

Alexander Eskuchen
Vergleich von Verfahren für die
Generierung von Kundenstandorten zur
Simulation von urbanen
Versorgungskonzepten
Masterarbeit

Studiengang	Maschinenbau
Matrikelnummer	148785
Thema ausgegeben am	29.03.2019
Arbeit eingereicht am	13.09.2019
Erstprüfer	Univ.-Prof. Dr.-Ing. Markus Rabe
Zweitprüfer	M. Sc. Astrid Klüter

Inhaltsverzeichnis

Inhaltsverzeichnis	I
1. Einleitung	1
2. Urbane Logistiknetzwerke.....	4
2.1 Aufbau urbaner Distributionsnetzwerke	4
2.2 Logistische Netzwerke der Kurier-, Express- und Paket-Dienstleister.....	7
2.3 Herausforderungen in der Last-Mile-Logistik	13
2.4 Verfahren und Konzepte um auf die Herausforderungen in der Last Mile zu reagieren.....	15
3. Modellierung und Simulation.....	23
3.1 Grundlagen einer Simulation	23
3.2 Vorgehen in einer Simulationsstudie	27
3.3 Datenbeschaffung und -aufbereitung	30
3.4 Analyse von Simulationsergebnissen	34
4. Überblick über Verfahren zur Generierung von Standortdaten	40
4.1 Übersicht über genutzte Verfahren der Standortgenerierung	40
4.2 Standortgenerierung auf Grundlage realer Daten	44
4.3 Erzeugung von Zufallszahlen.....	45
4.4 Erzeugung von Zufallsvariablen auf Basis von Verteilungsfunktionen	51
5. Vorbereitung der Simulationsstudie	56
5.1 Vorstellung des Testszenarios.....	56
5.2 Aufbau des Simulationsmodells	58
5.3 Einführung von Kriterien zur Bewertung der Generierungsverfahren	66
6. Versuchsdurchführung und Auswertung der Generierungsverfahren.....	69
6.1 Simulation mit einer Gleichverteilung.....	69
6.1.1 Gleichverteilung im Testset E-n22-k4	70
6.1.2 Gleichverteilung im Testset CMT14.....	74
6.1.3 Gleichverteilung im Testset C110_1.....	80
6.1.4 Gleichverteilung im Testset Antwerp1	86
6.2 Simulation mit einer Normalverteilung	91
6.2.1 Normalverteilung im Testset E-n22-k4.....	92
6.2.2 Normalverteilung im Testset CMT14	97
6.2.3 Normalverteilung im Testset C110_1	103
6.2.4 Normalverteilung im Testset Antwerp1	109
6.3 Simulation mit einer Dreiecksverteilung	115
6.3.1 Dreiecksverteilung im Testset E-n22-k4.....	116
6.3.2 Dreiecksverteilung im Testset CMT14	121

6.3.3	Dreiecksverteilung im Testset C110_1	126
6.3.4	Dreiecksverteilung im Testset Antwerp1	132
6.4	Simulation unter Berücksichtigung der Bevölkerungsdichte.....	137
6.4.1	Berücksichtigung der Bevölkerungsdichte Gleichverteilung.....	138
6.4.2	Berücksichtigung der Bevölkerungsdichte Normalverteilung	143
6.4.3	Berücksichtigung der Bevölkerungsdichte Dreiecksverteilung	149
7.	Vergleich und Bewertung der simulierten Generierungsverfahren.....	155
7.1	Vergleich der Generierungsverfahren.....	155
7.1.1	Vergleich Testset E-n22-k4.....	155
7.1.2	Vergleich Testset E-n22-k4 unter Berücksichtigung der Bevölkerungsdichte	159
7.1.3	Vergleich Testset CMT14	163
7.1.4	Vergleich Testset C110_1	166
7.1.5	Vergleich Testset Antwerp1	170
7.2	Bewertung der Generierungsverfahren	174
8.	Zusammenfassung und Ausblick.....	185
	Literaturverzeichnis	188
	Abkürzungsverzeichnis	205
	Tabellenverzeichnis	206
	Abbildungsverzeichnis	208
	Formelverzeichnis	211
	Anhang A: Weitere Verfahren Standortgenerierung	212
	Anhang B: Testsets und Simulationsergebnisse	213
	Anhang C: Code	229
	Eidesstattliche Versicherung	275

1. Einleitung

Ein Bericht der Vereinten Nationen sagt voraus, dass im Jahr 2050 68 Prozent der Weltbevölkerung in urbanen Gebieten leben oder dorthin ziehen werden (UN 2019). In einem Bericht, der von der Organisation für wirtschaftliche Zusammenarbeit und Entwicklung (OECD 2015) herausgegeben wurde, wird erwartet, dass dieser Wert im Jahr 2100 bei 85 Prozent liegen wird. Des Weiteren gehen Esser und Kurte (2018) von einem stärker wachsenden nationalen Paketmarkt aus. Passend dazu prognostiziert Göpfer (2019), dass der Online-Handel über das Jahr 2030 hinaus jedes Jahr um 10 bis 15 Prozent wachsen wird.

Durch den bereits eingangs erwähnten Zuzug von Menschen in urbane Gebiete sowie dem Anstieg von Bestellungen im Internet werden Staus und die Luftverschmutzung in diesen Gebieten verstärkt (Bates et al. 2018). Insbesondere die Last Mile, welche den Transport der Waren vom Warenhaus bis zum Endkunden umfasst (Gevaers et al. 2011), gilt als teuerster, ineffizientester und auch als der umweltschädigste Teil der gesamten Supply Chain (Gevaers et al. 2014). Von herausragender Bedeutung ist dabei die Last Mile besonders im Bereich des Business-to-Consumer (Wannenwetsch 2014, S. 649).

Um die Kunden umweltfreundlicher und effektiver beliefern zu können, werden weltweit neue Versorgungskonzepte entwickelt. Diese Versorgungskonzepte beinhalten zum Beispiel das Nutzen von Crowdsourcing (Castillo et al. 2018) sowie den Einsatz von Fahrradkurieren (Maes und Vanelsländer 2012) oder Drohnen (DHL 2018). Eine Möglichkeit diese Versorgungskonzepte auf ihre Effektivität und Umweltfreundlichkeit zu untersuchen, stellen Simulationsmodelle dar.

Damit die Untersuchungen der Versorgungskonzepte mittels Simulationsmodellen aussagekräftige Ergebnisse erzielen können, müssen qualitativ sowie quantitativ hochwertige Eingangsdaten vorliegen (Wenzel et al. 2008, S. 119 f.). Zu diesen Eingangsdaten gehören unter anderem die Standorte der Kunden (Köhler und Haferkamp 2019). Die Verwendung der Kundenstandorte, die durch die Datenschutz-Grundverordnung zu den personenbezogenen Daten gezählt werden, kann für den Anwender mit Schwierigkeiten verbunden sein, denn möglicherweise ist die Verwendung von Kundenstandorten ohne die direkte Einwilligung ebenjener aufgrund neuer Datenschutzgesetze rechtlich schwierig (Verordnung EU 2016). Auch können die

Kundenstandorte aufgrund einer unzureichenden Datenbasis fehlen, weil das betroffene Unternehmen beispielsweise die Adressen seiner Kunden gar nicht kennt (Yilmaz et al. 2017). Liegen die Daten nicht vor, so müssen sie geeignet generiert werden (Gutenschwager et al. 2017, S. 158).

Das Gesamtziel dieser Masterarbeit ist ein Vergleich und die Bewertung von Verfahren für die Generierung von Kundenstandorten, um diese in Simulationsmodellen im Rahmen von urbanen Versorgungskonzepten zu verwenden. Dabei sind fünf wichtige und aufeinander aufbauende Teilziele zu bearbeiten. Um einen Überblick über die Herausforderungen und Schwierigkeiten im Kontext der urbanen Logistiksysteme zu erhalten, findet als erstes Teilziel eine Einführung in die Distributionssysteme statt, die in urbanen Gebieten eingesetzt werden. Da sich die Simulation innerhalb der Produktion und Logistik als wichtige Analysemethode etabliert hat (Rabe et al. 2008, S. 1), werden im zweiten Teilziel die Grundlagen für die, in dieser Arbeit genutzte, Simulationsmethode und deren Auswertung geschaffen. Als Eingangsdaten für die Simulation werden die Standorte von Kunden verwendet, die vor dem Hintergrund des Datenmangels geeignet generiert werden müssen. Deswegen werden im dritten Teilziel Verfahren vorgestellt, die in der Literatur zur Generierung von (Kunden-)Standorten eingesetzt werden. Das vierte Teilziel verknüpft die vorherigen Teilziele miteinander, sodass sich ein experimentierbares Simulationsmodell ergibt. Im fünften und letzten Teilziel dieser Arbeit wird das Simulationsmodell genutzt, um Simulationen unter Verwendung unterschiedlicher Generierungsverfahren durchzuführen, die Ergebnisse zu analysieren und die verschiedenen Verfahren miteinander zu vergleichen.

Zur Erfüllung der Teilziele ist die vorliegende Arbeit in mehrere Kapitel unterteilt. Als Ausgangspunkt und Grundlage für die vorliegende Arbeit werden in Kapitel 2 mit Hilfe einer Literaturrecherche der Aufbau von Distributionsnetzwerken sowie der Netzwerke der Kurier,-Express- und Paketdienstleister (KEP) beschrieben. Des Weiteren werden die Herausforderungen in der Last-Mile-Logistik vorgestellt sowie entsprechende Verfahren und Konzepte eingeführt, um auf diese Herausforderungen zu reagieren. Kapitel 3 dient der Schaffung der methodischen Grundlagen für die Simulation in dieser Arbeit, um die vorliegende Arbeit im wissenschaftlichen Kontext zu verankern. Dazu werden zuerst grundsätzliche Begrifflichkeiten eingeführt und das Vorgehen in Simulationsstudien beschrieben. Darauf aufbauend wird die Datenbeschaffung und -aufbereitung eingeführt, wobei speziell auf das Vorgehen bei Datenmangel eingegangen wird. Auch die Analyse

der Ergebnisse wird separat in einem eigenen Abschnitt behandelt. Zur Verdeutlichung ihrer Funktionsweise werden in Kapitel 4 die Generierungsverfahren, die sich durch die Literaturrecherche ergeben haben, anhand ihrer Themengebiete, in denen sie veröffentlicht wurden, klassifiziert. Zudem wird untersucht, ob die Abbildung realer Standorte durch zufällig generierte Standorte in der Literatur bereits analysiert wurde. Des Weiteren erfolgt eine Betrachtung der Literatur dahingehend, ob bereits verwertbare Quellen im Hinblick auf die Generierung zufälliger Standorte unter Berücksichtigung vorliegender Daten sowie bezüglich der Erzeugung von Zufallszahlen herangezogen werden können. Da die zufällige Generierung von Kundenstandorten als gleichbedeutend mit der Erzeugung von Zufallsvariablen anzusehen ist, wird abschließend die Erzeugung derer unter Einbezug der zuvor eingeführten Zufallszahlen und durch die Verwendung von Verteilungsfunktionen erläutert.

Mit Hilfe der geschaffenen Grundlagen aus den vorherigen Kapiteln wird in Kapitel 5 ein Versuchsszenario aufgezeigt und darauf aufbauend ein Simulationsmodell entwickelt, um vergleichbare Ergebnisse mit praktischer Relevanz zu erhalten. Das entstandene Simulationsmodell wird in Kapitel 6 dazu verwendet, verschiedene Generierungsverfahren zu simulieren und die Simulationsergebnisse zu analysieren. Dies dient insbesondere dazu, die variierenden Stärken und Schwächen der jeweiligen Generierungsverfahren im Hinblick auf unterschiedliche geografische Gegebenheiten herauszustellen. Außerdem werden die Generierungsverfahren in Kapitel 7 auf Basis der Simulationsergebnisse miteinander verglichen und abschließend bewertet, um deren Verwendbarkeit als Alternative zu Realdaten beurteilen zu können. Abschließend werden die inhaltlichen Ergebnisse der vorliegenden Arbeit in Kapitel 8 zusammengefasst und darüber hinaus weitere Aspekte der Arbeit, die in Zukunft untersucht werden könnten, in einem Ausblick vorgestellt.

2. Urbane Logistiknetzwerke

Dieses Kapitel dient der Einführung von Grundlagen im Hinblick auf Distributionssysteme in städtischen Gebieten. Dazu wird eingangs der generelle Aufbau urbaner Distributionsnetzwerke erläutert. Danach werden die logistischen Netzwerke der KEP-Dienstleister vorgestellt, die in dem Bereich der Business-to-Consumer (B2C) Belieferung eine herausgestellte Position einnehmen. Anschließend erfolgt eine Einführung in die Herausforderungen der Last-Mile-Logistik, die durch die weltweit fortschreitende Urbanisierung entstehen. Das Kapitel schließt mit der Vorstellung von unterschiedlichen Konzepten und Verfahren ab, um die erläuterten Herausforderungen zu lösen.

2.1 Aufbau urbaner Distributionsnetzwerke

Zur Durchführung logistischer Prozesse wird ein logistisches System benötigt (Fleischmann 2008a). Logistische Systeme können nach Pfohl (2018, S. 16 f.) unter anderem in die Bereiche Beschaffungs- und Distributionslogistik eingeteilt werden. Die weiteren vom Autor genannten Bereiche, wie die Produktions-, die Entsorgungs- und die Ersatzteillogistik, werden hier nicht weiter betrachtet, da sie für die vorliegende Arbeit nicht relevant sind. Bei der Betrachtung eines Handelsunternehmens fügen Zentes und Schramm-Klein (2008) dem Bereich der Entsorgungslogistik die Retouren vom Konsumenten zum Unternehmen hinzu. Auch Retouren werden in dieser Arbeit nicht behandelt.

Aufgabe der Beschaffungslogistik ist die bedarfsgerechte Bereitstellung von beispielsweise Rohstoffen oder Handelswaren für ein Unternehmen (Pfohl 2018, S. 189). Mit Hilfe der Distributionslogistik werden die Kunden unter anderem mit Fertigfabrikaten oder Handelswaren aus den Produktionsstätten oder Auslieferungslagern eines Unternehmens beliefert (Pfohl 2018, S. 221). Dafür erfüllt die Distributionslogistik Aufgaben aus der Auftragsabwicklung, der Lagerhaltung, der Kommissionierung, der Verpackung, der Warenausgangsabwicklung und der inner- bzw. außerbetrieblichen Transportlogistik (Bichler et al. 2017, S. 56). Die Differenzierung in Beschaffungs- und Distributionslogistik wird unter anderem von Gudehus (2010, S. 7) und von Fleischmann (2008b, S. 12) als subjektiv bezeichnet, da sie vom Betrachtungswinkel des

Unternehmens abhängig ist. Für Unternehmen A gehört das logistische System, um Unternehmen B mit Handelswaren zu versorgen, zur Distributionslogistik. Das gleiche logistische System gehört für Unternehmen B hingegen zur Beschaffungslogistik.

Neben der Aufteilung in die verschiedenen Bereiche kann ein logistisches System auch über seine Struktur beschrieben werden. Diese Struktur kann als Netzwerk bezeichnet werden (Fleischmann 2008a; Ballou 2004, S. 41 ff.). Bretzke beschreibt ein logistisches Netzwerk als ein „geordnetes Durchflusssystem, das aus hierarchisch und geografisch angeordneten, im Hinblick auf ein Leistungsziel komplementären Ressourcenknoten und diese Knoten verbindenden Verkehrs- und Informationswegen („Kanten“) besteht(...)“ (Bretzke 2015, S. 102). Weiter führt Bretzke (2015, S. 102) aus, dass durch das logistische Netzwerk Quellen mit Senken verbunden werden. Diese Verbindungen sollen sowohl wirtschaftlich als auch bedarfsgerecht erfolgen. Die Quellen eines Netzwerks sind beispielsweise Produktionsstätten oder Händler. Als Senken werden unter anderem auch Händler oder Endkunden bezeichnet.

Nach Gudehus (2010, S. 17) können Logistiknetzwerke anhand ihrer Stufigkeit unterschieden werden. Stufigkeit bezeichnet, im Kontext der Logistiknetzwerke, die Anzahl der Zwischenstationen, die von den logistischen Objekten auf ihrem Weg zwischen Quelle(n) und Senke(n) passiert werden. Ein einstufiges Logistiknetzwerk zeichnet sich laut Pfohl (2018, S. 5) durch einen direkten Güterfluss aus. Dabei wird der Güterfluss zwischen Quelle und Senke nicht unterbrochen. Somit finden innerhalb eines einstufigen Logistiknetzwerks keine weiteren Lagerprozesse statt. Bei einem zweistufigen Logistiknetzwerk wird der Güterfluss nach Gudehus (2010, S. 18) zwischen Quelle und Senke einmalig durch eine Zwischenstation unterbrochen. Die Zwischenstation kann dazu benutzt werden, um entweder als Sammelstation in der Nähe der Quellen oder als Verteilstation in der Nähe der Senken eingesetzt zu werden. Von einer Verteilstation aus können kleinere Sendungsgrößen an mehrere Kunden mittels einer Liefertour zugestellt werden (Fleischmann et al. 2015). Zweistufige Logistiknetzwerke werden von Pfohl (2018, S. 6) als mehrstufige Logistiksysteme bezeichnet. Die zwei genannten Logistiknetzwerke sind in Abbildung 1 zu sehen.

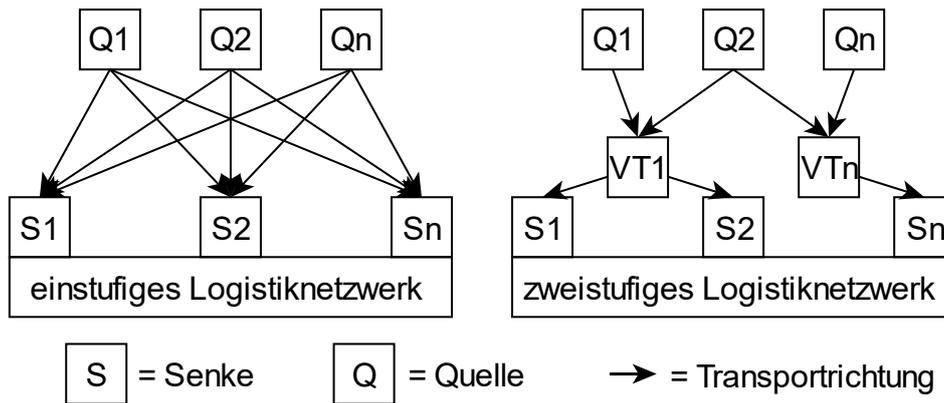


Abbildung 1: Ein- und zweistufige Logistiknetzwerke (nach Gudehus 2010, S. 18 f.)

In einem dreistufigen Netzwerk wird der Güterstrom laut Gudehus (2010, S. 18 f.) insgesamt zwei Mal unterbrochen. Dies kann durch eine Kombination von Sammel- und Verteilstationen geschehen. Es ist aber auch möglich eine Kombination aus Logistikzentren und Verteilstationen zu nutzen. Durch diese genannten Kombinationen kann eine hohe Bündelung der Transporte erfolgen, welche wiederum die Transportkosten senken kann. Wird der Güterfluss mehr als zwei Mal unterbrochen, spricht der Autor von mehrstufigen Netzwerken. Diese Netzwerkstrukturen entstehen bei Transporten über große Entfernungen wie beispielsweise der Luftfracht und werden aus diesem Grund hier nicht weiter behandelt. Die in diesem Absatz behandelten Logistiknetzwerke sind in Abbildung 2 dargestellt.

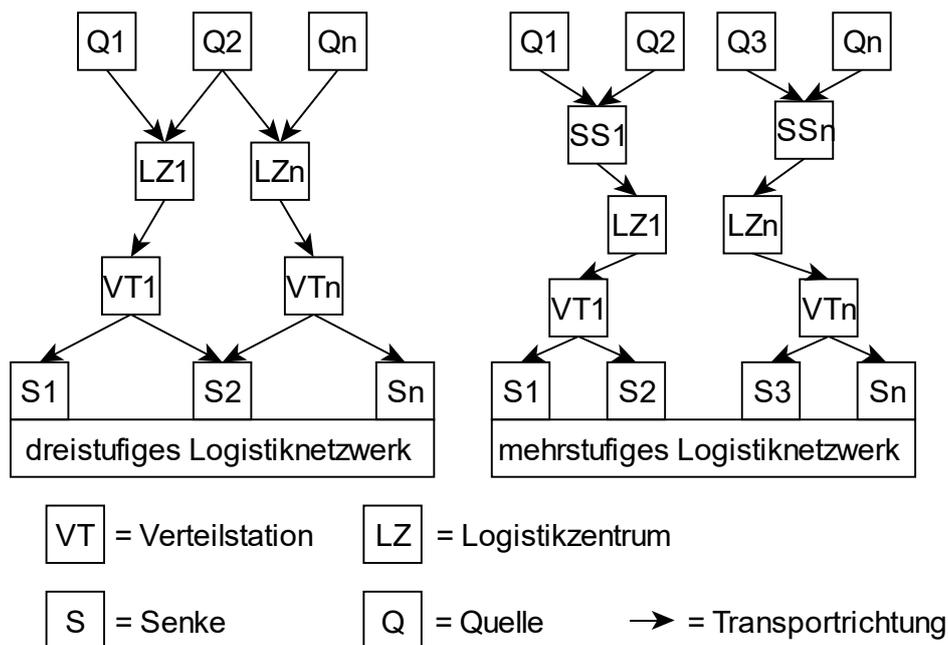


Abbildung 2: Drei- und mehrstufige Logistiknetzwerke (nach Gudehus 2010, S. 19 f.)

Die beschriebenen Logistiknetzwerke können ohne Kenntnisse der Standorte und Lieferfähigkeit der Quellen sowie der Standorte und Serviceerwartungen der Kunden nicht gestaltet werden (Pfohl 2018, S. 103). Gudehus (2010, S. 17) fasst die Parameter, welche die Struktur des Logistiknetzwerks beeinflussen, weiter. Diese sind Folgende:

- Anzahl, Standorte und Funktionen der Quellen und Lieferstellen
- Anzahl, Standorte, Funktionen und Zuordnung der Logistikstationen zwischen den Quellen und Senken
- Anzahl, Standorte und Funktionen der Senken und Empfangsstellen

Wird das Verständnis der logistischen Netzwerke mit dem Ziel der Distributionslogistik, „Güter zur richtigen Zeit, in richtiger Qualität und Quantität am richtigen Ort“ (Bichler et al. 2017, S. 56) bereitzustellen, verknüpft, so ergeben sich zur Erfüllung dieses Ziels Distributionsnetzwerke (Sonntag & Thulesius, 2015). Distributionsnetzwerke des Handels oder der Industrie können als One-To-Many-Netzwerke bezeichnet werden, weil der Güterfluss von einer begrenzten Anzahl an Quellen aus zu einer Vielzahl an Senken geht (Bretzke 2015, S. 192). Die Aufgaben eines Distributionsnetzwerks, hauptsächlich die der Transportlogistik, werden zunehmend an externe Logistikdienstleister outgesourced (Fleischmann 2008b, Zentes und Schramm-Klein 2008, S. 421). Auch Waren, die im Internet gekauft werden, brauchen eine physische Distributionsstruktur, die entweder selbst vorgehalten oder outgesourced wird (Agatz et al. 2013). Aufgrund der durch den E-Commerce notwendigen Distributionsstrukturen werden Sendungen hauptsächlich mit KEP-Dienstleistern versendet, wodurch die beauftragenden Unternehmen aber die Kontrolle über das Distributionsnetzwerk abgeben (Auffermann und Lange 2008; Sonntag und Thulesius 2015). Die eingesetzten KEP-Dienstleister können beispielsweise DHL oder GLS sein (Vahrenkamp und Kotzab 2012, S. 126).

2.2 Logistische Netzwerke der Kurier-, Express- und Paket-Dienstleister

Die KEP-Dienstleister versuchen Güterströme zu bündeln, um eine höhere Auslastung der Transportkapazitäten zu erreichen (Bretzke 2015, S. 362). Vahrenkamp und Kotzab (2012, S. 147 - 155) sowie Bretzke (2015, S. 362 f.) unterscheiden die KEP-Dienste anhand ihrer Marktsegmente sowie der Art der Zustellung. Paketdienste transportieren Pakete bis zu 31,5 Kilogramm und bündeln die, von mehreren Versendern aufgegebenen,

Pakete zu einer Sendung. Aufgrund der Bündelung werden keine Einzelsendungen durchgeführt, sondern die Sendungen erfolgen mengenorientiert. Zur Ermöglichung von kostengünstigen Transporten, nutzen Paketdienste überwiegend Depot- oder Hub-and-Spoke-Netzwerke. Expressdienstleister geben ihren Kunden keine Limitierung bezüglich der Größe oder des Gewichts vor. Komplett- und Teilladungen ab einer Tonne Gewicht werden im Direktverkehr zugestellt, mit der Ausnahme von Stückgut bis 2 Tonnen Gewicht, welches gebündelt und über mehrstufige Logistiknetzwerke versandt wird (Fleischmann 2008b). Kurierdienste hingegen haben sich auf eine individuelle Abholung und Zustellung von Sendungen unter 3 Kilogramm, zum Beispiel von Dokumenten, spezialisiert (Vahrenkamp und Kotzab 2012, S. 147). Laut Cardeneo (2008) erfolgt der regionale Transport im Direktverkehr vom Auftraggeber zum Empfänger, wobei es zu keinen Bündelungseffekten kommt. Im Falle eines national oder international durchgeführten Transports wird ein mehrstufiges Distributionsnetzwerk genutzt. Trotz der Unterscheidung in KEP-Dienstleister ist es durchaus üblich, dass ein im Paketmarkt tätiges Unternehmen zusätzlich auch Express- oder Kurierdienste oder beides anbietet (Vahrenkamp und Kotzab 2012, S. 147). Tripp (2019, S. 256 f.) teilt den Marktsegmenten der KEP-Dienstleister verschiedene Kombinationen aus Auftraggeber und Kunden zu, die Wannewetsch (2014, S. 207) auch als Interaktionsformen bezeichnet. Im Paketmarkt sind die Kombinationen Business-to-Business (B2B), B2C, Consumer-to-Consumer (C2C) sowie Consumer-to-Business (C2B) vertreten. Im Expressmarkt ist laut dem Autor nur B2B anzutreffen, während im Kuriermarkt zusätzlich noch der B2C vorkommt. Vahrenkamp und Kotzab (2012, S. 123) bezeichnen das B2B und das B2C als besonders relevant.

Die angesprochenen Logistikdienstleister können als eigenes logistisches System betrachtet werden (Fleischmann 2008a). Im Gegensatz zu den One-To-Many-Netzwerken der Industrie oder des Handels (vgl. Abschnitt 2.1), verbinden die Logistikdienstleister viele Versender, beispielsweise Unternehmen oder Haushalte, mit vielen Empfängern und werden deshalb als Many-To-Many-Netzwerke bezeichnet (Vahrenkamp und Kotzab 2012, S. 152; Bretzke 2015, S. 364). Diese Distributionsnetzwerke stellen einen wesentlichen Bestandteil innerhalb der E-Commerce-Logistik dar (Sonntag und Thulesius 2015). Aufgrund der verschiedenen Marktsegmente der KEP-Unternehmen haben die eingesetzten Distributionsnetzwerke unterschiedliche Strukturen (Vahrenkamp und Kotzab 2012, S. 152 f.). Da in dieser Arbeit Verfahren zur Generierung von Kundenstandorten in städtischen Gebieten

untersucht werden, folgt eine Erläuterung der typischen Netzwerkstrukturen der als „B2C-affin“ (Sonntag und Thulesius 2015) geltenden Paketdienste. Bei diesen typischen Netzwerkstrukturen handelt es sich, wie bei der Unterscheidung der KEP-Unternehmen am Beginn dieses Abschnitts schon angesprochen, um das Depot- und das Hub-and-Spoke-Netzwerk.

In einem Depotnetzwerk werden nach Vahrenkamp und Kotzab (2012, S. 153 f.) Depots zur Erschließung einzelner Regionen eingesetzt (vgl. Abbildung 3). Durch Sammelverkehre werden die Sendungen für andere Regionen eingesammelt, zum regionalen Depot zur Sortierung sowie Bündelung gebracht und auf Hauptrelationen mit LKWs zu ihrem Empfangsdepot transportiert. Somit sind alle Depots miteinander über Transportrelationen verbunden. Im Empfangsdepot werden die Sendungen in Auslieferungstouren zusammengestellt und durch Verteilverkehre mit Kleintransportern zu den Kunden gebracht. Gudehus (2010, S. 939 f.) bezeichnet diese Art der Netzwerkstruktur als dezentrales Netz. Vorteile dieser Struktur sind die kürzeren Wege zwischen den Depots und die dadurch bedingten niedrigeren Beförderungszeiten. Als Nachteil gilt die geringe Auslastung der großen Beförderungsmittel der Hauptrelationen.

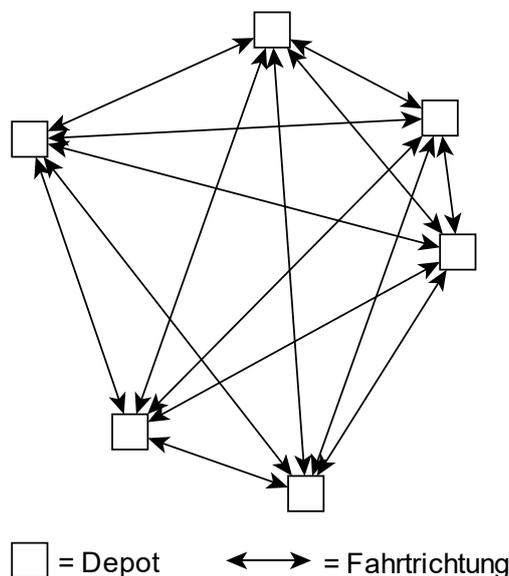


Abbildung 3: Ein beispielhaftes Depotnetzwerk

Das Hub-and-Spoke-Netzwerk zeichnet sich nach Cardeneo (2008) durch ein Zentralhub sowie durch von den Depots aus sternförmig auf das Zentralhub zulaufende Transportrelationen (Spoke) aus (vgl. Abbildung 4). Bei dieser Netzwerkstruktur gibt es

zwischen den einzelnen Depots, im Unterschied zu den Depotnetzwerken, keine direkten Transportrelationen. Alle Sendungen werden im Zentralhub sortiert und für den Transport zu den Depots gebündelt. An den Depots finden wie beim Depotnetzwerk Sammel- und Verteilverkehre statt. Diese Netzwerkstruktur wird von Gudehus (2010, S. 940 f.) als zentrales Netz bezeichnet. Vorteile gegenüber einem dezentralen Netz sieht der Autor in einem höheren Füllungsgrad größerer Transportmittel oder, bei gleicher Transportmittelgröße, in einer höheren Beförderungsfrequenz auf der Hauptrelation. Einen Nachteil stellt der längere Lieferweg von benachbarten Depots dar, da diese auch über das Zentralhub stattfinden.

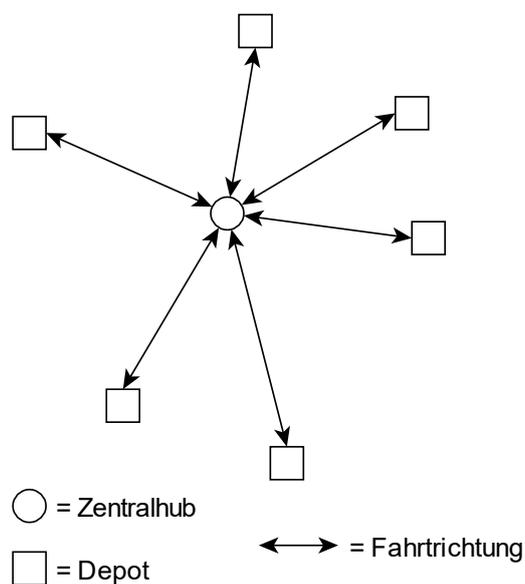


Abbildung 4: Ein beispielhaftes Hub-and-Spoke-Netzwerk

In der Distribution von Paketen wird nach Vahrenkamp und Kotzab (2012, S. 156 f.) sowie Gudehus (2010, 941 f.), aufgrund der unterschiedlichen Anforderungen und ungleichmäßig verteilter Frachtaufkommen, eine Kombination aus Depot- und Hub-and-Spoke-Netzwerken eingesetzt. Hierbei gilt es auch die Ergänzung des Hub-and-Spoke-Netzwerks durch regionale Hubs anzusprechen (vgl. Abbildung 5). Diese Regionalhubs werden in Regionen mit hohem Paketaufkommen, wie dem Ruhrgebiet oder Süddeutschland, eingesetzt. Dabei werden Sendungen mit der gleichen Zielregion vorsortiert und nicht über das Zentralhub befördert, sondern direkt an die Regionalhubs geliefert. Dadurch sollen die Haupttransportrelationen zum Zentralhub entlastet werden.

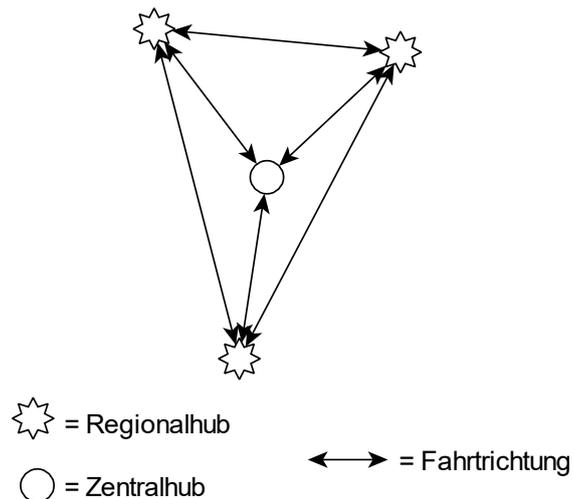


Abbildung 5: Hub-and-Spoke-Netzwerk mit Regionalhubs

Gemeinsam haben die genannten Netzwerke, dass sie aufgrund der mehrmals unterbrochenen Transportrelationen mehrstufige Logistiknetzwerke darstellen (Cardeneo 2008). Auch die in Abschnitt 2.1 angesprochene unterschiedlichen Sichtweisen auf das Logistiksystem kommen zum Tragen. Wird ein KEP-Unternehmen beauftragt, Pakete mit seinem Logistiknetzwerk auszuliefern, handelt es sich, aus der Sicht des beauftragenden Unternehmens, bei den Distributionsnetzwerken ihrer Logistikdienstleister um einstufige Distributionsnetzwerke (Bretzke 2015, S. 193). Aus der Sicht des Logistikdienstleisters ist sein eingesetztes Depot-, Hub-and-Spoke-Netzwerk oder die Kombination beider Netzwerke ein mehrstufiges Distributionsnetzwerk (Cardeneo 2008).

Eine weitere Gemeinsamkeit der im E-Commerce genutzten Distributionsnetzwerke ist, dass die Verteilverkehre als letzte Meile bezeichnet werden (Sonntag und Thulesius 2015). Die letzte Meile ist für die Belieferung von Städten besonders wichtig (Thaller et al. 2017). Im englischsprachigen Raum wird die letzte Meile auch als „Last Mile“ bezeichnet (ten Hompel und Heidenblut 2011, S. 181). Im deutschen Sprachraum wird sowohl der deutsche als auch der englische Begriff genutzt, im weiteren Verlauf der vorliegenden Arbeit wird der Begriff der Last Mile verwendet. Die Last Mile stammt ursprünglich aus der Telekommunikationsbranche und wird dort als Bezeichnung für den Weg zwischen dem Verteilerkasten des Telekommunikationsunternehmens und dem Hausanschluss des Kunden benutzt (Schulte 2017, S. 322). In der Distributionslogistik beschreibt die Last Mile die Lieferung der Ware von den Verteilungszentren oder Distributionszentren zum Kunden und beinhaltet damit die letzte Wegstrecke der Ware zum Kunden (Wegner und Wegner 2017, S. 266; Aized und Srari 2014). Die Auslieferung

auf der Last Mile übernehmen hauptsächlich KEP-Unternehmen (Wannenwetsch 2014, S. 649).

Bei der Verwendung des Begriffs Last Mile gilt es zu beachten, dass mit ihm nicht exakt eine Meile bezeichnet wird, sondern die Wegstrecke, die von den geografischen Gegebenheiten des Distributionsnetzwerks abhängig ist (Cardenas et al. 2017a). Weitere Einflüsse auf die Wegstrecke können die Ausgangs- oder Endpunkte des Distributionsnetzwerks darstellen (Jaller et al. 2018).

Nach Jaller et al. (2018) befinden sich Distributionszentren außerhalb von Städten, da mehr Land zur Verfügung steht und die Mieten günstiger sind. Goodman (2005) bestätigt dies, da der Autor angibt, dass zwischen Distributionszentrum und Endkunde nicht mehr als 50 Meilen liegen. Bogdanski (2017) spricht für die Stadt Berlin von einer Distanz von bis zu 30 Kilometern zwischen Depots und Stadtzentrum. Gemäß einer Studie von Dablanc und Rakotonarivo (2010) sind unter anderem die Distributionszentren der Paketunternehmen in Paris im Zeitraum von 1974 bis 2008 vom Zentrum der Stadt in die Vorstädte verlagert worden. Gleichzeitig beobachten die Autoren aber auch, dass die Paketunternehmen zunehmend versuchen, Distributionszentren wieder in der Innenstadt von Paris zu etablieren, um die Entfernung zu ihren Kunden zu reduzieren. Zu ähnlichen Ergebnissen kommen auch Jaller et al. (2017). Die Autoren untersuchen in ihrer Studie die Verteilung von Distributionszentren in Süd-Kalifornien im Zeitraum von 1998 bis 2014. Dabei stellen sie ebenfalls fest, dass bis zum Jahr 2007 Distributionszentren in die Vorstädte verlagert wurden. Ab dem Jahr 2008 werden die Distributionszentren aber wieder näher an die Städte verlegt. Dies deckt sich mit der Aussage aus einem Whitepaper von Aoun und Vatcheva (2018) für die Deutsche Post DHL Group, dass eine Verkürzung der Last Mile hauptsächlich durch eine Annäherung der Distributionszentren an den Kunden zu erreichen ist.

Aus den Distributionszentren erfolgen in der Last Mile die Lieferungen an die Endkunden. Dabei werden 20 bis 50 Sendungen verteilt und die Fahrzeuge legen dabei nicht mehr als 150 bis 200 Meilen zurück (Goodman 2005). Edwards et al. (2009) bezeichnen als typische Route eine Strecke von 50 Meilen und 120 Sendungen. Tripp (2019, S. 258) spricht im B2C von 80 bis 140 Paketen pro Auslieferungstour, ohne Angabe der Streckenlänge. Eine im Auftrag des Bundesverbands Paket & Express Logistik (BIEK) durchgeführte Studie (Bogdanski 2017) identifiziert für die Stadt Berlin eine Einteilung in drei verschiedene Streckenabschnitte. Auf Strecke A werden bis zu 50

Kilometer zurückgelegt und 160 Sendungen verteilt, Strecke B umfasst 60 bis 80 Kilometer mit 150 Sendungen und Strecke C wird auf 90 bis 130 Kilometern mit 130 Sendungen durchgeführt.

Auch die Struktur einer Stadt kann Einfluss auf die Wegstrecke nehmen (Jaller et al. 2018). Anhand ihrer Struktur klassifizieren Louf und Barthelemy (2010) insgesamt 131 Städte weltweit. Bei ihrer Klassifizierung teilen die Autoren die Städte in vier Gruppen ein. Diese Klassifizierung nutzt Raps (2017) um in fünf unterschiedlichen Städten Realdistanz mittels Distanzmetriken abzuschätzen. Dabei kommt der Autor unter anderem zu dem Ergebnis, dass die unterschiedlichen Stadtformen keinen erheblichen Einfluss auf die Wegstrecken haben und die gewichtete euklidische Distanzfunktion mit einem Faktor von 1,234 die Distanzen für alle betrachteten Städte gut approximiert. Es gilt aber zu beachten, dass bei diesem Vergleich die durchschnittliche Streckenlänge zwischen zwei Punkten bei maximal 10 Kilometern lag. Rabe et al. (2017) geben als optimalen Faktor für die Approximation von Distanzen mit der gewichteten euklidischen Distanzfunktion innerhalb der griechischen Stadt Athen den Faktor 1,25 an.

2.3 Herausforderungen in der Last-Mile-Logistik

Im Jahr 2018 lebten einem Bericht der Vereinten Nation (UN 2019) zufolge 55 Prozent der Weltbevölkerung in städtischen Räumen. Es wird vorausgesagt, dass im Jahr 2050 68 Prozent der Weltbevölkerung in diesen Gebieten wohnen wird. Dieser Zuwachs wird hauptsächlich in Asien und Afrika stattfinden, da im Jahr 2018 schon 82 Prozent der amerikanischen und 74 Prozent der europäischen Bevölkerung in städtischen Räumen leben. Durch diesen Zuwachs steigen auch die Individualverkehre innerhalb der Städte weiter an, wodurch es zu einer Überlastung des Stadtverkehrsnetz kommen kann (Lehmacher 2015, S. 7f.).

