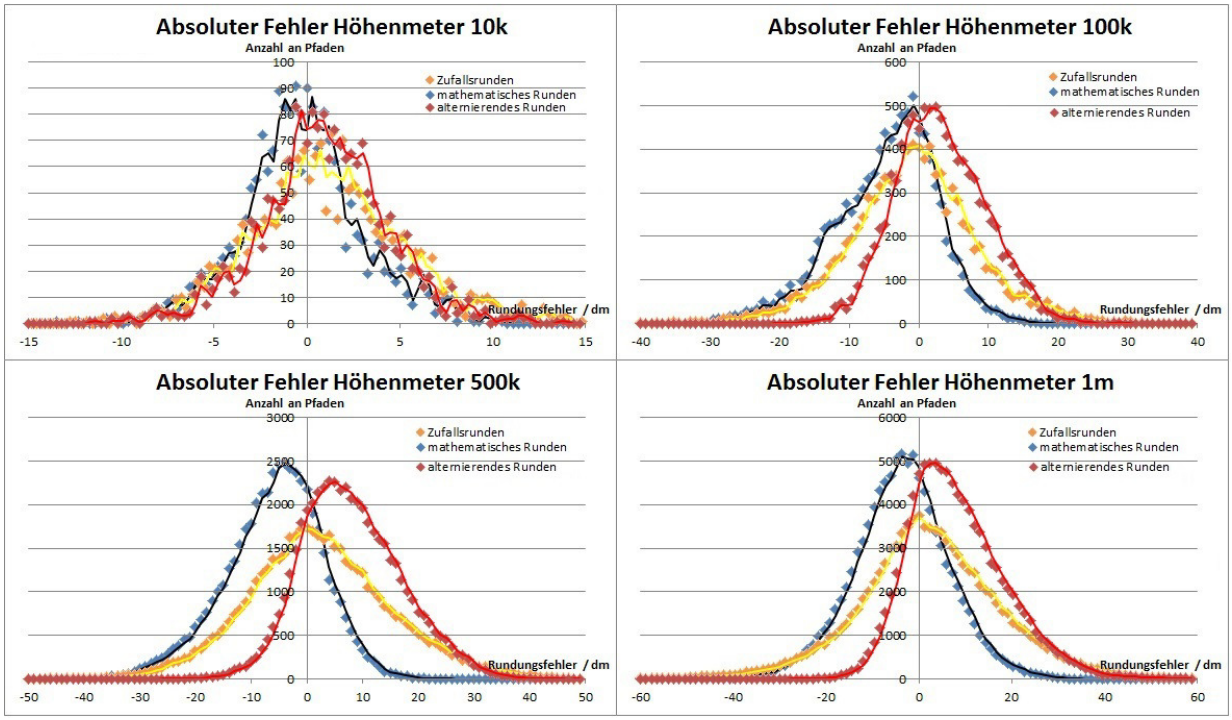

Abbildung 22: Verteilung des absoluten Fehlers der Höhenmeter

In den Graphiken in Abbildung 22 wird die Anzahl an Pfaden mit einem bestimmten gesamten Rundungsfehler der Höhenmeter aufgetragen. Die x-Achse bildet dabei den gesamten Rundungsfehler in Dezimetern ab, die y-Achse die Anzahl der Pfade. Die vier Graphiken bilden die Verteilung auf den vier Graphen ab. Welcher Graph gezeigt wird, kann dem jeweiligen Titel entnommen werden. In Blau mit schwarzer Trendlinie ist die Verteilung beim mathematischen Runden aufgezeichnet. In Orange mit gelber Trendlinie ist die Verteilung beim Zufallsrunden aufgezeichnet. In Rot mit roter Trendlinie ist die Verteilung beim alternierenden

Runden aufgezeichnet. Es ist zu sehen, dass das alternierende Runden ähnliche Ergebnisse liefert, wie das mathematische Runden. Im Vergleich zum mathematischen Runden ist allerdings der Mittelwert der Abweichung der Stichprobe verschoben. Es ist allerdings besser als das Zufallsrunden.