Eine zusätzliche Belastung des Stadtverkehrsnetz tritt durch die ansteigende Sendungsmenge im KEP-Markt auf (Thaller 2017), welche durch die Entwicklung des E-Commerce entsteht (Lehmacher 2015, S. 10). Dabei sorgen nicht nur die Privatkunden für eine ansteigende Nachfrage, sondern auch der Einzelhandel nutzt verstärkt Paketdienste, um einer kapital- und platzintensiven Lagerhaltung vorzubeugen (Gerdes und Heinemann 2019). Esser und Kurte (2018) geben in einer Studie für den BIEK an, dass sich das Paketvolumen auf dem deutschen Markt in den Jahren 2000 bis 2017 von

1,69 Milliarden Sendungen auf 3,35 Milliarden beinahe verdoppelt hat. Dabei liegt der Anteil der B2C Sendungen bei 61 Prozent, der des B2B bei 33 Prozent und der im C2C bei 6 Prozent. Dies ist gegenüber dem Jahr 2009 eine Verschiebung in Richtung des B2C, denn in jenem Jahr lag der Marktanteil des B2C bei 45 Prozent. Deswegen wird der B2C als Markttreiber bezeichnet und die Autoren geben für den nationalen Paketmarkt eine Wachstumsprognose um 5 bis 6 Prozent im Jahr 2018 an. Der Geschäftsbericht der Deutsche Post DHL Group (DPDHL 2018, S. 51) für das Jahr 2018 unterstützt diese Prognose, denn das Paketvolumen von DHL für nationale Sendungen stieg im betroffenen Zeitraum von 1,376 auf 1,479 Milliarden Pakete. Dies entspricht einem Wachstum von 7 Prozent.

Der steigende Individualverkehr und der Anstieg des Lieferverkehrs aufgrund des erhöhten Paketvolumens, führt nach Gerdes und Heinemann (2019) unter anderem zu einer Konkurrenzsituation beider Verkehre um knappe Verkehrsflächen. Diese Verkehrsdichte bezeichnet Rall (2019) als Engpass der Distributionslogistik. Zusätzlich führt die Verkehrsbelastung zu Staus, Lärm und Luftverschmutzungen. Um die Belastung der Bevölkerung durch CO₂-Emissionen einzudämmen, hat sich die EU (2011, S. 9) zum Ziel gesetzt, dass städtische Zentren bis 2030 durch eine CO₂ freie Logistik beliefert werden. Auch äußern Kunden vermehrt den Wunsch, dass Logistikunternehmen ihre CO₂ Bilanz verbessern, ohne jedoch als Kunden durch höhere Gebühren oder Wartezeiten für diese Verbesserungen zu bezahlen (Gevaers et al. 2011). Des Weiteren wünschen sich Kunden schnellere Lieferungen (Wegner und Wegner 2017, S. 267). Betrug die Lieferzeiten von Paketen vor einigen Jahren noch mehrere Tage, so betragen diese heute in 99 Prozent der Fälle nur noch 24 Stunden (Löhr und Schäfer 2018). Um sich von Wettbewerbern zu differenzieren und abzuheben, versuchen Logistikdienstleister die Lieferzeiten weiter zu verkürzen und dem Kunden Lieferungen am gleichen Tag (Same Day Delivery) anzubieten (Rall 2019). Zusätzlich hat sich die Erreichbarkeit der Kunden verschlechtert (Metzler 2013). Dadurch müssen unter Umständen mehrfache Zustellversuche unternommen werden (Auffermann und Lange 2008). Um der Nichtanwesenheit ihrer Kunden vorzubeugen, können die Logistikdienstleister mit ihren Kunden die Belieferung innerhalb von Zeitfenstern vereinbaren (Vahrenkamp und Kotzab 2012, S. 128 f.). 47 Prozent der deutschen Konsumenten erwarten, dass diese Zeitfenster eine maximale Spannweite von 2 Stunden haben (Handelsblatt 2019).

Aufgrund des genannten Anstiegs des Paketvolumens, den Wünschen der Kunden bzw. der Politik sowie den Fehlversuchen in der Zustellung, gilt die Last Mile als einer der teuersten und ineffizientesten Teile der Supply Chain (Gevaers et al. 2011). Zusätzlich führt die Bündelung der Sendungen auf der Haupttransportrelation (vgl. Abschnitt 2.2) dazu, dass die Transportkosten überwiegend von den Kosten für die Sammel- und Verteilverkehre abhängen (Koether 2012, S. 104). Diese Kosten werden in der Literatur unterschiedlich beziffert. Goodman (2005) gibt an, dass 28 Prozent aller Transportkosten auf der Last Mile anfallen. Wogegen Metzler (2013) und Joerss et al. (2016) von einem Anteil von bis zu 50 Prozent ausgehen. Deswegen wird die Wirtschaftlichkeit der Last Mile besonders durch die Transportkosten beeinflusst (Rall 2019). Die Transportkosten und die ökologische Effizienz sind nach Vahrenkamp und Kotzab (2012, S. 458 f.) sowie Kellner (2016) unter anderem von der Transportstrecke, der Transportzeit und der minimalen Anzahl an Auslieferungstouren abhängig. Der Einfluss der Transportstrecke auf die Kosten resultiert aus den Kosten für jeden gefahrenen Kilometer eines Transportfahrzeugs. Zudem erhöhen jeder gefahrene Kilometer sowie die Transportzeit den CO₂ Ausstoß. Die Transportzeit ist dabei von der Infrastruktur des Straßennetzes, beispielsweise Bundesstraße oder Autobahn, und den Verkehrsbedingungen, wie Staus, abhängig. Für die Ausführung der Auslieferungstouren werden Transportfahrzeuge benötigt, die wiederum Kapital darstellen. Eine kleinere Fahrzeugflotte aufgrund weniger bedienter Touren führt somit zu weniger Kosten. Auch die niedrigen Stoppfaktoren im B2C haben nach Tripp (2019, S. 258) Auswirkungen auf den operativen Erfolg des Logistikunternehmens. Denn zumeist erhält der Endkunde im B2C ein Paket pro Stopp, während im B2B pro Stopp 3 bis 5 Pakete verteilt werden. Metzler (2013) spricht im B2C von durchschnittlich 1,1 Paketen pro Stopp. Neben dem höheren Stoppfaktor sind Erstzustellungen im B2B Bereich erfolgreicher, da Firmenkunden in der Regel häufiger an der Firmenadresse anzutreffen sind (Umundum 2015). Zudem wird von Sonntag und Thulesius (2015) erwartet, dass sich für die Beförderung neuer Sortimente wie beispielsweise Tiefkühlprodukte spezialisierte Transportnetzwerke bilden werden.

2.4 Verfahren und Konzepte um auf die Herausforderungen in der Last Mile zu reagieren

Durch den sehr starken Wettbewerb zwischen den Paketunternehmen und dem dadurch entstehenden Preisdruck, ist es für die Unternehmen eine Herausforderung die

Belieferung auf der Last Mile kosteneffizient und gleichzeitig nachhaltig durchzuführen (Ehmke und Mattfeld 2012; Bretzke 2015, S. 363). Auf die wirtschaftliche und ökologische Effizienz haben unter anderem sowohl die Dauer der Fahrzeiten als auch die Länge der Transportstrecken großen Einfluss (Kellner 2016). Nach van Duin et al. (2016) kann die Effizienz und die Nachhaltigkeit der Belieferung auf der Last Mile mit verschiedenen Verfahren und Konzepten verbessert werden. Dies können beispielsweise Abholstellen (Pick Up Points), eine Verlegung der Lieferung ins Wochenende oder kürzere Transportwege durch beispielsweise eine verbesserte Tourenplanung sein.

Edwards et al (2009) untersuchen, ob der Einsatz von Abholstellen nach einem ersten fehlgeschlagenen Zustellversuch zu einer Reduzierung der CO₂ Emissionen führt. Dabei kommen die Autoren zu dem Ergebnis, dass der Einsatz von Abholstellen wie Supermärkten oder Bahnhöfen zu einer Reduktion des CO₂ Ausstoßes führen kann. Punakivi et al. (2001) zeigen in ihrer Studie, dass eine Lieferbox, die entweder beim Kunden fest angebracht oder vom Paketboten zum Kunden gebracht und dann mit einem Schließmechanismus gegen Diebstahl gesichert wird, die Kosten für die Logistikdienstleister um bis zu 60 Prozent senken kann. Außerdem würde so eine Erstzustellungsquote von 100 Prozent erreicht, wodurch mehrfache Zustellversuche vermieden werden können. Im Jahr 2014 startete DHL mit dem Verkauf von Paketboxen, die beim Kunden fest installiert werden und nur von DHL genutzt werden können (Süddeutsche Zeitung 2014). Dieses Angebot wurde im Jahr 2019 eingestellt, weil dieses Angebot von den Kunden wenig angenommen wurde (WinFuture 2019). Im Jahr 2002 startete die Tochtergesellschaft der Deutsche Post AG (2010, S. 94) DHL ein System mit Packstationen. Die Packstationen stehen an zentralen, öffentlichen Orten und ermöglichen den Paketboten von DHL eine Kundenunabhängige Lieferung. Nach der Lieferung kann der Kunde sein Paket mit einem Code aus der Packstation holen. Nach Angaben der Deutschen Post AG konnten so im Jahr 2005 in Köln mit 33 Packstationen insgesamt 40000 gefahrene Kilometer und 11 Tonnen CO₂ Emissionen eingespart werden. Im Jahr 2019 existieren deutschlandweit 3500 Packstationen, deren Anzahl in den nächsten Jahren noch weiter erhöht werden soll (WinFuture 2019). Die Konkurrenzunternehmen GLS, Hermes und DPD führten im Jahr 2015, im Gegensatz zu DHL, ein Anbieter offenes System von Packstationen und Paketboxen ein (Wirtschaftswoche 2015). Innerhalb dieses Systems sollen Paketboxen vor Ein- sowie Mehrfamilienhäusern eingerichtet und die öffentlichen Packstationen an zentralen Orten errichtet werden. Die ersten Stationen wurden beim Einzelhändler dm installiert

(Eurotransport 2019). Kuwok und Asdecker (2015) untersuchen in ihrer Studie aus dem Jahr 2015 die Nachhaltigkeit von Packstationen. Werden alle Pakete einer Auslieferungstour in Packstationen abgelegt, können, im Vergleich zu einer Auslieferung ohne Fehlzustellungsquote, die CO₂ Emissionen um 27 Prozent gesenkt werden. Steigt die Fehlzustellungsquote im Vergleichsfall an, können durch die Packstationen weitere CO₂ Emissionen eingespart werden. Die errechneten Einsparungen setzen voraus, dass die Kunden die Pakete an den Packstationen nicht zu 100 Prozent mit dem Auto abholen. Bei 100 Prozent erhöhen sich die ausgestoßenen CO₂ Emissionen um 10 Prozent. Die Autoren haben für die Abholung mit dem Auto durch eine Umfrage einen Wert von 66 Prozent ermittelt. Die restlichen Kunden holen ihre Pakete zu Fuß ab oder nutzen dazu das Fahrrad, den Bus oder das Motorrad. Die Reduzierung der CO₂ Emissionen mit Packstationen ist damit auch von der kundenseitigen Nutzung ökologisch vorteilhafterer Transportmittel abhängig.

Neben Packstationen und Paketboxen können die Transportkosten sowie die ökologische Effizienz auch mit Hilfe von Tourenplanungsverfahren und durch eine höhere Auslastung der Transportmittel minimiert werden (Kellner 2016; Cleophas und Ehmke 2014). Das Tourenplanungsproblem wird im englischsprachigen Raum als Vehicle Routing Problem bezeichnet oder kurz VRP (Rieck et al. 2007). Nach Vahrenkamp und Kotzab (2012, S. 454 - 459) ist die Tourenplanung für die Paketdienste aufgrund der steigenden Bedeutung des E-Commerce besonders relevant. Bei der Tourenplanung wird versucht ein Auslieferungsproblem zu lösen. Dabei sollen die Kunden einer Region mit Fahrzeugen möglichst unter Einhaltung aller Restriktionen, wie die Kapazität der Fahrzeuge oder die maximal gefahrene Strecke pro Fahrzeug, auf einer Tour beliefert werden. Die Kunden und die Ausgangspunkte werden auch als Knoten bezeichnet, während die Entfernung zwischen zwei Knoten Kante genannt wird. Die Länge der Kanten wird entweder durch geografische Informationssysteme (GIS) oder durch Distanzmetriken, wie der euklidischen Distanz oder der Manhattan-Distanz, bestimmt. Längen, die mit Hilfe eines GIS, also auf der Basis von digitalen Straßenkarten, bestimmt werden, bilden die reale Distanz des Straßennetzes ab (Schulte 2017, S. 316). Bei der Bestimmung mittels Distanzmetrik handelt es sich nur um eine Annäherung an die reale Distanz (Apparicio et al. 2003).

Das Ziel einer Tourenplanung ist nach Vahrenkamp und Kotzab (2012, S. 454 - 459) vom jeweiligen Unternehmen abhängig. Die Tourenplanung kann unter der Berücksichtigung

unterschiedlicher Ziele durchgeführt werden. Die Ziele können unter anderem eine Minimierung der Kosten oder der gesamten Transportstrecke sein. Weitere Ziele können die Ermittlung der Mindestanzahl benötigter Fahrzeuge oder der Touren mit der geringsten Fahrzeit darstellen. Die Verbesserung des Lieferservices oder höhere Auslastung der Transportmittel können auch Ziele einer Tourenplanung sein. Für die Tourenplanung wichtige Kennzahlen sind nach Becker und Winkelmann (2019, S. 276) die Tourenlänge (Gesamtsumme gefahrener Kilometer), die durchschnittliche Tourendauer (Summe Tourzeiten/Anzahl Auslieferungstouren) und die durchschnittliche Auslastung der Transportfläche (Summe genutztes Transportvolumen/vorhandenes Transportvolumen).

Bei dem Tourenplanungsproblemen handelt es sich um NP-vollständige Probleme (Gietz 2008). Mattfeld und Vahrenkamp (2014, S. 229f.) benutzen den Begriff NP-hart. Beide Begriffe bedeuten, dass es schwierig ist, für die Tourenplanungsprobleme optimale Ergebnisse zu berechnen. Für eine Knotenzahl von $n=15$ können $(n-1)!$ verschiedene Touren gebildet werden, was 1.307,67 Milliarden Möglichkeiten entspricht. Momentan existiert kein Algorithmus, der die optimale Lösung eines Tourenplanungsproblems in polynomialer Rechenzeit berechnen kann. Stattdessen steigt die Rechenzeit exponentiell an. In der Praxis wurden deshalb sogenannte heuristische Verfahren entwickelt, die sich der optimalen Lösung annähern (Mattfeld und Vahrenkamp 2014, S. 49 f.). Um die entwickelten heuristischen Verfahren miteinander vergleichen zu können, werden mit den Verfahren Benchmark-Testsets gelöst und als Kennzahlen die benötigten Fahrzeuge, die benötigte Fahrstrecke sowie die benötigte Zeit zum Lösen des Testsets aufgenommen (Pisinger und Ropke 2007). Diese Testsets, mit unterschiedlicher Quantität an Kundenstandorten, werden unter anderem auf der Webseite von Uchoa et al. (2014) bereitgestellt. Weitere Benchmark-Testsets lassen sich auf der Webseite der, zur Universität von Malaga gehörenden, Networking and Emerging Optimization Research Group (2013) finden.

Die Auswirkungen einer Erhöhung des Stoppfaktors auf der Last Mile untersuchen Edwards et al. (2010). Dabei analysieren sie eine Tour mit der Länge von 50 Meilen und 120 Lieferungen. Wird bei jedem Stopp ein Paket ausgeliefert, werden 181 Gramm CO₂ pro Lieferung ausgestoßen. Die Erhöhung des Stoppfaktors auf 1,4 führt zu einer Verringerung des Ausstoßes auf 137 Gramm CO₂. Eine weitere Verringerung auf 72 Gramm CO₂ lässt sich mit einer Erhöhung des Stoppfaktors auf 2,5 erreichen. Aus diesem

Grund geben die Autoren den Logistikdienstleistern die Handlungsempfehlung, ihre Stoppfaktoren zu maximieren.

Eine weitere Idee, um die Belieferung auf der Last Mile zu verbessern, ist zum Beispiel die Zusammenarbeit zwischen Logistikdienstleistern. Um die Transporteffizienz in Athen zu verbessern, untersuchen Rabe et al. (2018) mit Hilfe einer ereignisdiskreten Simulation den Einsatz von Urban Consolidation Centers (UCC) unter der Voraussetzung der horizontalen Kollaboration der beteiligten Unternehmen. Die Simulationen wurden mit Realdaten von griechischen Logistikdienstleistern durchgeführt. Die Fahrstrecke jeder Tour wird mit Hilfe der gewichteten euklidischen Distanz abgeschätzt. Um mögliche Verbesserungen quantifizieren zu können, werden von den Autoren verschiedene Kennzahlen eingesetzt. Unter anderem werden die Anzahl der Touren, die Anzahl der Lieferungen insgesamt und pro Tour, die Länge der Fahrstrecke insgesamt und pro Tour sowie die Anzahl der gelieferten Boxen insgesamt und pro Tour aufgenommen. Mit einer Division aus der Kennzahl der gelieferten Boxen pro Tour und der Kapazität der Lieferfahrzeuge wird außerdem die Auslastung der Fahrzeuge pro Tour berechnet. Nach der Durchführung und Auswertung der Simulationen kommen die Autoren zu dem Ergebnis, dass der Einsatz von UCCs ökologisch und wirtschaftlich erfolgreich sein kann. Denn die Anzahl der Touren, die zur Belieferung der Kunden nötig sind, sowie die Länge der Fahrstrecke können reduziert werden. Die Autoren geben aber zu bedenken, dass die Ergebnisse der Studie aufgrund der zur Verfügung gestellten Daten der Logistikdienstleister limitiert sind, da beispielsweise keine Informationen über die Betriebskosten eines UCC oder die Zeitfenster, in denen die Kunden beliefert werden wollten, zur Verfügung standen.

Auch die Zustellung durch andere Lieferfahrzeuge werden in der Literatur untersucht. In einer Simulationsstudie von van Duin et al. (2014) wird der Einsatz von Transportbooten bei der Belieferung von Einzelhändlern, Gastronomiebetrieben und Hotels entlang der Kanäle in Amsterdam analysiert. Auch sollte in der Simulationsstudie herausgefunden werden, ob der Einsatz von Transportbooten auf den Kanälen Einflüsse auf die Transitzeiten der Haus- und Touristenboote hat. Als Kennzahlen wurden unter anderem die Durchlaufzeit, die Auslastung der eingesetzten Transportboote genutzt und die durchschnittlichen Transitzeiten der Haus- bzw. Touristenboote aufgenommen. Mit Hilfe der Simulationsstudie wird aufgezeigt, dass der Einsatz von Transportbooten den restlichen Verkehr auf den Kanälen um 3,6 Minuten verzögert. Vor dem Hintergrund der

auf Freizeitaktivitäten ausgelegten Haus- und Touristenboote betrachten die Autoren diese Verzögerungen als vernachlässigbar. Die Autoren kommen zu dem Ergebnis, dass Transportboote als ernstzunehmende Transportalternative zu betrachten sind und möglicherweise in weiteren Städten mit Kanalsystemen genutzt werden können. Die ökonomische sowie ökologische Durchführbarkeit gilt es aber noch zu erforschen. Die ökologische Durchführbarkeit wird in einer Simulationsstudie von van Duin et al (2018) untersucht und baut auf van Duin et al. (2014) auf. Darin wird die Belieferung mit Transportbooten simuliert und anschließend mit einer berechneten Anzahl an LKW Touren verglichen. Dabei kommen die Autoren zu dem Ergebnis, dass sich je nach Szenario 5 bis 55 LKWs einsparen lassen. Unter der Voraussetzung eines elektrischen Antriebs der Transportboote bezeichnen die Autoren die Anlieferung mit Transportbooten als ein innovatives Konzept, um Emissionen einzusparen. Die Auslieferung von Paketen mittels Transportbooten halten auch Janjevic und Ndiaye (2014) für möglich. Wird der letzte Weg zum Kunden beispielsweise mit einem Lastenfahrrad zurückgelegt, könnte diese Art des Transports auch in Städten mit weniger dichten Wasserwegen wie Paris eingesetzt werden.

Arnold et al. (2018) untersuchen die Zustellung von Paketen mit einem Lastenfahrrad. In der Simulationsstudie wurden drei Zustellweisen betrachtet. Die Zustellung per Transportvan, die Selbstabholung der Pakete durch die Kunden an Abholstellen oder die Zustellung mit Lastenfahrrädern von den Abholstellen aus. Bei der Selbstabholung durch die Kunden und der Zustellung per Lastenfahrrädern werden die Pakete mit Transportvans bis zu den Abholstellen transportiert. Die Zustellung mit Hilfe von Transportvans reduzieren die Kosten für die Logistikdienstleister mit jedem ausgelieferten Paket. Diese Kosten und die Transportwege können durch eine höhere Anzahl der Selbstabholung durch den Kunden weiter verringert werden. Dabei gilt es jedoch zu beachten, dass dabei der Individualverkehr durch die Kunden in der Stadt ansteigt. Der Einsatz der Transportfahrräder führt zu einer Erhöhung der Kosten für die Logistikdienstleister, die gefahrenen Kilometer durch die Transportvans reduzieren sich dabei jedoch um mehr als 50 Prozent. Um die Kosten für die Dienstleister zu senken und trotzdem weniger Kilometer mit Transportvans zurückzulegen, schlagen die Autoren einen hybriden Einsatz der Selbstabholung durch die Kunden und der Lastenfahrräder vor. Eine von Melo und Baptista (2017) durchgeführte Simulationsstudie zeigt ebenfalls positive Auswirkungen auf den CO₂-Ausstoß, wenn Transportvans durch Lastenfahrräder ersetzt werden. In beiden vorgestellten Simulationsstudien gehen die

Autoren allerdings davon aus, dass die Abmaße und das Gewicht der Pakete gering genug sind, um sie auf Lastenfahrrädern zu transportieren.

Zu den ausgefalleneren Lieferfahrzeugen gehören unter anderem Lieferroboter oder Drohnen. Poeting et al. (2019) untersuchen den Einsatz von Lieferrobotern auf der Last Mile mit Hilfe von agentenbasierter Simulation in der Stadt Köln. Um die ökonomischen und ökologischen Auswirkungen der Liefertouren messen zu können, werden unter anderem die Kilometer pro Tour, die Dauer jeder Tour und der CO₂ Ausstoß aufgenommen. Die Fahrstrecke wird mit Hilfe des GIS OpenStreetMaps (OSM) als reale Distanz berechnet. In der Studie werden vier Szenarien betrachtet. Im Ersten werden keine Auslieferungen mit den Lieferrobotern durchgeführt, im Zweiten werden ein Prozent, im Dritten zwei Prozent und im Vierten drei Prozent der Lieferungen mit Hilfe von Lieferrobotern durchgeführt. Als Ergebnis geben die Autoren an, dass der Einsatz der Lieferroboter keinen signifikanten Einfluss auf die Gesamtfahrstrecken und damit die CO₂ Bilanz haben.

Auch die Möglichkeit, Pakete per Drohne auszuliefern, wird von einer Vielzahl an Unternehmen untersucht und getestet. Dabei ist der Einsatz der Drohnen nicht nur auf die Last Mile beschränkt. Im Jahr 2013 wurde diese Technologie noch eingesetzt, um ein Paket einen Kilometer über den Rhein zum Hauptquartier des Logistikdienstleisters DHL (2018) zu schicken. Der Logistikdienstleister entwickelte die Technologie weiter und setzte die Drohne im Jahr 2014 für die Belieferung der Nordseeinsel Juist und im Jahr 2016 in der Alpenregion Reit im Winkl ein. Im Jahr 2019 startete DHL mit der Testzustellung in China (Golem 2019). Der Konzern Amazon startete seine Feldversuche mit Drohnen in Europa im Jahr 2013 und möchte im Jahr 2019 im Rahmen eines regulären Testbetriebs Kleinteile wie zum Beispiel Zahnpasta auf diese Weise ausliefern (Amazon 2016; Tagesschau 2019). Das zum Google Mutterkonzern Alphabet gehörende Unternehmen Wing hat im Jahr 2019 angekündigt, im australischen Canberra Ware an zugängliche Häuser zu liefern, indem die Ware aus geringer Höhe über dem Grundstück abgeworfen wird (WirtschaftsWoche 2019). Nach Joerss et al. (2016) liegt die Zukunft der Drohnenbelieferung in ländlichen Gebieten. Die Drohnentechnologie wird aber nicht nur im Kontext der Paketlieferungen betrachtet, sondern stellt auch innerhalb der Katastrophenhilfe für schwer zugängliche Gebiete ein aktuelles Forschungsthema dar (Rabta et al. 2018).

Eine Fülle an Zukunftskonzepten für die Belieferung auf der Last Mile zeigt DHL (2014) in einem Bericht aus dem Jahr 2014 auf. Unter anderem wird in dem Bericht die Unterstützung des Paketboten mittels autonomer Belieferungsfahrzeuge erläutert. Dabei fährt das Fahrzeug neben dem Paketboten her und dieser entnimmt die Pakete aus dem fahrenden Fahrzeug. Wenn die Strecke zum nächsten Kunden weiter ist, kann der Fahrer in das Fahrzeug steigen und zu diesem Kunden selbst fahren. Auch wird die Idee der selbstfahrenden Packstation aufgebracht. Bei diesem Konzept fährt eine Packstation autonom zu festgelegten Zeiten an vorher bestimmte Orte und steht den Kunden zur Paketentnahme oder -annahme zur Verfügung. In höherem Maße futuristisch wirken die nun folgenden beiden Konzepte. Das erste Konzept beschreibt Pakete die selbst zum Kunden fahren und im zweiten Konzept fährt ein autonomes Transportfahrzeug in die Nähe der Kunden und entlädt dort mehrere kleine Fahrzeuge mit Schwarmverhalten, die die Pakete zu den Kunden bringen. Im Jahr 2019 ist durch Post Chef Frank Appel verkündet worden, dass die Unterstützung der Paketboten mittels autonomer Belieferungsfahrzeuge, Postbot genannt, zwar technisch funktioniert, in der Anschaffung jedoch zu teuer ist und der Einsatz bis auf weiteres aufgeschoben wird (WinFuture 2019).

3. Modellierung und Simulation

Zu Beginn dieses Kapitels werden die Grundlagen vorgestellt, auf denen die in der vorliegenden Arbeit durchgeführte Simulationsstudie aufbaut. Daran anschließend folgt eine Einführung in das strukturierte Vorgehen innerhalb einer Simulationsstudie. Da die Datenbeschaffung und -aufbereitung sowie die Analyse von Simulationsergebnissen für diese Arbeit von besonderer Bedeutung sind, werden diese jeweils in einem eigenen Abschnitt erklärt.

3.1 Grundlagen einer Simulation

Ein System wird durch die VDI - Richtlinie 3633 Blatt 1 (2014, S. 4) als eine Anzahl von Elementen bezeichnet, die miteinander in Verbindung stehen und von ihrer Umwelt abgegrenzt sind. Die Abgrenzung erfolgt laut Gutenschwager et al. (2017, S. 11) mittels Systemgrenzen über die mit Hilfe von Schnittstellen Materie, Energie und Informationen, die sogenannten Ein - und Ausgangsgrößen, ausgetauscht werden können. Wird ein vorhandenes oder geplantes System in eine vereinfachte Nachbildung überführt, so wird diese Abbildung als Modell bezeichnet und unterscheidet sich nur innerhalb eines, von der Untersuchung abhängigem, Toleranzrahmens vom abgebildeten System (VDI - Richtlinie 3633 Blatt 1 2014, S. 3). Werden die Attribute eines Modells oder Modellelements von außen mit Werten belegt, so werden sie als Parameter bezeichnet. Als Simulation wird das „nachbilden eines Systems mit seinen dynamischen Prozessen in einem experimentierbaren Modell, um zu Erkenntnissen zu gelangen, die auf die Wirklichkeit übertragbar sind“ (VDI - Richtlinie 3633 Blatt 1 2014, S. 3) bezeichnet, wobei sich das Modell über die Zeit entwickelt. Die experimentierbaren Modelle werden auch Simulationsmodelle genannt (Gutenschwager et a. 2017, S. 22).

Die einmalige Ausführung des Simulationsmodells mit festgelegter Parameterkonfiguration ist nach Gutenschwager et al. (2017, S. 23) ein Simulationslauf. Dabei können die Ergebnisse eines jeden Simulationslaufs bei gleicher Parametereinstellung reproduziert werden. Um statistisch abgesicherte Ergebnisse zu erhalten, wird ein Simulationslauf wiederholt. Diese Wiederholung nennt sich Replikation. Die Replikation unterscheidet sich vom Simulationslauf insofern, dass die Startwerte für die Zufallszahlengenerierung verändert werden (Rabe et al. 2008, S. 12f.).

In einem Simulationsexperiment werden mehrere Simulationsläufe mit jeweils systematischer Parametervariation durchgeführt (VDI - Richtlinie 3633 Blatt 1 2014, S. 3).

Simulationsmodelle werden von Law (2015, S. 6) und Gutenschwager et al. (2017, S. 15 ff.) in statische oder dynamische, deterministische oder stochastische sowie kontinuierliche oder diskrete Simulationsmodelle eingeteilt. Ein statisches Simulationsmodell spiegelt ein System zu einem bestimmten Zeitpunkt wider oder es bildet ein System ab, in dem Zeit keine Rolle spielt. Dementsprechend repräsentiert ein dynamisches Simulationsmodell ein System, das sich über die Zeit entwickelt. Ein Simulationsmodelle ohne jegliche zufälligen Komponenten ist deterministisch. Sobald die Eingangsgrößen des Simulationsmodells spezifiziert sind, sind die Ausgangsgrößen vorherbestimmt, auch wenn viel Rechenzeit benötigt wird, um die Ausgangsgrößen zu bestimmen. In einem stochastischen Simulationsmodell hingegen werden zumindest einige zufällige Eingangsgrößen verarbeitet, sodass auch zufällige Ausgangsgrößen entstehen. Die Zustände in kontinuierlichen Simulationsmodellen verändern sich kontinuierlich über der Zeit. Im Gegensatz dazu finden die Zustandsänderungen in diskreten Simulationsmodellen nur zu diskreten Zeitpunkten statt. Weiter wird der Zeitfortschritt in diskreten Simulationsmodellen entweder durch das Eintreten von Ereignissen, dann als ereignisdiskrete Simulation oder in der englischsprachigen Literatur discrete event simulation (Law 2015, S.6) bezeichnet, oder in festgelegten zeitlichen Intervallen, zeitdiskrete Simulation genannt, erfolgen (Wenzel 2018).

Eine weitere Klassifizierung der Simulation kann nach Robinson (2014, S. 168) und Law (2015, S. 494 f.) in terminierende und nicht terminierende Simulation erfolgen. In der terminierenden Simulation wird die Länge des Simulationslaufs durch ein natürliches Ereignis begrenzt. Dieses Ereignis kann zum Beispiel der Zeitpunkt sein, an dem das System leergelaufen ist. Bei einem Produktionsauftrag kann dieses Ereignis beispielsweise die Fertigstellung des gesamten Produktionsauftrags sein. Auch kann das Ereignis durch das Management vorgegeben werden. In einer nicht terminierenden Simulation existiert kein natürliches Ereignis, welches die Simulation beendet. Ein typisches Beispiel ist die Untersuchung eines Produktionssystems, um das Verhalten im Langzeitbetrieb zu betrachten. Die Simulation in der vorliegenden Arbeit wird durch die Generierung der Standorte begrenzt. Ist die Generierung aller Kundenstandorte abgeschlossen, so wird auch die Simulation beendet. Damit gehört die in dieser Arbeit

getätigte Simulation zur Gruppe der terminierenden Simulationen. Des Weiteren ist das Simulationsmodell in der vorliegenden Arbeit ein statisches, da die Zeit in diesem nicht relevant ist.

Bei der Analyse von stochastischen und statischen Simulationsmodellen kommen Monte-Carlo-Simulationen zum Einsatz (Domschke et al. 2015, S. 234 f.). Arroyo et al. (2019) nutzen die Monte-Carlo-Simulation in einem Simulationsmodell für die Last Mile, um die Gesamtkosten im Betrieb (Englisch: total cost of ownership), welche in ihrem Modell von 14 Variablen abhängig sind, zu berechnen. Gonzalea-Martin et al. (2018) nutzen die Monte-Carlo-Simulation innerhalb ihrer Simulationsrahmenstruktur für ein Tourenplanungsproblem, um die zu erwartenden Kosten aufgrund von Ausfällen einer Tour sowie die Zuverlässigkeit, dass eine Tour nicht ausfällt, zu berechnen. Aufgrund ihrer statischen Natur bilden Monte-Carlo-Simulationen die Zeit nicht explizit ab (Gutenschwager et al. 2017, S. 23). In einer Monte-Carlo-Simulation werden die Inputparameter nach Hofstadler und Kummer (2017, S. 197 ff.) durch Zufallszahlen abgebildet. Durch ein vorgegebenes Berechnungsmodell werden die Inputparameter in zufällige Ergebnisgrößen überführt. Die Änderung der Zufallszahlen und die anschließende Berechnung der Ergebnisgrößen wird für eine festgelegte Anzahl, auch Iterationen genannt, wiederholt. Die Iterationen werden in stufenweise und wiederholende Iterationen unterteilt. Bei einer stufenweisen Iteration ändern sich die Inputs in Abhängigkeit des Outputs der vorherigen Iteration. Im Gegensatz dazu sind die Inputs bei wiederholenden Iterationen von den Outputs unabhängig. Das heißt, dass die Werte der Zufallszahlen vom Ergebnis vorheriger Iterationen unabhängig sind. Bei eingehender Betrachtung der Definition einer wiederholenden Iteration fällt auf, dass sich diese mit der Definition einer Replikation zu Beginn dieses Abschnitts deckt. Um Verwechslungen vorzubeugen, wird im weiteren Verlauf der vorliegenden Arbeit die Bezeichnung „Replikationen“ für wiederholende Iterationen genutzt. Mundform et al. (2011) empfehlen für Monte-Carlo-Simulationen zwischen 5000 und 8000 Replikationen. Law (2015, S. 507) empfiehlt für stochastische Simulationen mindestens drei bis fünf Replikationen durchzuführen.

Domschke et al. (2015, S. 234 f.) nennen die Unabhängigkeit aufeinander folgender Ereignisse des zugrundeliegenden Simulationsmodells sowie das Wissen über die Wahrscheinlichkeitsverteilungen der Parameter als Grundvoraussetzung einer Monte-Carlo-Simulation. Für Brandimarte (2014, S. 35 f.) stellen Monte-Carlo-Simulationen

vier Anforderungen. Die Möglichkeit das Systemverhalten zu beschreiben, ein Zufallszahlengenerator, statistische Werkzeuge zum Sammeln und Auswerten der Ergebnisse sowie die Möglichkeit die Ergebnisse zu visualisieren. Diese Anforderungen werden von Universalsprachen wie C++ oder Java und von Softwareprogrammen wie MATLAB oder R erfüllt. Auch Microsoft Excel kann allein, in Kombination mit der enthaltenen Programmiersprache VBA oder durch Erweiterungen (add-ins), wie das proprietäre @Risk (Palisade 2019), für Monte-Carlo-Simulationen eingesetzt werden (Law 2015, S. 717). Für den Autor liegt der Vorteil bei der Verwendung einer universellen Programmiersprache in ihrer Ausführungsgeschwindigkeit, bei der Verwendung eines Tabellenkalkulationsprogrammes wie Excel in der bekannten Nutzeroberfläche. Botchkarev (2015) gibt zu bedenken, dass für Monte-Carlo-Simulationen in Excel externe Zufallszahlengeneratoren genutzt werden sollen, da die in Excel vorhandenen Funktionen für die Zufallszahlengenerierung unzureichend sind. Nach Banks et al. (2014a, S. 27) haben die von Excel bereitgestellten Zufallszahlengeneratoren bekannte Defizite, bezüglich der Gleichverteilung und Unabhängigkeit der Zufallszahlen. Sawilowksy (2003) formuliert vier Faktoren, die erfüllt werden müssen, um eine hohe Qualität der Monte-Carlo-Simulation sicherzustellen: Der Zufallszahlengenerator hat eine große Periode bis sich die Zahlenwerte wiederholen, die Anzahl der Replikationen ist groß genug, um akkurate Ergebnisse zu ermöglichen, das Berechnungsmodell spiegelt das Simulationsmodell wider und die Simulationsstudie untersucht die Fragestellung. Nach Gentle (2003, S. 310) müssen die Parameter, die genutzte Software sowie Computer und die Startwerte der Zufallszahlengeneratoren der Monte-Carlo-Simulation notiert werden, um die Simulation exakt reproduzieren zu können.

In Abbildung 6 ist die Struktur einer Monte-Carlo-Simulation zu sehen. Nach Klein (2010) müssen zu Beginn einer Monte-Carlo-Simulation die Wahrscheinlichkeitsverteilungen der mit Unsicherheit behafteten Parameter, auch Inputgrößen genannt, geschätzt werden. Danach wird eine Menge von U_i Zufallszahlen auf dem Intervall $(0; 1)$ generiert und in die geforderte Wahrscheinlichkeitsverteilung transformiert (Brandimarte 2014, S. 254 f.). Klein (2010) zufolge werden die transformierten Zufallsvariablen X_a in das Simulationsmodell eingegeben und darauf aufbauend der Ergebniswert Y_k berechnet. Dieses Vorgehen wird, beginnend mit der Generierung von Zufallszahlen, k -mal wiederholt und wird als die Durchführung von k Simulationsläufen bezeichnet. Ist die gewünschte Anzahl an Simulationsläufen erreicht, wird die berechnete

Ergebnisverteilung (Y_1 bis Y_k) analysiert und statistisch ausgewertet. Brandimarte (2014, S. 255) nennt hierfür beispielsweise den Punktschätzer und das Konfidenzintervall.

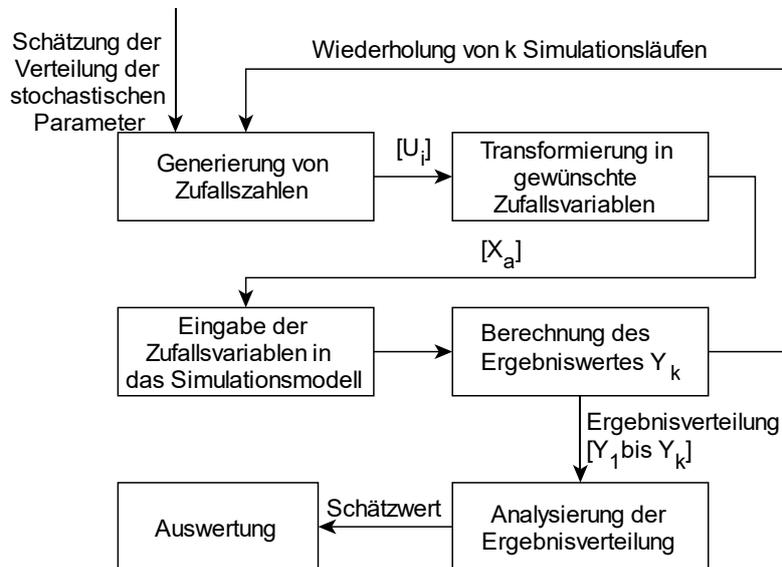


Abbildung 6: Struktur einer Monte-Carlo-Simulation (nach Brandimarte 2014, S. 254 und Klein 2010)

3.2 Vorgehen in einer Simulationsstudie

Um eine Simulationsstudie erfolgreich durchführen zu können, ist es von zentraler Bedeutung, dass die Projektvorbereitung und -durchführung systematisch aufgebaut wird (Gutenschwager et al. 2017, S. 142 f.). Dafür haben Rabe et al. (2008, S. 4 - 8) ein Vorgehensmodell (Abbildung 7) entwickelt. Das Modell ist dabei in Phasen (durch Ellipsen dargestellt) und Phasenergebnisse (durch Rechtecke dargestellt) aufgeteilt. Zu den Phasen gehören die Aufgabendefinition, die Systemanalyse, die Datenbeschaffung, die Modellformalisierung, die Datenaufbereitung, die Implementierung sowie die Experimente und Analyse. Jede der genannten Phasen besitzt ein Phasenergebnis. Diese Ergebnisse sind die Aufgabenspezifikation, das Konzeptmodell, die Rohdaten, das formale Modell, die aufbereiteten Daten, das ausführbare Modell und die Simulationsergebnisse. Alle Phasenergebnisse sind einer detaillierten Verifikation und Validierung (V&V) zu unterziehen, um wirtschaftlichem Schaden vorzubeugen. Durch diese Vorgehensweise können Fehler frühzeitig erkannt, behoben und so die Auswirkungen auf das Simulationsmodell begrenzt werden. Zwei Beispiele für V&V-Techniken sind nach Rabe et al. (2009, S. 102 - 105) die Validierung im Dialog und der

Schreibtischtest. Beim Schreibtischtest wird die eigene Arbeit einer sorgfältigen Überprüfung, unter anderem auf Vollständigkeit und Korrektheit, unterzogen. Dabei werden jedoch häufig eigene Fehler nicht erkannt, weshalb es empfehlenswert ist, eine andere Person hinzuzuziehen. Das Hinzuziehen dieser zweiten Person stellt die V&V-Technik Validierung im Dialog dar. In dieser wird zusammen mit Fachexperten, denen das reale System bekannt ist, das Modell diskutiert. Dabei schätzen die Fachexperten das Simulationsmodell aufgrund ihrer Erfahrung ab.

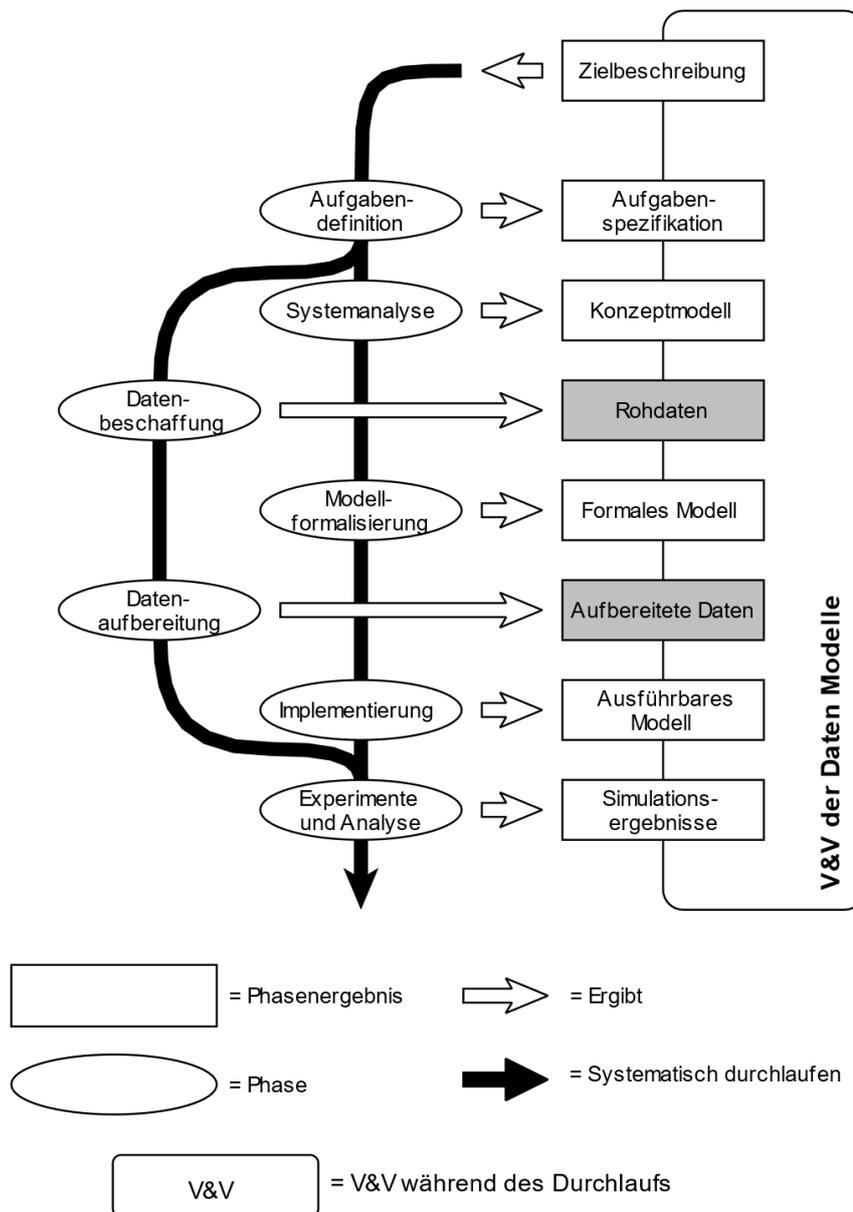


Abbildung 7: Vorgehensmodell für Simulationsstudien (nach Rabe et al. 2008, S. 5)

Die Zielbeschreibung umfasst unter anderem den Untersuchungszweck, die wesentlichen Ziele, die erwarteten Ergebnisse der Simulationsstudie und die geplante Nutzung des

erstellten Modells sowie die Beantwortung der Fragestellung, ob die Simulation die geeignete Analysemethode für den Untersuchungszweck darstellt (Gutenschwager et al., S. 147 f.). Nachfolgend werden die Phasen des Vorgehensmodell kurz vorgestellt. Nach Gutenschwager et al. (2017, S. 144) wird in der Aufgabendefinition die Zielbeschreibung präzisiert. So wird mit der Aufgabenspezifikation ein Grundverständnis der zu bearbeitenden Aufgabe erzeugt. Durch die Systemanalyse wird das Modell bezüglich seines Detaillierungsgrads und den Systemgrenzen hin eingeschränkt. Dabei wird festgelegt, welche Elemente des Systems mit welcher Genauigkeit modelliert werden. Zum Beispiel können innerhalb einer Supply Chain nur einzelne Segmente anstatt der gesamten Supply Chain abgebildet werden (Stefanovic et al. 2009). Die Ergebnisse der Modellformalisierung werden unter anderem in Textform oder mit Ablaufdiagrammen im Konzeptmodell festgehalten, welches damit diese Phase abschließt. Mit Hilfe der Modellformalisierung wird beispielsweise das in Textform beschriebene Konzeptmodell in einen Programmablaufplan überführt. So soll sich das formale Modell als Ergebnis dieser Phase ohne weitere Erläuterungen zum abzubildenden System implementieren lassen. Diese Phase ist vom verwendeten Simulationswerkzeug unabhängig. In der Implementierung entsteht das eigentliche Simulationsmodell, das ausführbare Modell. Diese Phase ist maßgeblich vom verwendeten Simulationswerkzeug abhängig, da das Simulationsmodell beispielsweise mit Hilfe von vordefinierten Bausteinen oder einer Programmiersprache erstellt wird. Die Systemanalyse, die Modellformalisierung und die Implementierung ergeben zusammen die Modellbildung (Wenzel et al. 2008 S. 125).

Parallel zur Modellbildung finden nach Gutenschwager et al. (2017, S. 145) die Datenbeschaffung und -aufbereitung statt. In der Phase der Datenbeschaffung werden jegliche Daten, die in der Simulationsstudie benötigt werden, beschafft. Quellen für diese Daten können zum Beispiel die Datenbestände von Unternehmen darstellen. Sind benötigte Daten nicht vorhanden, so müssen sie in dieser Phase erhoben werden. Sämtliche Daten werden als Rohdaten, dem Phasenergebnis der Datenbeschaffung, zusammengestellt. Die Rohdaten werden in der Phase der Datenaufbereitung für die Nutzung im ausführbaren Modell aufbereitet. In der Aufbereitung werden zum Beispiel die relevanten Daten aus allen Rohdaten herausgefiltert oder statistische Verteilungen der Rohdaten ermittelt. Die so aufbereiteten Daten stellen das Ergebnis dieser Phase dar und können in der Experimentedurchführung, der nächsten Phase des Vorgehensmodells, verwendet werden. In der Experimentedurchführung werden die aufbereiteten Daten mit dem ausführbaren Modell zusammengeführt und die Experimente nach einem

Experimenteplan ausgeführt. Die dabei entstehenden Ergebnisse sind aufzuzeichnen und zu analysieren. Die Analyse kann Aufschluss über Schlussfolgerungen für das zugrundeliegende System geben. In den Simulationsergebnissen werden die Schlussfolgerungen, die Daten und die Analyse der Daten festgehalten.

3.3 Datenbeschaffung und -aufbereitung

Arnold et al. (2018) weisen darauf hin, dass die akkurate Bereitstellung von Eingangsdaten für die Simulation zu den häufigsten Hindernissen in Simulationsstudien gehören. Für Gißrau und Gereke (2017) ist die Datenerhebung in Simulationsstudien ein großes Problem. Daher ist der Aufwand für die Datenbeschaffung für Wenzel (2018) nicht zu unterschätzen, denn dieser ist von der Komplexität und dem Detailierungsgrad der benötigten Daten abhängig. Die benötigten Daten ergeben sich wiederum aus dem Untersuchungsziel, das in der Aufgabenspezifikation definiert wird. Des Weiteren geben Gutenschwager et al. (2017, S. 158) zu bedenken, dass für Aufträge, die als Eingangsdaten in der Simulation genutzt werden, für die mangels Vergangenheitsdaten aber keine Daten vorliegen, im Konzeptmodell eine entsprechende Generierung definiert werden muss. Robinson (2014, S. 121 f.) teilt die Daten anhand ihrer Verfügbarkeit und Beschaffbarkeit in drei Kategorien ein. Die Kategorie A beinhaltet Daten, die verfügbar sind, weil sie bekannt sind oder schon erhoben wurden. In Kategorie B befinden sich Daten, welche momentan nicht verfügbar sind, aber erhoben werden können. Die Daten in Kategorie C hingegen sind weder verfügbar noch können sie durch den Simulationsexperten in einer Datenerhebung gesammelt werden.

Die Datenerhebung wird nach Gutenschwager et al. (2017, S. 160 f.) in zwei Kategorien eingeteilt. Bei der Sekundärerhebung nutzt der Simulationsexperte die bestehenden Datenquellen des Unternehmens. Werden die Daten spezifisch für die Simulationsstudie erhoben, wird dies als Primärerhebung bezeichnet. Beide Formen der Datenerhebung können auf eine Vielzahl an Methoden zurückgreifen, um an die benötigten Daten zu gelangen. Diese Methoden werden von Gutenschwager et al. (2017, S. 165-168) erläutert, sind für die vorliegende Arbeit aber nicht relevant und werden aus diesem Grund nicht weiter thematisiert. Für die Datenerhebung können laut Gutenschwager et al. (2017, S. 161 f.) eine Vielzahl an Datenquellen herangezogen werden, um den Datenbedarf der Simulationsstudie zu decken. Eine von den Autoren als wesentlich bezeichnete Quelle sind Systeme der Produktionsplanung und -steuerung, die unter anderem die Stammdaten

des Unternehmens enthalten. Weitere Quellen sind in Gutenschwager et al. (2017, S. 161 - 165) zu finden, werden hier aber nicht weiter erläutert, da sie für die vorliegende Arbeit nicht relevant sind. Die benötigten Daten können, wie eingangs erwähnt, der Aufgabenspezifikation entnommen werden. Deshalb ist es notwendig, dass der Simulationsexperte möglichst früh die für das Simulationsmodell benötigten Daten identifiziert (Kelton et al. 2015, S. 180). Im Folgenden werden einige Beispiele dieser Daten gegeben. Die benötigten Daten können zum Beispiel Produktionsdaten oder Transportaufträge aus der Kategorie der Systemlastdaten sein VDI - Richtlinie 3633 Blatt 1 (2014, S. 34). Weitere Daten sind nach van Bonn (2013) die Kundenstammdaten, welche die Lieferanschrift des Kunden mit Land, Ort, Postleitzahl, Straße und Hausnummer beinhaltet. Zudem werden die Adressinformationen für Lager oder Depots in den Standorten vorgehalten. Beispielhafte Daten für ein Tourenplanungsproblem sind unter anderem die Standorte der Depots, Lager und Kunden (Pfeilsticker et al. 2015). Für die Modellierung einer Distributionsstruktur werden nach Schulte (2017, S: 707) unter anderem aus den Kundendaten die Kundenstandorte, aus den Lagerdaten die Lagerstandorte sowie die Transportkosten aus den Transportdaten benötigt. Bei der Datenbeschaffung ist darauf zu achten, dass dem Simulationsexperte nicht der Fehler unterläuft, mehr Daten aufzunehmen als notwendig, da Quantität nicht zu einer höheren Qualität der Simulationsstudie führt, sondern nur den Aufwand für die Beschaffung und Aufbereitung erhöht (Wenzel und Bernhard 2008). Ist die Datenerfassung abgeschlossen, liegen die Daten als Rohdaten vor (Gutenschwager et al. 2017, S. 161).

Die Rohdaten aus der Datenerfassung müssen für die Simulationsstudie nach Gutenschwager et al. (2017, S. 168 - 171) entsprechend aufbereitet werden und in eine für die Simulationsstudie nutzbare Form überführt werden. Dabei werden zum Beispiel falsche oder fehlerhafte Einträge bereinigt. Außerdem sind die Werte und Definitionsbereiche der Variablen zu überprüfen, um beispielsweise auszuschließen, dass die Werte reeller Variablen nur durch natürliche Variablenwerte ausgedrückt werden. Des Weiteren ist die Granularität der Daten zu untersuchen. Das bedeutet, dass eine Überprüfung der Variablen stattfindet, um zwei verschiedene Maßeinheiten, zum Beispiel Minuten und Sekunden, für eine Variable auszuschließen. Auch Daten, die durch Wahrscheinlichkeitsverteilungen erzeugt wurden, müssen auf Vollständigkeit und Widersprüche hin überprüft werden. Häufig wird diese Überprüfung laut den Autoren mittels der Verifikations- und Validierungstechniken des Schreibtischtests oder des Strukturierten Durchgehens durchgeführt.

Für den Simulationsexperten kann es unmöglich sein, diese Daten zu sammeln, da beispielsweise die untersuchten Prozesse nicht existieren oder Gesetze die Datensammlung verhindern oder erschweren (Banks et al. 2014a, S. 331). Daten die hierunter fallen wurden am Beginn dieses Abschnitts als Daten der Kategorie C bezeichnet. Ein Beispiel für ein Gesetz gegen die Sammlung von Daten stellt die am 27. April 2016 in Kraft getretene Datenschutz-Grundverordnung (Verordnung EU 2016) dar. In dieser wird die Nutzung aller Daten, durch die eine Person direkt oder indirekt identifiziert werden kann, reglementiert. Diese Daten werden auch als personenbezogene Daten bezeichnet. Unter die personenbezogenen Daten fallen auch Standortdaten. Unternehmen müssen bei der Verwendung dieser Daten nachweisen können, dass der Kunde dem Zweck dieser Verarbeitung explizit eingewilligt hat. Darüber hinaus muss der Kunde unter anderem über den Zweck der Verarbeitung, weitere Empfänger seiner Daten sowie der Dauer der Datenspeicherung informiert werden. Ändern sich diese Angaben, muss erneut die explizite Zustimmung des Kunden eingeholt werden. Personenbezogene Daten können darüber hinaus auch einem öffentlichen Zweck dienend genutzt werden, so etwa in der Forschung. Dabei ist keine explizite Zustimmung der betroffenen Personen notwendig, die Daten sind aber soweit zu anonymisieren und der zugriffsberechtigte Personenkreis einzuschränken, wie es der Zweck der Verwendung ermöglicht. Außerdem gilt es nach der Europäischen Kommission (2019) zu beachten, dass Einverständniserklärungen, die vor dem 25.5.2018 von Kunden eingeholt wurden und nicht mit den Regeln der Datenschutz-Grundverordnung übereinstimmen, ihre Wirkung verlieren. Somit müssen betroffene Unternehmen jeden einzelnen ihrer Kunden kontaktieren und deren explizites Einverständnis, im Einklang mit der Datenschutz-Grundverordnung, einholen, um weiterhin die Daten ihrer Kunden verarbeiten zu dürfen. So gibt beispielsweise Umundum (2015) an, dass die Österreichische Post Kundendaten anonym auswertet. Dabei können die Kunden wählen, ob ihre Daten für die Optimierung der Zustellung eingesetzt werden. Vor dem Hintergrund der angesprochenen Anonymisierung der Daten ist es zudem erwähnenswert, dass Rocher et al. (2019) in ihrer Veröffentlichung nur 15 demografische Attribute benötigen, um 99,98 Prozent aller Amerikaner in jeglichen, selbst in unvollständigen, Datensätzen zu re-identifizieren. Für eine 78,7 prozentige Wahrscheinlichkeit eine Person zu identifizieren reichen gemäß der Forscher fünf Attribute aus. Dies lässt die Autoren daran zweifeln, dass die momentan verwendeten Verfahren zur Anonymisierung von Daten vor dem Hintergrund der Datenschutz-Grundverordnung ausreichend sind.

Zur Erhebung fehlender Daten empfehlen Gutenschwager et al (2017, S. 166) mit Experten des Prozesses zusammen zu arbeiten und diese einen typischen, einen minimalen und einen maximalen Wert schätzen zu lassen. Mit diesen Angaben kann dann eine Dreiecksverteilung erstellt werden. Können die Daten nicht mit Hilfe von Experten ermittelt werden, schlagen Banks et al. (2014a, S. 363 ff.) die Nutzung von Wahrscheinlichkeitsverteilungen vor. Namentlich nennen die Autoren die Gleichverteilung, die Dreiecksverteilung und die Beta-Verteilung. Nach Robinson (2014, S. 133f.) wird unter anderem die Gleichverteilung genutzt, um Daten anzunähern, wenn diese Daten nicht beschaffbar sind. Kelton et al. (2015, S. 190 ff.) empfehlen für den Fall, dass keine Daten vorliegen, entweder deterministische Werte festzulegen, deren Effekte auf das Ergebnis der Simulation mit Sensitivitätsanalysen untersucht werden, oder Wahrscheinlichkeitsverteilungen wie die Exponential-, Dreiecks- und Gleichverteilung zu nutzen.

Die Entscheidung, wie mit zufälligen Daten zu verfahren ist, erweist sich als bedeutsam, da zum Beispiel nach Sörensen (2008) die Standortentscheidung für ein Distributionszentrum eine wichtige und kostenintensive Entscheidung darstellt, die schwer rückgängig gemacht werden kann. Dabei basiert diese Entscheidung auf volatilen Informationen, wie Kundenstandorten, und es ist deswegen besonders wichtig, dass diese Unsicherheit der Informationen in die Entscheidung für einen Standort einbezogen wird. Zudem ist es sehr schwierig abzuschätzen, wie sich die Simulationsergebnisse vom realen System unterscheiden, wenn aus diesem keine Vergleichsdaten vorliegen (Wenzel et al. 2008, S. 120). So sollten die Daten für eine geplante Anlage sehr vorsichtig abgeschätzt werden (Wenzel 2018).

Außerdem sollte bei der Datenauswahl auf die Qualität der Eingangsdaten geachtet werden, denn diese „(...) ist von essentieller Bedeutung für den Erfolg einer Simulationsstudie“ (Wenzel et al. 2008, S. 119). Es ist aber nicht möglich die benötigte Qualität der Eingangsdaten unabhängig vom Anwendungsszenario oder Nutzer der Daten zu betrachten (Wand und Wang 1996). Denn verschiedene Nutzer werden die gesammelten Daten unterschiedlich beurteilen (Cappiello et al. 2004). Die unterschiedliche Beurteilung ist laut Strong et al. (1997) dadurch begründet, dass die Nutzer die Qualität der Daten im Kontext ihrer zu bearbeitenden Aufgabe beurteilen. Außerdem können die gleichen Daten zu unterschiedlichen Zeitpunkten in verschiedenen Prozessen benötigt werden. Deswegen ist es aus der Sicht der Autoren wichtig, dass im

Zusammenhang der Datenqualität die verschiedenen Nutzer der Daten nicht außen vorgelassen werden und die Datenqualität nicht nur im Kontext der Spezifikationen betrachtet wird, sondern auch für alle Nutzer in der benötigten Bandbreite zur Verfügung stehen. Dies gilt besonders vor dem Hintergrund, dass beispielsweise nicht alle Simulationen eine hohe Genauigkeit der Daten benötigen (Skoogh und Johansson 2008). Aus diesem Grund müssen Daten immer vor dem Hintergrund des Untersuchungsziels evaluiert werden (Williams und Ülgen 2012).

So verwendet Thonemann (2015, S. 103) zur Darstellung von Lager- und Kundenstandorten ein x-y-Koordinatensystem. Das Simulationsprogramm AnyLogic (The AnyLogic Company 2019) benötigt für die Darstellung von Standorten mit Geo-Koordinaten nach AnyLogic(2019) die Längen- und Breitengrade in Gradzahlen. Dabei gilt es zu beachten, dass sich die Breitengrade von -90° (Süd) über 0° bis $+90^\circ$ (Nord) und die Längengrade von -180° (West) über 0° bis $+180^\circ$ (Ost) erstrecken. Ebenfalls benötigt das Simulationsprogramm SimChain (SimPlanAG 2019) diese Darstellung der Koordinaten (SimChain 2014, S. 6). Die Umrechnung der Koordinaten der Stadt Madrid, mit $40^\circ 29'$ Nord und $3^\circ 55'$ West, in Gradzahlen ergibt nach Mattfeld und Vahrenkamp (2014, S.20) einen Breitgrade von 40,29 und einen Längengrad von -3,55.

Weitere Ausführungen zur benötigten Datenqualität führen über das Ziel der Arbeit hinaus. Eine Vertiefung dieses Themas kann in der Veröffentlichung von Bokrantz et al. (2018) gefunden werden.

3.4 Analyse von Simulationsergebnissen

In der Experimentedurchführung werden die Ergebnisse, die aus den Experimenten resultieren, aufgezeichnet und ausgewertet (vgl. Abschnitt 3.2). Für die Auswertung empfehlen Gutenschwager et al. (2017, S. 193) die Verwendung von Kennzahlen, wobei es vom Ziel der Simulationsstudie abhängig ist, welche Kennzahlen verwendet werden. Law (2015, S. 497) unterscheidet die statistische Auswertung von Simulation danach, ob sie der Gruppe der terminierenden oder nicht-terminierenden Simulationen angehören. Da die Simulation in der vorliegenden Arbeit der terminierenden Simulation angehört (vgl. Abschnitt 3.1), werden nur Auswertungsverfahren für diese Gruppe vorgestellt. Zur Ansicht von Auswertungsverfahren für nicht-terminierende Simulationen kann beispielsweise Law (2015, S. 511 – 542) herangezogen werden.

Die Auswertungsverfahren differenzieren Gutenschwager et al. (2017, S. 193 - 197) zum einen in die Auswertung einer Parameterkonfiguration und zum anderen in den Vergleich verschiedener Parameterkonfigurationen. Für die Auswertung einer einzigen Parameterkonfiguration mit verschiedenen Replikationen ist unter anderem der Mittelwert einer verwendeten Kennzahl heranzuziehen. Des Weiteren sind die Standardabweichung sowie der minimale bzw. der maximale Wert der Kennzahl aufzunehmen. Durch die Auswertung dieser Streuungs- und Lagemaße kann der Simulationsexperte herausfinden, wie stark die Simulationsergebnisse schwanken. Der Mittelwert, genauer gesprochen der arithmetische Mittelwert \bar{X} , wird nach Formel (1) berechnet (Kosfeld et al. 2016, S. 79).

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

Weiter führen Gutenschwager et al. (2017, S. 193 - 197) aus, dass in einer Simulationsstudie, in der mehrere, beispielsweise zwei, Parameterkonfigurationen miteinander zu vergleichen sind, es nicht ausreicht, die Mittelwerte beider Konfigurationen einander gegenüberzustellen. Für eine statistisch abgesicherte Auswertung muss untersucht werden, ob sich aus den unterschiedlichen Parameterkonfigurationen signifikant unterschiedliche Ergebnisse ergeben. Hierfür kann auf Konfidenzintervalle zurückgegriffen und untersucht werden, ob sich die beiden Intervalle, unter der Berücksichtigung der Irrtumswahrscheinlichkeit, überschneiden. Überschneiden sich beide Konfidenzintervalle nicht, so ist der Erwartungswert für die eine Kennzahl höher als für die andere. Die Aussagekraft hiervon hängt von der verwendeten Kennzahl ab. Wird zum Beispiel die Kennzahl „Durchsatz“ ermittelt, ist ein höherer Erwartungswert besser. Überschneiden sich die Konfidenzintervalle jedoch, kann die vorangegangene Aussage nicht getroffen werden und die Ergebnisse müssen mit einer anderen Methode ausgewertet werden.

In Produktions- und Logistiksystemen ist nicht davon auszugehen, dass die Ausgangsgrößen, also auch die aufgenommenen Kennzahlen, der Standardnormalverteilung unterliegen (Gutenschwager et al. 2017, S. 115). Für diesen Fall müssen nach Stocker und Steinke (2017, S.469 - 479) approximative Konfidenzintervalle verwendet werden. Allgemein wird mit einem Konfidenzintervall die zu schätzende Zufallsgröße mit einer gewissen Sicherheitswahrscheinlichkeit, auch Konfidenzniveau $1 - \alpha$ genannt, überdeckt. Ein Konfidenzniveau von beispielsweise 95

% entspricht einer Irrtumswahrscheinlichkeit α von 5 %. Zur Berechnung eines Konfidenzintervalls werden n Ausprägungen der voneinander unabhängigen und identisch verteilten Zufallsgrößen genutzt (Gutenschwager et al. 2017, S. 114). In der vorliegenden Arbeit werden die Zufallsgrößen durch die aufgenommenen Kennzahlen ausgedrückt. Mit approximativen Konfidenzintervallen wird das vorgegebene Niveau $1-\alpha$ nicht erreicht. Jedoch wird die Approximation nach dem Gesetz der großen Zahlen (Law, 2015, S. 233) genauer, je größer die Stichprobengröße n der zu approximierenden Zufallsgröße X ist. Mindestens sollte $n > 30$ sein. Die linke und rechte Grenze des Konfidenzintervalls werden dann mit Hilfe von Formel (2) berechnet. In Formel (2) bezeichnet S^2 die korrigierte empirische Varianz, die in Formel (3) zu finden ist (Lange und Mosler 2017, S. 70). Die Standardabweichung S lässt sich durch Ziehen der Quadratwurzel aus S^2 berechnen (Kosfeld et al. 2016, S. 120).

$$\left[\bar{X} - z_{1-\alpha/2} * \sqrt{\frac{S^2}{n}}, \bar{X} + z_{1-\alpha/2} * \sqrt{\frac{S^2}{n}} \right] \quad (2)$$

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2 \quad (3)$$

Ist $n > 30$ können die Werte für $z_{1-\alpha/2}$ aus Wertetabellen für die Standardnormalverteilung entnommen werden, denn für große n ist das standardisierte Stichprobenmittel approximativ standardnormalverteilt (Stocker und Steinke 2017, S. 477). Tabellen für diese Werte sind zum Beispiel bei Stocker und Steinke (2017, S.678) zu finden. Ist ein Konfidenzintervall berechnet, so kann ausgesagt werden, dass der Erwartungswert der Zufallsgröße mit dem gewählten Konfidenzniveau vom berechneten Konfidenzintervall überdeckt wird (Law 2015, S. 234).

Law (2015, S. 560 - 565) empfiehlt für den Vergleich von zwei Parameterkonfigurationen die paarweise Auswertung aller Einzelergebnisse x_i und y_i , auch Paired-t-Konfidenzintervall genannt. Dazu wird zuerst für jede Replikation i $z_i = x_i - y_i$ berechnet und mit den z-Werten ein neues Konfidenzintervall gebildet. Ist die linke Grenze in dem neuen Konfidenzintervall größer oder die rechte Grenze < 0 , so unterscheiden sich die Erwartungswerte der beiden Parameterkonfigurationen um den angegebenen Wert. Sind beide Werte des Konfidenzintervalls > 0 , gibt die linke Grenze den Wert an, mit dem sich die Konfiguration x_i mindestens von y_i unterscheidet. Befindet sich die 0 im Konfidenzintervall, kann nur die Aussage getroffen werden, dass keine der

Parameterkonfigurationen der anderen überlegen ist. Um einer 0 im Konfidenzintervall oder sich überschneidenden Konfidenzintervallen vorzubeugen, empfehlen Gutenschwager et al. (2017, S. 198) weitere Replikationen durchzuführen.

Bei der Visualisierung der Ergebnisse ist nach Gutenschwager et al. (2017, S. 199 - 202) darauf zu achten, dass die Ergebnisse unverfälscht wiedergegeben werden. Dies wird auch Expressivität genannt. Des Weiteren soll der Betrachter die Aussage, die die Visualisierung unterstützen soll, verstehen, was auch Effektivität genannt wird. Deswegen sollte die Visualisierung intuitiv, beispielsweise unterstützt durch eine Legende, verstanden werden können. Als letztes muss der Simulationsexperte die Angemessenheit seiner Visualisierung beachten. Hierbei darf der Aufwand bei der Erstellung und der Interpretation der Visualisierung den Nutzen nicht übersteigen. Die bekannteste Visualisierung ist die Darstellung der Simulationsergebnisse in einer Tabelle. Darüber hinaus können Box-Plot-Diagramme und Diagramme zur vergleichenden Darstellung von beispielsweise unterschiedlichen Konfidenzintervallen eingesetzt werden.

Zum Vergleich einzelner absoluter Häufigkeiten, also wie häufig das zu vergleichende Merkmal X auftritt (Lange und Mosler 2017, S. 6), können nach Stocker und Steinke (2017, S. 33) Säulendiagramme eingesetzt werden. Sind diese Säulen horizontal aufgetragen, wird das Diagramm Balkendiagramm genannt. In Balkendiagrammen werden die Häufigkeiten ihrer Größe nach geordnet, um so einzelne Kategorien hervorzuheben.

Zur Konstruktion von Box-Plot-Diagrammen werden nach Stocker und Steinke 2017 (S. 40 f.) der Median, der Mittelwert, der minimale- und der maximale Wert sowie das obere und untere Quartil benötigt. Ein Box-Plot besteht generell aus der Box und den Whiskers, die an jeder kurzen Seite der Box angebracht sind. Die Darstellung des Box-Plots kann sowohl horizontal als auch vertikal erfolgen. Somit befindet sich die Messskala entweder neben oder unter der Box. Die Werte, die dem Box-Plot zu Grunde liegen sollen, müssen zu Beginn nach ihrer Größe aufsteigend geordnet werden. Weiterhin hängt die Lage der Box von der Größe des Werts des 1. und 3. Quartil ab. Beide Werte der Quartile werden mit Hilfe von Formel (4) berechnet, indem für α entweder 0,25, für das 1. Quartil, oder 0,75, entsprechend für das 3. Quartil, eingesetzt und der Wert des Quartils in der geordneten Liste abgelesen wird (Lange und Mosler 2017, S. 8 f.). Die oberen Gaußklammern $\lceil \cdot \rceil$ werden in Formel (4) eingesetzt, um auf die nächste ganze Zahl

aufzurunden (Groß 2010, S. 34). Die Länge der Box wird durch den Interquartilsabstand dQ , durch Formel (5) berechnet, bestimmt. Innerhalb der Box befinden sich somit 50 Prozent der Werte und außerhalb der Box auf jeder Seite jeweils 25 Prozent. Des Weiteren wird die Box an der Stelle des Medians mit einer Linie geteilt. Der Median teilt nach Groß (2015, S. 55) die geordneten Werte in zwei Teile. Der eine Teil ist 50 Prozent größer als der Median, der andere Teil 50 Prozent kleiner. Meist weicht der arithmetische Mittelwert deutlich vom Median ab. Daraus lässt sich ableiten, dass einige verhältnismäßig größere oder kleinere Ausreißer existieren, da sich der arithmetische Mittelwert unverhältnismäßig stark in die Richtung der Ausreißer orientiert (Stocker und Steinke 2017, S. 55).

$$q_{\alpha} = \begin{cases} X_{[\alpha * n]} & \text{falls } \alpha * n \neq \mathbb{N} \\ X_{(\alpha * n)} & \text{falls } \alpha * n = \mathbb{N} \end{cases} \quad (4)$$

$$dQ = q_{0,75} - q_{0,25} \quad (5)$$

$$\text{Median} = \begin{cases} X_{\left(\frac{x+1}{2}\right)} & \text{für } n \text{ ungerade} \\ \frac{1}{2} \left[X_{\frac{n}{2}} + X_{\left(\frac{n}{2}+1\right)} \right] & \text{für } n \text{ gerade} \end{cases} \quad (6)$$

Weiter führen Stocker und Steinke (2017, S. 39 - 43) aus, dass die Länge der Whisker des Box-Plots unterschiedlich berechnet wird. Die verbreitetste Art der Berechnung ist es, den Interquartilsabstand mit 1,5 zu multiplizieren und dann vom Wert des 1. Quartils abzuziehen, beziehungsweise auf den Wert des 3. Quartils zu addieren. Dabei wird der kleinere Wert auch unterer Zaun, der größere korrespondierend oberer Zaun genannt. Alle Werte, die unterhalb des unteren Zauns oder oberhalb des oberen Zauns liegen, werden Ausreißer genannt. Box-Plots eignen sich besonders für die Auswertung eines Merkmals für unterschiedliche Kategorien. Die Autoren führen als Beispiel die Verteilung der monatlichen Renditen verschiedener Aktienindizes an. Bei der Auswertung des Box-Plot-Diagramms kann die Lage des Medians innerhalb der Box aufschlussreich sein. Liegt der Median in der Mitte der Box, deutet dies auf eine im Mittel symmetrische Verteilung hin. Befindet sich der Median links vom Zentrum, deutet sich eine rechtsschiefe Verteilung an. Wohingegen die Lage des Medians rechts vom Zentrum, die Vermutung einer linksschiefen Verteilung zulässt. Dabei gibt die Schiefe nach Stocker und Steinke (2017, S. 49 -60) an, ob die Verteilung einer Wertereihe Abweichungen zur Symmetrie aufweist. Dabei bedeutet linksschief, dass rechts mehr Werte vorliegen als links. Bei der Rechtsschiefe ist dies umgekehrt. Die Schiefe einer Verteilung kann auch mit Hilfe des

Medians und des Mittelwerts bestimmt werden. Sind die Werte symmetrisch verteilt sind Median und Mittelwert gleich groß. Bei einer linksschiefen Verteilung der Werte ist der Median größer als der arithmetische Mittelwert, bei einer rechtsschiefen Verteilung ist dies umgekehrt.

4. Überblick über Verfahren zur Generierung von Standortdaten

In diesem Kapitel werden die Verfahren zur Generierung von Kundenstandorten vorgestellt. Dazu erfolgt in Abschnitt 4.1 eine Übersicht über Verfahren, die in wissenschaftlichen Veröffentlichungen genutzt wurden, um Standorte zu generieren. Weiterführend wird die Standortgenerierung unter Berücksichtigung von Realdaten in Abschnitt 4.2 erläutert. Für die Generierung von Zufallsvariablen mit Hilfe von Verteilungsfunktionen wird eine Einführung in die Zufallszahlen benötigt, diese wird in Abschnitt 4.3 getätigt. In Abschnitt 4.4 wird die Generierung von Zufallsvariablen durch eine Gleich-, eine Normal- sowie eine Dreiecksverteilung erläutert.

4.1 Übersicht über genutzte Verfahren der Standortgenerierung

In wissenschaftlichen Veröffentlichungen werden Wahrscheinlichkeitsverteilungen oder Realdaten zur Generierung von Standorten in einer Vielzahl von Themenbereichen eingesetzt. Für einen angemessenen Überblick zu den Veröffentlichungen zu diesem Thema, die sich aus der Literaturrecherche zu dieser Arbeit ergeben haben, sind diese in Tabelle 1 bis Tabelle 4 aufgeführt. Die Quellen sind entsprechend ihrer Themenbereiche in die Tabellen eingeordnet. Dabei sind die Tabellen dreigeteilt. In der ersten Spalte werden die Autoren genannt, in der mittleren Spalte sind die eingesetzten Verfahren aufgeführt und die letzte Spalte beinhaltet Hinweise zu den Verfahren. Unter Verfahren werden in den Tabellen die Generierung der Kundenstandorte mit Hilfe von Wahrscheinlichkeitsfunktionen oder Realdaten eingeordnet. Auch wird die Standortgenerierung in einigen Veröffentlichungen als zufällig bezeichnet, ohne auf das zugrunde liegende Verfahren einzugehen oder dieses zu benennen. Um dies zu kennzeichnen, wird bei den betreffenden Veröffentlichungen in der Tabellenspalte Verfahren „Zufällig“ eingetragen. In der Spalte Hinweise werden unter anderem der Name einer Software, wenn diese für die Generierung der Standorte genutzt wurde, das Intervall, in dem die Parameter für die genutzte Wahrscheinlichkeitsfunktion liegen, oder die verwendeten Koordinatensysteme angegeben. Werden keine Hinweise zu der Veröffentlichung gegeben, wird in der Spalte Hinweise „-“ eingetragen. Die Auflistung

bezweckt einen Überblick über die Verfahren und erhebt deswegen keinen Anspruch auf Vollständigkeit.

Die Generierung zufälliger Standorte wird unter anderem in der Untersuchung von Fragestellungen im Themenbereich der Kommunikationsnetzwerke genutzt. Die hierzu gefundene Literatur ist in Tabelle 1 aufgelistet. Exemplarisch wird aus zwei Literaturquellen die Nutzung der Standortgenerierung erläutert. Sun und Samaan (2015) untersuchen die Auswirkungen eines Knotenausfalls auf die Fähigkeit eines Funknetzwerks den Kunden weiterhin eine Internetverbindung zu ermöglichen. Dazu werden die Knoten des Netzwerks so generiert, dass ihre Verteilung im zweidimensionalen Raum mit einer stetigen Gleichverteilung sowie einer Poisson-Verteilung übereinstimmt. Zarimpas et al. (2007) generieren die Breiten- und Längengrade von Kundenstandorten so, dass sie eine stetige Gleichverteilung abbilden. Weitere Anwendungsfälle sind Tabelle 1 zu entnehmen.

Tabelle 1: Verfahren zur Standortgenerierung in Kommunikationsnetzwerken

Autor(en)	Verfahren	Hinweise
Bettstetter et al. 2004	Stetige Gleichverteilung	1000 x 1000 m ²
Samarasinghe et al. 2013	Poisson Prozess	Lambda
Sboui et al. 2016	Zufällig	Koordinatensystem (x; y)
Sun und Samaan 2015	Stetige Gleichverteilung Poisson-Verteilung	Rechteck 20 x 20, Lambda der Poisson-Verteilung 1,0 bis 6,0
Zarimpas et al. 2007	Stetige Gleichverteilung	Breiten- (Lat) und Längengrad (Lon)

Des Weiteren ist die Nutzung zufallsgenerierter Standorte bei Forschungen in Themenbereichen, die der Flora und Fauna zugeordnet werden können, verbreitet (siehe Tabelle 2). In diesem Themengebiet wird diese Nutzung exemplarisch anhand einer Literaturquelle vorgestellt. Zum Beispiel nutzen Chhetri et al. (2018) zufallsgenerierte Standorte, um die Genauigkeit ihres entworfenen Modells zur möglichen Ausweitung der subalpinen Waldgrenze zu überprüfen. Die zufälligen Standorte werden von den Autoren mit Hilfe der proprietären Software ArcGIS (ArcGIS 2019) erzeugt. Es werden keine Angaben zu den verwendeten Formeln gemacht, mit denen die Software die zufälligen Standorte berechnet.

Tabelle 2: Verfahren zur Standortgenerierung in Flora und Fauna

Autor(en)	Verfahren	Hinweise
Chhetri et al. 2018	Zufällig	ArcGIS
Deb et al. 2018	Zufällig	-
Foley et al. 2012	Zufällig	ArcGIS
Legge et al. 2013	Zufällig	ArcGIS
Richard & Côté 2016	Zufällig	-

In Tabelle 3 sind beispielhafte Anwendungen der Standortgenerierung aufgelistet, welche dem Themenbereich der Standortplanung zugeordnet werden. Eine Literaturquelle wurde diesem Themenbereich zugeordnet, wenn in dieser, der Definition von ten Hompel und Heidenblut (2011, S. 292) folgend, anhand von Standortfaktoren geografisch Standorte geplant wurden. Aus diesem Themenbereich werden exemplarisch drei Literaturquellen vorgestellt. Yousefli et al. (2018) entwickeln ein Modell für das Transfer Punkt Problem, in welchem die Kundenstandorte so erzeugt werden, dass sie eine Normalverteilung abbilden. Die Autoren geben an, dass in ihrem Modell jede Wahrscheinlichkeitsfunktion abgebildet werden kann, testen dieses jedoch nur anhand einer Normalverteilung. Davoodi et al. (2011) entwickeln einen Algorithmus für das P-Center-Problem und testen diesen mit gleichverteilten Nachfragepunkten, welche in einem Rechteck der Dimension 450 x 550, ohne Angabe einer Einheit, erzeugt werden. Post et al. (2018) generieren die x- und y-Koordinaten der Kundenstandorte anhand einer Gleichverteilung. Dabei startet die Generierung im Intervall (0; 50) Kilometer und wird für jedes weitere Experiment um 50 Kilometer erhöht, bis sich das Intervall (0; 300) Kilometer ergibt. Tabelle 35 im Anhang listet weitere Anwendungsfälle aus der Standortplanung auf.