Tabelle 5: Daten zur empirischen Auswertung des Zufallsrundens

Graph	10k	100k	500k	1m
Stichprobengröße	2.000 Paare	10.000 Paare	50.000 Paare	100.000 Paare
kein Pfad	69 Paare	126 Paare	232 Paare	563 Paare
durchschn. Anzahl Kanten pro Pfad	113,36	534,14	908,53	1.323,89
durchschn. max. mögl. Fehler	57,08 dm	267,97 dm	454,67 dm	662,84 dm
mittlere Abweichung	1,03 dm	3,21 dm	8,36 dm	8,64 dm
korr. Stichprobenvarianz	11,90 dm	42,43 dm	81,97 dm	112,26 dm

In Tabelle 5 werden Daten von der Auswertung des alternierenden Rundens auf den vier Graphen dargestellt. Mit Paare werden Paare aus Start- und Zielknoten bezeichnet. Mit steigender Anzahl an Knoten im Graph steigt auch die durchschnittliche Anzahl an Kanten pro Pfad. Dies wiederum erhöht den durchschnittlichen maximal möglichen Fehler. Je größer dieser ist, desto größer auch die tatsächlichen Fehler. Die mittlere Abweichung bezeichnet die durchschnittliche Abweichung zwischen gerundeten Höhenmetern und exakten Höhenmetern auf einer Stichprobe.

Tabelle 6: Vergleich korrigierte Stichprobenvarianz

Graph	10k	100k	500k	1m
mathematisches Runden	10,70 dm	54,89 dm	76,88 dm	109,34 dm
Zufallsrunden	18,81 dm	83,93 dm	168,60 dm	244,41 dm
alternierendes Runden	11,90 dm	42,43 dm	81,97 dm	112,26 dm
Verhältnis altern. zu math. Runden*	1,11	0,77	1,14	1,03
Verhältnis altern. zu Zufallsrunden**	0,63	0,51	0,49	0,46

* Verhältnis zwischen alternierendem und mathematischem Runden

** Verhältnis zwischen alternierendem Runden und Zufallsrunden

Aus Tabelle 6 ist ersichtlich, dass das alternierende Runden in drei von vier Auswertungen etwa doppelt so genau ist wie das Zufallsrunden. Im Vergleich zum mathematischen Runden zeigt sich allerdings, dass es in den meisten Fällen etwa gleich weit streut. Es gibt hier also noch keine nennenswerte Verbesserung zu verzeichnen. Daher wird der Algorithmus auch durch Zusätze erweitert.

6.3.2 Zusatz: Superkanten

Das alternierende Runden mit dem Zusatz der Superkanten wird auf den unterschiedlichen Graphen angewandt, anschließend wird der arithmetische Mittelwert der Abweichungen auf der Stichprobe und die korrigierte Stichprobenvarianz errechnet. Diese Auswertung wird mit den Auswertungen des mathematischen und des Zufallsrundens verglichen.