Tabelle 3: Verfahren zur Standortgenerierung in der Standortplanung

Autor(en)	Verfahren	Hinweise
Yousefli et al. 2018	Normalverteilung	-
Kalantari et al. 2014	Zufällig	-
Davoodi et al. 2011	Gleichverteilung	Verteilt auf Rechteck der Dimension 450 x 550
Bertazzi und Maggioni 2015	Stetige Gleichverteilung	-
Post et al. 2018	Stetige Gleichverteilung	Verteilung auf 300 x 300 in Kilometern

Einen Überblick über die verwendeten Generierungsverfahren im Zusammenhang mit dem Themenbereich der Last Mile ermöglicht Tabelle 4. Boysen et al. (2016) untersuchen die Belieferung mit Drohnen auf der Last Mile. Die Drohnen starten ihre Touren von LKWs aus. Die Haltepunkte der LKWs erzeugen die Autoren so, dass eine Gleichverteilung der Koordinaten x und y im Intervall $(-1000; 1000)$ entsteht. Köhler und Haferkamp (2019) evaluieren Auftragsannahmeverfahren und deren Einfluss auf die Lieferkosten bei Hauslieferungen in der Last Mile unter anderem im Großraum der Stadt Hamburg. Die Kundenstandorte in Hamburg werden so generiert, dass eine Gleichverteilung abgebildet wird. Franz (2018) erzeugt im Intervall $(0; 30)$ gleichverteilte ganzzahlige x - und y -Koordinaten für die Depot- und Kundenstandorte. Der Autor gibt an, dass dieses Intervall bei einer Betrachtung in der Längeneinheit (LE) Kilometer sich zu 900 Quadratkilometer multipliziert und damit ungefähr der Fläche Berlins entspricht. Weitergehend wird der Einfluss zufälliger Standorte auf die Länge der Fahrstrecke untersucht. Die untersuchte Literaturquelle stellt die einzige Quelle dar, die im Rahmen der vorliegenden Arbeit gefunden wurde und sich mit der Auswirkung zufälliger Kundenstandorte auf die Länge der Fahrstrecke beschäftigt. Da die Dissertation aber gleichzeitig das Standort- und Tourenplanungsproblem behandelt, die Standorte der Depots also ebenfalls verschoben werden und sowohl die Anzahl als auch die Standorte der Depots von den Kundenstandorten abhängig sind, können die Ergebnisse nicht auf diese Arbeit übertragen werden. Die gleichzeitige Behandlung des Standort- und Tourenplanungsproblem wird auch Location Routing Problem genannt und stellt einen Spezialfall der Tourenplanungsprobleme dar (Irnich et al. 2014, S. 22). In Tabelle 36 im Anhang sind weitere Anwendungsfälle aus der Last Mile zu finden.

Tabelle 4: Verfahren zur Standortgenerierung in der Last Mile

Autor(en)	Verfahren	Hinweise
Boysen et al. 2016	Gleichverteilung	Koordinatensystem $(x; y)$, Verteilt auf $(-1000; 1000)$
Köhler und Haferkamp 2019	Gleichverteilung mit Realdaten	-
Franz 2018	Gleichverteilung	Koordinatensystem $(x; y)$, Verteilt auf $(0; 30)$
Arnold et al. 2018	Reales Paketaufkommen	-
Cleophas und Ehmke 2014	Gleichverteilung	-
Poeting et al. 2019	Realdaten	-

4.2 Standortgenerierung auf Grundlage realer Daten

Neben der Nutzung von Verteilungsfunktionen werden Standorte auch auf Grundlage realer Daten generiert. Beispielsweise nutzen Köhler und Haferkamp (2019) die Lieferdaten eines Online-Supermarkts, um Kundenstandorte für ihr Modell zu erzeugen. Poeting et al. (2019) nutzen als Basis ihres Modells die öffentlich zugänglichen Daten von Hermes für die Paketbelieferung der Stadt Köln (vgl. Abschnitt 2.4). Die Verteilung der Kundenstandorte erfolgt in dem Modell in zwei Schritten. Zuerst wird der Stadtteil, in dem der Kunde wohnt, über eine stochastische Wahrscheinlichkeit bestimmt. Diese Wahrscheinlichkeit ist als Anteil der Pakettieferungen des jeweiligen Stadtteils am Gesamtpaketvolumen der Stadt Köln aus dem Realdatensatz definiert. Im zweiten Schritt wird der Kundenstandort zufällig im ausgewählten Stadtteil platziert, wobei eine Erläuterung der Zufälligkeit nicht stattfindet. Durch die zufällige Platzierung der Kundenstandorte kann es vorkommen, dass die Standorte unrealistisch platziert werden, zum Beispiel im Rhein. Aus diesem Grund geben die Autoren die Empfehlung, dass eine realistischere Kundenstandortgenerierung im städtischen Kontext entwickelt werden muss.

Arnold et al. (2018) gehen ähnlich wie Poeting et al. (2019) vor (vgl. Abschnitt 2.4). Die Autoren nutzen einen realen Datensatz, in dem alle Kundenstandorte einer dreimonatigen Zeitperiode hinterlegt sind und berechnen mit diesem die geografische Verteilung der Paketnachfrage in Antwerpen. Nach dieser Berechnung wird der realen Datensatz nicht weiterverwendet. Fortfahrend unterteilen sie das städtische Gebiet Antwerpens mit einer Fläche von vier Quadratkilometern in 100 gleich große Distrikte und berechnen für jeden Distrikt, mit Hilfe der berechneten geografischen Verteilung der Paketnachfrage, den Anteil am Gesamtpaketvolumen. Der berechnete Anteil stellt die Wahrscheinlichkeit eines Kundenauftrags aus dem zugehörigen Distrikt dar. Der Kundenstandort innerhalb der Distrikte wird zufällig ausgewählt, wobei zufällig nicht weiter spezifiziert wird. Zusätzlich stoßen die Autoren auf das identische Problem wie Poeting et al. (2019), denn die zufällige Generierung von Kundenstandorten kann zu einer ungültigen Adresse führen, wie zum Beispiel in einem Fluss. Als Lösung wird angeführt, dass diese ungültigen Adressen in der folgenden Tourenplanung der jeweils naheliegendsten Adresse neu zugewiesen werden. Wie diese Zuweisung erfolgt, wird nicht erläutert.

Cardenas et al. (2017b) stellen fest, dass die Anzahl an Pakettieferungen anscheinend mit einer hohen Bevölkerungsdichte korrelieren. Ein Vergleich der Lieferdichte pro

Quadratkilometer zeigt eine doppelte Lieferdichte in städtischen gegenüber ländlichen Gebieten. Bei weiterer Betrachtung dieses Umstands fällt auf, dass die größere Anzahl der Paketlieferungen auf die höhere Bevölkerungsanzahl in Städten zurückzuführen ist, ländliche Gebiete aber mehr Paketlieferungen pro Kopf erhalten. Clarke et al. (2015) stellen ein gesteigertes Interesse der ländlichen Bevölkerung an Bestellungen über das Internet fest. Diese Beobachtungen decken sich auch mit dem Gutachten von Manner-Romberg et al. (2017), welches im Auftrag der Bundesnetzagentur angefertigt wurde. Darin stellen die Autoren fest, dass das Paketvolumen des B2C die Bevölkerungsverteilung in Deutschland widerspiegelt und die bevölkerungsreichsten Großstädte, namentlich Berlin, Hamburg bzw. die Großstadtregion Essen, Mülheim an der Ruhr und Gelsenkirchen, das höchste Paketaufkommen verzeichnen. Bei Betrachtung der Pakete pro Kopf werden diese Großstädte jedoch von Städten in ländlichen Gebieten überholt. Zusätzlich sollte sich nach Cardenas et al (2016) nicht nur auf Wohngebiete fokussiert werden, da viele Konsumenten Pakete zu ihrem Arbeitsplatz bestellen. Cleophas und Ehmke (2014) teilen die Stadt Stuttgart in 9 Liefergebiete ein, deren Größe von der Bevölkerungsdichte abhängig ist. Die Nachfrageverteilung über die Liefergebiete erfolgt anhand der Einkommensverteilung in Stuttgart, wobei die Autoren diese nicht weiter spezifizieren.

4.3 Erzeugung von Zufallszahlen

Die Auswertung der Literatur aus Abschnitt 4.1 hat ergeben, dass zufällige Standorte unter anderem einer Gleichverteilung folgend generiert werden. Außer Franz (2018) gibt keine der ausgewerteten Literaturquellen Auskunft darüber, wie die Gleichverteilung der Standorte erreicht wird. Franz (2018) erzeugt alle zufälligen Datensätze mit der Excel Funktion „Zufallsbereich“. Diese Funktion erzeugt in einem selbst definierten Bereich ganzzahlige Zufallszahlen (Microsoft 2019a).

Zum besseren Verständnis folgt in diesem Abschnitt eine Einführung in die Erzeugung von Zufallszahlen. Nach Gutenschwager et al. (2017, S. 127) unterliegen reale System zufälligen Vorgängen, welche mit Zufallsvariablen beschrieben werden können. Die Abbildung dieser Zufallsvariablen kann mit Hilfe von Verteilungsfunktionen geschehen, wobei die Zufallsvariablen von Gutenschwager et al. (2017, S. 98 f.) in diskrete und stetige Zufallsvariablen eingeteilt werden. Nehmen die Zufallsvariablen endlich oder abzählbar unendlich viele Werte an, werden sie als diskret bezeichnet. Stetig ist eine

Zufallsvariable, wenn sie jeden Wert in einem Intervall $(a; b)$ mit $a, b \in \mathbb{R}$ und $a < b$, annehmen kann. Weiter führen Gutenschwager et al. (2017, S. 127) aus, dass die Darstellung der Zufallsvariablen in Simulationsmodellen durch Zufallszahlen erfolgt. Die Zufallszahlen werden mit Hilfe von mathematisch deterministischen Rechenvorschriften als Zahlenfolge generiert, dem Zufallszahlenstrom. So erzeugte Zufallszahlen werden deterministische Zufallszahlen genannt. Durch die deterministische Berechnung können die Zufallszahlen und damit die Simulationsergebnisse reproduziert werden. Deterministische Zufallszahlen werden auch als Pseudo-Zufallszahlen bezeichnet (Waldmann und Helm 2016, S. 21). Gerlach (2019, S. 162) führt aus, dass der identische Zufallszahlenstrom immer identische Zufallszahlen generieren wird. Um das zu verhindern, kann der Start der Zufallszahlenfolge durch variierende Startwerte (Seeds) modifiziert werden. Die Zufallszahlen werden durch Zufallszahlengeneratoren erzeugt. Dabei ist die Auswahl eines geeigneten Zufallszahlengenerators dem Autor zufolge sehr schwer und es sollten bereits etablierte Zufallszahlengeneratoren verwendet werden.

Neben der deterministischen Berechnung von Zufallszahlen, können Zufallszahlen nach Bierhorst et al. (2018) zudem auf Basis von Quantum Rauschen oder anderen elementaren Quantenmessungen erzeugt werden. Die so erzeugten Zufallszahlen sind unberechenbar. Gabriel et al. (2010) bezeichnen Zufallszahlen, die aus Systemen der Quantenmechanik gewonnen werden, als „ultimative Zufallszahlen“. Weil die so erzeugten Zufallszahlenströme nicht wiederholt werden können, sind diese jedoch für Simulationen ungeeignet (L'Ecuyer 2015). Aus diesem Grund wird auf diese Art der Zufallszahlen im Weiteren nicht eingegangen und der Term Zufallszahl beziehungsweise Zufallszahlen wird im Verlauf der vorliegenden Arbeit ausnahmslos für Zufallszahlen genutzt, die mit Hilfe von deterministischen Rechenvorschriften erzeugt wurden.

Die Zufallszahlen, die mit einem Zufallszahlengenerator erzeugt werden, müssen nach Robinson (2014, S. 37) gleichverteilt und unabhängig sein. Gleichverteilt sagt aus, dass jede Zufallszahl die gleiche Wahrscheinlichkeit besitzt, an jedem Punkt im Zufallszahlenstrom aufzutreten. Unabhängig voneinander sind die Zufallszahlen, wenn eine berechnete Zufallszahl nicht die Wahrscheinlichkeit der Berechnung der folgenden Zufallszahlen beeinflusst. Des Weiteren ist darauf zu achten, dass für unterschiedliche Konfigurationen eines Simulationsmodells, die exakt gleichen Zufallszahlen verwendet werden, damit mögliche Unterschiede in den Ergebnissen auf die Unterschiede zwischen

den Konfigurationen und nicht die der Zufallszahlen zurückzuführen sind (Law 2015, S. 589).

Im Unterschied dazu, definieren Law (2015, S. 396) als auch Banks et al. (2014a, S. 277) unter anderem die folgenden Anforderungen für Zufallszahlengeneratoren:

- Die erzeugten Zufallszahlen sollten auf dem Intervall (0; 1) gleichverteilt erscheinen und keine Korrelation zueinander aufweisen.
- Der Zufallszahlengenerator sollte möglichst schnell sein, um auch Millionen von Zufallszahlen effizient erzeugen zu können.
- Es sollte möglich sein, den Zufallszahlenstrom zu reproduzieren.
- Die Periode oder Zykluslänge, bis zu einer Wiederholung der Zufallszahlen, sollte möglichst groß sein.

Wie im vorherigen Absatz angesprochen, werden gleichverteilte Zufallszahlen durch deterministische Rechenvorschriften, auch Algorithmen genannt, erzeugt (Gerlach 2019, S. 162). Dazu verwenden die Zufallszahlengeneratoren rekursive Funktionen (Gutenschwager et al. 2017, S. 128). In der Literatur sind verschiedene Zufallszahlengeneratoren veröffentlicht worden, die unterschiedliche rekursive Funktionen nutzen. Nach Law (2015, S. 397 - 409) sind dies unter anderem Lineare Kongruenzgeneratoren, Allgemeine Kongruenzgeneratoren, Kompositionsgeneratoren und Linear Rückgekoppelte Schieberegistergeneratoren, auch Tausworthe-Generatoren genannt. Ein ausführlicher Vergleich der genannten Zufallszahlengeneratoren und die Erläuterung der genutzten rekursiven Funktionen ist nicht Ziel dieser Arbeit. Zu diesem Zweck wird auf einschlägige Literatur wie Law (2015, S. 397 - 409), Gutenschwager et al. (2017, S. 128 - 131) sowie Banks et al. (2014a, S. 277 - 283) verwiesen.

Bevor auf die Erzeugung von Zufallszahlen eingegangen wird, folgt eine kurze Erläuterung der Testmethoden, mit denen die vom Zufallszahlengenerator erzeugten Zufallszahlen auf ihre statistischen Eigenschaften hin untersucht werden können. Diese statistischen Eigenschaften sind nach Banks et al. (2014a, S. 284 f.) die Gleichverteilung und die Unabhängigkeit und wurden am Beginn dieses Abschnitts schon als grundsätzliche Anforderungen für die Zufallszahlen und die Zufallszahlengeneratoren angesprochen. Die Autoren empfehlen die Tests bei unbekanntem oder undokumentiertem Zufallszahlengeneratoren auszuführen. Bekannte Zufallszahlengeneratoren zu testen ist für die

Autoren nicht notwendig. Law (2015, S. 409) spricht sich ebenfalls für die Überprüfung von Zufallsgeneratoren aus, die in der wissenschaftlichen Literatur nicht getestet wurden. Im Verlauf der vorliegenden Arbeit, werden Zufallsgeneratoren eingesetzt, die in wissenschaftlichen Veröffentlichungen intensiv getestet wurden. Aus diesem Grund werden die Testmethoden nicht weiter erläutert. Für eine vertiefende Darstellung wird auf die angesprochene Literatur verwiesen. Wird im weiteren Verlauf der Arbeit von den statistischen Eigenschaften eines Zufallsgenerators gesprochen, ist hiermit die Eigenschaft des Generators gemeint, Zufallszahlen gleichverteilt und unabhängig voneinander zu erzeugen.

Im Folgenden wird die Berechnung von Zufallszahlen mit einem Linearen Kongruenzgenerator, einem Kompositionsgenerator und dem Mersenne-Twister erläutert. Einen der bekanntesten Zufallszahlengeneratoren stellt der Lineare Kongruenzgenerator dar (Waldmann und Helm 2016, S. 21 f.). Nach Gutenschwager et al. (2017, S. 128 f.) wird eine Menge an n_i gleichverteilten Zufallszahlen mit linearen Kongruenzgeneratoren mit der folgenden Formel erzeugt:

$$n_i = (a * n_{i-1} + c) \text{ mod } m \quad (7)$$

Die ganzzahligen Parameter a , c und m in Formel (7) beeinflussen das Verhalten des Generators und sind dementsprechend zu wählen. Die Verteilung auf das Intervall $(0; 1)$ wird erreicht, indem jede erzeugte Zufallszahl n_i durch m dividiert wird. Selbst bei bewusster Wahl der Parameter, können lineare Kongruenzgeneratoren auch bei voller Periode nicht gleichverteiltes Verhalten aufweisen (Law 2015, S. 398). Aus diesem Grund wird die Wahl geeigneter Parameter als „nicht triviale Angelegenheit“ (Gutenschwager et al. 2017, S. 129) bezeichnet. Gerlach (2019, S. 162) bewertet die Qualität der vom linearen Kongruenzgenerator erzeugten Zufallszahlen als „nicht besonders gut“. L’Ecuyer et al. (2002) stellen fest, dass lineare Kongruenzgeneratoren mit $m \leq 2^{31}$ von Simulationsexperten nicht mehr empfohlen werden, da sie durch schwache statistische Eigenschaften Simulationsergebnisse verfälschen können. Zudem erfüllen lineare Kongruenzgeneratoren die Anforderungen moderner Monte-Carlo-Simulationen nicht (Rubinstein und Kroese 2017, S. 51). Law (2015, S. 402) gibt an, dass Kompositionsgeneratoren und der Mersenne-Twister bessere statistische Eigenschaften aufweisen als lineare Kongruenzgeneratoren und deswegen zu bevorzugen sind. L’Ecuyer et al. (2002) empfehlen ebenfalls unter anderem diese beiden Generatoren.

Gutenschwager et al. (2017, S. 130) geben an, dass Kompositionsgeneratoren zwei oder mehr unterschiedliche Generatoren miteinander kombinieren. Das Ziel dieser Kombination ist die Entstehung eines Zufallszahlengenerators mit besseren statistischen Eigenschaften als vor der Kombination. Banks et al. (2014a, S. 281 ff.) sowie Law et al. (2015, S. 403 ff.) empfehlen den in C programmierten und aus zwei Zufallszahlengeneratoren bestehenden Kompositionsgenerator von L'Ecuyer (1999). Nach Rubinstein und Kroese (2017, S. 51 f.) besteht der genannte Kompositionsgenerator alle relevanten statistischen Tests. Die folgenden Formeln verdeutlichen, wie die Zufallszahlen durch diesen Generator erzeugt werden:

$$x_{1,n} = (a_{12} * x_{1,n-2} - a_{13} * x_{1,n-3}) \bmod m_1 \quad (8)$$

$$x_{2,n} = (a_{21} * x_{2,n-2} - a_{23} * x_{2,n-3}) \bmod m_2 \quad (9)$$

$$Y_n = (x_{1,n} - x_{2,n}) \bmod m_1 \quad (10)$$

$$U_i = \begin{cases} \frac{Y_i}{m_1 + 1} & \text{wenn } Y_n > 0 \\ \frac{m_1}{m_1 + 1} & \text{wenn } Y_n = 0 \end{cases} \quad (11)$$

Formel (8) und (9) bilden die beiden Zufallszahlengeneratoren ab, die miteinander kombiniert werden. Für die Parameter a_{12} , a_{13} , a_{21} , a_{23} , m_1 und m_2 hat L'Ecuyer (1999) in seiner Veröffentlichung verschiedene Werte bereitgestellt. Diese Parameterwerte erzeugen nach Law (2015, S. 403) Zufallszahlen mit „exzellenten statistischen Eigenschaften“. Eine Besonderheit dieses Ansatzes ist, dass der Startwert durch den Vektor $(x_{1,0}, x_{1,1}, x_{1,2}, x_{2,0}, x_{2,1}, x_{2,2})$ beschrieben und die erste Zufallszahl mit U_3 bezeichnet wird. Die ersten drei Werte des Startwerte Vektors $(x_{1,0}, x_{1,1}, x_{1,2})$ liegen zwischen 0 und m_1 , die letzten drei $(x_{2,0}, x_{2,1}, x_{2,2})$ zwischen 0 und m_2 . Durch Formel (10) findet die eigentliche Kombination der Zufallszahlengeneratoren statt und Formel (11) stellt sicher, dass die erzeugte Zufallszahl auf dem Intervall $(0; 1)$ verteilt und keine Null erzeugt wird (Waldmann und Helm 2016, S. 25). Der erläuterte Kompositionsgenerator wurde von Banks et al. (2014a, S. 284) aus der Programmiersprache C in die Programmiersprache VBA übersetzt und auf der zum Buch gehörenden Webseite (Banks et al. 2014b) zum Download bereitgestellt. Für Simulationsstudien, die mehrere Zufallszahlenströme benötigen, wurde der genannte Kompositionsgenerator durch L'Ecuyer et al. (2002) erweitert. Diese modifizierte Variante dieses

Zufallszahlengenerators wird im Simulationsprogramm Arena (Arena 2019) eingesetzt (Kelton et al. 2015, S. 523).

Eine weitere Möglichkeit, um Zufallszahlen zu erzeugen, sind nach Gutenschwager et al. (2017, S. 131) sowie Law (2015, S. 405 - 409) sogenannte Tausworthe-Generatoren, basierend auf der Veröffentlichung des namensgebenden Robert C. Tausworthe (Tausworthe 1965). Die Zufallsgeneratoren dieser Familie setzen auf der Ebene der Bits an. Dazu wird eine Sequenz aus Binärzahlen erzeugt und aus dieser mit jeweils den L rechten Binärstellen, mit $L \in \mathbb{N}$, eine Dezimalzahl gebildet. Die Zufallszahlen im Intervall $(0; 1)$ ergeben sich, indem die Dezimalzahlen durch 2^L dividiert werden. Matsumoto und Kurita (1996) geben zu bedenken, dass für bestimmte Startwerte die Verteilung der Zufallszahlen sowohl nicht zufällig als auch nicht gleichverteilt sind. Aufgrund dieser statistischen Defizite sind nach Law (2015, S. 407 ff.) weitere Zufallszahlengeneratoren auf dem Ansatz von Tausworthe entstanden. Einer davon ist der Mersenne-Twister, in der Literatur mit MT19937 abgekürzt, von Matsumoto und Nishimura (1998). Diesem Zufallszahlengenerator werden von Law „exzellente statistische Eigenschaften“ (Law 2015, S. 408) bescheinigt. Nach Gerlach (2019, S. 162 f.) wird der Mersenne-Twister oft in Monte-Carlo-Simulationen verwendet. Waldmann und Helm (2016, S. 27) bezeichnen Mersenne-Twister als „Quasi-Standard“. Die Funktion „Zufallszahl“ in Microsofts Excel (Microsoft 2019b), das Simulationsprogramm SIMUL8 (SIMUL8 2019), das Programm MATLAB (MATLAB 2019) sowie unter anderem die Programmiersprachen Free Pascal (Free Pascal 2019), C++ (C++ 2019) und Julia (Julia 2019) nutzen zur Zufallszahlenerzeugung den Mersenne-Twister von Matsumoto und Nishimura. Wang (2015) hat den Mersenne-Twister aus der Programmiersprache C in VBA übersetzt.

Sowohl Panneton et al. (2006) als auch Waldmann und Helm (2016, S. 27) geben zu bedenken, dass bei der Nutzung des Mersenne-Twister darauf geachtet werden muss, einen Startwert zu wählen, der durch unabhängige und zufällig verteilte Bits erzeugt wird. Wird ein Startwert gewählt, der wenige Einsen enthält, entstehen Gebiete, in denen die eigentlich im Intervall $(0; 1)$ gleichverteilt erzeugten Zufallszahlen für die nächsten tausend Generierungen Werte unter 0,5 annehmen und somit keiner Gleichverteilung entsprechen. Um dies zu vermeiden, haben sowohl Saito und Matsumoto (2008) als auch Panneton et al. (2006) eine verbesserte Version des Mersenne-Twister entwickelt. Beide Weiterentwicklungen vermeiden das angesprochene Startwerteproblem nicht, sondern verlassen die angesprochenen Gebiete nur schneller als der Mersenne-Twister von

Matsumoto und Nishimura (1998). Waldmann und Helm (2016, S. 27) bezeichnen diese Zufallszahlengeneratoren als Standard für Monte-Carlo-Simulationen.

4.4 Erzeugung von Zufallsvariablen auf Basis von Verteilungsfunktionen

Die Übersicht in Abschnitt 4.1 hat aufgezeigt, dass in der Literatur Standorte unter anderem durch entsprechende Verteilungsfunktionen erzeugt werden. Neben einigen anderen Autoren generieren zudem Post et al. (2018) und Boysen et al. (2016) sowohl die x- als auch die y-Koordinate wie eine Zufallsvariable. Des Weiteren wurden in den Literaturquellen Gleich-, Normal- und Dreiecksverteilungen zur Generierung von Standorten genutzt. Im weiteren Verlauf dieser Arbeit werden die Koordinaten der Kundenstandorte exemplarisch mit Hilfe dieser Verteilungsfunktionen erzeugt. Aus diesem Grund erfolgt in diesem Abschnitt eine Einführung in die Erzeugung von Zufallsvariablen auf Basis dieser drei Verteilungsfunktionen.

Sollen Zufallsvariablen erzeugt werden, die in einem anderen Intervall liegen als $(0; 1)$, bieten sich für stetige und diskrete Zufallsvariablen nach Law et al. (2015, S. 451 - 470) sowie Gutenschwager et al. (2017, S. 131 - 134) verschiedene Methoden an. In diesen Methoden werden die im Intervall $(0; 1)$ erzeugten gleichverteilten Zufallszahlen aus den vorherigen Absätzen genutzt. Ist der Wertebereich der ausgesuchten Verteilungsfunktion das Intervall $(0; 1)$, können Zufallsvariablen in einem beliebigen Intervall mit der Inversionsmethode erzeugt werden. Dabei ist es zwingend erforderlich, dass die Umkehrfunktion der Verteilungsfunktion bekannt ist oder berechnet werden kann.

Zufallsvariablen können nach Gutenschwager et al. (2017, S. 108 ff.) beispielsweise mit einer Gleichverteilung abgebildet werden. Diese sind gleichverteilt, wenn alle Zufallsvariablen mit der gleichen Wahrscheinlichkeit im Intervall $(a; b)$ auftreten (Lange und Mosler 2017, S. 51 f.). Die untere Intervallgrenze a entspricht dabei dem kleinsten Wert, die obere Intervallgrenze b dem größten Wert. In Abbildung 8 können die Dichtefunktionen von zwei Gleichverteilungen mit unterschiedlichen Intervallen betrachtet werden.

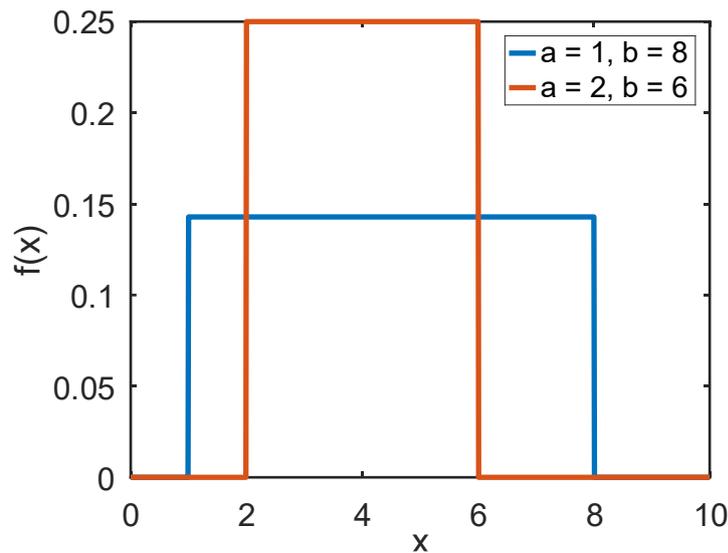


Abbildung 8: Dichtefunktionen von zwei stetigen Gleichverteilungen

Nach Gutenschwager et al. (2017, S. 133 f.) können gleichverteilte stetige Zufallsvariablen mit Hilfe der Inversionsmethode erzeugt werden. Für die Erzeugung dieser Zufallsvariablen auf dem Intervall $(a; b)$ ergibt sich die Umkehrfunktion in Formel (12). Mit dieser Formel werden mit g_i Zufallszahlen x_i gleichverteilte stetige Zufallsvariablen auf dem Intervall $(a; b)$ erzeugt.

$$x_i = a + (b - a) * g_i \quad (12)$$

Für diskrete Zufallsvariablen, die auf dem Intervall $(a; b)$ gleichverteilt sind, kann nach Law (2015, S. 464), Formel (13) genutzt werden. Genau wie im Fall der gleichverteilt stetigen Zufallsvariablen, ergeben g_i Zufallszahlen x_i gleichverteilte diskrete Zufallsvariablen. Die unteren Gaußklammern $\lfloor \cdot \rfloor$ werden in der Formel eingesetzt, um auf die nächste ganze Zahl abzurunden (Groß 2010, S. 33).

$$x_i = a + \lfloor (b - a + 1) * g_i \rfloor \quad (13)$$

Des Weiteren lassen sich Zufallsvariablen nach Gutenschwager et al. (2017, S. 107 f.) mit einer Normalverteilung abbilden. Auffällig ist dabei der glockenförmige Verlauf der Dichtefunktion der Normalverteilung (vgl. Abbildung 9). Für eine Normalverteilung ist der Erwartungswert μ und die Varianz σ^2 anzugeben. Die Normalverteilung mit dem Erwartungswert 0 und der Varianz 1 wird Standardnormalverteilung genannt. Drei Dichtefunktionen der Normalverteilung mit verschiedenen Erwartungswerten und Varianzen können Abbildung 9 entnommen werden.

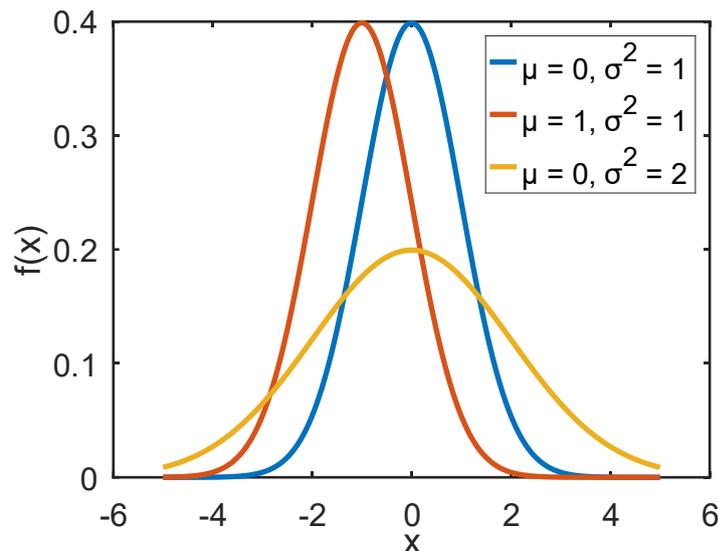


Abbildung 9: Dichtefunktionen von drei Normalverteilungen

Für die Erzeugung von normalverteilten Zufallsvariablen X_i empfehlen Gutenschwager et al. (2017, S. 134 f.) sowie Waldmann und Helm (2016, S. 63) im ersten Schritt die Erzeugung von Werten x_i der Standardnormalverteilung $N(0; 1)$. Im zweiten Schritt werden die erzeugten Werte durch Formel (14) auf die gewünschte Normalverteilung $N(\mu; \sigma^2)$ transformiert.

$$X_i = \mu + \sigma^2 * x_i \quad (14)$$

Zur Erzeugung der standardnormalverteilten Werte empfehlen Gutenschwager et al. (2017, S. 134 f.) sowie Waldmann und Helm (2016, S. 63) beispielsweise die Methode von Box und Muller (Box und Muller 1958). Bei dieser Methode werden für zwei standardnormalverteilte Zufallsvariablen x_1 und x_2 zwei gleichverteilte Zufallszahlen g_1 und g_2 benötigt. Mit den erzeugten Zufallszahlen können unter Zuhilfenahme der Formeln (15) und (16) die Zufallsvariablen x_1 und x_2 berechnet werden.

$$x_1 = \sqrt{-2 * \ln (g_1)} * \cos (2 * \pi * g_2) \quad (15)$$

$$x_2 = \sqrt{-2 * \ln (g_1)} * \sin (2 * \pi * g_2) \quad (16)$$

Law (2015, S. 457 f.) gibt zu bedenken, dass die genutzten gleichverteilten Zufallszahlen voneinander unabhängig sein müssen, da sonst ein Spiralverhalten anstatt einer Normalverteilung beobachtet werden kann. Zudem wird für die Auswertung der trigonometrischen Figuren ein hoher Rechenaufwand benötigt (Gutenschwager et al. 2017, S. 135).

Aufgrund dessen empfehlen Waldmann und Helm (2016, S. 64) sowie Law (2015, S. 457 f.) die Polarmethode nach Marsaglia und Bray (1964). Für diese Methode werden wieder zwei gleichverteilte Zufallszahlen g_1 und g_2 benötigt. Mit diesen beiden Zufallszahlen werden unter Verwendung von Formel (17) s_1 und s_2 berechnet.

$$s_i = 2 * g_i - 1 \text{ für } i = 1,2 \quad (17)$$

Danach wird $W = s_1^2 + s_2^2$ berechnet und solange $W \leq 1$ können die standardnormalverteilten Zufallsvariablen x_1 und x_2 mit Formel (18) berechnet werden, andernfalls muss wieder mit der Berechnung der Zufallszahlen gestartet werden. Da die berechneten Zufallszahlen durch $W \leq 1$ abgewiesen werden können, benötigt diese Methode mehr Zufallszahlen als Zufallsvariablen berechnet werden. Die Ablehnung der Zufallszahlen erfolgt mit der Wahrscheinlichkeit $1 - \pi/4$, was 21,46 Prozent entspricht.

$$x_i = \sqrt{\frac{-2 * \ln(W)}{W}} * s_i \text{ für } i = 1,2 \quad (18)$$

Weiterhin kann die Abbildung von Zufallsvariablen nach Gutenschwager et al. (2017, S. 108 ff.) mit einer Dreiecksverteilung erfolgen. Wie bei der Gleichverteilung treten die Zufallsvariablen in einem Intervall (a; b) auf. Zusätzlich dazu wird der am häufigsten auftretende Wert m der Zufallsvariable angegeben. Drei Dichtefunktionen mit unterschiedlichen Parametern sind in Abbildung 10 zu sehen.

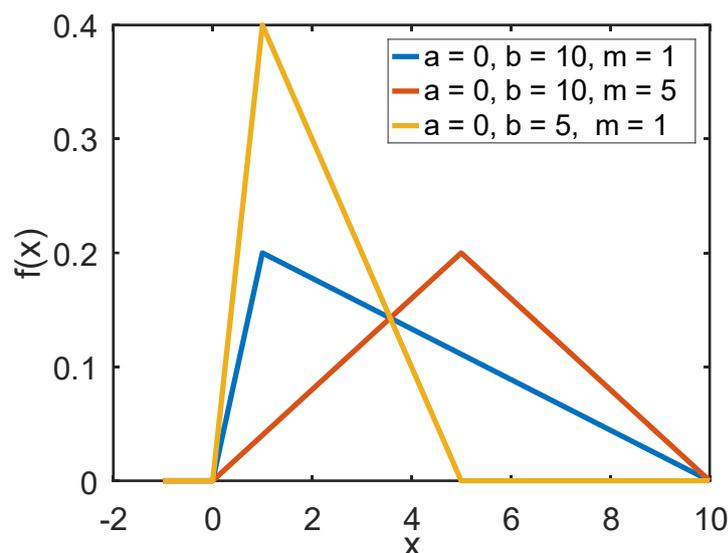


Abbildung 10: Dichtefunktionen von drei Dreiecksverteilungen

Für die Erzeugung von Zufallsvariablen, die dreiecksverteilt sind, empfehlen Gutenschwager et al. (2017, S. 133 f.) sowie Law (2015, S. 461) wiederum die

Inversionsmethode. Mit den Formeln (19), (20) und (21) werden g_i gleichverteilte Zufallszahlen in x_i dreiecksverteilte Zufallsvariablen im Intervall $(a; b)$ und mit dem häufigsten Wert m überführt.

$$x_i = a + (b - a) * h_i \quad (19)$$

$$m' = \frac{m - a}{b - a} \quad (20)$$

$$h_i = \begin{cases} \sqrt{m' * g_i} & \text{für } 0 \leq g_i \leq m' \\ 1 - \sqrt{(1 - m') * (1 - g_i)} & \text{für } m' \leq g_i \leq 1 \end{cases} \quad (21)$$

5. Vorbereitung der Simulationsstudie

Die Literaturrecherche im Grundlagenteil der vorliegenden Arbeit hat gezeigt, dass in der Literatur keine Quelle existiert, in der untersucht wird, ob reale Kundenstandorte mit Hilfe von zufällig generierten Standorten abgebildet werden können. Aus diesem Grund wird in diesem Kapitel zuerst das zugrunde liegende Szenario für die Simulation eingeführt. Danach wird darauf aufbauend ein Simulationsmodell entworfen. Im Abschluss dieses Kapitels erfolgt die Erläuterung der Kriterien zur Bewertung der Generierungsverfahren.

5.1 Vorstellung des Testszenarios

Im Verlauf dieses Abschnitts werden Einschränkungen getroffen, die das Versuchsszenario hinsichtlich des Umfangs dieser Arbeit begrenzen. In der vorliegenden Arbeit wird ein fiktiver Logistikdienstleister aus dem Bereich der KEP-Dienstleister (vgl. Abschnitt 2.1) betrachtet. Dieser Logistikdienstleister erfüllt die Distributionsaufgaben für verschiedene Unternehmen, die innerhalb des E-Commerce tätig sind, und beliefert die Kunden dieser Unternehmen. Im Rahmen seiner Dienstleistungen bietet der KEP-Dienstleister nur Paketlieferungen an. Er übernimmt keine Express- oder Kurierdienste, wie im KEP-Markt sonst durchaus üblich (vgl. Abschnitt 2.2). Des Weiteren wird der Kundenkreis auf das B2C eingeschränkt, da dieser Bereich einen höheren Anteil am Gesamtpaketmarkt besitzt als B2B und C2C insgesamt und dem B2C zudem für die nächsten Jahre weiter hohe Wachstumsraten vorausgesagt werden (vgl. Abschnitt 2.3). Außerdem werden Retouren nicht betrachtet (vgl. Abschnitt 2.1). Aufgrund des niedrigen Stoppfaktors von 1,1 im B2C, wird die Belieferung der Kunden im Versuchsszenario auf ein Paket pro Kunde beschränkt (vgl. Abschnitt 2.3).

Bei dem Netzwerk des Logistikdienstleisters handelt es sich darüber hinaus um ein mehrstufiges Logistiknetzwerk (vgl. Abschnitt 2.1). Daneben kann das Netzwerk auch als Many-to-Many-Netzwerk bezeichnet werden, da es viele Versender, in diesem Fall verschiedene Unternehmen, mit vielen Empfängern, hier die Kunden, verbindet (vgl. Abschnitt 2.2). Many-to-Many-Netzwerke sind ein wesentlicher Bestandteil des E-Commerce.

Der fiktive Logistikdienstleister möchte die Distributionsaufgaben für seine Auftraggeber in weiteren Städten übernehmen. Aufgrund der zunehmenden Urbanisierung der Weltbevölkerung und den damit einhergehenden Konsequenzen, wie überlasteten Innenstädten oder der Forderung der Kunden, logistische Netzwerke umweltfreundlicher zu gestalten, aber auch dem Wettbewerb zwischen den Logistikunternehmen selbst, möchte der fiktive Logistikdienstleister die Belieferung seiner Kunden ökologisch effizienter gestalten und außerdem seine Kosten senken. Wegen der Bündelungseffekte auf der Haupttransportrelation können sowohl die Kosten als auch die Effektivität der Belieferung vor allem bei der Belieferung der Kunden, der Last Mile, gesenkt beziehungsweise erhöht werden (vgl. Abschnitt 2.3). Aus diesem Grund wird in der vorliegenden Arbeit nur die Last Mile, der letzte Schritt im Logistiknetzwerk, des fiktiven KEP-Dienstleisters betrachtet. In der Last Mile werden die Kunden des KEP-Dienstleisters aus einem Distributionszentrum beliefert, welches sich am Stadtrand befindet (vgl. Abschnitt 2.3). Mit dieser Einschränkung kann die Betrachtung des Netzwerks als Many-to-Many-Netzwerk nicht mehr aufrechterhalten werden, denn die Belieferung der Kunden aus einem Distributionszentrum stellt ein einstufiges Logistiknetzwerk dar (vgl. Abschnitt 2.1). Das Logistiknetzwerk kann somit auch als One-to-Many-Netzwerk bezeichnet werden. Die Kundenstandorte in den Städten sind dem Logistikdienstleister nicht bekannt, da er bis jetzt nicht in diesen Städten tätig war und somit über keine Kundendaten verfügt. Die Nutzung dieser Daten wäre vor dem Hintergrund der Datenschutz-Grundverordnung ohne eine entsprechende Einwilligung der Kunden rechtlich zweifelhaft (vgl. Abschnitt 3.3). Des Weiteren wollen die Auftraggeber dem Logistikdienstleister, aufgrund von Zweifeln an den gängigen Anonymisierungsverfahren für Kundendaten, keine Daten zur Verfügung stellen (vgl. Abschnitt 3.3). Den Simulationsexperten des Logistikdienstleisters ist bewusst, dass sowohl die Dauer der Fahrzeit als auch die Fahrstrecke Einfluss auf die wirtschaftliche und ökologische Effizienz haben. Des Weiteren wissen sie um die Abhängigkeit der Fahrstrecke von den geographischen Gegebenheiten des Netzwerks sowie der Lage der Kundenstandorte und des Distributionszentrum (vgl. Abschnitt 2.2 und Abschnitt 2.3). Aus diesem Grund wollen die Simulationsexperten herausfinden, ob sie bei dem vorliegenden Datenmangel die Kundenstandorte mit Hilfe von Wahrscheinlichkeitsverteilungen generieren können und wie diese Generierung die Fahrstrecke beeinflusst.

5.2 Aufbau des Simulationsmodells

Der Aufbau des Simulationsmodells wird anhand des Vorgehensmodells für Simulationsstudien in Abschnitt 3.2 entwickelt (vgl. Abbildung 7). Der Untersuchungszweck des Simulationsmodells, der in der Zielbeschreibung festgelegt wird, besteht in der Beantwortung der Frage, ob Verfahren zur Standortgenerierung bei Datenmangel (vgl. Abschnitt 3.3) eingesetzt werden können, um Kundenstandorte zu generieren. Die Methode der Simulation stellt hierbei eine geeignete Analysemethode dar, da mit ihr ein fiktives System in einem experimentierbaren Simulationsmodell abgebildet werden kann. Das experimentierbare Simulationsmodell wird benötigt, um die Einflüsse zufällig generierter Kundenstandorte auf das Logistiknetzwerk des Versuchsszenarios (vgl. Abschnitt 5.1) zu untersuchen.

Für die Aufgabenspezifikation lässt sich festhalten, dass in dem Simulationsmodell nur die Last Mile des Versuchsszenarios abgebildet werden muss. Des Weiteren müssen die zufälligen Kundenstandorte für das Simulationsmodell geeignet generiert und in das Simulationsmodell eingefügt werden. Die Darstellung des zu beliefernden Stadtgebietes erfolgt über ein x-y-Koordinatensystem. Da die Zeit keinen Einfluss auf die Kundenstandorte nimmt und die Generierung der Kundenstandorte zufällig ist, ist das Simulationsmodell statisch sowie stochastisch. Für stochastische und statische Simulationsmodelle wird die Monte-Carlo-Simulation eingesetzt (vgl. Abschnitt 3.1). Für die Umsetzung der Monte-Carlo-Simulation wird das Programm Excel von Microsoft genutzt. Die nachfolgenden Angaben zum verwendeten Rechner dienen dazu, die Simulation reproduzieren zu können (vgl. Abschnitt 3.1): Für die Simulationen wird ein Rechner mit Windows 10 Betriebssystem, dem Prozessor Intel Core i7-7700k und 16 Gigabyte Arbeitsspeicher verwendet.

Da in der Literatur vor der Verwendung der internen Zufallszahlengeneratoren von Excel gewarnt wird (vgl. Abschnitt 3.1), wird zur Berechnung der Zufallszahlen in diesem Simulationsmodell der in VBA übersetzte Kompositionsgenerator aus Abschnitt 4.3 genutzt. Das Simulationsmodell muss außerdem eine Funktion besitzen, die es dem Simulationsexperten möglich macht, beliebig viele Replikationen eines Simulationslaufes durchzuführen, damit die Startwerte der Zufallszahlengeneratoren geändert werden. Durch eine weitere Funktion muss jede einzelne Replikation der Monte-Carlo-Simulation nach Wahl und Notwendigkeit wiederholbar sein, damit die Nachvollziehbarkeit der Ergebnisse gegeben ist (vgl. Abschnitt 3.1). Auch ist

sicherzustellen, dass mindestens drei bis fünf aber besser 5000 bis 8000 Replikationen mit dem Simulationsmodell durchführbar sind, um der Empfehlung für die benötigten Replikationen in einer Monte-Carlo-Simulation folgen zu können (vgl. Abschnitt 3.1). Zudem muss die Anzahl der Kundenstandorte veränderbar sein, um festzustellen, ob die Anzahl der Kundenstandorte Einfluss auf das Ergebnis hat. Zur Auswertung werden, wie in der Literatur empfohlen, Kennzahlen eingesetzt (vgl. Abschnitt 3.4). Als Kennzahlen zur Auswertung werden die Tourenlänge, die der Gesamtstrecke aller gefahrenen Touren in einer Replikation entspricht, und die Zeit, die eine Replikation benötigt, aufgenommen (vgl. Abschnitt 2.4). Der Simulationslauf endet, nachdem alle Kundenstandorte berechnet und alle Ergebnisgrößen aufgenommen wurden, daher handelt es sich um eine terminierende Simulation (vgl. Abschnitt 3.1). Somit können die Verfahren zur Analyse von Simulationsergebnissen aus Abschnitt 3.4 zur Untersuchung des Simulationsmodells eingesetzt werden.

Mit Hilfe des Simulationsmodells untersucht werden, ob reale Kundenstandorte mit zufällig generierten Kundenstandorten abgebildet werden können. Um diesen Einfluss zu isolieren, werden nur die Kundenstandorte betrachtet und beispielsweise nicht der Bedarf der Kunden. Das heißt, dass die einzigen beiden Elemente des Simulationsmodells der Standort des Distributionszentrum und die Standorte der Kunden sind. Aufgrund der Beschränkung, dass jeder Kunde nur ein Paket erhält (vgl. Abschnitt 5.1), muss im Simulationsmodell eine Funktion hinterlegt werden, die die Generierung doppelter Kundenstandorte verhindert. Die Generierung doppelter Standorte wird als Fehler aufgenommen, um diesen Faktor später in die Bewertung der Generierungsverfahren mit einfließen zu lassen. Wird ein Standort außerhalb des Stadtgebietes generiert, wird dieser verworfen und ein neuer generiert. Zudem wird dies als Fehler „außerhalb des Stadtgebietes“ bezeichnet und für die Bewertung aufgenommen. Des Weiteren werden bei der Berechnung der normalverteilten Zufallsvariablen Fehler erzeugt, welche als Fehler mit der Bezeichnung „Berechnungsfehler“ aufgenommen werden (vgl. Abschnitt 4.4). Da sich Tourenplanungsverfahren der optimalen Lösung nur annähern und diese nicht exakt berechnen (vgl. Abschnitt 2.4), wird auf die Nutzung von Tourenplanungsverfahren zur Berechnung der Kennzahl Tourenlänge verzichtet. Als eine Tour wird die Belieferung eines einzigen Kunden vom Distributionszentrum aus bezeichnet. Die zurückgelegte Fahrstrecke auf jeder Tour wird aufgezeichnet und die Kennzahl Tourenlänge aus diesen einzelnen Fahrstrecken der Touren aufsummiert. Die einzelnen Fahrstrecken werden mit Hilfe der gewichteten euklidischen Distanz berechnet. Als Gewichtungsfaktor wird 1,234

genutzt (vgl. Abschnitt 2.2). Die grafische Darstellung in der vorliegenden Arbeit beinhaltet Box-Plots und Balkendiagramme (vgl. Abschnitt 3.4). Mit Hilfe der Box-Plots wird die Kennzahl Tourenlänge dargestellt. Für die Konstruktion und Auswertung der Box-Plots müssen der Mittelwert, der Median sowie die Werte des 1. und des 3. Quartils berechnet werden. Die Formeln für die benötigten Werte sind in Abschnitt 3.4 zu finden. Die Darstellung der Fehler, der benötigten Zeit für die Generierung der Standorte und die Konfidenzintervalle der Tourenlänge erfolgt mit Balkendiagrammen. Dazu werden jeweils der minimale und maximale Wert aufgenommen, beziehungsweise im Fall des Konfidenzintervalls wird dieses anhand Formel (2) in Abschnitt 3.4 berechnet. Außerdem werden die Paired-t-Konfidenzintervalle für den Vergleich von zwei Parameterkonfigurationen eines oder verschiedener Generierungsverfahren genutzt. Das Konfidenzniveau für alle Konfidenzintervalle in der vorliegenden Arbeit beträgt 95 Prozent. Für dieses Konfidenzniveau ergibt sich für $z_{1-\alpha/2}$ ein tabellierter Wert von 1,96 (vgl. Abschnitt 3.4).

Innerhalb der Datenbeschaffung müssen nach Abschnitt 3.3 die Standorte der Lager und der Kunden beschafft werden. In der vorliegenden Arbeit sind dies die x- und y-Koordinaten der Standorte, im Folgenden (x; y). Das Distributionszentrum liegt am Rand des Stadtgebietes (vgl. Abschnitt 5.1). Aus diesem Grund wird der Standort des Distributionszentrum auf die Koordinaten (0; 0) gelegt. Die (x; y)-Koordinaten der Kundenstandorte gehören in der vorliegenden Arbeit zu den Daten der Kategorie C (vgl. Abschnitt 3.3) und sind deswegen weder verfügbar noch können sie durch eine Datenerhebung verfügbar gemacht werden. Aus diesem Grund werden die Daten mit Wahrscheinlichkeitsverteilungen erzeugt. In der Literatur werden unter anderem die Gleichverteilung, die Normalverteilung sowie die Dreiecksverteilung empfohlen und auch bereits genutzt (vgl. Abschnitt 4.1). Deswegen werden die Gleichverteilung, die Normalverteilung und die Dreiecksverteilung verwendet (vgl. Abschnitt 4.4), um die x- und y-Koordinaten für die Kundenstandorte zu generieren. Weil die Verteilungsfunktionen zur Generierung der Koordinaten von Kundenstandorten eingesetzt werden, werden diese im weiteren Verlauf der Arbeit als Generierungsverfahren bezeichnet. Zuletzt wird vorausgesetzt, dass dem fiktiven Logistikdienstleister aus Abschnitt 5.1 die Bevölkerungsdichten in den Städten, in welche er expandieren möchte, bekannt sind und er die Kundenstandorte unter Berücksichtigung dieser generiert (vgl. Abschnitt 4.2). Die Parameter der Generierungsverfahren werden anhand ausgewählter Benchmark-Testsets aus der Tourenplanung bestimmt (vgl.

Abschnitt 2.4). Die Auswahl der Testsets erfolgt exemplarisch und sind in Tabelle 5 abgebildet. Die erste Spalte der Tabelle enthält die Autoren, die das Testset veröffentlicht haben. In der zweiten Spalte ist die Bezeichnung des Testsets eingetragen. Dieser Name wird auch im weiteren Verlauf der vorliegenden Arbeit zur Kennzeichnung der Testsets genutzt.

Tabelle 5: Ausgewählte Benchmark-Testsets

Autoren	Bezeichnung	Anzahl Standorte	Begrenzende Koordinaten (x; y)	Anordnung	Kunden pro QLE	Abbildung (im Anhang)
Christofides und Eilon (1969)	E-n22-k4	22	(165; 265)	Verteilt auf (120; 180) bis (165; 264)	0,0005	Abbildung 72 Seite: 213
Christofides et al. (1979)	CMT14	100	(95; 85)	Gruppiert	0,012	Abbildung 73 Seite: 213
Homberger und Gehring (2004)	C110_1	1000	(500; 500)	Gruppiert	0,004	Abbildung 74 Seite: 214
Homberger und Gehring (2004)	R110_1	1000	(500; 500)	Gleichverteilt	0,004	Abbildung 75 Seite: 215
Homberger und Gehring (2004)	RC110_1	1000	(500; 500)	Gruppiert und gleichverteilt	0,004	Abbildung 76 Seite: 216
Arnold et al. (2019)	Antwerp1	6001	(2000; 2000)	Gruppiert und gleichverteilt	0,0015	Abbildung 77 Seite: 217

Die Anzahl der Standorte, die von den Generierungsverfahren erzeugt werden, sind gleich der Anzahl der Standorte aus den Testsets zu wählen, damit die Ergebnisse der Simulation im weiteren Verlauf der vorliegenden Arbeit mit den Testsets verglichen werden können. Diese sind in Spalte drei von Tabelle 5 zu sehen. Außerdem beinhalten die Testsets nur ganzzahlige Koordinaten für die Standorte, weswegen in der Simulation ebenfalls nur ganzzahlige Koordinaten generiert werden. Die Grenzen der Stadtgebiete spannen sich von (0; 0), dem Standort des Distributionszentrums, bis zu den begrenzenden Koordinaten der Testsets auf, die sich in Spalte vier befinden. Für die Koordinatensysteme werden keine LE angegeben. Aus diesem Grund werden die Fahrstrecken und somit auch die Kennzahl Tourenlänge in der vorliegenden Arbeit in LE

gemessen. Innerhalb der fünften Spalte ist aufgelistet, wie die Standorte in den Testsets angeordnet sind. Dabei sind die Standorte entweder in einer Ecke des Koordinatensystems oder in Gruppen angeordnet oder aber gleichmäßig über das gesamte Koordinatensystem verteilt. Die sechste Spalte gibt das Verhältnis der Standorte zur Fläche des Koordinatensystems an. Dabei gilt es zu beachten, dass die Fläche beispielsweise nicht in Quadratkilometern gemessen wird, sondern in Quadratlängeinheiten (QLE). Die Standorte sind für jedes Testset jeweils in ein (x; y)-Koordinatensystem eingetragen und, um Platz einzusparen, im Anhang dieser Arbeit eingefügt worden. In der letzten Spalte sind diese Abbildungen für jedes Testset mit ihrer Nummer und Seitenzahl eingetragen.

In Abbildung 11 ist der Programmablaufplan für die Simulation in dieser Arbeit dargestellt. Vom Anwender werden die Anzahl der zu erzeugenden Kundenstandorte, die Anzahl der Replikationen, die Koordinaten des Distributionszentrums sowie alle Parameter, die das ausgesuchte Generierungsverfahren benötigt, eingegeben. Ist beispielsweise die Anzahl der Standorte unbekannt, kann jede beliebige ganze Zahl bis 20000 eingegeben werden. Jenseits dieses Zahlenbereichs nimmt die Simulation sehr viel Zeit in Anspruch. Bei der Anzahl der Replikationen ist sich an der Empfehlung aus der Literatur zu orientieren. Die Schritte zur Generierung der Kundenstandorte sind für alle Verfahren gleich. Der einzige Unterschied zwischen den Verfahren besteht in den genutzten Formeln zur Verteilung der Standorte. In einem ersten Schritt werden zwei Zufallszahlen mit dem Kompositionsgenerator erzeugt und mit der ersten Zufallszahl eine x-Koordinate, mit der zweiten Zufallszahl eine y-Koordinate erzeugt. Die erzeugten Koordinaten werden im nächsten Schritt zu einem Standort zusammengefügt. Um doppelten Standorten vorzubeugen wird in einer Liste überprüft, ob dieser Standort schon existiert. Dieses Vorgehen wird so lange wiederholt, bis alle benötigten Standorte generiert wurden.

Daraufhin werden die Kennzahlen Tourenlänge und Generierungszeit sowie die entstandenen Fehler bestimmt. Diese Werte werden in ein dafür vorgesehenes Tabellenblatt eingetragen. Möchte der Simulationsexperte mehr als eine Replikation durchführen, wird wieder bei der Erzeugung der Kundenstandorte gestartet. Dieses Vorgehen wird so lange wiederholt, bis die Anzahl der gewünschten Replikationen erreicht ist. Nach dem Erreichen der benötigten Replikationen werden aus den Kennzahlen und den erzeugten Fehlern die benötigten Werte für die grafische

Auswertung berechnet. Des Weiteren wird das Konfidenzintervall bestimmt. Zuletzt werden die Kennzahlen, die entstandenen Fehler, sowie die berechneten Werte für die grafische Auswertung und das Konfidenzintervall ausgegeben.

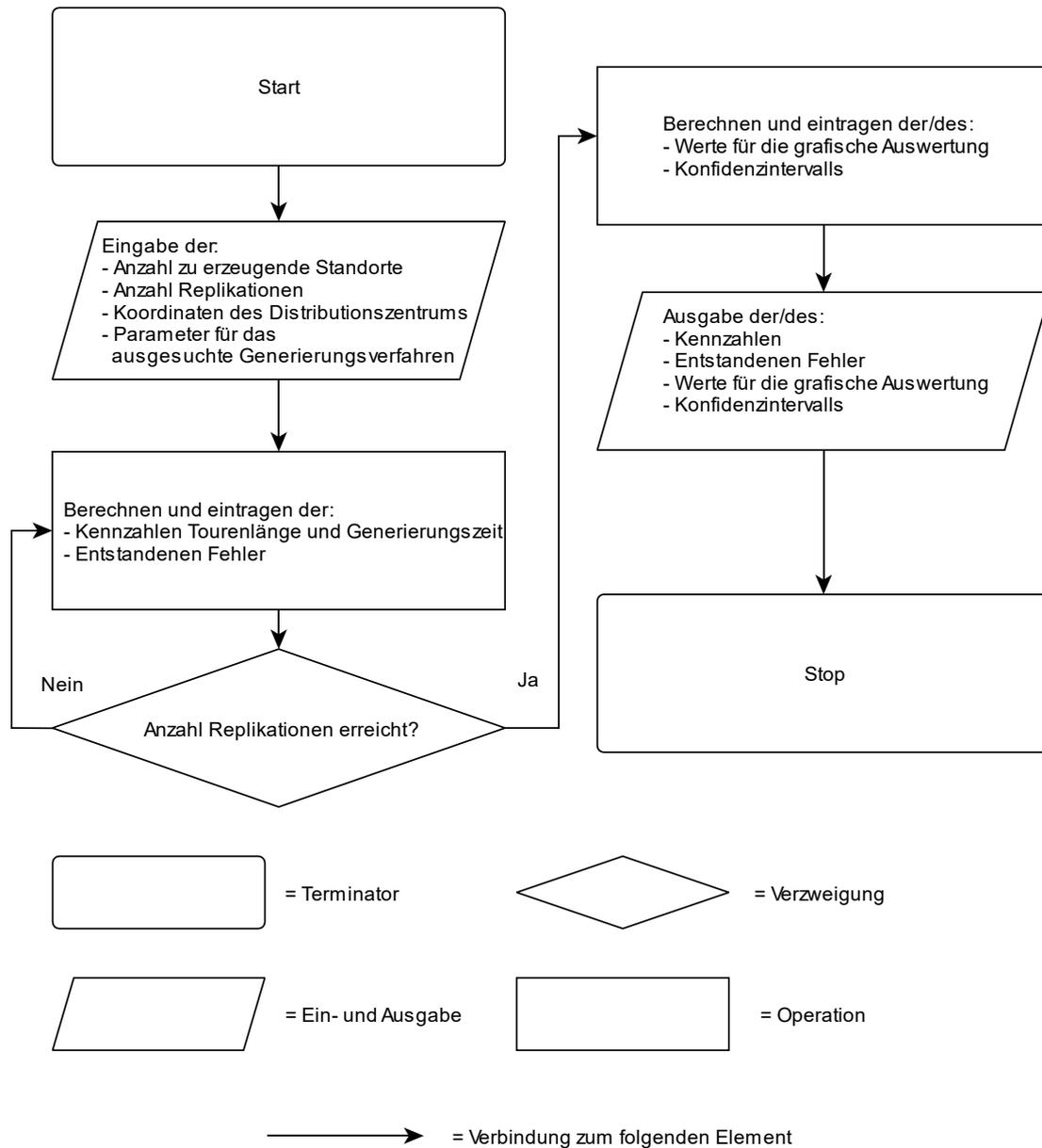


Abbildung 11: Programmablaufplan

Bei der Implementierung, der Überführung des formalen Modells in ein ausführbares Modell (vgl. Abschnitt 3.2), ist der genutzte Kompositionsgenerator in geeigneter Weise zu parametrisieren. Dazu werden Werte verwendet, die in der Literatur veröffentlicht und deren Eignung durch statistische Tests untersucht wurden (vgl. Abschnitt 4.3). Die Werte der Parameter sind in Tabelle 6 abgebildet.

Table 6: *Parameter des genutzten Kompositionsgenerators*

Parameter	Wert
a_{12}	1403580
a_{13}	810728
a_{21}	527612
a_{23}	1370589
m_1	4294967087
m_2	4294944443

Des Weiteren ist zu beachten, dass der Kompositionsgenerator zur Generierung der Zufallszahlen für jede Replikation einen Startwertvektor mit sechs Startwerten ($x_{1,0}$, $x_{1,1}$, $x_{1,2}$, $x_{2,0}$, $x_{2,1}$, $x_{2,2}$) benötigt (vgl. Abschnitt 4.3). Diese Startwerte werden nicht bei jeder Replikation erzeugt, sondern bereits im Vorhinein berechnet und der Monte-Carlo-Simulation als Eingangsdaten zur Verfügung gestellt. Die ersten drei Startwerte ($x_{1,0}$, $x_{1,1}$, $x_{1,2}$) liegen zwischen 0 und m_1 , also im Intervall $(0, 4294967087)$, die letzten drei Startwerten liegen zwischen 0 und m_2 und damit im Intervall $(0, 4294944443)$. Zur Berechnung der Startwerte werden mit der Formel „Zufallszahl“ des Programms Excel Zufallszahlen auf Basis des Mersenne-Twisters erzeugt und mit Hilfe von Formel (13) auf das benötigte Intervall transformiert. Mit diesem Vorgehen werden insgesamt 20000 Startwertvektoren mit jeweils sechs Startwerten erzeugt, dies ermöglicht die Durchführung von insgesamt 20000 Replikationen zu jedem Generierungsverfahren. Um gleichen Startwertvektoren vorzubeugen, werden diese im Anschluss an die Berechnung auf mehrfach vorkommende Startwertvektoren überprüft. Mit diesem Vorgehen wird die aufwendige Überprüfung auf gleiche Startwertvektoren aus der Standortgenerierung herausgenommen. Zur Standortgenerierung greift ein Generierungsverfahren a mit 100 Replikationen auf die ersten 100 Startwertvektoren zu, bei 1000 Replikationen auf die ersten 1000 und bei 10000 Replikationen entsprechend auf die ersten 10000 Startwertvektoren. Für ein anderes Generierungsverfahren b ist das Vorgehen äquivalent. Dadurch greifen alle Standortgenerierungsverfahren auf die gleichen Startwerte zu und erzeugen dadurch die gleichen Zufallszahlen für die Standortgenerierung. Somit können verschiedene Verfahren miteinander verglichen werden und es ist sichergestellt, dass mögliche Unterschiede in den Ergebnissen der Verfahren nicht auf unterschiedliche Zufallszahlen zurückzuführen sind (vgl. Abschnitt 4.3). Zusätzlich können durch dieses Vorgehen einzelne Replikationen oder der gesamte Simulationslauf wiederholt werden.

Bei der Umsetzung in das ausführbare Modell unterscheiden sich die Generierungsverfahren in den Formeln für die Berechnung der Verteilung der Kundenstandorte und in den von den Formeln benötigten Parametern. Ansonsten folgt das Modell dem Programmablaufplan aus Abbildung 11. Die Generierung von gleichverteilten Kundenstandorten erfolgt mit Hilfe von Formel (13) aus Abschnitt 4.4. Normalverteilte Kundenstandorte werden durch die Polarmethode aus Abschnitt 4.4 erzeugt. Mit Formel (19) werden die Kundenstandorte einer Dreiecksverteilung folgend generiert. Die benötigten Parameter für die verschiedenen Verfahren werden in den Abschnitten erläutert, in denen diese benötigt werden.

Zur V&V wurde die Technik Strukturiertes Durchgehen eingesetzt (vgl. Abschnitt 3.2). Dafür wurden mit dem Testset Antwerp1 (vgl. Tabelle 5), weil es die meisten Kundenstandorte enthält und aufgrund der Größe auch die meisten Kundestandorte in diesem Stadtgebiet erzeugt werden können, jeweils eine Simulation für die Gleich-, die Normal- und die Gleichverteilung durchgeführt. Die Replikations Nummer dieser Simulation war 17. Die Parameter betragen für die Gleichverteilung, jeweils für die x- und die y-Koordinate, das Intervall (0; 2000) und für die Normalverteilung dasselbe Intervall sowie sowohl für die x- als auch die y-Koordinate ein Erwartungswert von 1000 LE und eine Varianz von 300 LE. Die Dreiecksverteilung nutzt auch das identische Intervall wie die vorherigen Generierungsverfahren und zusätzlich dazu liegt der wahrscheinlichste Wert bei 1000 LE, sowohl für die x- als auch die y-Koordinate (vgl. Abschnitt 4.4). Nach Abschluss der Simulationen wurden die generierten x- und y-Koordinaten der Kundenstandorte für jedes Generierungsverfahren in ein x-y-Koordinatensystem eingetragen und auf Auffälligkeiten hinsichtlich der Verteilung der Kundenstandorte untersucht (vgl. Abbildung 78, Abbildung 79 sowie Abbildung 80). In den Abbildungen konnten keine Auffälligkeiten gefunden werden. Des Weiteren erfolgte eine Untersuchung jedes einzelnen Testsets bezüglich doppelt generierter Standorte. In der Replikation werden zwar doppelte Standorte erzeugt, diese nimmt der Algorithmus jedoch, wie vorgesehen, als Fehler auf und berechnet neue Koordinaten für den betroffenen Standort, sodass keine doppelten Standorte zu finden waren. Abschließend folgte die Überprüfung der Kennzahl Tourenlänge, welche hier händisch berechnet wurde. Wie die vorherigen Untersuchungen, war diese ohne Fund. Mit diesen Ergebnissen kann ein Fehler nicht ausgeschlossen werden, der Algorithmus scheint jedoch seine Arbeit zu verrichten.

5.3 Einführung von Kriterien zur Bewertung der Generierungsverfahren

Nach Abschluss aller Simulationen werden die eingesetzten Generierungsverfahren miteinander verglichen, um die möglichen Unterschiede in den Generierungsverfahren aufzuzeigen. Dafür wird für jedes Testset betrachtet, ob sich die verschiedenen Generierungsverfahren in dem erzeugten Mittelwert der Tourenlänge oder der Spannweite des Konfidenzintervalls der Tourenlänge unterscheiden. In den Vergleich fließen außerdem die Generierungszeiten und die erzeugten Fehler bei der Generierung ein. Am Ende jedes Vergleichs erfolgt eine Platzierung der Generierungsverfahren. Weil das fiktive Unternehmen seine Tourenlänge verringern möchte (vgl. Abschnitt 5.1) richtet sich die Platzierung danach, welches Generierungsverfahren die kürzeste Tourenlänge erzeugt. Diese Platzierung geschieht ohne Betrachtung der Tourenlänge der Testsets.

Basierend auf dem Vergleich folgt abschließend eine Bewertung der Generierungsverfahren. Die Bewertung wird anhand von Kriterien vorgenommen, die im nächsten Absatz erläutert werden. Dabei können in jedem Kriterium, basierend auf der Gesamtanzahl der Generierungsverfahren, insgesamt sechs Punkte erreicht werden. Das Verfahren, welches das Kriterium am besten erfüllt, bekommt drei Punkte, das mittlere Verfahren zwei und äquivalent das letzte Verfahren einen Punkt. Bei Gleichstand erhalten die drei Generierungsverfahren jeweils zwei Punkte. Das erste Kriterium für die Bewertung behandelt die Parameter der Generierungsverfahren. Vor der Anwendung der Generierungsverfahren müssen die zugrundeliegenden Verteilungsfunktionen geeignet parametrisiert werden (vgl. Abschnitt 4.4). Je mehr Parameter ein Verfahren benötigt, umso mehr müssen vom Simulationsexperten, sofern keine Daten vorliegen, geschätzt werden. Damit hat eine größere Anzahl an benötigten Parametern einen größeren Einfluss auf das Ergebnis der Simulation (vgl. Abschnitt 3.1). Aus diesem Grund ist das erste Bewertungskriterium die Anzahl der benötigten Parameter für das Generierungsverfahren. Je weniger Parameter das Verfahren benötigt, desto besser wird es gewichtet. Durch das zweite Kriterium wird überprüft, welches Generierungsverfahren genutzt werden kann, um Standortdaten für Simulationsprogramme, wie AnyLogic oder SimChain, zu erzeugen. Beide genannten Simulationsprogramme benötigen zur Darstellung von Standorten die Geo-Koordinaten in Gradzahlen (vgl. Abschnitt 3.3), wobei die Gradzahlen mit reellen Zahlen dargestellt werden. Für die Erzeugung von Gradzahlen werden demzufolge Generierungsverfahren benötigt, die stetige

Zufallsvariablen erzeugen können (vgl. Abschnitt 4.3). Hierbei werden die Generierungsverfahren danach gewichtet, ob sie das Kriterium erfüllen oder nicht.

Ab dem Folgenden Kriterium erfolgt die Bewertung der Generierungsverfahren einzeln für jedes Testset. Für das dritte Kriterium wird die Anzahl der Replikationen bewertet, die durchgeführt werden. Die geringste Anzahl an benötigten Replikationen erhält die beste Beurteilung. Mit dem vierten Kriterium wird die Zeit gewichtet, welche die Generierungsverfahren benötigen, um die Anzahl der Replikationen aus Kriterium drei durchzuführen. Hierbei bekommt die kürzeste Dauer die volle Punktzahl. Im fünften Kriterium werden die Breiten der Konfidenzintervalle der Tourenlänge betrachtet. Je größer die Breite dieser Konfidenzintervalle ist, desto mehr schwankt der Erwartungswert der Tourenlänge (vgl. Abschnitt 3.4). Aus diesem Grund werden kleine Breiten der Konfidenzintervalle besser beurteilt als große.

Mit Hilfe des sechsten Kriteriums erfolgt die Beurteilung darüber, inwiefern die Generierungsverfahren die Tourenlängen der Testsets, im weiteren als wahre Tourenlänge bezeichnet, mit ihren Konfidenzintervallen überdecken und damit am besten annähern (vgl. Abschnitt 5.2). Hierdurch wird auch die Eignung des Generierungsverfahren, die wahre Tourenlänge des jeweiligen Testsets abzubilden, bewertet. Enthalten ein oder mehrere Konfidenzintervalle die wahre Tourenlänge des Testsets nicht, so wird der Abstand der oberen Grenze der Konfidenzintervalle zur wahren Tourenlänge des Testsets bewertet. Hierbei gilt, je kürzer der Abstand desto besser wird das Generierungsverfahren bewertet. Das siebte Kriterium behandelt die Gesamtanzahl aller Fehler, die bei der Generierung entstehen (vgl. Abschnitt 5.2). Je weniger Fehler, desto besser ist die Bewertung des Generierungsverfahren. Als achttes Kriterium wird die Anzahl der Ausreißer bewertet, die sich außerhalb der Zäune der Box-Plots befinden (vgl. Abschnitt 3.4). Da die Ausreißer Einfluss auf den Mittelwert haben können, führen weniger Ausreißer zu einer besseren Bewertung, als viele Ausreißer. Die Platzierung der Generierungsverfahren, die in den Vergleichen getätigt wurde, wird im neunten und letzten Kriterium bewertet. Eine Platzierung auf Platz eins wird am besten bewertet. Für die weiteren Platzierung erfolgt die Bewertung entsprechend. Aus den Kriterien Nummer drei bis neun ergibt sich für die Bewertung eines Testsets der in Tabelle 7 dargestellte Unterkatalog.

Tabelle 7: Unterkatalog für die Bewertung der Generierungsverfahren eines Testsets

Kriterium [Name]	Verfahren a [Punktzahl]	Verfahren b [Punktzahl]
Anzahl Replikationen		
Generierungszeit		
Breite der Konfidenzintervalle		
Konfidenzintervall überdeckt die Tourenlänge des Testsets		
Gesamtanzahl der Fehler		
Anzahl erzeugter Ausreißer		
Platzierung des Verfahrens		
Gesamtpunktzahl		

Für alle Generierungsverfahren werden die Ergebnisse der Unterkataloge im Kriterienkatalog, der in Tabelle 8 zu sehen ist, zusammengefasst. Des Weiteren erfolgt die Gesamtbewertung der Generierungsverfahren anhand dieses Kriterienkatalogs.

Tabelle 8: Kriterienkatalog für die Bewertung der Generierungsverfahren

Kriterium [Name]	Verfahren a [Punktzahl]	Verfahren b [Punktzahl]
Benötigte Parameter		
Nutzung in Simulationsprogrammen		
Anzahl Replikationen		
Generierungszeit		
Breite der Konfidenzintervalle		
Konfidenzintervall überdeckt die Tourenlänge des Testsets		
Gesamtanzahl der Fehler		
Anzahl erzeugter Ausreißer		
Platzierung des Verfahrens		
Gesamtpunktzahl		

6. Versuchsdurchführung und Auswertung der Generierungsverfahren

In diesem Kapitel werden die Kundenstandorte mit den ausgesuchten Generierungsverfahren erzeugt und ausgewertet. Die Monte-Carlo-Simulation wird dabei mit dem Programm Excel von Microsoft durchgeführt (vgl. Abschnitt 5.2). Im ersten Abschnitt dieses Kapitels werden Kundenstandorte mit einer Gleichverteilung erzeugt. Der zweite Abschnitt behandelt die Erzeugung der Kundenstandorte mit einer Normalverteilung. Innerhalb des dritten Abschnitts werden die Kundenstandorte einer Dreiecksverteilung folgend generiert. Sind dagegen einige Informationen, wie beispielsweise die Dichte der Kundenstandorte in der ausgewählten Region, bekannt, können die Parameter der Generierungsverfahren, die in den vorherigen drei Abschnitten genutzt wurden, darauf angepasst werden. Dieses Verfahren der Kundenstandortgenerierung wird im vierten Abschnitt dieses Kapitels behandelt.

6.1 Simulation mit einer Gleichverteilung

In diesem Abschnitt erfolgt die Generierung der Kundenstandorte mit Hilfe einer Gleichverteilung. Dieser Abschnitt ist dafür in vier Unterabschnitte, 6.1.1 bis 6.1.4, eingeteilt. Pro Unterabschnitt wird die Generierung der Kundenstandorte in einem Testset durchgeführt und ausgewertet. Neben der Anzahl der zu generierenden Kundenstandorte und der durchzuführenden Replikationen muss der zur Berechnung verwendeten Formel (vgl. Abschnitt 5.2) das Intervall $(a; b)$, auf dem die Generierung der x - und y -Koordinaten der Standorte stattfindet, vorgegeben werden. Dieses Intervall ist von der Begrenzung des jeweils verwendeten Testsets abhängig (vgl. Tabelle 5). Um die Entwicklung der Kennzahlen über eine steigende Anzahl an Replikationen beobachten zu können, werden für jedes Testset drei Simulationsläufe mit 100, 1000 und 10000 Replikationen durchgeführt (vgl. Abschnitt 5.2). Die Kennzahl Generierungsfehler setzt sich in diesem Abschnitt nur aus dem Fehler „doppelte Standorte“ zusammen, da bei Verwendung von Formel (13) keine „Berechnungsfehler“ entstehen und durch das Intervall $(a; b)$ zudem keine Standorte „außerhalb des Stadtgebietes“ erzeugt werden. Aus Gründen der Übersichtlichkeit sind die vollständigen Werte der berechneten Kennzahlen

und der zur Erstellung der Grafiken verwendeten Werte im Anhang in Tabelle 37 und Tabelle 38 zu finden.

6.1.1 Gleichverteilung im Testset E-n22-k4

Zuerst folgt die Auswertung der Gleichverteilung über das Testset E-n22-k4. Die begrenzenden Koordinaten für dieses Testset sind 165 LE auf der x-Achse und 265 LE auf der y-Achse (vgl. Tabelle 5). Daraus ergibt sich in der Simulation für die Generierung der x-Koordinate das Intervall (0; 165) und für die y-Koordinate das Intervall (0; 265). Das Testset beinhaltet 22 Kundenstandorte. Aus diesem Grund sind in der Simulation 22 Kundenstandorte zu erzeugen. In Tabelle 9 befindet sich eine Auswahl an Ergebniswerten der drei Simulationsläufe. Diese Auswahl stellt keine Wertung über die Bedeutung der einzelnen Werte dar, sondern ermöglicht einen ersten Überblick über die generierten Kennzahlen. Alle berechneten Werte sind, wie bereits angesprochen, vollständig im Anhang zu finden.

Tabelle 9: Gleichverteilung Testset E-n22-k4

Replikationen [Anzahl]	Tourenlänge [LE]		Doppelte Standorte [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	4530,39	4548,09	0	0,594
1000	4528,83	4533,02	6	2,360
10000	4542,45	4539,18	44	12,230

Für einen Simulationslauf mit 100 Replikationen ergibt sich eine Generierungszeit von 0,594 Sekunden (vgl. Tabelle 9). Die kürzeste Dauer für eine Replikation liegt bei 0 Sekunden, die längste bei 0,016 Sekunden. Der Mittelwert für die Generierungszeit von 0,001 Sekunden und der Median von 0 Sekunden zeigen, dass die benötigte Generierungszeit für 22 Standorte sehr kurz ist. In diesem Simulationslauf werden bei der Standortgenerierung keine doppelten Standorte erzeugt (vgl. Abbildung 12). Der Mittelwert der Tourenlänge beträgt 4530,39 LE, während der Median bei 4548,09 LE liegt und somit den Mittelwert übersteigt. Dies lässt die Vermutung einer linksschiefen Verteilung der Tourenlänge zu. Aufgrund dessen werden Werte, die größer als der Mittelwert sind, häufiger beobachtet. Der Abstand zwischen Median und Mittelwert

beträgt dabei 17,7 LE. Dies entspricht, bezogen auf den Mittelwert, einer Abweichung von 0,39 Prozent. Deswegen ist die Linksschiefe der Verteilung der Tourenlänge nicht stark ausgeprägt. Insofern kann die Verteilung der Tourenlänge auch beinahe als symmetrisch bezeichnet werden.

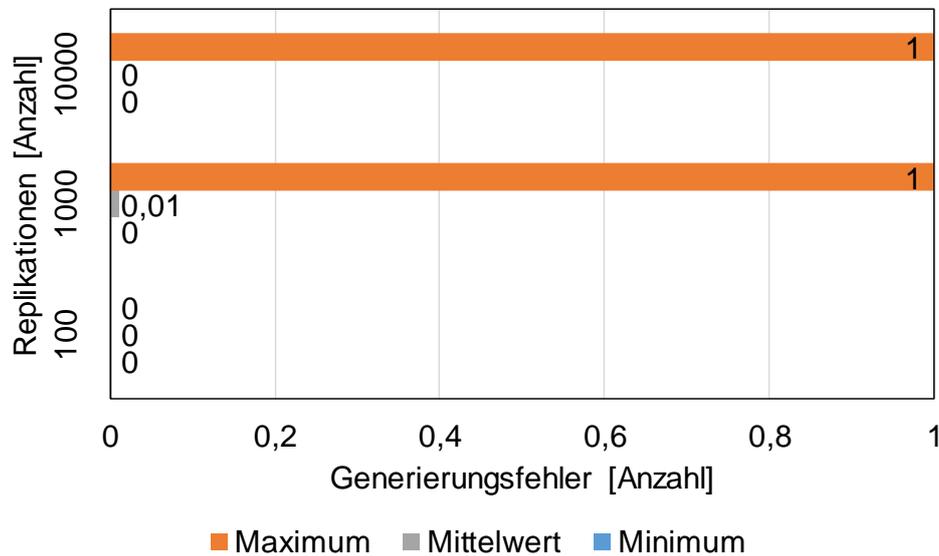


Abbildung 12: Gleichverteilung Generierungsfehler pro Replikation Testset E-n22-k4

Ein Simulationslauf mit 1000 Replikationen benötigt 2,36 Sekunden mit wiederum einer minimalen Generierungszeit von 0 Sekunden und einer maximalen von 0,016 Sekunden. Der Mittelwert und der Median liegen beide bei 0 Sekunden. Hinsichtlich der Generierungszeit lässt sich im Vergleich zu der Dauer der Generierung von 22 Standorten aus den 100 Replikationen kein Unterschied ausmachen. Im Gegensatz zum vorherigen Simulationslauf ergeben sich in diesem über alle 1000 Replikationen sechs doppelte Standorte als Fehler. Dabei liegt pro Replikation die minimale Fehleranzahl bei 0 und die maximale Fehleranzahl bei 1 (vgl. Abbildung 12). Der Mittelwert der Tourenlänge sinkt im Vergleich zum vorherigen Simulationslauf auf 4528,83 LE. Zusätzlich sinkt im Vergleich der Median auf 4533,02 LE. Die Vermutung einer linksschiefen Verteilung der Tourenlänge bleibt damit bestehen. Der Abstand zwischen Mittelwert und Median sinkt zudem auf 4,19 LE und entspricht damit, bezogen auf den Mittelwert, einer Abweichung von 0,09 Prozent. Auch hier ist die Linksschiefe der Verteilung der Tourenlänge nicht stark ausgeprägt und beinahe symmetrisch.