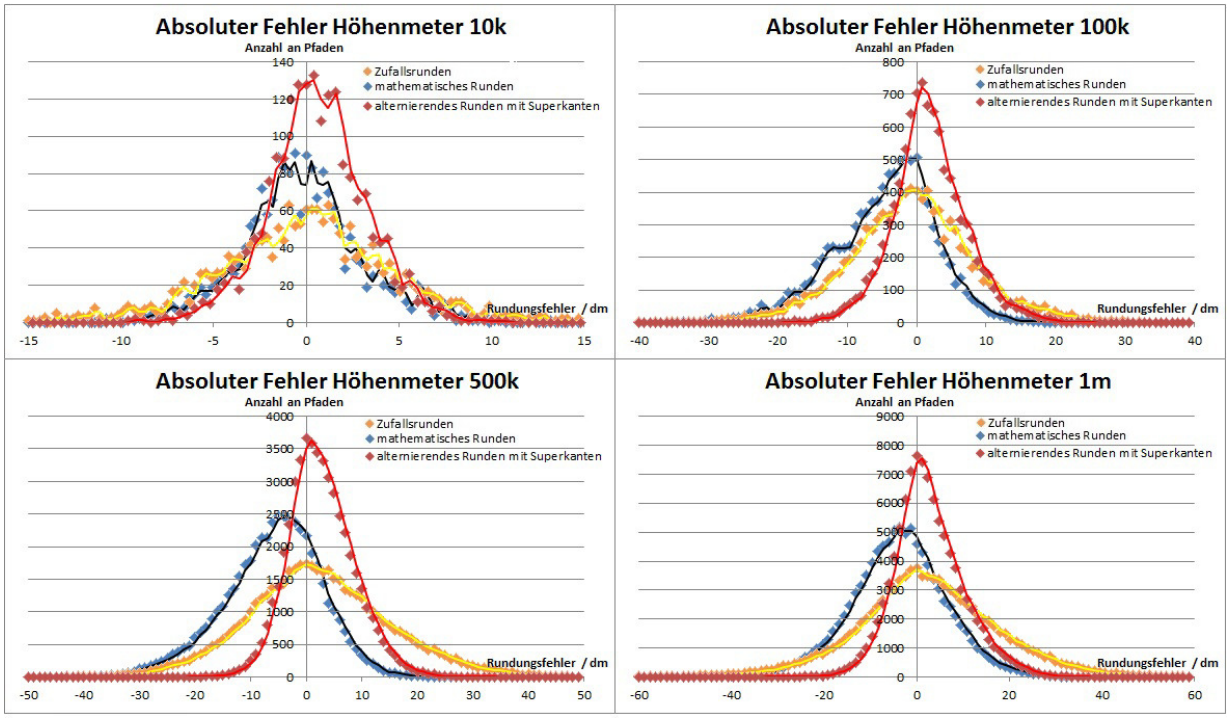

Abbildung 23: Verteilung des absoluten Fehlers der Höhenmeter

In den Graphiken in Abbildung 23 wird die Anzahl an Pfaden mit einem bestimmten gesamten Rundungsfehler der Höhenmeter aufgetragen. Die x-Achse bildet dabei den gesamten Rundungsfehler in Dezimetern ab, die y-Achse die Anzahl der Pfade. Im Vergleich zu den Abbildungen 20, 21 und 22 ist die y-Achse nun anders skaliert. Die vier Graphiken bilden die Verteilung auf den vier Graphen ab. Welcher Graph gezeigt wird, kann dem jeweiligen Titel entnommen werden. In Blau mit schwarzer Trendlinie ist die Verteilung beim mathematischen Runden aufgezeichnet. In Orange mit gelber Trendlinie ist die Verteilung beim Zufallsrunden aufgezeichnet. In Rot mit roter Trendlinie ist die Verteilung beim alternierenden Runden mit Superkanten aufgezeichnet. Es ist zu sehen, dass das alternierende Runden mit Superkanten deutlich bessere Ergebnisse erzielt.

Tabelle 7: Daten zur empirischen Auswertung des Zufallsrundens

Graph	10k	100k	500k	1m
Stichprobengröße	2.000 Paare	10.000 Paare	50.000 Paare	100.000 Paare
kein Pfad	69 Paare	126 Paare	232 Paare	563 Paare
durchschn. Anzahl Kanten pro Pfad	113,36	534,14	908,53	1.323,89
durchschn. max. mögl. Fehler	57,08 dm	267,97 dm	454,67 dm	662,84 dm
mittlere Abweichung	0,69 dm	2,06 dm	3,08 dm	3,08 dm
korr. Stichprobenvarianz	7,30 dm	28,53 dm	35,38 dm	57,96 dm

In Tabelle 7 werden Daten von der Auswertung des alternierenden Rundens auf den vier Graphen dargestellt. Mit Paare werden Paare aus Start- und Zielknoten bezeichnet. Mit steigender Anzahl an Knoten im Graph steigt auch die durchschnittliche Anzahl an Kanten pro Pfad. Dies wiederum erhöht den durchschnittlichen maximal möglichen Fehler. Je größer dieser ist, desto größer auch die tatsächlichen Fehler. Es lässt sich nicht sagen, wie viele Kanten eines Pfades bei dieser Methode auf Superkanten fallen. Die mittlere Abweichung bezeichnet die durchschnittliche Abweichung zwischen gerundeten Höhenmetern und exakten Höhenmetern auf einer Stichprobe.

Tabelle 8: Vergleich korrigierte Stichprobenvarianz

Graph	10k	100k	500k	1m
mathematisches Runden	10,70 dm	54,89 dm	76,88 dm	109,34 dm
Zufallsrunden	18,81 dm	83,93 dm	168,60 dm	244,41 dm
alternierendes Runden + Superkanten	7,30 dm	28,53 dm	35,38 dm	57,96 dm
Verhältnis altern. zu math. Runden*	0,68	0,52	0,46	0,53
Verhältnis altern. zu Zufallsrunden**	0,39	0,34	0,21	0,24

* Verhältnis zwischen alternierendem Runden mit Superkanten und mathematischem Runden

** Verhältnis zwischen alternierendem Runden mit Superkanten und Zufallsrunden

Aus Tabelle 8 geht hervor, dass das alternierende Runden mit Superkanten sowohl gegenüber dem Zufallsrunden als auch gegenüber dem mathematischen Runden genauer ist. Das Zufallsrunden streut 3- bis 4-mal so weit, wie das alternierende Runden mit Superkanten. Das mathematische Runden streut ungefähr 1,5- bis 2-mal so weit, wie das alternierende Runden mit Superkanten. Diese Rundungsmethode ist also besser als das mathematische Runden und das auf allen Graphen

6.3.3 Zusatz: Gewichtung

Für das alternierende Runden mit dem Zusatz der Gewichtung werden zunächst unterschiedliche Gewichtungen und Grenzwerte ausprobiert. Dies erfolgt zum Zweck der schnelleren Auswertung auf dem 10k Graphen. Eine der Kombinationen aus Gewichtungen und Grenzwerten wird ausgewählt. Mit dieser Kombination wird der Algorithmus auf den unterschiedlichen Graphen angewandt, anschließend wird der arithmetische Mittelwert der Abweichungen auf der Stichprobe und die korrigierte Stichprobenvarianz errechnet. Diese Auswertung wird mit den Auswertungen des mathematischen und des Zufallsrundens verglichen.

Tabelle 9: Kombinationen aus Gewichtungen und Grenzwerten

Vorgängergewichtung	1	1	1	0,5	0,5	0,5
Eingangsgewichtung	1	1	1	1	1	1
Eigengewichtung	1	1	1	1	2	2
Grenzschwelle	1	2	0,5	1	1	2
korrigierte Stichprobenvarianz	9,10 dm	9,58 dm	8,98 dm	10,67 dm	10,67 dm	10,67 dm

In Tabelle 9 sind verschiedene Kombinationen für Gewichtungen und Grenzwerte angegeben. Außerdem die mit der jeweiligen Kombination erzielte korrigierte Stichprobenvarianz. Für alle Auswertungen der Kombinationen wurden die gleichen Pfade verwendet. Für die erste Auswertung wurden diese zufällig ausgewählt. Die Zufallspfade werden mit Start- und Zielknoten abgespeichert. Dadurch sind die Zufallspfade rekonstruierbar und für die weiteren Auswertungen verwendbar. Es fällt auf, dass die korrigierte Stichprobenvarianz für gleiche Gewichtung und geringste gewählte Schwelle am geringsten ist. Das heißt, wenn das alternierende Muster am einfachsten zu durchbrechen ist, ist die Streuung am geringsten. Es wird also für die weitere Auswertung die Kombination mit gleicher Gewichtung 1 und 0,5 als Schwellwert verwendet.