Für den Simulationslauf mit 10000 Replikationen werden 12,23 Sekunden benötigt. Die minimale Generierungszeit liegt bei 0 Sekunden, die maximale bei 0,016. Diese Werte sind mit denen aus den vorherigen Simulationsläufen identisch. Zusätzlich liegen der

Mittelwert und der Median bei 0 Sekunden. In diesem Simulationslauf werden insgesamt 44 doppelte Standorte als Fehler erzeugt. Die minimale Anzahl an Fehlern pro Replikation liegt bei 0, die maximale Anzahl bei 1 (vgl. Abbildung 12). Für den Mittelwert ergibt sich in diesem Simulationslauf ein Wert von 4542,45 LE und ist damit im Vergleich zu dem vorherigen Simulationslauf um 13,72 LE angestiegen. Der Median beträgt 4539,18 LE und ist ebenfalls angestiegen, jedoch nur um 6,16 LE. Damit ist der Mittelwert für diesen Simulationslauf größer als der Median. Dadurch ist die Vermutung einer rechtsschiefen Verteilung der Werte der Tourenlänge zulässig. Aufgrund dieser Vermutung, können für die Tourenlänge Werte die kleiner als der Mittelwert von 4542,45 LE sind, häufiger beobachtet werden. Die Differenz zwischen Mittelwert und Median beträgt jedoch nur 3,27 LE und entspricht damit einer Abweichung von 0,07 Prozent, bezogen auf den Mittelwert. Die rechtsschiefe Verteilung ist damit, wie die linksschiefe Verteilung in den vorherigen Simulationsläufen, kaum ausgeprägt und kann insofern als beinahe symmetrisch bezeichnet werden. Über alle Simulationsläufe betrachtet, verändern sich die Lagemaße der Tourenlänge, der Generierungsfehler und der Generierungszeit nicht signifikant.

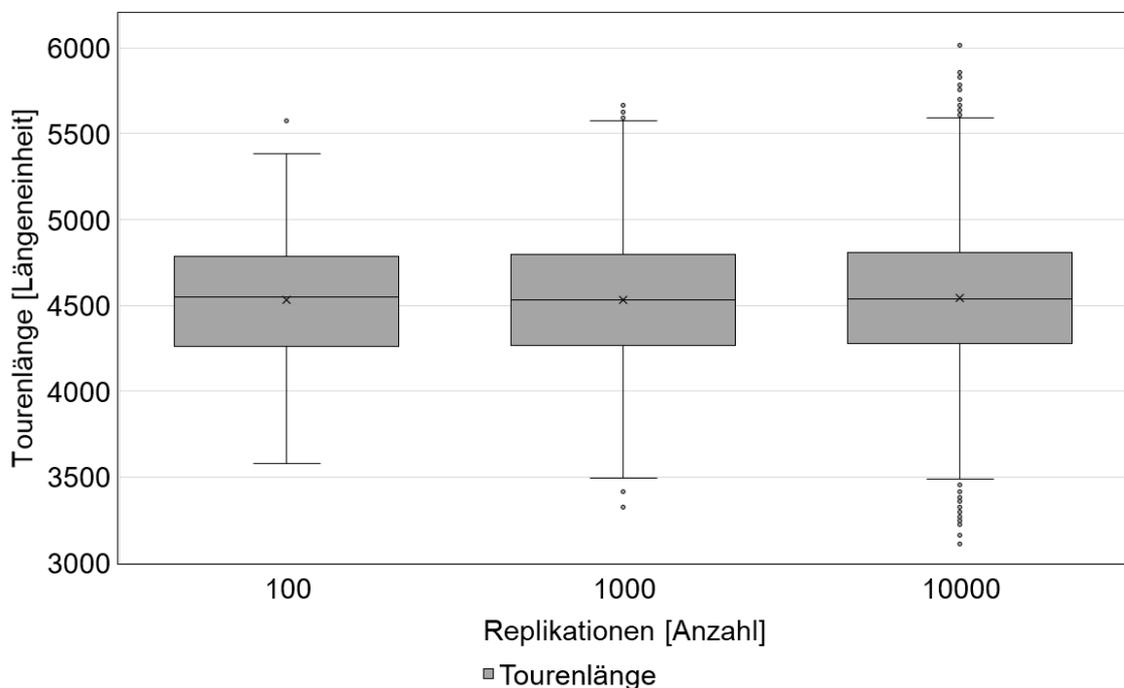


Abbildung 13: Gleichverteilung Box-Plots Testset E-n22-k4

In Abbildung 13 ist die Tourenlänge für jede Replikation der drei Simulationsläufe in ein Box-Plot-Diagramm eingetragen. Die im Vorhinein beobachtete geringe Abweichung des Mittelwerts, in der Abbildung jeweils mit einem x gekennzeichnet, vom Median, in der

Abbildung durch den horizontalen Strich in der Box gekennzeichnet, ist auch im Box-Plot-Diagramm zu erkennen. Zudem ist auch ersichtlich, dass die Verteilung der Tourenlänge von einer linksschiefen Verteilung in den ersten beiden Simulationsläufen, da der Mittelwert unterhalb des Medians eingezeichnet ist, im letzten Simulationslauf in eine rechtsschiefe Verteilung übergeht, weil sich der Mittelwert knapp über dem Median befindet. Im Diagramm sehen die Boxen der drei Replikationen beinahe gleich groß aus. Die Werte für den Interquartilsabstand bestätigten diese Beobachtung, für 100 Replikationen ergibt sich ein Wert von 526,77 LE, für 1000 Replikationen ein Wert von 526,75 und für 10000 Replikationen ein Wert von 530,71. Zudem liegt die untere Grenze der Box für 100 Replikationen bei 4263,47 LE, für 1000 Replikationen bei 4267,25 LE sowie für 10000 Replikationen bei 4279,92 LE. Daraus kann abgeleitet werden, dass sich die Werte der Tourenlänge zwischen dem ersten und dritten Quartil nicht signifikant voneinander unterscheiden und dies sich auch bei einer höheren Anzahl an Replikationen nicht ändert. Des Weiteren unterscheiden sich die Box-Plot-Diagramme hinsichtlich der Grenzen ihrer Whisker. Im ersten Simulationslauf beginnt der untere bei 3578,87 LE und der obere endet bei 5382,79 LE. Für den zweiten Simulationslauf ergeben sich 3495,32 LE und 5573,84 LE sowie für den Dritten 3488,2 LE und 5591,38 LE. Daraus ist erkennbar, dass sich die Streubreite der Tourenlänge für eine steigende Anzahl Replikationen vergrößert. Ein Unterschied zwischen den Replikationen lässt sich anhand der Abbildung jedoch ausmachen. Die Anzahl der Ausreißer steigt mit der Anzahl der Replikationen an. Bei 100 Replikationen existiert ein einziger Ausreißer, bei 1000 Replikationen sind es fünf Ausreißer und für 10000 Replikationen ergeben sich 61 Ausreißer. Die Auswirkungen der Ausreißer und der unterschiedlichen Streubreiten sind jedoch gering, da der Mittelwert von 4530,39 LE bei 100 Replikationen auf 4542,45 LE bei 10000 Replikationen steigt. Damit liegen die beiden Mittelwerte 12,06 LE auseinander, was einer Abweichung von 0,2 Prozent, bezogen auf den Mittelwert der 10000 Replikationen, entspricht.

In Abbildung 14 sind die approximierten 95% Konfidenzintervalle für die drei Simulationsläufe zu sehen. Aus der Abbildung kann entnommen werden, dass die Größe des Konfidenzintervalls mit einer größer werdenden Anzahl an Replikationen kleiner wird. Bei 100 Replikationen reicht die Größe des Konfidenzintervalls von 4412,92 bis 4647,86 LE, dies entspricht einer Breite des Intervalls von 234,94 LE. Für 1000 Replikationen verringert sich die Breite des Konfidenzintervalls auf 50,42 LE und verringert sich für 10000 Replikationen weiter auf 15,34 LE. Hieraus ist ableitbar, dass

die Ausreißer, die im Box-Plot-Diagramm in Abbildung 13 zu erkennen sind, keine Auswirkungen auf das Konfidenzintervall haben. Zudem kann für die Streuung der Werte der Tourenlänge innerhalb der 1000 und 10000 Replikationen ausgesagt werden, dass sich diese verringert, je größer die Anzahl der Werte ist.

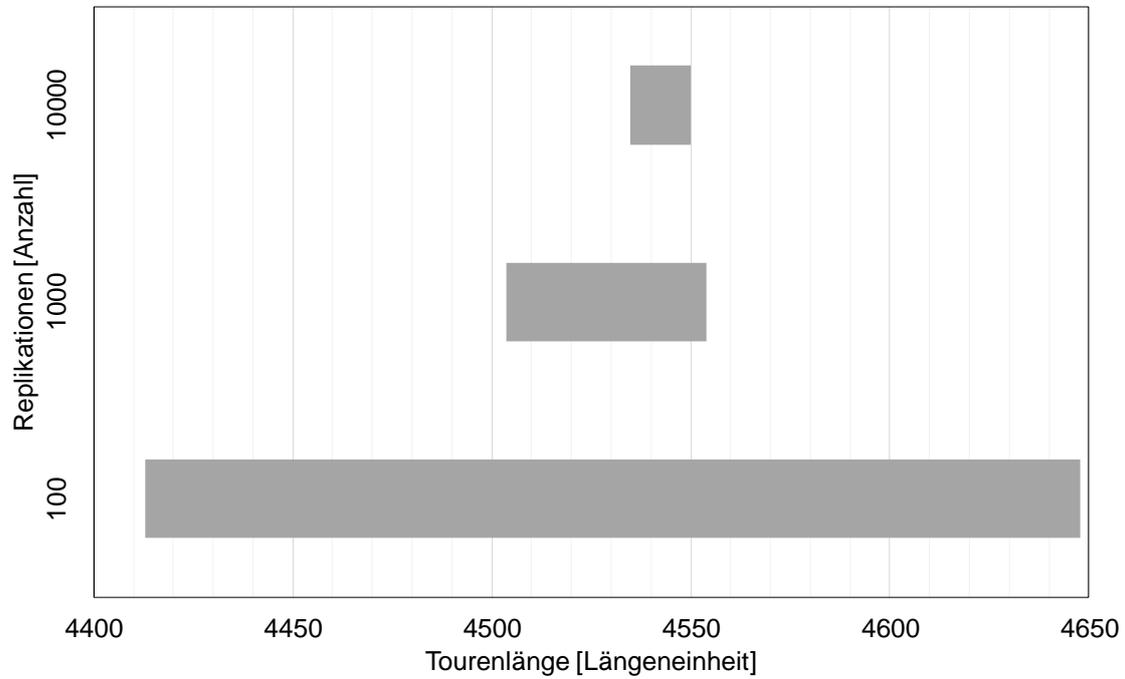


Abbildung 14: Gleichverteilung 95% Konfidenzintervalle Testset E-n22-k4

Mit dem Konfidenzintervall der 10000 Replikationen kann ausgesagt werden, dass der Wert der Tourenlänge mit einer Wahrscheinlichkeit von 95 Prozent vom approximierten Konfidenzintervall (4537,78;4550,12) überdeckt wird. Die Auswertung der drei Simulationsläufe hat auch hinsichtlich einer Erhöhung der Replikationen keine signifikanten Veränderungen der Lagemaße der Tourenlänge, der Generierungsfehler und der Generierungszeit ergeben. Deshalb können die Ergebnisse jedes Simulationslaufs im Vergleich der Generierungsverfahren in Abschnitt 7.1 genutzt werden. Weil die Größenordnung der Tourenlänge mit dem approximierten 95% Konfidenzintervall des dritten Simulationslaufs am präzisesten abgeschätzt werden kann, werden die Ergebnisse dieses Simulationslaufs für den Vergleich herangezogen.

6.1.2 Gleichverteilung im Testset CMT14

Nachfolgend wird die Gleichverteilung der Kundenstandorte im Testset CMT14 ausgewertet. Für dieses Testset sind die begrenzenden Koordinaten 95 LE auf der x-

Achse und 85 LE auf der y-Achse (vgl. Tabelle 5). Damit werden die x-Koordinaten der generierten Kundenstandorte im Intervall (0; 95) und die y-Koordinaten im Intervall (0; 85) erzeugt. Da das Testset 100 Kundenstandorte enthält, werden in jeder Replikation 100 Kundenstandorte erzeugt. Tabelle 10 beinhaltet eine Auswahl an Ergebniswerten aus den drei Simulationsläufen. Diese Auswahl stellt keine Bewertung der abgebildeten Ergebniswerte dar. Sämtliche berechneten Werte können, wie Eingangs dieses Abschnitts beschrieben, im Anhang eingesehen werden.

Tabelle 10: Gleichverteilung Testset CMT14

Replikationen [Anzahl]	Tourenlänge [LE]		Doppelte Standorte [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	8508,62	8538,57	65	0,359
1000	8514,98	8521,08	589	3,297
10000	8519,63	8517,40	6015	33,469

Der Simulationslauf mit 100 Replikationen dauert 0,359 Sekunden. Dabei liegt die kürzeste Dauer für die Durchführung einer Replikation bei 0 Sekunden, die längste bei 0,016 Sekunden. Als Mittelwert ergibt sich eine Generierungszeit von 0,001 Sekunden. Wird zusätzlich der Median von 0 Sekunden für die Generierungszeit betrachtet, so lässt sich feststellen, dass die Generierungszeit für 100 Standorte, und damit für eine Replikation, sehr gering ist. Insgesamt werden als Fehler 65 doppelte Standorte erzeugt. Die minimale Anzahl an Fehlern innerhalb einer Replikation liegt bei 0, maximal werden 4 Fehler in einer Replikation erzeugt (vgl. Abbildung 15). Dass es sich bei diesem Maximalwert für die Anzahl der Fehler um einen Ausreißer handelt, ist am Mittelwert von 0,65 Fehlern und dem Median von 0 Fehlern zu erkennen. Bei 100 Replikationen beträgt der Mittelwert der Tourenlänge 8508,62 LE. Der Median liegt bei 8538,57 LE und ist damit größer als der Mittelwert. Auf Basis dieser Beobachtung ist von einer linksschiefen Verteilung der Tourenlänge auszugehen. Mehr als 50 Prozent der Werte für die Tourenlänge sind damit größer als der Mittelwert. Die Ausprägung dieser Linksschiefe ist jedoch nicht stark, da die Abweichung zwischen Mittelwert sowie Median 29,95 LE beträgt und damit, bezogen auf den Mittelwert, eine Abweichung von lediglich 0,35 Prozent darstellt. Deswegen kann die Verteilung der Tourenlänge auch als beinahe symmetrisch bezeichnet werden.

Für den Simulationslauf mit 1000 Replikationen ergibt sich eine Generierungszeit von 3,297 Sekunden (vgl. Tabelle 10). Die minimale Generierungszeit liegt, wie im vorherigen Simulationslauf, bei 0 Sekunden und genauso liegt die maximale Generierungszeit bei 0,016 Sekunden. Die Werte des Medians, mit 0 Sekunden, und des Mittelwerts, mit 0,001 Sekunden, stimmen mit denen aus dem vorherigen Simulationslauf für 100 Replikationen überein. Die Generierungszeit für eine Replikation ist damit gering. Bei der Generierung werden insgesamt 589 doppelte Standorte erzeugt. Die minimale Anzahl dieses Fehlers beträgt 0, die maximale Anzahl 4 (vgl. Abbildung 15).

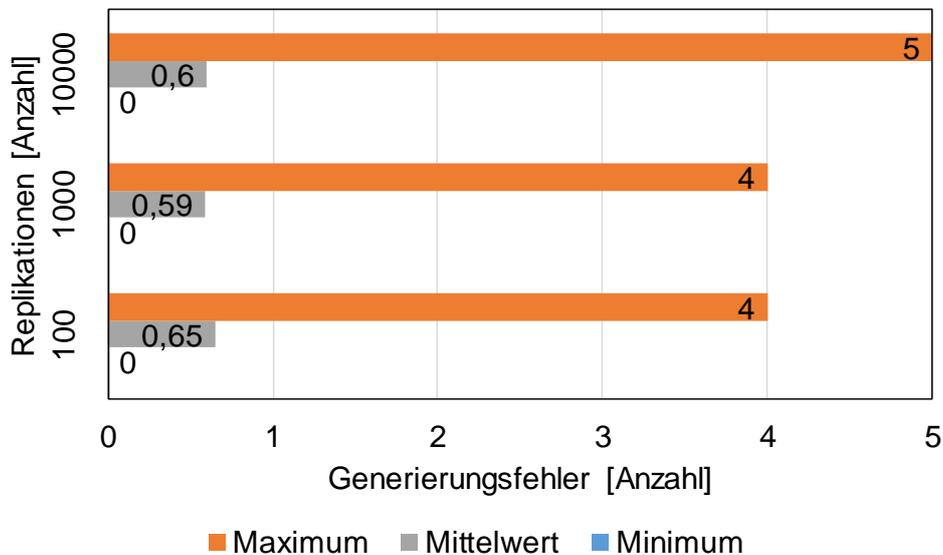


Abbildung 15: Gleichverteilung Generierungsfehler pro Replikation Testset CMT14

Bei dieser maximalen Anzahl handelt es sich um einen Ausreißer, da der Mittelwert bei 0,59 und der Median bei 0 liegt. Im Vergleich zum vorherigen Simulationslauf mit 100 Replikationen steigt der Mittelwert der Tourenlänge auf 8514,98 LE an und der Median sinkt auf 8521,08 LE. Damit ist der Mittelwert auch in diesem Simulationslauf kleiner als der Median und die Beobachtung einer linksschiefen Verteilung bleibt bestehen. Da sich der Abstand zwischen Mittelwert und Median weiter auf 6,1 LE verkürzt und so, bezogen auf den Mittelwert, einer Abweichung von 0,07 Prozent entspricht, ist diese linksschiefe Verteilung weiterhin schwach ausgeprägt und kann auch als beinahe symmetrisch bezeichnet werden.

Die 10000 Replikationen des dritten Simulationslaufs werden in 33,469 Sekunden durchgeführt (vgl. Tabelle 10). Die minimal Generierungszeit einer Replikation liegt bei 0 Sekunden. Maximal wird eine Generierungszeit von 0,016 Sekunden benötigt. Wie in den vorherigen beiden Simulationsläufen beträgt der Median 0 Sekunden und der

Mittelwert 0,001 Sekunden. Damit tritt im Vergleich zu den vorherigen Simulationsläufen keine Veränderung der Generierungszeit pro Replikation auf. Innerhalb dieses Simulationslaufs werden 6015 doppelte Standorte erzeugt. Die minimale Anzahl dieses Fehlers liegt bei 0 doppelten Standorten, die maximale bei 5 (vgl. Abbildung 15). Anhand des Medians von 0 Fehlern und dem Mittelwert von 0,6 Fehlern ist erkennbar, dass es sich bei dem Maximalwert von 5 um einen Ausreißer handelt. Für die Tourenlänge ergibt sich in diesem Simulationslauf ein Mittelwert von 8519,63 und der Median liegt bei 8517,4. Damit ist der Mittelwert größer als der Median. Die linksschiefe Verteilung der Tourenlänge aus den vorherigen Simulationsläufen wird damit zu einer rechtsschiefen Verteilung. Entsprechend können mehr Werte beobachtet werden, die kleiner als der Mittelwert sind. Da die Abweichung des Medians vom Mittelwert bei 2,23 LE liegt, was einer Abweichung, bezogen auf den Mittelwert, von 0,03 Prozent entspricht, ist die Rechtsschiefe der Verteilung der Tourenlänge schwach ausgeprägt. Aus diesem Grund kann auch hier von einer beinahe symmetrischen Verteilung der Tourenlänge gesprochen werden. Über alle Simulationsläufe betrachtet, führt eine Erhöhung der Replikationsanzahl um das Zehnfache ungefähr zu einer Verzehnfachung der erzeugten doppelten Standorte sowie einer Verzehnfachung der Generierungszeit (vgl. Tabelle 10). Außerdem verändern sich die Lagemaße der Tourenlänge, der Generierungszeit sowie der Generierungsfehler über alle Simulationsläufe nicht signifikant.

Die Tourenlängen aller Replikationen der drei Simulationsläufe sind in Abbildung 16 für jeden Simulationslauf in ein separates Box-Plot-Diagramm eingetragen. Die über die drei Replikationen kleiner werdende Abweichung des Mittelwerts, in der Abbildung durch ein x gekennzeichnet, vom Median, mit einem horizontalen Strich innerhalb der Box gekennzeichnet, ist in der Abbildung zu erkennen. Zudem ist auch die Veränderung der linksschiefen Verteilung der Tourenplanung aus den ersten beiden Simulationsläufen zu einer rechtsschiefen Verteilung im dritten und letzten Simulationslauf ersichtlich, denn innerhalb der Box-Plot-Diagramme für 100 und 1000 Replikationen befindet sich das x des Mittelwerts unterhalb der horizontalen Linie des Medians. Bei dem Simulationslauf mit 10000 Replikationen ist das x des Mittelwerts knapp oberhalb der horizontalen Linie des Medians. Die Box des Box-Plot-Diagramms für 100 Replikationen erscheint in Abbildung 16 größer zu sein als die Box der Diagramme für 1000 und 10000 Replikationen. Diese Beobachtung wird durch die Werte für die Interquartilsabstände der drei Simulationsläufe bestätigt. Für 100 Replikationen beträgt dieser 430,3 LE, bei 1000

Replikationen 414,8 LE und für 10000 Replikationen 417,66 LE. Insgesamt sind die unteren Grenzen und damit die Intervalle, innerhalb derer sich 50 Prozent der Werte für die Tourenlänge befinden, bei allen Simulationsläufen nah beieinander. Beim Simulationslauf mit 100 Replikationen beträgt die untere Grenze 8287,87 LE, für 1000 Replikationen 8307,57 LE sowie für 10000 Replikationen 8312,13 LE.

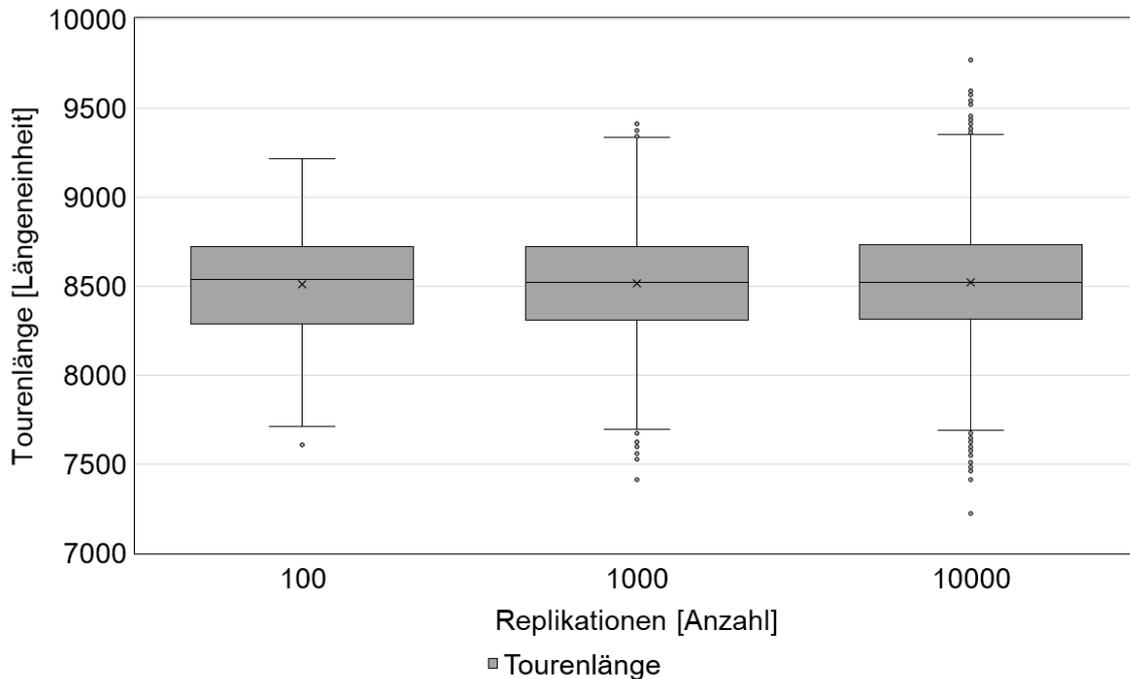


Abbildung 16: Gleichverteilung Box-Plots Testset CMT14

Außerdem weisen die Grenzen der Whisker für die jeweiligen Box-Plot-Diagramme eine wertmäßige Ähnlichkeit auf, denn diese liegen für den ersten Simulationslauf bei 7707,82 und 9218,2 LE, für den Zweiten bei 7691,46 und 9337,51 LE sowie für den Dritten bei 7686,79 und 9353,18 LE. Daraus kann geschlossen werden, dass sich die Grenzen, in denen die Werte für die Tourenlänge liegen, über alle Replikationen hinweg in einem ähnlichen Rahmen bewegen und sich nicht signifikant unterscheiden, obwohl die Replikationen vom ersten auf den dritten Simulationslauf ver Hundertfacht werden. Der sichtbarste Unterschied für die drei Simulationsläufe in Abbildung 16 ist die Anzahl der Ausreißer. Bei 100 Replikationen existiert ein Ausreißer, für 1000 Replikationen ergeben sich 14 Ausreißer und bei 10000 Replikationen entstehen 90 Ausreißer. In den Ausreißern der 10000 Replikationen befinden sich 55 Ausreißer unterhalb des unteren Whiskers und 35 oberhalb des oberen Whiskers. Da sich der für Ausreißer anfällige Mittelwert von 8508,62 LE bei 100 Replikationen auf 8519,63 LE bei 10000 Replikationen erhöht, lässt sich auf einen geringen Einfluss der Ausreißer schließen.

Die approximierten 95% Konfidenzintervalle der drei Simulationsläufe sind in Abbildung 17 zu sehen. Die Breite der Konfidenzintervall wird bei einer steigenden Anzahl an Replikationen kleiner. Bei 100 Replikationen spannt sich das Konfidenzintervall von 8328,83 bis 8688,42 LE auf, das entspricht einer Breite von 359,59 LE. Für 1000 Replikationen liegen die Grenzen des Konfidenzintervalls bei 8488,87 bis 8541,08 LE, die Breite sinkt damit auf 52,21 LE. Die geringste Breite des Konfidenzintervalls mit 12,92 LE wird bei 10000 Replikationen im Intervall von 8513,17 bis 8526,09 LE erreicht. Daraus kann geschlossen werden, dass die Ausreißer, welche im Box-Plott-Diagramm in Abbildung 16 sichtbar sind, keine Auswirkungen auf die Konfidenzintervalle haben.

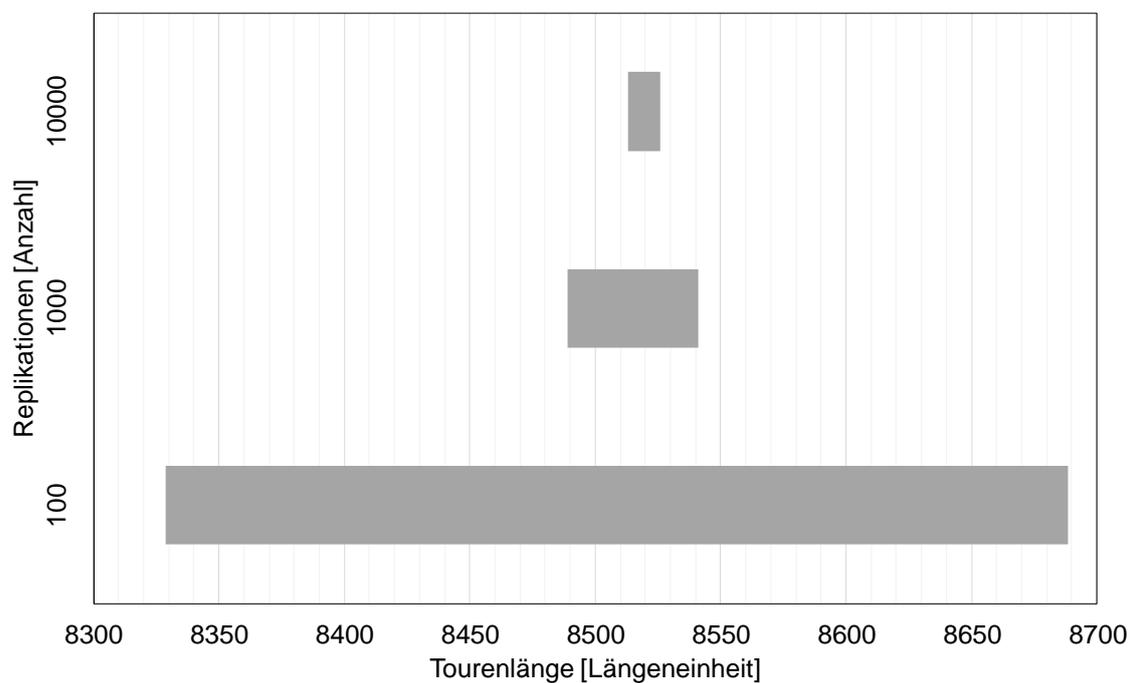


Abbildung 17: Gleichverteilung 95% Konfidenzintervalle Testset CMT14

Die Breite der Konfidenzintervalle nähert sich mit der steigenden Anzahl an Replikationen dem Mittelwert der Tourenlänge des zugehörigen Simulationslaufs an. Aus diesem Grund kann mit Hilfe des Konfidenzintervalls des dritten Simulationslaufs mit 10000 Replikationen ausgesagt werden, dass der Wert der Tourenlänge mit einer Fehlerwahrscheinlichkeit von 5 Prozent durch das approximierte Konfidenzintervall mit den Grenzen von 8513,17 bis 8526,09 LE überdeckt wird. Die Auswertung hat auch für eine hohe Anzahl an Replikationen keine signifikante Veränderung hinsichtlich der Lagemaße der Tourenlänge, der Generierungsfehler und der Generierungszeit ergeben. Deswegen können im Vergleich der Generierungsverfahren im Abschnitt 7.1 die Ergebnisse aller Simulationsläufe genutzt werden. Jedoch wird die Größenordnung der

Tourenlänge am präzisesten mit dem approximierten 95% Konfidenzintervall des dritten Simulationslaufs abgeschätzt, weswegen die Ergebnisse dieses Simulationslaufs in den Vergleich einfließen.

6.1.3 Gleichverteilung im Testset C110_1

Im Folgenden wird die Auswertung der Gleichverteilung der Kundenstandorte über das Testset C110_1 vorgenommen. Die begrenzenden Koordinaten dieses Testsets liegen für die x-Achse bei 500 LE und für die y-Achse ebenfalls bei 500 LE (vgl. Tabelle 5). Da die begrenzenden Koordinaten der x- und y-Achse gleich sind, werden die x- und y-Koordinaten der Standorte im selben Intervall erzeugt. So ergibt sich für die Generierung der gleichverteilten Kundenstandorte sowohl für die x-Koordinate als auch die y-Koordinate das Intervall (0,500). Das Testset enthält 1000 Kundenstandorte. Deswegen werden innerhalb jeder Replikation 1000 Kundenstandorte erzeugt. Für die Testsets C110_1 und RC110_1 werden dieselbe Anzahl an Kundenstandorten in denselben Intervallen erzeugt. Bei identischer Parametrisierung der Gleichverteilung ergeben sich immer identische Werte. Aus diesem Grund werden sich die Ergebnisse der Generierung dieser beiden Testsets nicht zu den Ergebnissen des Testsets C110_1 unterscheiden. Deswegen wird die Standortgenerierung nur für das Testset C110_1 durchgeführt und die Ergebnisse gelten entsprechend für die oben genannten Testsets. Innerhalb von Tabelle 11 ist eine Auswahl an Ergebniswerten für die drei durchgeführten Simulationsläufe abgebildet. Bei dieser Auswahl handelt es sich, wie auch zuvor, um keine Bewertung der dargestellten Werte. Für alle berechneten Ergebniswerte wird auf Tabelle 37 im Anhang verwiesen.

Tabelle 11: Gleichverteilung Testset C110_1

Replikationen [Anzahl]	Tourenlänge [LE]		Doppelte Standorte [Anzahl]	Generierungszeit [s]
	Mittelwert	Median		
			Gesamt	Gesamt
100	471929,83	472303,42	191	1,485
1000	472261,27	472327,66	1978	14,969
10000	472313,07	472246,20	20079	148,109

Die Generierung für 100 Replikationen nimmt insgesamt 1,485 Sekunden in Anspruch. Die minimale Zeit für eine Replikation liegt bei 0 Sekunden und die maximale bei 0,016 Sekunden. Für den Mittelwert ergibt sich eine Generierungszeit von 0,012 Sekunden, der Median beträgt 0,016 Sekunden, wodurch die Lagemaße dieses Testsets von den zuvor besprochenen Testsets E-n22-k4 und CMT14 abweichen. Die Kunden pro QLE und damit die Dichte der Kundenstandorte der Testsets ist als Grund für diese Abweichung auszuschließen, da diese im Falle des Testsets E-n22-k4 bei 0,0005 und im Falle des Testsets CMT14 bei 0,012 liegt. Für das vorliegende Testset liegt diese jedoch bei 0,004 Kunden pro QLE und damit genau zwischen den Werten der vorherigen Testsets. Die im Mittel und Median abweichende Generierungszeit ist durch die erhöhte Anzahl der zu generierenden Kundenstandorte begründet. Diese liegt für E-n22-k4 bei 22, für CMT14 bei 100 und für das vorliegende Testset bei 1000. Damit ist diese Anzahl um das zehnfache größer und spiegelt sich in einer höheren Generierungszeit wider. Bei der Generierung der Standorte für die 100 Replikationen werden 191 doppelte Standorte erzeugt. Minimal wird kein doppelter Standort erzeugt, maximal ergeben sich 7 doppelte Standorte (vgl. Abbildung 18). Da der Median bei 2 Fehlern und der Mittelwert bei 1,91 Fehlern liegt, handelt es sich bei dem Maximalwert um einen Ausreißer. Für diesen Simulationslauf ergibt sich ein Mittelwert der Tourenlänge von 471929,83 LE. Der Median liegt bei 472303,42 LE und ist damit um 373,59 LE größer als der Mittelwert. Hieraus ergibt sich eine linksschiefe Verteilung der Tourenlänge. Damit sind für die Tourenlänge bei 100 Replikationen mehr Werte zu beobachten, die größer als der Mittelwert sind. Bezogen auf den Mittelwert ergibt sich jedoch eine Abweichung von 0,08 Prozent, woraus zu schließen ist, dass die Linksschiefe schwach ausgeprägt ist und die Werte für die Tourenlänge somit beinahe symmetrisch verteilt sind.

Der Simulationslauf für 1000 Replikationen dauert 14,969 Sekunden. Die minimale Generierungszeit beträgt 0 Sekunden. Anders als im vorherigen Simulationslauf mit 100 Replikationen ergibt sich für die maximale Generierungszeit ein Wert von 0,031 Sekunden. Der Median bleibt mit 0,016 Sekunden hingegen gleich und für den Mittelwert kann ein Wert von 0,013 Sekunden bestimmt werden. Daraus wird ersichtlich, dass es sich bei dem Maximalwert der Generierungszeit um einen Ausreißer handelt. Bei der Generierung der Kundenstandorte für diesen Simulationslauf ergeben sich insgesamt 1978 doppelte Standorte. Minimal wird dieser Fehler 0-mal erzeugt und maximal 8-mal (vgl. Abbildung 18). Anhand des Medians von 2 Fehlern und dem Mittelwert mit 1,98 Fehlern ist festzustellen, dass es sich bei der maximalen Fehleranzahl um einen Ausreißer

handelt. Der Mittelwert der Tourenlänge steigt, im Vergleich zum vorherigen Simulationslauf, auf 472261,27 LE. Gleichzeitig erhöht sich auch der Median auf 472327,66 LE. Dabei verringert sich der Abstand zwischen Mittelwert und Median auf 66,39 LE. Bezogen auf den Mittelwert ergibt sich eine Abweichung von 0,01 Prozent. Somit kann eine linksschiefe Verteilung, die aus dem gegenüber dem Mittelwert höheren Wert des Medians resultiert, als schwach oder auch, aufgrund der geringen Abweichung, als beinahe symmetrische Verteilung bezeichnet werden.

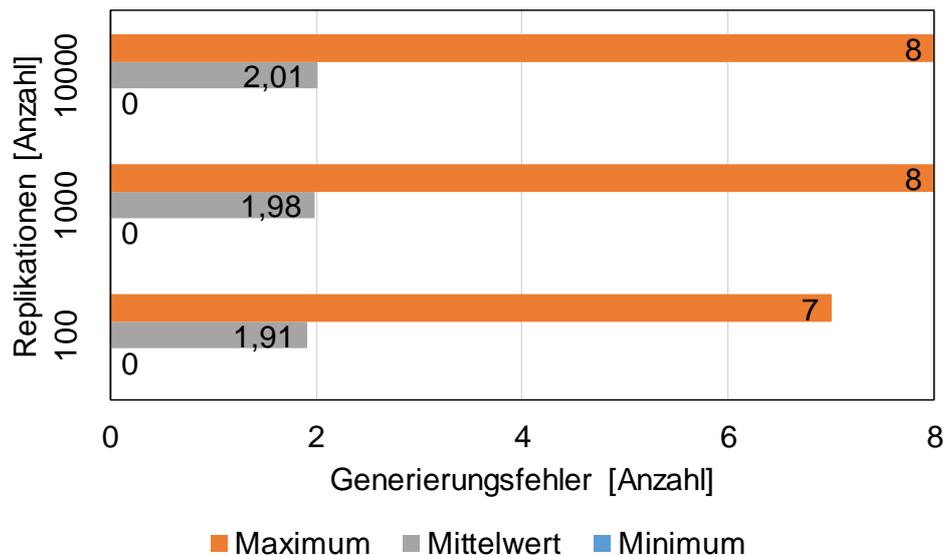


Abbildung 18: Gleichverteilung Generierungsfehler pro Replikation Testset C110_1

Für den Simulationslauf mit 10000 Replikationen wird eine Generierungszeit von 148,109 Sekunden benötigt. Dabei beträgt die minimal benötigte Zeit 0 Sekunden. Wie schon im vorherigen Simulationslauf steigt die maximale Generierungszeit, in diesem Simulationslauf bis auf 0,094 Sekunden, an. Der Median bleibt mit 0,016 Sekunden unverändert, der Mittelwert steigt auf 0,014. Anhand der Größen für den Mittelwert und dem Median lässt sich aussagen, dass die maximale Generierungszeit einen Ausreißer darstellt. Über die 10000 Replikationen werden insgesamt 20079 doppelte Standorte erzeugt. Dabei liegt die minimale Anzahl für diesen Fehler bei 0 und die maximale bei 8 (vgl. Abbildung 18). Da der Median bei 2 und der Mittelwert bei 2,01 Fehlern liegt, handelt es sich bei dem maximalen Fehlerwert um einen Ausreißer. Der Mittelwert steigt im Vergleich zum vorherigen Simulationslauf auf 472313,07 LE und der Median sinkt auf 472246,2 LE. Dies führt zu einer rechtsschiefen Verteilung. Der Abstand zwischen Median und Mittelwert verändert sich nicht signifikant von 66,39 auf 66,87 LE. Dies entspricht, auf den Mittelwert bezogen, einer Abweichung von 0,01 Prozent. Deswegen

kann auch von einer beinahe symmetrischen Verteilung gesprochen werden. Über alle Simulationsläufe betrachtet, führt eine Verzehnfachung der Anzahl der Replikationen zu einer Verzehnfachung der doppelten Standorte sowie einer Verzehnfachung der benötigten Gesamtgenerierungszeit. Zudem verändern sich die Lagemaße der Tourenlänge, der Generierungszeit und der Generierungsfehler über alle Simulationsläufe nicht signifikant.