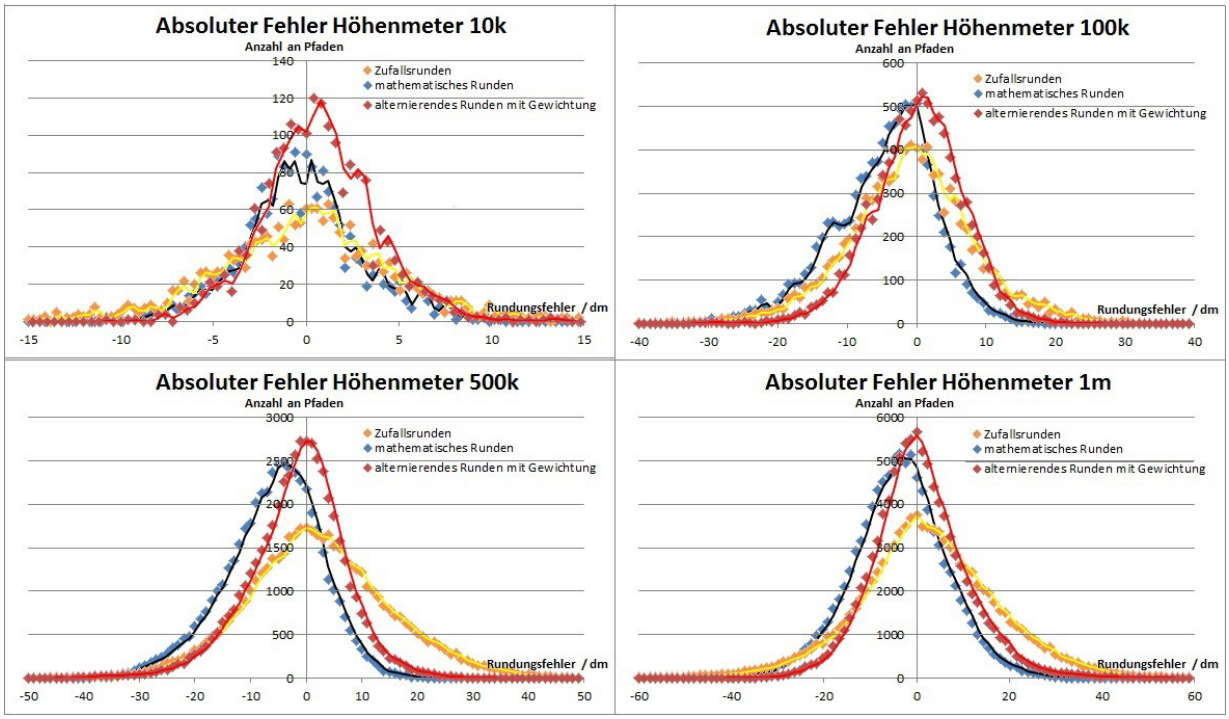

Abbildung 24: Verteilung des absoluten Fehlers der Höhenmeter

In den Graphiken in Abbildung 24 wird die Anzahl an Pfaden mit einem bestimmten gesamten Rundungsfehler der Höhenmeter aufgetragen. Die x-Achse bildet dabei den gesamten Rundungsfehler in Dezimetern ab, die y-Achse die Anzahl der Pfade. Im Vergleich zu den Abbildungen 20, 21, 22 und 23 ist die y-Achse anders skaliert. Die vier Graphiken bilden die Verteilung auf den vier Graphen ab. Welcher Graph gezeigt wird, kann dem jeweiligen Titel entnommen werden. In Blau mit schwarzer Trendlinie ist die Verteilung beim mathematischen Runden aufgezeichnet. In Orange mit gelber Trendlinie ist die Verteilung beim Zufallsrunden aufgezeichnet. In Rot mit roter Trendlinie ist die Verteilung beim alternierenden Runden mit Gewichtung aufgezeichnet. Es ist zu sehen, dass das alternierende Runden mit Gewichtung ähnliche bis etwas bessere Ergebnisse erzielt als das mathematische Runden.