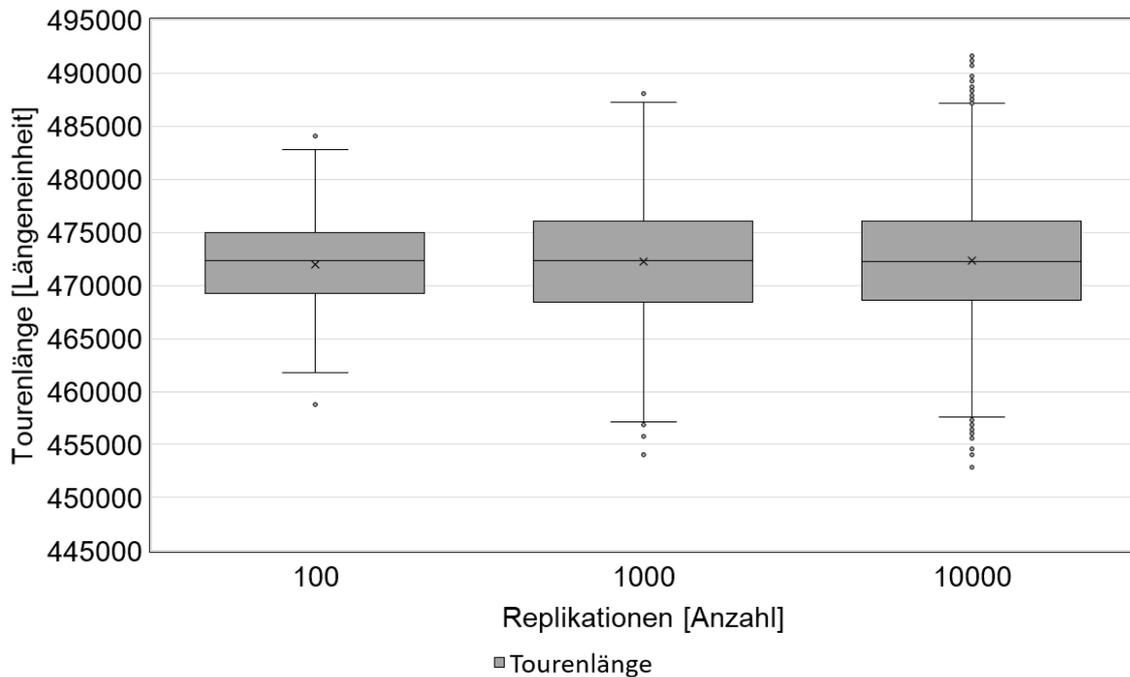


Abbildung 19: Gleichverteilung Box-Plots Testset C110_1

Für alle drei Simulationsläufe sind die zugehörigen Tourenlängen in Abbildung 19 in ein eigenes Box-Plot-Diagramm eingetragen. Der Mittelwert ist in den Box-Plots durch ein x gekennzeichnet, wohingegen der Median durch die horizontale Linie innerhalb der Box markiert ist. In den Box-Plot-Diagrammen ist über den Verlauf der drei Simulationsläufe erkennbar, dass sich der Mittelwert und der Median einander annähern. Auch die Veränderung von einer linksschiefen Verteilung aus den ersten beiden Simulationsläufen hin zu einer rechtsschiefen Verteilung im letzten Simulationslauf, kann in den Box-Plots erkannt werden. Dies ist der Fall, weil sich der Mittelwert für 100 und 1000 Replikationen unterhalb des Medians und für 10000 Replikationen oberhalb ebenjenes befindet. Auch die geringen Abstände zwischen beiden Lagemaßen ist den Diagrammen zu entnehmen. Die Größe der Box für den ersten Simulationslauf mit 100 Replikationen erscheint in Abbildung 19 kleiner zu sein als die Boxen für die folgenden beiden Simulationsläufe, die sich in ihrer Größe ähneln. Die Werte der Interquartilsabstände bestätigen diese

Beobachtung. Für 100 Replikationen liegt der Interquartilsabstand bei 6120,6 LE, für 1000 Replikationen bei 7627,6 und für 10000 Replikationen bei 7392,24 LE. Damit befindet sich die untere Grenze der Box für 100 Replikationen bei 469260,27 LE, für 1000 Replikationen bei 468448,81 LE sowie für 10000 Replikationen bei 468637,76 LE. Die Werte der Tourenlänge zwischen dem ersten und dem dritten Quartil unterscheiden sich für 1000 und 10000 Replikationen kaum. Im Vergleich zu 100 Replikationen unterscheiden sich die Interquartilsabstände um 1000 LE. Daraus lässt sich schließen, dass die Unterschiede der Interquartilsabstände und unteren Grenzen der Boxen zwischen dem ersten und zweiten Simulationslauf auffällig sind. Ein Unterschied in den Box-Plot-Diagrammen, der in Abbildung 19 erkennbar ist, ist die Länge der Whisker. Für 100 Replikationen sind der obere und untere Whisker sichtbar kleiner als die für 1000 und 10000 Replikationen. In Werten ausgedrückt bedeutet dies, dass der untere Whisker 461774,73 und der obere 482766,65 LE beträgt. Dies ist ein deutlicher Unterschied zu den beinahe identischen Werten der Whisker für 1000 und 10000 Replikationen. Bei 1000 Replikationen betragen die Werte 457092,33 LE für den unteren und 487204,61 LE für den oberen Whisker. Bei 10000 Replikationen endet der untere bei 457608,71 LE und der obere bei 487109,93 LE. Ein weiterer sichtbarer Unterschied ist die Anzahl der Ausreißer. Im ersten Simulationslauf existiert 1 oberer und 1 unterer Ausreißer, im zweiten 3 untere und 1 oberer, im letzten Simulationslauf existieren 37 untere und 39 obere Ausreißer. Dies lässt die Vermutung zu, dass sich bei einer Erhöhung von 100 Replikationen auf 10000 Replikationen die Streubreite der Tourenlänge vergrößert und mehr Extremwerte erzeugt werden. Der Einfluss sowohl dieser beiden Faktoren als auch der signifikanten Abweichungen der Interquartilsabstände ist dennoch als gering einzuschätzen, da sich der Mittelwert, der für Ausreißer anfällig ist, von 471929,83 LE bei 100 Replikationen auf 472313,07 LE bei 10000 Replikationen erhöht und dies einem Unterschied von 383,24 LE oder, bezogen auf den Mittelwert der 10000 Replikationen, einem Unterschied von 0,08 Prozent entspricht.

In Abbildung 20 sind die approximierten 95% Konfidenzintervalle der drei durchgeführten Simulationsläufe zu sehen. Für eine steigende Anzahl an Replikationen reduziert sich die Breite des Konfidenzintervalls. Beim ersten Simulationslauf mit 100 Replikationen beträgt die Breite des 95% Konfidenzintervalls 18718,66 LE. Im zweiten Simulationslauf mit 1000 Replikationen sinkt die Breite des 95% Konfidenzintervalls auf 1981,27 und erreicht im dritten Simulationslauf mit 10000 Replikationen und einer Breite von 284,63 LE ihr Minimum. Die Ausreißer der Tourenlänge sowie die Vergrößerung der

Streubreite, die in Abbildung 19 sichtbar sind, nehmen damit keinen signifikanten Einfluss auf die Breite des Konfidenzintervalls.

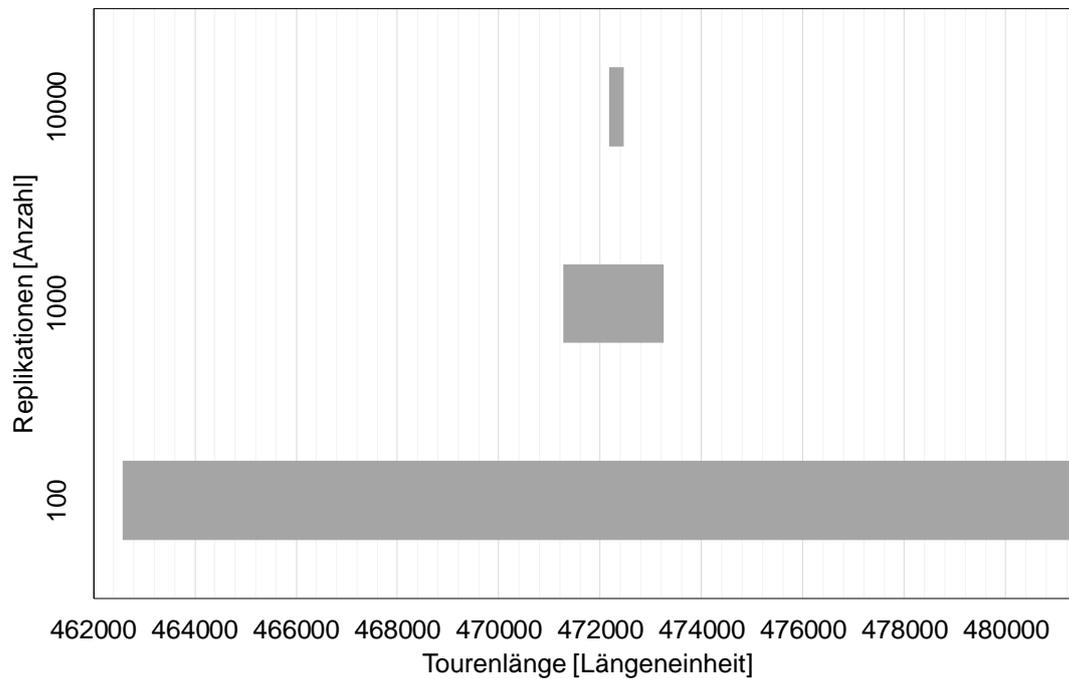


Abbildung 20: Gleichverteilung 95% Konfidenzintervalle Testset C110_1

Weiterhin nähert sich die Breite des Konfidenzintervalls mit einer steigenden Anzahl an Replikationen dem Mittelwert der Tourenlänge an. Deswegen kann auf Basis des 95% Konfidenzintervalls für die 10000 Replikationen ausgesagt werden, dass der Wert der Tourenlänge mit einer Wahrscheinlichkeit von 95 Prozent vom approximierten Konfidenzintervall der Dimension (472170,75;472455,38) überdeckt wird. Bei der Auswertung der drei Simulationsläufe kann, auch bei einer Erhöhung der Replikationen, keine signifikante Veränderung der Lagemaße der Tourenplanung, der Generierungszeit und der Generierungsfehler festgestellt werden. Deswegen können die Ergebnisse aus jedem Simulationslauf für den Vergleich der Generierungsverfahren im Abschnitt 7.1 herangezogen werden. Für die Größenordnung der Tourenlänge kann die präziseste Voraussage mit Hilfe des approximierten Konfidenzintervall des dritten Simulationslaufs getroffen werden, weswegen die Ergebnisse dieses Simulationslaufs im Vergleich genutzt werden.

6.1.4 Gleichverteilung im Testset Antwerp1

Zuletzt erfolgt in diesem Abschnitt die Auswertung der Gleichverteilung über das Testset Antwerp1. In dem angesprochenen Testset sind die begrenzenden Koordinaten der x- und y-Achse jeweils 2000 LE (vgl. Tabelle 5). Da die begrenzenden Koordinaten für beide Achsen identisch sind, werden die Koordinaten der Standorte auch auf demselben Intervall und damit auf (0,2000) erzeugt. Das Testset enthält 6001 Standorte. Deswegen werden in jeder Replikation 6001 Kundenstandorte generiert. Innerhalb von Tabelle 12 befindet sich eine Auswahl an Ergebniswerten. Diese Auswahl stellt keine Bewertung der Ergebniswerte dar und alle berechneten Werte für dieses Testset können im Anhang in Tabelle 38 eingesehen werden.

Tabelle 12: Gleichverteilung Testset Antwerp1

Replikationen [Anzahl]	Tourenlänge [LE]		Doppelte Standorte [Anzahl]	Generierungszeit [s]
	Mittelwert	Median		
			Gesamt	Gesamt
100	11331931,77	11333757,59	500	7,859
1000	11333827,46	11333268,87	4602	77,406
10000	11334315,97	11333328,41	45300	788,047

Der erste Simulationslauf mit 100 Replikationen benötigt eine Generierungszeit von 7,859 Sekunden (vgl. Tabelle 38). Der minimale Wert der Generierungszeit liegt dabei bei 0,062 Sekunden und der maximale Wert bei 0,079 Sekunden für eine Replikation. Als Mittelwert der Generierungszeit ergeben sich 0,075 Sekunden, wobei der Median bei 0,078 Sekunden liegt. Werden die beiden Lageparameter berücksichtigt, kann festgestellt werden, dass es sich bei der minimalen Generierungszeit um einen Ausreißer handelt und die Generierungszeit zumeist in der Nähe des maximalen Wertes liegen wird. Auch wird die Vermutung aus der vorherigen Auswertung des Testsets „C110_1“ bestätigt, dass die Dichte der Kundenstandorte nicht der Grund für eine erhöhte Generierungszeit ist. Denn das vorliegende Testset hat eine Kundendichte von 0,0015 Kunden pro QLE und liegt damit unterhalb des Wertes von 0,004 des Testsets „C110_1“. Das vorliegende Testset benötigt aber eine Generierungszeit von 7,859 Sekunden, wohingegen die Generierungszeit für „C110_1“ bei 1,485 Sekunden liegt. Die Anzahl der Kundenstandorte ist ausschlaggebend für die Dauer der Generierungszeit, da die Anzahl der Kunden für dieses Testset bei 6001 und für „C110_1“ bei 1000 liegt. Bei der

Generierung der Kundenstandorte für 100 Replikationen werden insgesamt 500 doppelte Standorte erzeugt. Die minimale Fehleranzahl liegt dabei bei 1 und die Maximale bei 11 (vgl. Abbildung 21). Bei der maximalen Anzahl der Fehler handelt es sich um einen Ausreißer, da sowohl der Median als auch der Mittelwert bei 5 liegen. Der Mittelwert der Tourenlänge für diesen Simulationslauf ergibt 11331931,77 LE und der Median liegt bei 11333757,59 LE. Damit ist der Median größer als der Mittelwert und es kann eine linksschiefe Verteilung der Tourenlänge angenommen werden. Der Abstand zwischen Mittelwert und Median beträgt dabei 1825,82 LE. Dies ergibt, bezogen auf den Mittelwert, eine Abweichung von 0,02 Prozent und lässt den Schluss auf eine schwach ausgeprägte linksschiefe Verteilung zu. Damit ist die Verteilung der Tourenlänge beinahe symmetrisch.

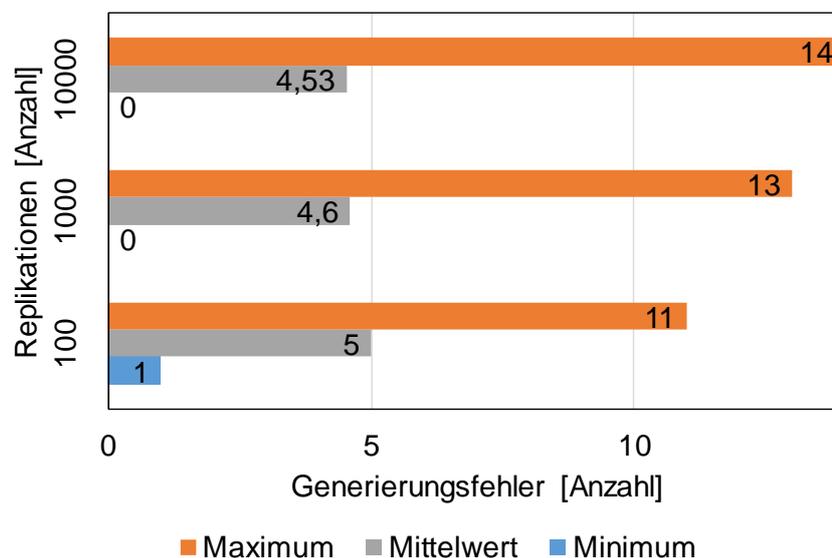


Abbildung 21: Gleichverteilung Generierungsfehler pro Replikation Testset Antwerp1

Im Simulationslauf mit 1000 Replikationen wird eine Gesamtgenerierungszeit von 77,406 Sekunden benötigt (vgl. Tabelle 38). Dabei liegt die minimal benötigte Generierungszeit einer Replikation bei 0,062 und die Maximale bei 0,125 Sekunden. Bei beiden Werten handelt es sich um Ausreißer, da der Mittelwert 0,076 Sekunden und der Median 0,078 Sekunden betragen. Innerhalb des Simulationslaufes werden 4602 doppelte Standorte generiert. Als minimale Fehleranzahl ergeben sich 0 Fehler und die maximale Anzahl an Fehlern ist 13 (vgl. Abbildung 21). Der Maximalwert ist ein Ausreißer, weil sich im Mittelwert 4,6 Fehler ergeben und der Median 4 Fehler beträgt. Als Mittelwert für die Tourenlänge ergibt sich ein Wert von 11333827,46 LE und für den Median liegt ein Wert von 11333268,87 LE vor. Damit ist der Mittelwert größer als der Median und

es kann eine rechtsschiefe Verteilung der Tourenlänge angenommen werden. Der Abstand zwischen Mittelwert und Median beträgt jedoch nur 558,59 LE und entspricht damit, bezogen auf den Mittelwert, einer Abweichung von 0,005 Prozent. Deswegen ist die Rechtsschiefe schwach ausgeprägt und die Tourenlänge ist beinahe symmetrisch verteilt.

Die Generierung der Standorte im dritten Simulationslauf mit 10000 Replikationen nimmt 788,047 Sekunden in Anspruch. Die minimal benötigte Zeit für die Generierung in einer Replikation beträgt 0,062 Sekunden und die Maximale 0,297 Sekunden. Sowohl der minimale als auch der maximale Wert für die Generierung stellen beide Ausreißer dar, da der Mittelwert bei 0,077 Sekunden und der Median bei 0,078 Sekunden liegt. Bei der Durchführung von 10000 Replikationen werden insgesamt 45300 doppelte Standorte erzeugt, wobei die minimale Anzahl dieses Fehlers bei 0 und die maximale Anzahl bei 14 liegt (vgl. Abbildung 21). Dabei handelt es sich bei dem Maximalwert der Fehler um einen Ausreißer, da der Mittelwert bei 4,53 Fehlern und der Median bei 4 Fehlern liegt. Für die Tourenlänge ergibt sich ein Mittelwert von 11334315,97 LE und ein Median von 11333328,41 LE. Dadurch kann, wie in dem vorherigen Simulationslauf, von einer Rechtsschiefe der Verteilung ausgegangen werden. Die Differenz aus Mittelwert und Median beträgt 987,56 und entspricht damit, bezogen auf den Mittelwert, einer Abweichung von 0,009 Prozent. Deswegen ist die Rechtsschiefe schwach ausgeprägt und es liegt beinahe eine symmetrische Verteilung vor. Werden alle drei Simulationsläufe zusammen betrachtet, ist auffällig, dass eine Verzehnfachung der Replikationen zu einer Verzehnfachung der benötigten Gesamtgenerierungszeit führt. Eine Verzehnfachung der doppelten Standorte, wie bei dem zuvor untersuchten Testset, findet hingegen nicht statt. Des Weiteren verändern sich sowohl der Mittelwert als auch der Median der Tourenlänge, der Generierungszeit sowie der Generierungsfehler nicht signifikant.

Die Tourenlängen für die drei durchgeführten Simulationsläufe sind in Abbildung 19 in Box-Plot-Diagramme eingetragen. In den Box-Plots ist der Mittelwert mit einem x gekennzeichnet, der Median hingegen ist durch eine horizontale Linie innerhalb der Box gekennzeichnet. Der Übergang von einer linksschiefen Verteilung im ersten Simulationslauf zu einer rechtsschiefen Verteilung in den letzten beiden Simulationsläufen, der zu Beginn analysiert wurde, kann auch in Abbildung 19 beobachtet werden. Denn der Mittelwert liegt für 100 Replikationen unterhalb des Medians. Beginnend bei 1000 Replikationen liegt dieser jedoch oberhalb des Medians.

Des Weiteren können auch die geringen Abstände zwischen dem Mittelwert und dem Median den Box-Plot-Diagrammen entnommen werden.

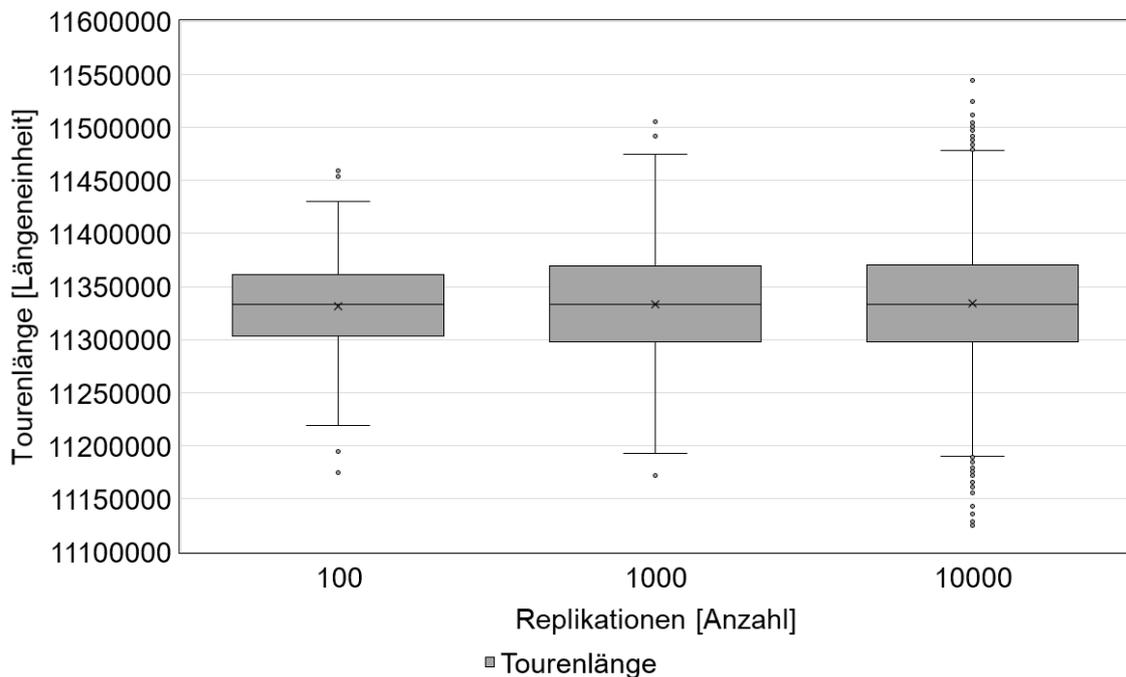


Abbildung 22: Gleichverteilung Box-Plots Testset Antwerp1

Eine weitere Beobachtung betrifft die Boxen des Diagramms, die 50 Prozent aller Werte enthält. Für den ersten Simulationslauf erscheint die Box kleiner zu sein als für die weiteren beiden Simulationsläufe. Mit Hilfe der Werte für die Interquartilsabstände kann diese Beobachtung bestätigt werden. Der Interquartilsabstand für den Simulationslauf mit 100 Replikationen beträgt 58383,77 LE. Für 1000 Replikationen steigt dieser Wert auf 71688,22 LE an, bis er bei 10000 durchgeführten Replikationen sein Maximum mit 72091,25 LE erreicht. Die untere Grenze der Box bei 100 Replikationen befindet sich bei 11303760,2 LE, für 1000 Replikationen bei 11297879,96 LE sowie für 10000 Replikationen bei 11298247,13 LE. Anhand dieser Werte und den Interquartilsabständen ist erkenntlich, dass sich die Werte der Tourenlänge zwischen dem ersten und dritten Quartal kaum unterscheiden. In Abbildung 19 können Unterschiede in der Länge der Whisker beobachtet werden, denn für 100 Replikationen sind die Whisker sichtbar kürzer als für die 1000 und 10000 Replikationen. In Werten ausgedrückt bedeutet dies, dass für den ersten Simulationslauf der untere Whisker bei 11219259,52 LE und der obere Whisker bei 11430073,26 LE enden. Bei 1000 Replikationen enden der untere Whisker bei 11192849,09 LE und der Obere bei 11474471,71 LE. Für 10000 durchgeführte Replikationen enden die Whisker bei ähnlichen Werten, der untere Whisker bei

11190260,05 LE und der Obere bei 11478407,12 LE. Des Weiteren unterscheiden sich die drei Simulationsläufe hinsichtlich ihrer Ausreißer, die auch in Abbildung 19 eingetragen sind. Für den ersten Simulationslauf ergeben sich 2 obere und 2 untere Ausreißer, für den zweiten Simulationslauf 2 obere und 1 unterer Ausreißer sowie für den dritten Simulationslauf 35 obere und 39 untere Ausreißer. Daraus kann geschlossen werden, dass bei einer Erhöhung von 100 auf 10000 Replikationen eine größere Streubreite der Tourenlänge und mehr Ausreißer erzeugt werden. Deren Einfluss kann jedoch als gering eingeschätzt werden, weil der für Ausreißer anfällige Mittelwert zwar von 11331931,77 LE auf 11334315,97 LE steigt. Dieser Anstieg entspricht jedoch nur 2384,2 LE und damit, bezogen auf den Mittelwert der 10000 Replikationen, einer Abweichung von 0,02 Prozent.

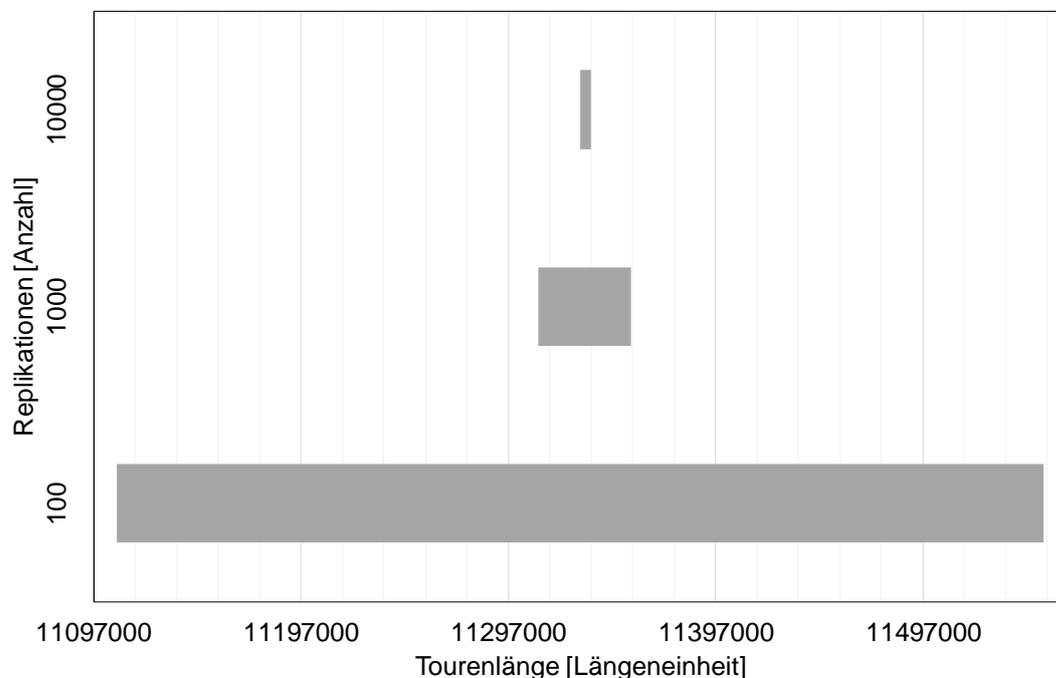


Abbildung 23: Gleichverteilung 95% Konfidenzintervalle Testset Antwerp1

Die approximierten 95% Konfidenzintervalle der drei Simulationsläufe sind in Abbildung 23 dargestellt. Bei einer steigenden Anzahl an Replikationen wird die Breite der Konfidenzintervalle reduziert. Für den ersten Simulationslauf beträgt die Breite des Konfidenzintervalls 446921,61 LE. Diese sinkt im zweiten Simulationslauf auf 44957,58 LE und erreicht im dritten Simulationslauf mit einer Breite von 4929,35 LE ihr Minimum. Die steigende Anzahl der Ausreißer und die Vergrößerung der Streubreite der Tourenlänge, die im vorherigen Absatz analysiert wurden, nehmen damit keinen sichtbaren Einfluss auf die Breite der Konfidenzintervalle. Je höher die Anzahl der

durchgeführten Replikationen ist, umso mehr nähert sich die Breite des Konfidenzintervalls dem Mittelwert des zugehörigen Simulationslaufs, welcher bei 10000 Replikationen 11334315,97 LE beträgt, an. Auf Basis des 95% Konfidenzintervalls der 10000 Replikationen kann ausgesagt werden, dass das approximierte Konfidenzintervall der Dimension (11331851,3;11336780,65) den wahren Wert der Tourenlänge mit einer Fehlerwahrscheinlichkeit von 5 Prozent überdeckt. Die Auswertung der drei Simulationsläufe lässt, auch für eine hohe Anzahl Replikationen, keine signifikanten Veränderungen in den Lagemaßen der Tourenlänge, der Generierungszeit und der Generierungsfehler erkennen. Aus diesem Grund können im Vergleich der Generierungsverfahren in Abschnitt 7.1 die Ergebnisse aller drei Simulationsläufe genutzt werden. Da mit dem approximierten 95% Konfidenzintervall aus dem dritten Simulationslauf die Größenordnung der Tourenlänge am präzisesten abgeschätzt werden kann, werden die Ergebnisse dieses Simulationslauf für den Vergleich genutzt.

6.2 Simulation mit einer Normalverteilung

Innerhalb dieses Abschnitts wird die Generierung der Kundenstandorte mit Hilfe einer Normalverteilung durchgeführt und im Anschluss ausgewertet. Dabei ist dieser Abschnitt in vier Unterabschnitte, 6.2.1 bis 6.2.4, unterteilt. In jedem Unterabschnitt wird die Generierung der Kundenstandorte in einem Testset besprochen. Die Formel, welche zur Generierung der x- und y-Koordinaten der Standorte eingesetzt wird, benötigt als Parameter den Erwartungswert μ und die Varianz σ^2 . Diese Parameter werden für jedes Testset so festgelegt, dass die Normalverteilung das gesamte Stadtgebiet, das von den begrenzenden Koordinaten aufgespannt wird, überdeckt. Damit keine Werte außerhalb des Stadtgebietes erzeugt werden, sind auch die begrenzenden Koordinaten als Parameter mit aufzunehmen. Um die Kennzahlen über eine steigende Anzahl an Replikationen beobachten zu können, erfolgen für jedes Testset insgesamt drei Simulationsläufe mit 100, 1000 und 10000 Replikationen (vgl. Abschnitt 5.2). Die Kennzahl Generierungsfehler setzt sich bei dieser Standortgenerierung aus den Fehlern „doppelte Standorte“, „außerhalb der Stadtgrenze“ und „Berechnungsfehler“ zusammen (vgl. Abschnitt 5.2). Damit die Übersichtlichkeit der Arbeit gewahrt bleibt, befinden sich die Werte aller Kennzahlen und die Werte, die für die grafische Auswertung benötigt werden, im Anhang in Tabelle 39 und Tabelle 40.

6.2.1 Normalverteilung im Testset E-n22-k4

Im Folgenden werden die Kundenstandorte mit einer Normalverteilung im Stadtgebiet des Testset E-n22-k4 erzeugt und ausgewertet. Für dieses Testset sind die begrenzenden Koordinaten des Stadtgebietes auf der x-Achse 165 LE und 265 LE auf der y-Achse (vgl. Tabelle 5). Deswegen werden alle Koordinaten der Standorte, die außerhalb der begrenzenden Koordinaten liegen, wobei dies für die x-Koordinate das Intervall (0; 165) und für die y-Koordinate das Intervall (0; 265) ist, verworfen und als Fehler aufgenommen. Damit die Koordinaten auf dem gesamten Stadtgebiet erzeugt werden, liegt für die x-Koordinate der Erwartungswert bei 82 LE und die Varianz bei 40 LE sowie für die y-Koordinate der Erwartungswert bei 132 LE und die Varianz bei 50 LE. Da das Testset 22 Kundenstandorte beinhaltet, werden in jeder Replikation 22 Standorte erzeugt. Tabelle 13 beinhaltet eine Auswahl an Ergebniswerten. Diese Auswahl stellt keine Bewertung der einzelnen Werte dar, sondern ermöglicht einen ersten Überblick über die berechneten Kennzahlen. Sämtliche Ergebniswerte sind im Anhang in Tabelle 39 zu finden.

Tabelle 13: Normalverteilung Testset E-n22-k4

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	4360,00	4320,03	789	0,093
1000	4368,16	4366,18	7619	0,531
10000	4362,06	4359,94	75016	5,453

Der erste Simulationslauf benötigt für 100 Replikationen eine Generierungszeit von 0,093 Sekunden. Die minimale Generierungszeit liegt bei 0 Sekunden, die Maximale bei 0,015 Sekunden. Da sowohl der Mittelwert als auch der Median 0 Sekunden betragen, kann ausgesagt werden, dass die Generierung von 22 normalverteilten Kundenstandorten wenig Zeit in Anspruch nimmt. Bei der Durchführung von 100 Replikationen werden insgesamt 789 Fehler erzeugt. Dabei wird 1 doppelter Standort erzeugt, bei dem es sich somit um einen Ausreißer handelt. 132 Standorte liegen außerhalb der Stadtgrenzen. Die minimale Anzahl dieses Fehlers liegt bei 0 Fehlern und die Maximale bei 6 Fehlern, wobei es sich bei der maximalen Anzahl um einen Extremwert handelt, weil der Mittelwert bei 1,32 Fehlern und der Median bei 1 Fehler liegt. Zudem erzeugt die

Berechnung Kundenstandorte 656 Berechnungsfehler. Die minimale Anzahl beträgt 0 Fehler sowie die maximale Anzahl 16 Fehler. Wiederum handelt es sich bei dem maximalen Fehler um einen Extremwert, da der Mittelwert bei 6,56 Fehlern und der Median bei 6 Fehlern liegt. Die minimalen und maximalen Fehler für 100 Replikationen können auch Abbildung 24 entnommen werden. Für die Tourenlänge ergibt sich ein Mittelwert von 4360 LE sowie ein Median von 4320,03 LE. Der Mittelwert ist damit größer als der Median. Dies deutet auf eine rechtsschiefe Verteilung der Tourenlänge hin. Dabei unterscheiden sich Mittelwert und Median um 39,97 LE, was einer Abweichung, bezogen auf den Mittelwert, von 0,9167 Prozent entspricht. Aus diesem Grund ist die Rechtsschiefe nur schwach ausgeprägt und kann auch als beinahe symmetrisch bezeichnet werden.

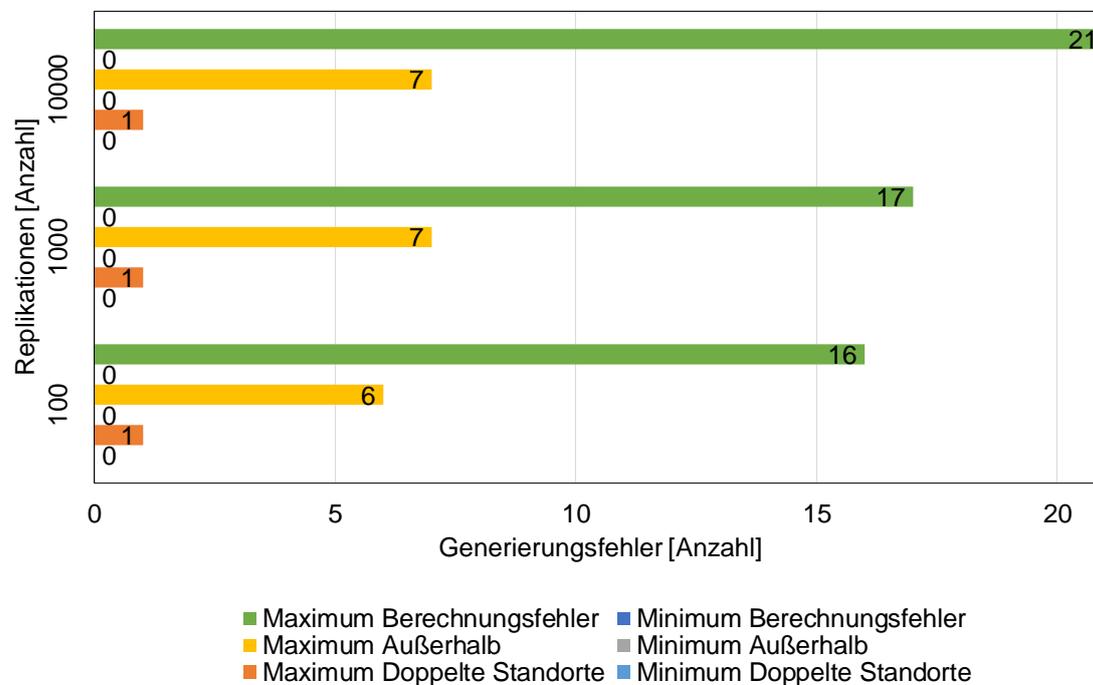


Abbildung 24: Normalverteilung Generierungsfehler pro Replikation Testset E-n22-k4

Für den Simulationslauf mit 1000 Replikationen wird eine Gesamtgenerierungszeit von 0,531 Sekunden benötigt. Auch hier liegt die minimale Generierungszeit einer Replikation bei 0 Sekunden, während maximal 0,016 Sekunden benötigt werden. Wie im ersten Simulationslauf, ergeben sich sowohl für den Mittelwert als auch den Median eine Generierungszeit von 0 Sekunden. Die Anzahl aller Fehler steigt insgesamt auf einen Wert von 7619. Bei der Generierung werden insgesamt 17 doppelte Standorte erzeugt, wobei das Minimum bei 0 und das Maximum bei 1 liegen. Anhand des Medians von 0 und des Mittelwerts von 0,02 ist erkennbar, dass der maximale Fehlerwert selten

vorkommt. Es werden insgesamt 1205 Kundenstandorte erzeugt, die außerhalb der Stadtgrenzen liegen. Dabei liegt die Mindestanzahl bei 0 und der Maximalwert bei 7. Aufgrund des Medians von 1 Fehler und dem Mittelwert von 1,2 Fehlern kann der Maximalwert als Extremwert bezeichnet werden. Bei der Generierung der Kundenstandorte entstehen 6397 Berechnungsfehler. Die minimale Anzahl an Fehlern liegt wiederum bei 0 Fehlern und die maximale Anzahl bei 17 Fehlern. Auch hier ist die maximale Anzahl an Fehlern als Extremwert zu bezeichnen, da sich als Mittelwert 6,4 Fehler und für den Median 6 Fehler ergeben. Die beschriebenen minimalen und maximalen Werte der Fehler können auch in Abbildung 24 nachvollzogen werden. Im Vergleich zum vorherigen Simulationslauf erhöhen sich sowohl der Mittelwert auf 4368,16 LE als auch der Median auf 4366,18 LE. Mit diesen Werten ist der Mittelwert noch immer größer als der Median. Die rechtsschiefe Verteilung der Tourenlänge ist somit weiterhin gegeben. Da der Abstand zwischen Mittelwert und Median 1,98 LE beträgt, was, bezogen auf den Mittelwert, einer Abweichung von 0,0453 Prozent entspricht, kann von einer schwachen Ausprägung der Rechtsschiefe ausgegangen beziehungsweise die Verteilung der Tourenlänge als beinahe symmetrisch bezeichnet werden.

Die 10000 Replikationen des dritten Simulationslaufs benötigten eine Gesamtgenerierungszeit von 5,453 Sekunden. Wie im vorherigen Simulationslauf liegt der minimale Wert der Generierungszeit bei 0 Sekunden, der Maximale bei 0,016 Sekunden und sowohl der Mittelwert als auch der Median bei 0 Sekunden. Insgesamt entstehen bei der Generierung der Kundenstandorte 75016 Fehler. Die Anzahl der doppelt erzeugten Standorte liegt bei 122, wobei die minimale Anzahl 0 und die maximale Anzahl 1 beträgt. Da der Median 0 Fehler und der Mittelwert 0,01 Fehler beträgt, kann ausgesagt werden, da bei 10000 Replikationen doppelte Standorte selten erzeugt werden. Außerhalb der Stadtgrenze liegen 11580 Standorte. Die Mindestanzahl dieses Fehlers liegt bei 0 Fehlern und die maximale Anzahl bei 7 Fehlern. Bei dem maximalen Wert dieses Fehlers handelt es sich um einen Extremwert, da der Median 1 und der Mittelwert 1,16 betragen. Des Weiteren werden bei der Generierung 63314 Berechnungsfehler erzeugt, dies entspricht im Minimum 0 Fehlern und im Maximum 21 Fehlern. Die 21 Fehler sind wiederum ein Extremwert, weil der Mittelwert 6,33 Fehler und der Median einen Wert von 6 Fehlern aufweist. Sowohl der Mittelwert der Tourenlänge sinkt, verglichen mit dem vorherigen Simulationslauf, auf 4362,06 LE als auch der Median auf 4359,94. Auch hier ist der Mittelwert größer als der Median. Aus diesem Grund kann auch weiterhin von

einer rechtsschiefen Verteilung der Tourenlänge ausgegangen werden. Der Abstand zwischen Mittelwert und Median beträgt 2,12 LE und entspricht so, auf den Mittelwert bezogen, einer Abweichung von 0,0486 Prozent. Bei einer gemeinsamen Betrachtung der drei Simulationsläufe ist auffällig, dass sich sowohl der Mittelwert als auch der Median der Tourenlänge, der Generierungszeit und der Generierungsfehler nicht signifikant verändern. Des Weiteren führt eine Verzehnfachung der Replikationen ungefähr zu einer Verzehnfachung von Gesamtfehleranzahl und Generierungszeit.