Tabelle 10: Daten zur empirischen Auswertung des Zufallsrundens

Graph	10k	100k	500k	1m
Stichprobengröße	2.000 Paare	10.000 Paare	50.000 Paare	100.000 Paare
kein Pfad	69 Paare	126 Paare	232 Paare	563 Paare
durchschn. Anzahl Kanten pro Pfad	113,36	534,14	908,53	1.323,89
mittlere Abweichung	0,78 dm	0,31 dm	-1,43 dm	1,10 dm
korr. Stichprobenvarianz	9,60 dm	43,26 dm	78,08 dm	105,33 dm

In Tabelle 10 werden Daten von der Auswertung des alternierenden Rundens auf den vier Graphen dargestellt. Mit Paare werden Paare aus Start- und Zielknoten bezeichnet. Mit steigender Anzahl an Knoten im Graph, steigt auch die durchschnittliche Anzahl an Kanten pro Pfad. Die mittlere Abweichung bezeichnet die durchschnittliche Abweichung zwischen gerundeten Höhenmetern und exakten Höhenmetern auf einer Stichprobe.

Tabelle 11: Vergleich korrigierte Stichprobenvarianz

Graph	10k	100k	500k	1m
mathematisches Runden	10,70 dm	54,89 dm	76,88 dm	109,34 dm
Zufallsrunden	18,81 dm	83,93 dm	168,60 dm	244,41 dm
alternierendes Runden + Gewichtung	9,60 dm	43,26 dm	78,08 dm	105,33 dm
Verhältnis altern. zu math. Runden*	0,90	0,79	1,02	0,96
Verhältnis altern. zu Zufallsrunden**	0,51	0,52	0,46	0,43

* Verhältnis zwischen alternierendem Runden mit Gewichtung und mathematischem Runden

** Verhältnis zwischen alternierendem Runden mit Gewichtung und Zufallsrunden

Aus Tabelle 11 geht hervor, dass das alternierende Runden mit Gewichtung etwa doppelt so genau ist wie das Zufallsrunden. Gegenüber dem mathematischen Runden ist das alternierende Runden mit Gewichtung überwiegend etwas besser. Lediglich bei einem Graphen schneidet es etwas schlechter ab.

6.4 Gewichtetes Runden

Das gewichtete Runden wird auf den unterschiedlichen Graphen angewandt, anschließend wird der arithmetische Mittelwert der Abweichungen auf der Stichprobe und die korrigierte Stichprobenvarianz errechnet. Diese Auswertung wird mit den Auswertungen des mathematischen und des Zufallsrundens verglichen.

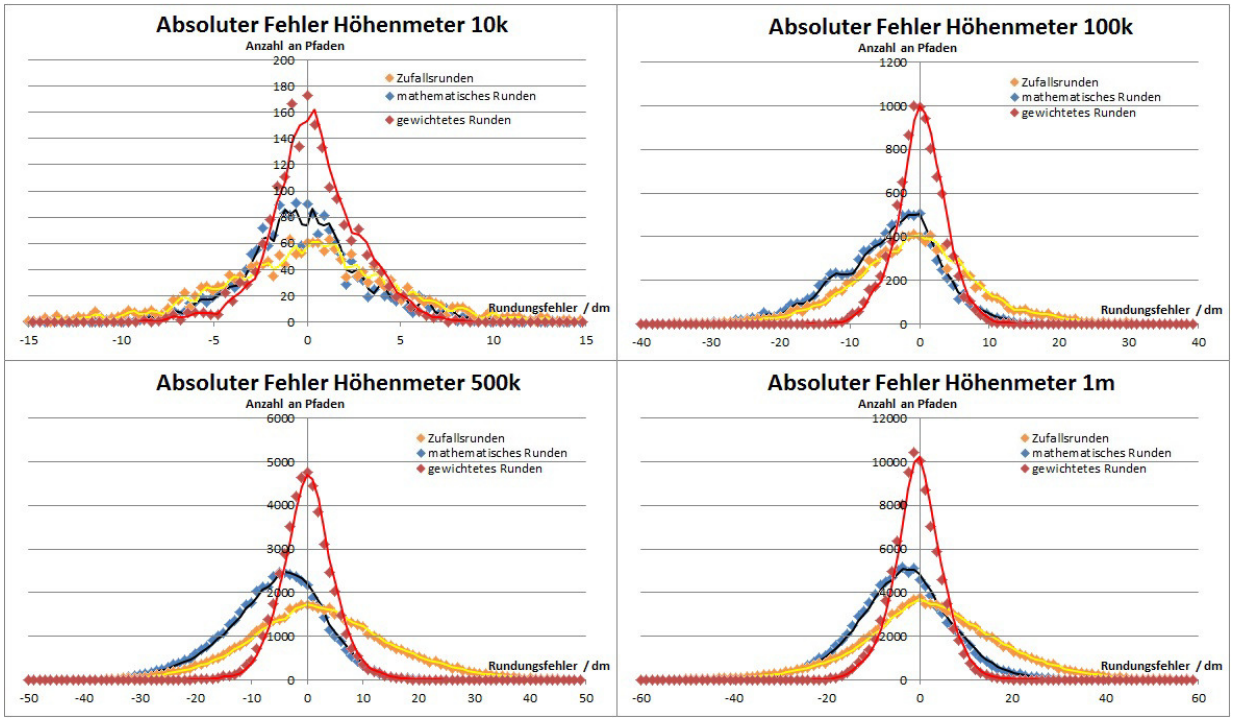

Abbildung 25: Verteilung des absoluten Fehlers der Höhenmeter

In den Graphiken in Abbildung 25 wird die Anzahl an Pfaden mit einem bestimmten gesamten Rundungsfehler der Höhenmeter aufgetragen. Die x-Achse bildet dabei den gesamten Rundungsfehler in Dezimetern ab, die y-Achse die Anzahl der Pfade. Im Vergleich zu den Abbildungen 20 bis 24 ist die Skalierung der y-Achse anders. Die vier Graphiken bilden die Verteilung auf den vier Graphen ab. Welcher Graph gezeigt wird, kann dem jeweiligen Titel entnommen werden. In Blau mit schwarzer Trendlinie ist die Verteilung beim mathematischen Runden aufgezeichnet. In Orange mit gelber Trendlinie ist die Verteilung beim Zufallsrunden aufgezeichnet. In Rot mit roter Trendlinie ist die Verteilung beim gewichteten Runden aufgezeichnet. Es ist zu erkennen, dass das gewichtete Runden deutlich besser ist, als das mathematische Runden oder Zufallsrunden.

Tabelle 12: Daten zur empirischen Auswertung des Zufallsrundens

Graph	10k	100k	500k	1m
Stichprobengröße	2.000 Paare	10.000 Paare	50.000 Paare	100.000 Paare
kein Pfad	69 Paare	126 Paare	232 Paare	563 Paare
durchschn. Anzahl Kanten pro Pfad	113,36	534,14	908,53	1.323,89
mittlere Abweichung	0,41 dm	0,26 dm	0,19 dm	-0,28 dm
korr. Stichprobenvarianz	5,67 dm	12,59 dm	23,23 dm	28,81 dm

In Tabelle 12 werden Daten von der Auswertung des gewichteten Rundens auf den vier Graphen dargestellt. Mit Paare werden Paare aus Start- und Zielknoten bezeichnet. Die mittlere Abweichung bezeichnet die durchschnittliche Abweichung zwischen gerundeten Höhenmetern und exakten Höhenmetern auf einer Stichprobe. Die korrigierte Stichprobenvarianz wächst deutlich langsamer an, als bei den anderen Methoden.

Tabelle 13: Vergleich mittlere Abweichung

Graph	10k	100k	500k	1m
mathematisches Runden	-0,08 dm	-0,02 dm	-5,60 dm	-3,42 dm
Zufallsrunden	1,12 dm	-0,66 dm	2,66 dm	2,78 dm
alternierendes Runden	1,03 dm	3,21 dm	8,36 dm	8,64 dm
altern. Runden + Superkanten	0,69 dm	2,06 dm	3,08 dm	3,08 dm
altern. Runden + Gewichtung	0,78 dm	0,31 dm	-1,43 dm	1,10 dm
gewichtetes Runden	0,41 dm	0,26 dm	0,19 dm	-0,28 dm

Aus Tabelle 13 ist ersichtlich, dass das gewichtete Runden der einzige Algorithmus ist, für den der Mittelwert der Abweichung auf der Stichprobe auch für die beiden größten Graphen konsistent nah an 0 dm bleibt.

Tabelle 14: Vergleich korrigierte Stichprobenvarianz

Graph	10k	100k	500k	1m
mathematisches Runden	10,70 dm	54,89 dm	76,88 dm	109,34 dm
Zufallsrunden	18,81 dm	83,93 dm	168,60 dm	244,41 dm
gewichtetes Runden	5,67 dm	12,59 dm	23,23 dm	28,81 dm
Verhältnis gew. zu math. Runden*	0,53	0,23	0,30	0,17
Verhältnis gew. zu Zufallsrunden**	0,30	0,15	0,14	0,12

* Verhältnis zwischen gewichtetem und mathematischem Runden

** Verhältnis zwischen gewichtetem Runden und Zufallsrunden

Aus Tabelle 14 ist ersichtlich, dass das gewichtete Runden deutlich besser abschneidet als das mathematische Runden oder Zufallsrunden. Die Verbesserung ist dabei konsistent für die größer werdenden Graphen. Das gewichtete Runden liefert von allen beschriebenen und ausgewerteten Algorithmen die besten Ergebnisse. Für den 1m Graphen benötigt das Ausführen des Algorithmus etwa 2,14 Sekunden. Das bedeutet, es wird in 2,14 Sekunden eine Verbesserung um mehr als 80 % erzielt. Aus diesen Gründen wird der Algorithmus des gewichteten Rundens als Ergebnis dieser Arbeit betrachtet.

7 Erweiterungen

Dieser Abschnitt befasst sich mit möglichen Erweiterungen für den endgültigen Algorithmus.

Das Ergebnis des Algorithmus des gewichteten Rundens ist vom Startknoten abhängig. Hier lässt sich eine einfache Erweiterung ansetzen: Der Algorithmus kann z.B. 1.000-mal ausgeführt werden. Für jede Ausführung merkt man sich den Startknoten und errechnet über Zufallspfade die korrigierte Stichprobenvarianz. Am Ende wird der Startknoten, der zur kleinsten korrigierten Stichprobenvarianz führt, ausgewählt.

Bei der angewandten Pfadkorrektur über die Superkanten kann eine bessere Verteilung für die veränderten Kanten entwickelt werden. In der aktuellen Variante tendieren die veränderten Kanten dazu, am Ende der Superkante zu liegen. Fängt ein Pfad an einer solchen Stelle an, werden seine Höhenmeter verfälscht.

Es kann für die Erkennung des Rundungsfehlers bis zu einer Kante beim gewichteten Runden eine bessere Alternative gefunden werden. Die aktuelle Art und Weise kann zurückliegende Kanten mehrfach beachten. Dies könnte zu einer Verfälschung der Gewichtung führen.

Literaturverzeichnis

[1] Behrens, Jan (2008 - 2009): Algorithmus zum summenerhaltenden Runden unter Berücksichtigung von beliebig ineinandergeschachtelten Teilsummen (Fassung von 2009). Online: http://www.magnetkern.de/tree-round.html (letzter Abruf: 03.01.2015)

[2] Zervos, Michalis (2013): Image dithering in Matlab (Fassung vom 09.03.2013) Online: http://michal.is/projects/image-dithering-in-matlab (letzter Abruf: 05.01.2015)