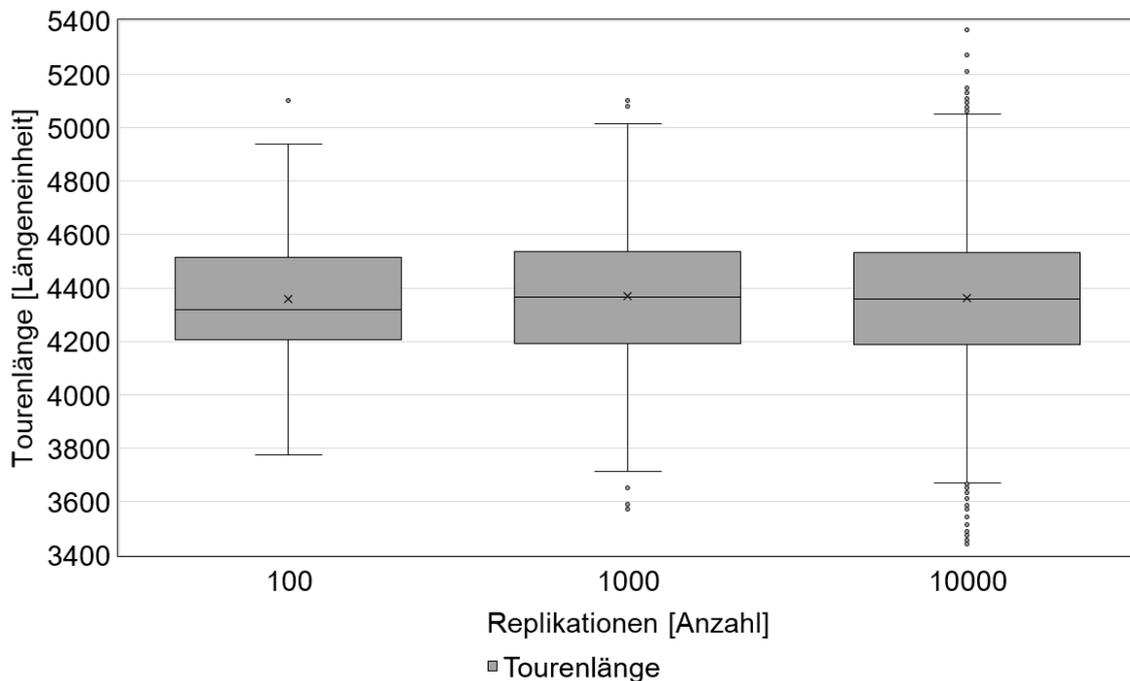


Abbildung 25: Normalverteilung Box-Plots Testset E-n22-k4

Für alle drei Simulationsläufe werden die Werte für die Tourenlänge jeweils in ein Box-Plot-Diagramm eingetragen (vgl. Abbildung 25). Der Mittelwert ist hier durch x gekennzeichnet, wohingegen der Median durch die horizontale Linie innerhalb der Box kenntlich gemacht ist. Die kleiner werdenden Abstände zwischen dem Mittelwert und dem Median sind so in der Abbildung zu erkennen. Auch die Rechtsschiefe der Verteilung der Tourenlänge ist zu sehen, denn der Mittelwert befindet sich in jedem Box-Plot oberhalb des Medians. Die Box des Simulationslaufs mit 100 Replikationen erscheint in der Abbildung kleiner als die Boxen für die letzten beiden Simulationsläufe zu sein. Die Werte für die Interquartilsabstände stützen diese Beobachtung. Bei 100 Replikationen beträgt der Interquartilsabstand 318,33 LE, für 1000 Replikationen steigt er auf 344,66 LE und für 10000 Replikationen beträgt er 346,98 LE. Die unteren Grenzen der Boxen liegen für alle drei Simulationsläufe nah beieinander. Die Grenze für den ersten

Simulationslauf liegt bei 4206,87 LE, für den zweiten Simulationslauf bei 4193,16 LE sowie für den Dritten bei 4187,06 LE. Somit ändern sich die Grenzen für 50 Prozent der Werte auch für eine höhere Anzahl an Replikationen nicht signifikant. Ein erkennbarer Unterschied ist die Länge der Whisker, die für den ersten Simulationslauf kürzer als für die anderen beiden Simulationsläufe ist. Für den ersten Simulationslauf liegt der Wert für den unteren Whisker bei 3776,4 LE und für den oberen Whisker bei 4937,79 LE. Im Unterschied dazu sind die Werte für die Whisker des zweiten und dritten Simulationslauf beinahe identisch. Diese betragen im zweiten Simulationslauf für den unteren Whisker 3713,49 LE und den Oberen 5015,65 LE sowie im dritten Simulationslauf für den Unteren 3668,77 LE und den Oberen 5052,53 LE. Zudem besteht ein weiterer in Abbildung 25 sichtbarer Unterschied in der Anzahl der Ausreißer. Bei 100 Replikationen entsteht ein oberer Ausreißer, bei 1000 Replikationen entstehen 4 untere und 2 obere Ausreißer. Im letzten Simulationslauf existieren 32 obere und 26 untere Ausreißer. Die Erhöhung von 100 auf 10000 Replikationen führt damit zu einer Erhöhung der Streubreite, der Tourenlänge und der Extremwerte. Der Einfluss dieser beiden Faktoren ist als gering einzuschätzen, da sich der, für Extremwerte anfällige, Mittelwert von 4360 LE bei 100 Replikationen auf 4362,06 LE erhöht und damit einem Unterschied von 2,06 LE entspricht. Bezogen auf den Mittelwert der 10000 Replikationen ist dies eine Abweichung von 0,0472 Prozent.

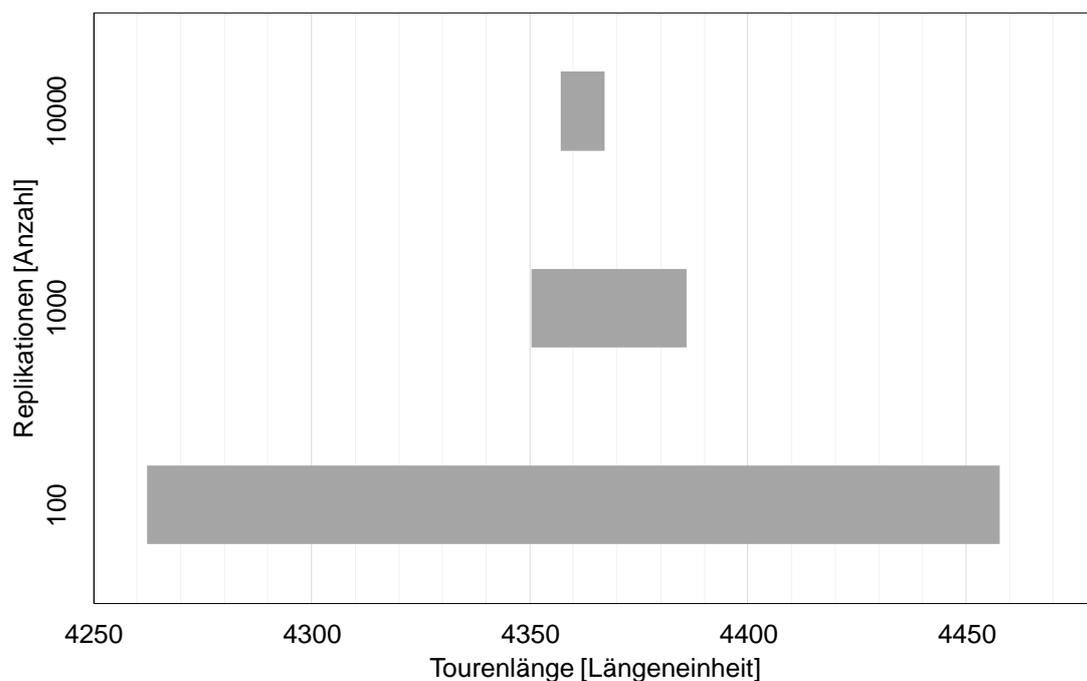


Abbildung 26: Normalverteilung 95% Konfidenzintervalle Testset E-n22-k4

Die approximierten 95% Konfidenzintervalle für die drei Simulationsläufe sind in Abbildung 26 dargestellt. Steigt die Anzahl der durchgeführten Replikationen, sinkt die Breite des Konfidenzintervalls. Im ersten Simulationslauf ergibt sich für die Breite des Konfidenzintervalls ein Wert von 195,29 LE. Die Breite des Konfidenzintervalls sinkt im zweiten Simulationslauf auf 35,55 LE und erreicht im dritten Simulationslauf mit 10,12 LE ihren minimalen Wert. Mit einer steigenden Anzahl der Replikationen nähert sich die Breite des Konfidenzintervall dem Mittelwert an. Auf Basis des 95% Konfidenzintervall der 10000 Replikationen kann ausgesagt werden, dass der wahre Wert der Tourenlänge mit einer Wahrscheinlichkeit von 95% vom approximierten Konfidenzintervall der Dimension (4357; 4367,12) überdeckt wird. In der Auswertung der drei Simulationsläufe lässt sich keine signifikante Veränderung der Mittelwerte sowie der Mediane der Tourenlänge, der Generierungszeit und der Generierungsfehler erkennen. In Abschnitt 7.1 können deswegen die Ergebnisse aller Simulationsläufe im Vergleich der Generierungsverfahren eingesetzt werden. Für die Größenordnung der Tourenlänge ist, aufgrund der minimalen Breite, die präziseste Vorhersage mit dem approximierten Konfidenzintervall des dritten Simulationslaufs möglich. Aus diesem Grund werden die Ergebnisse dieses Simulationslaufs im Vergleich genutzt.

6.2.2 Normalverteilung im Testset CMT14

Nachfolgend wird die Generierung der Kundenstandorte mit einer Normalverteilung über das Testset CMT14 durchgeführt und ausgewertet. Die Koordinaten, die das Stadtgebiet begrenzen, betragen auf der x-Achse 95 LE und auf der y-Achse 85 LE (vgl. Tabelle 5). Deswegen ergeben sich die x-Koordinaten der Standorte aus dem Intervall (0; 95) und die y-Koordinaten aus dem Intervall (0; 85). Um die Koordinaten auf dem ganzen Stadtgebiet erzeugen zu können, werden der Erwartungswert der x-Koordinate auf 50 LE und die Varianz auf 35 LE gesetzt. Für die y-Koordinate ergeben sich für den Erwartungswert 40 LE und die Varianz 25 LE. Das Testset beinhaltet 100 Kundenstandorte. Deswegen werden in jeder Replikation auch 100 Kundenstandorte erzeugt. In Tabelle 14 kann eine Auswahl der Ergebniswerte dieses Testsets betrachtet werden. Diese Auswahl stellt keine Bewertung der Ergebnisse dar. Alle Ergebnisse können vollständig im Anhang in Tabelle 39 gefunden werden.

Table 14: Normalverteilung Testset CMT14

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	8320,97	8328,61	7197	0,187
1000	8314,95	8330,16	72421	1,453
10000	8306,28	8308,85	725900	16,313

Die 100 Replikationen des ersten Simulationslaufs werden innerhalb einer Gesamtgenerierungszeit von 0,187 Sekunden erzeugt. Die minimale Generierungszeit für eine Replikation liegt bei 0 Sekunden und die Maximale bei 0,016 Sekunden. Für den Mittelwert ergeben sich 0,001 Sekunden sowie ein Wert von 0 Sekunden für den Median. Die Erzeugung von 100 normalverteilten Standorten nimmt somit sehr wenig Zeit in Anspruch. Insgesamt werden 7197 Generierungsfehler erzeugt. Auf die doppelten Standorte entfallen dabei im Ganzen 61 Fehler, wobei die minimale Anzahl bei 0 Fehlern und das Maximum bei 2 Fehlern liegt. Bei dem Maximalwert dieses Fehlers handelt es sich um einen Extremwert, da der Median bei 1 Fehler und der Mittelwert bei 0,61 Fehlern liegt. Außerhalb der Stadtgrenzen werden 3424 Standorte erzeugt. Dabei werden pro Replikation mindestens 19 Standorte und maximal 53 außerhalb der Stadtgrenzen erzeugt. Die minimalen und maximalen Werte stellen hier Extremwerte dar, weil der Median bei 34 Fehlern und der Mittelwert bei 34,24 Fehlern liegt. Bei der Generierung der Standorte entstehen 3712 Berechnungsfehler. Pro Replikation ergeben sich ein minimaler Wert von 23 Fehlern und ein maximaler Wert von 61 Fehlern. Auch hier handelt es sich bei dem minimal und dem maximal Wert um Extremwerte, da der Median bei 36 Fehlern und der Mittelwert bei 37,12 Fehlern liegen. In Abbildung 27 sind die genannten Werte zur Veranschaulichung abgebildet. Für die Tourenlänge ergibt sich bei 100 Replikationen ein Mittelwert von 8320,97 LE und ein Median von 8328,61 LE. Der Mittelwert ist damit kleiner als der Median. Daraus kann eine linksschiefe Verteilung der Tourenlänge abgeleitet werden. Der Abstand zwischen Mittelwert und Median beträgt 7,64 LE und entspricht damit, bezogen auf den Mittelwert, einer Abweichung von 0,0918 Prozent. Aufgrund dieser geringen Abweichung der beiden Lagemaße ist die Linksschiefe der Verteilung schwach ausgeprägt, die Verteilung der Tourenlänge ist beinahe symmetrisch.

Der zweite Simulationslauf mit 1000 Replikationen benötigt eine Gesamtgenerierungszeit von 1,453 Sekunden. Wie im vorherigen Simulationslauf, ist die minimale Generierungszeit einer Replikation 0 Sekunden, die Maximale 0,016 Sekunden, der Median 0 Sekunden sowie der Mittelwert 0,001 Sekunden. Damit gibt es keine Veränderung an der Erwartung, dass die normalverteilte Generierung von 100 Kundenstandorten pro Replikation wenig Zeit in Anspruch nimmt. Im zweiten Simulationslauf werden insgesamt 72421 Fehler verursacht, wovon 730 doppelte Standorte sind. In einer Replikation werden mindestens 0 und maximal 4 doppelte Standorte erzeugt. Da der Median 1 Fehler und der Mittelwert 0,73 Fehler betragen, handelt es sich bei dem Maximalwert von 4 Fehlern um einen Extremwert.

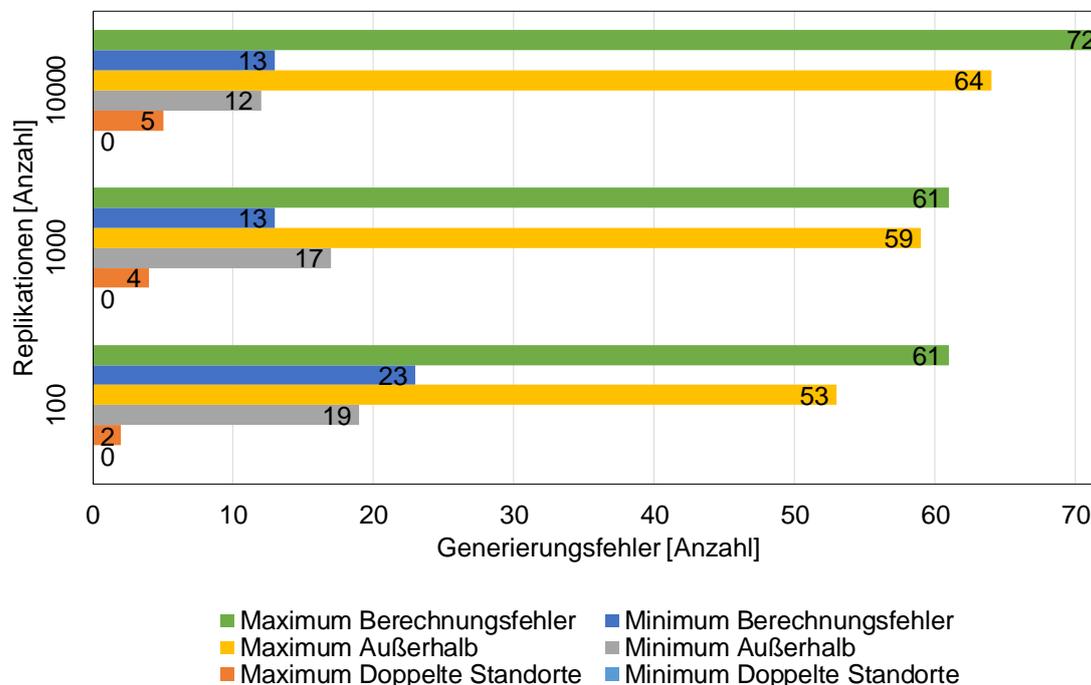


Abbildung 27: Normalverteilung Generierungsfehler pro Replikation Testset CMT14

Insgesamt werden 34982 Standorte erzeugt, die sich außerhalb der Stadtgrenzen befinden. Dabei liegen in einer Replikation mindestens 17 Standorte sowie maximal 59 Standorte außerhalb. Bei beiden Werten handelt es sich um Extremwerte, weil der Median bei 34,5 Fehlern und der Mittelwert bei 34,98 Fehlern liegt. Die Generierung der Standorte verursacht 36709 Berechnungsfehler, wobei mindestens 13 Fehler und maximal 61 Fehler entstehen. Auch hier sind beide Werte Extremwerte, da der Mittelwert bei 36,71 Fehlern und der Median bei 36 Fehlern liegt. Die angesprochenen Fehlerwerte sind in Abbildung 27 aufgezeigt. Der Mittelwert sinkt, im Vergleich zum vorherigen Simulationslauf, auf 8314,95 LE, wohingegen der Median im Vergleich auf 8330,16 LE steigt. Die

linksschiefe Verteilung der Tourenlänge bleibt damit bestehen. Der Abstand zwischen Mittelwert und Median verdoppelt sich auf 15,21 LE und ergibt, bezogen auf den Mittelwert, eine Abweichung von 0,1829 Prozent. Aufgrund dieses Wertes bleibt die schwache Linksschiefe bestehen und es liegt beinahe eine symmetrische Verteilung der Tourenlänge vor.

Der dritte Simulationslauf benötigt für die 10000 Replikation eine Gesamtgenerierungszeit von 16,313 Sekunden. Wie in den beiden vorherigen Simulationsläufen, betragen die Mindestgenerierungszeit 0, die maximale Generierungszeit 0,016, der Median 0 sowie der Mittelwert 0,001 Sekunden. Die Erwartung einer kurzen Generierungszeit für 100 Standorte bleibt damit bestehen. Insgesamt werden über alle Replikationen hinweg bei einer Gesamtfehleranzahl von 725900 Fehlern 7558 doppelte Standorte erzeugt. Wie im vorherigen Simulationslauf, ergibt sich ein Minimalwert von 0 Fehlern für eine Replikation. Der Maximalwert liegt bei 5 Fehlern, wobei es sich jedoch um einen Extremwert handelt, weil der Median 1 Fehler und der Mittelwert 0,73 Fehler betragen. Außerhalb der Stadtgrenzen werden 348903 Standorte erzeugt. Dabei werden in einer Replikation mindestens 12 dieser Fehler und Maximal 64 Fehler erzeugt. Bei beiden Werten handelt es sich um Extremwerte, da der Median bei 34 Fehlern und der Mittelwert bei 34,89 Fehlern liegen. Die Generierung der Standorte erzeugt bei 10000 Replikationen insgesamt 369439 Berechnungsfehler. In einer Replikation werden mindestens 13 Fehler und maximal 72 Fehler erzeugt. Auch diese beiden Werte stellen Extremwerte dar, weil der Median 37 Fehler und der Mittelwert 36,94 Fehler betragen. Für die Tourenlänge sinken, im Vergleich zum vorherigen Simulationslauf, sowohl der Mittelwert auf 8306,28 LE als auch der Median auf 8308,85. Weiterhin ist die Verteilung der Tourenlänge linksschief. Der Abstand zwischen den beiden Lagemaßen sinkt auf 2,57 LE und entspricht damit, bezogen auf den Mittelwert, einer Abweichung von 0,0309 Prozent. Aufgrund dieser geringen Abweichung ist die Linksschiefe schwach ausgeprägt und die Verteilung der Tourenlänge beinahe symmetrisch. Über alle Simulationsläufe betrachtet, führt eine Verzehnfachung der Anzahl an Replikationen zu einer Verzehnfachung der Gesamtgenerierungszeit sowie der Gesamtfehleranzahl. Außerdem verändern sich sowohl der Mittelwert als auch der Median der Tourenlänge, der Generierungszeit und der Generierungsfehler nicht signifikant.

Die Tourenlängen der drei Simulationsläufe sind in Abbildung 28 jeweils in ein Box-Plot-Diagramm eingetragen. In den Diagrammen ist der Mittelwert durch x gekennzeichnet, der Median durch die horizontale Linie innerhalb der Box. Der geringe Abstand zwischen Mittelwert und Median kann auch im Box-Plot für 100 Replikationen beobachtet werden. Des Weiteren ist der vergrößerte Abstand zwischen beiden Lagemaßen im Box-Plot für 1000 Replikationen sowie das Minimum dieses Abstands bei 10000 Replikationen zu erkennen. Die linksschiefe Verteilung der Tourenplanung kann in den Box-Plots daran erkannt werden, dass das x des Mittelwerts sich für alle drei Simulationsläufe unterhalb der horizontalen Linie des Medians befindet. Für den zweiten Simulationslauf erscheint die Box, in der sich 50 Prozent der Werte für die Tourenlänge befinden, kleiner zu sein, als die Boxen des ersten und dritten Simulationslaufes. Diese Vermutung kann mit Hilfe der Werte des Interquartilsabstandes bestätigt werden. Für 100 Replikationen beträgt der Interquartilsabstand 357,11 LE, für 1000 Replikationen sinkt der Interquartilsabstand auf 339,81 LE und für 10000 Replikationen steigt der Interquartilsabstand auf 358,86 LE, womit die Abstände des ersten und dritten Simulationslaufes beinahe identisch sind. Der Interquartilsabstand für den zweiten Simulationslauf ist dabei nur etwa 19 LE kleiner als für die anderen beiden Simulationsläufe.

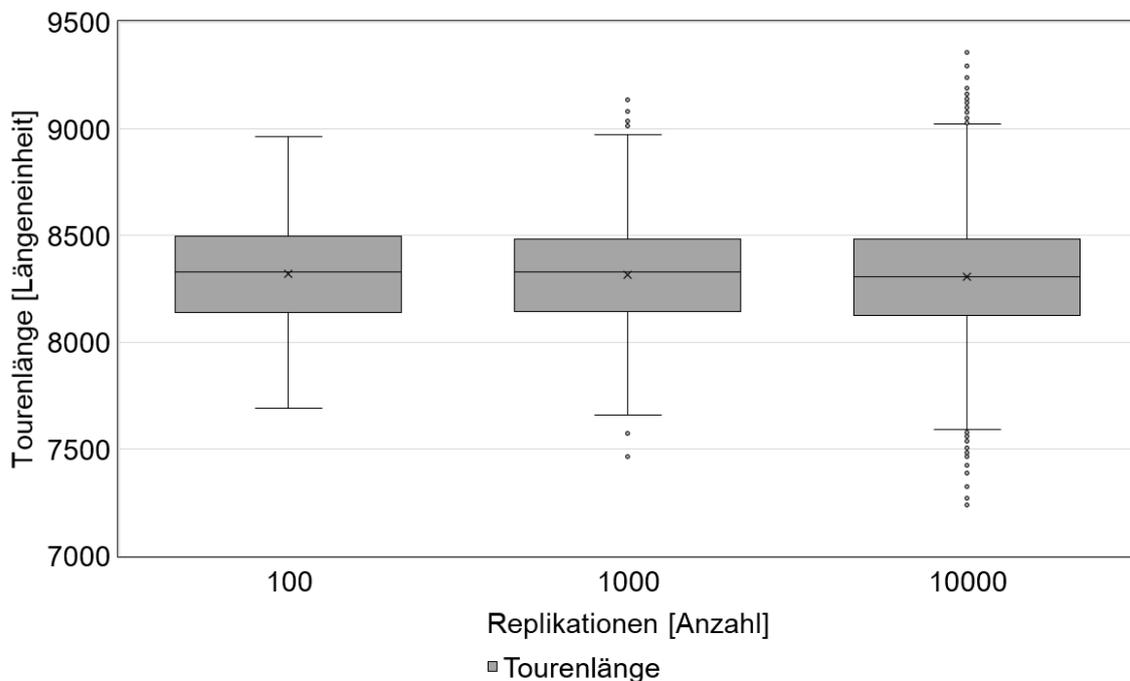


Abbildung 28: Normalverteilung Box-Plots Testset CMT14

Dies spiegelt sich auch in den unteren Grenzen der Boxen wider. Für den ersten Simulationslauf liegt die untere Grenze bei 8139,74 LE, die untere Grenze des zweiten Simulationslaufs bei 8143,14 LE sowie für den dritten Simulationslauf bei 8126,49 LE. Anhand dieser geringen Abweichungen ist erkennbar, dass sich die Grenzen, in denen 50% der Werte liegen, auch für eine hohe Anzahl an Replikationen nicht signifikant verändern. Die Differenz in der Länge der Whisker ist ein Unterschied der drei Simulationsläufe, der in Abbildung 28 erkennbar ist. Für 100 Replikationen wird der untere Whisker durch 7689,37 LE begrenzt und der Obere durch 8962,98 LE. Die Grenzen der Whisker des zweiten Simulationslaufs mit 1000 Replikationen liegen hinsichtlich der Unteren bei 7660,06 LE und hinsichtlich der Oberen bei 8973,88 LE. Innerhalb des dritten Simulationslaufs ergeben sich für die untere Grenze 7590,23 LE und für die obere Grenze 9023,6 LE. Einen weiteren Unterschied zwischen den Simulationsläufen stellt die unterschiedliche Anzahl der Ausreißer dar. Im ersten Simulationslauf existieren keine Ausreißer, im zweiten Simulationslauf existieren 5 untere sowie 5 obere Ausreißer und für den letzten Simulationslauf ergeben sich 47 untere sowie 39 obere Ausreißer. Eine Erhöhung der Replikationen von 100 auf 10000 führt zu einer höheren Streubreite der Tourenlänge sowie zu einer größeren Anzahl an Ausreißern. Für beide Faktoren ist der Einfluss als gering einzuschätzen, da sich der für Ausreißer anfällige Mittelwert von 8320,97 LE bei 100 Replikationen auf 8306,28 LE bei 10000 Replikationen verringert. Dies entspricht einer Senkung um 14,69 LE und damit, bezogen auf den Mittelwert der 10000 Replikationen, einer Abweichung von 0,1768 Prozent.

Für die drei Simulationsläufe sind in Abbildung 29 die approximierten 95% Konfidenzintervalle aufgetragen. Für eine steigende Anzahl an Replikationen sinkt die Breite der Konfidenzintervalle. Die Breite beträgt im ersten Simulationslauf 343,21 LE. Im zweiten Simulationslauf mit 1000 Replikationen sinkt die Breite des Konfidenzintervalls auf 46,39 LE und erreicht im dritten Simulationslauf mit 11,09 LE ihr Minimum. Bei einer höheren Anzahl an Replikationen nähert sich die Breite des Konfidenzintervall dem Mittelwert des zugehörigen Simulationslaufes an. Anhand des approximierten 95% Konfidenzintervall der 10000 Replikationen kann ausgesagt werden, dass der wahre Wert der Tourenlänge mit einer Wahrscheinlichkeit von 95% vom approximierten Konfidenzintervall der Dimension (8300,74; 8311,83) überdeckt wird. Aus der Auswertung der drei Simulationsläufe ergibt sich keine signifikante Veränderung sowohl für den Mittelwert als auch den Median der Tourenlänge, der Generierungsfehler und der Generierungszeit. Deswegen können die Ergebnisse aller Simulationsläufe für

den Vergleich der Generierungsverfahren in Abschnitt 7.1 genutzt werden. Weil die präziseste Aussage zur Größenordnung der Tourenlänge mit dem approximierten 95% Konfidenzintervall des dritten Simulationslauf getroffen werden kann, werden die Ergebnisse dieses Simulationslauf in den Vergleich eingebracht.

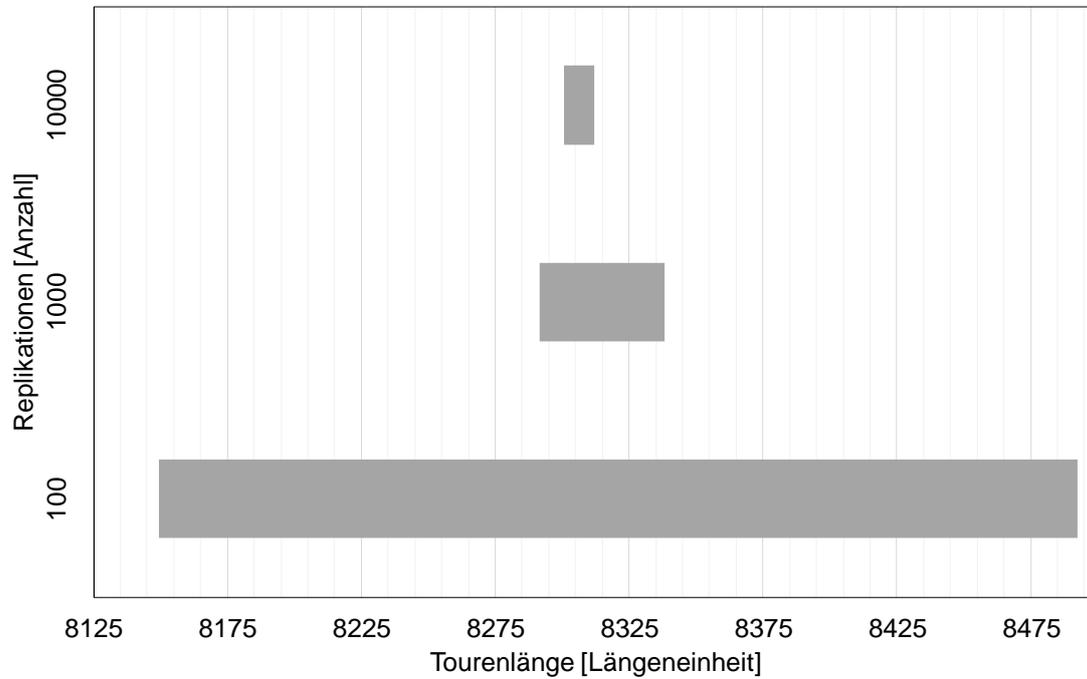


Abbildung 29: Normalverteilung 95% Konfidenzintervalle Testset CMT14

6.2.3 Normalverteilung im Testset C110_1

Im nachfolgenden werden die Kundestandorte im Stadtgebiet des Testset C110_1 mit einer Normalverteilung erzeugt und ausgewertet. Die Koordinaten, die das Stadtgebiet begrenzen, sind jeweils für die x- und die y-Achse 500 LE (vgl. Tabelle 5). Aus diesem Grund werden sowohl die x- als auch die y-Koordinaten der Standorte im Intervall (0; 500) erzeugt. Um eine Generierung der Kundenstandorte auf dem gesamten Stadtgebiet zu ermöglichen, liegen der Erwartungswert bei 250 LE und die Varianz bei 85 LE. Diese Werte gelten sowohl für die x- als auch die y-Koordinate. Da das Testset 1000 Kunden enthält, werden innerhalb jeder Replikation 1000 Kundenstandorte erzeugt. Für die Testsets C110_1 und RC110_1 würde die Normalverteilung identisch parametrisiert. Bei einer identischen Parametrisierung der Normalverteilung ergeben sich immer dieselben Ergebnisse. Deswegen wird die Generierung normalverteilter Standorte nur für das Testset C110_1 durchgeführt. Die Ergebnisse gelten äquivalent auch für die anderen beiden genannten Testsets. Tabelle 15 beinhaltet eine Auswahl von Ergebniswerten.

Diese Auswahl ist nicht als Bewertung der Ergebnisse zu sehen, sondern sie ermöglicht einen Überblick über die Ergebnisse. Alle Ergebniswerte können im Anhang in Tabelle 39 eingesehen werden.

Tabelle 15: Normalverteilung Testset C110_1

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	448497,12	448295,57	29040	1,078
1000	448420,17	448424,79	288425	10,296
10000	448649,69	448650,94	2885845	118,984

Der erste Simulationslauf mit 100 Replikationen erfordert eine Gesamtgenerierungszeit von 1,078 Sekunden. Minimal wird in einer Replikation eine Generierungszeit von 0 Sekunden benötigt, maximal 0,016 Sekunden. Der Mittelwert der Generierungszeit liegt bei 0,01 Sekunden und der Median bei 0,015 Sekunden. Damit ergibt sich, abweichend von den vorherigen Testsets E-n22-k4 und CMT14, eine höhere Generierungszeit pro Replikation. Da sich die Anzahl der Kundenstandorte beziehungsweise der Kundendichte pro QLE in Bezug auf die Generierung mit einer Gleich- oder Normalverteilung nicht verändern, ist wie, im Abschnitt 6.1 festgestellt, die höhere Generierungszeit pro Replikation auf die erhöhte Anzahl der Kundenstandorte in diesem Testset zurückzuführen. Insgesamt entstehen in diesem Simulationslauf 29040 Fehler, wovon 550 auf die Erzeugung doppelter Standorte zurückzuführen sind. Dabei liegt die Mindestanzahl einer Replikation bei 0 Fehlern und Maximal bei 16 Fehlern. Aufgrund des Mittelwerts von 5,5 Fehlern und des Medians von 5 Fehlern, ist der Wert der maximalen Fehleranzahl als Extremwert zu bezeichnen. Außerhalb der Stadtgrenzen werden 702 Standorte erzeugt. Der Minimalwert dieses Fehlers liegt bei 2, der Maximalwert bei 13. Da der Mittelwert bei 7,02 Fehlern und der Median bei 7 Fehlern liegen, ist auch dieser Maximalwert als Extremwert zu bezeichnen. Bei der Generierung der Standorte entstehen insgesamt 27788 Berechnungsfehler, mit einem Minimalwert von 226 Fehlern sowie einem Maximalwert von 324 Fehlern innerhalb einer Replikation. Bei einem Median von 280 Fehlern und einem Mittelwert von 277,88 können sowohl der Minimal- als auch der Maximalwert der Berechnungsfehler als Extremwert bezeichnet werden. Die angesprochenen minimalen und maximalen Werte aller Fehler sind in

Abbildung 30 aufgezeigt. Für die Tourenlänge ergeben sich ein Mittelwert von 448497,12 LE und ein Median von 448295,57 LE, womit der Mittelwert größer als der Median ist. Aufgrund dessen ist von einer rechtsschiefen Verteilung der Tourenlänge auszugehen. Der Abstand zwischen Mittelwert und Median beträgt 201,55 LE und entspricht damit einer Abweichung von 0,0449 Prozent, bezogen auf den Mittelwert. Aus diesem Grund ist die Rechtsschiefe der Tourenlänge schwach ausgeprägt und beinahe symmetrisch verteilt.

Für den zweiten Simulationslauf wird eine Gesamtgenerierungszeit von 10,296 Sekunden benötigt. Dabei ist für eine Replikation eine minimale Generierungszeit von 0 Sekunden und eine Maximale von 0,032 Sekunden erforderlich. Wie im vorherigen Simulationslauf, betragen der Median 0,015 und der Mittelwert 0,01 Sekunden. Bei diesem Simulationslauf entstehen insgesamt 288425 Fehler. 5625 dieser Fehler sind auf die Generierung doppelter Standorte zurückzuführen. Die Mindestanzahl des Fehlers liegt bei 0 und die maximale Anzahl bei 16. Der Maximalwert stellt einen Extremwert dar, weil der Median bei 5 und der Mittelwert bei 5,62 Fehlern liegen. Insgesamt 6834 Standorte werden außerhalb der Stadtgrenzen erzeugt, wobei in jeder Replikation mindestens 1 Fehler dieser Art passiert. Der maximale Fehlerwert liegt bei 16. Auch bei diesem Maximalwert handelt es sich um einen Ausreißer, da der Median 7 Fehler und der Mittelwert 6,83 Fehler betragen.

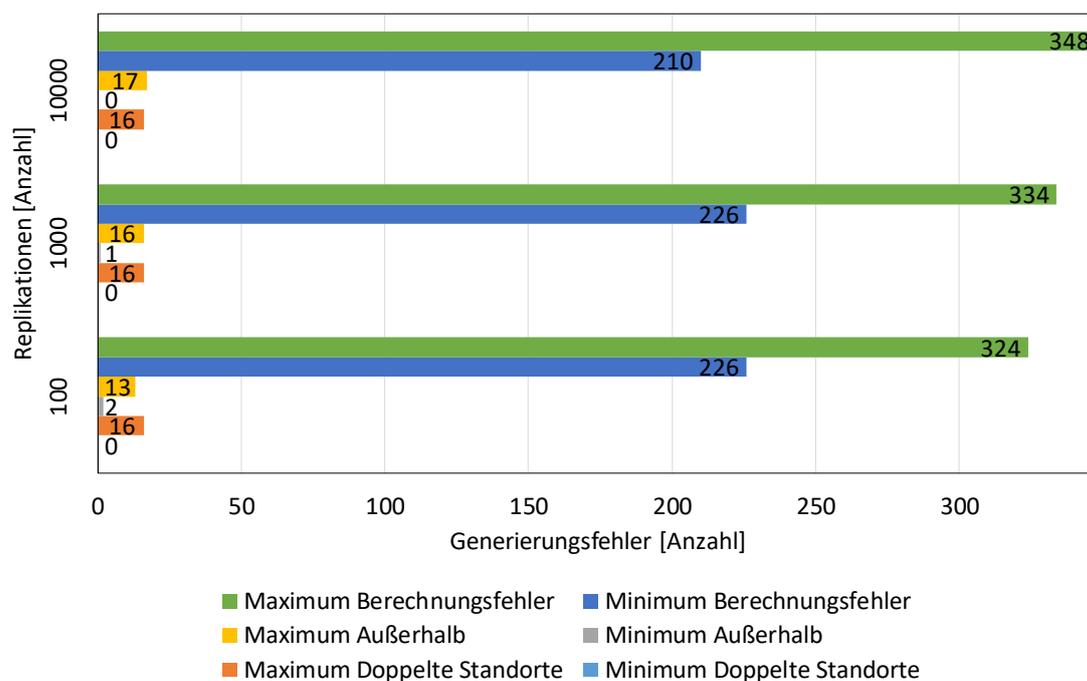


Abbildung 30: Normalverteilung Generierungsfehler pro Replikation Testset C110_1

Bei der Generierung der normalverteilten Standorte passieren 275966 Berechnungsfehler. Die Mindestanzahl beträgt 226, die Minimalanzahl 334. Sowohl bei der Mindest- als auch der Maximalanzahl handelt es sich um Extremwerte, weil der Mittelwert 275,97 Fehler und der Median 276 Fehler betragen. Minimal- und Maximalwerte der Fehler sind auch in Abbildung 30 zu betrachten. Die Tourenlänge besitzt einen Mittelwert von 448420,17 LE, der Median liegt bei 448424,79 LE. Somit ist der Median größer als der Mittelwert. Die rechtsschiefe Verteilung der Tourenlänge aus dem vorherigen Simulationslauf geht in eine linksschiefe Verteilung über. Da der Abstand zwischen Mittelwert und Median 4,62 LE beträgt und damit, bezogen auf den Mittelwert, einer Abweichung von 0,001 Prozent entspricht, ist die Linksschiefe der Verteilung schwach ausgeprägt und die Tourenlänge ist beinahe symmetrisch verteilt.

Der dritte Simulationslauf benötigt für 10000 Replikationen eine Gesamtgenerierungszeit von 118,984 Sekunden. Die minimal erforderliche Generierungszeit liegt bei 0, die maximale bei 0,032 Sekunden. Da der Median bei 0,015 und der Mittelwert bei 0,012 Sekunden liegen, stellt der maximale Wert der Generierungszeit einen Extremwert dar. Über die 10000 Replikationen hinweg werden bei insgesamt 2885845 Fehlern 55943 doppelte Standorte erzeugt. Der minimale Wert dieses Fehlers liegt bei 0, der Maximale bei 16 Fehlern. Bei dem Maximalwert handelt es sich um einen Extremwert, da der Median bei 5 und der Mittelwert bei 5,59 Fehlern liegen. Außerhalb der Stadtgrenzen werden 67430 Standorte generiert. In einer Replikation werden dabei mindestens 0 und maximal 17 Standorte außerhalb des Stadtgebietes erzeugt. Sowohl bei der minimalen als auch der maximalen Anzahl dieses Fehlers handelt es sich um Extremwerte, da der Median 7 Fehler und der Mittelwert 6,74 Fehler beträgt. Insgesamt werden 2762472 Berechnungsfehler erzeugt. Dies bedeutet eine Mindestanzahl von 210 und eine Maximalanzahl von 348 Fehlern innerhalb einer Replikation. Beide Werte sind ebenfalls Extremwerte, da der Median bei 276 Fehlern und der Mittelwert bei 276,25 Fehlern liegen. Für die Tourenlänge ergibt sich ein Mittelwert von 448649,69 LE sowie ein Median von 448650,49 LE. Wie im vorherigen Simulationslauf ist der Median damit größer als der Mittelwert und die linksschiefe Verteilung der Tourenlänge bleibt bestehen. Der Abstand zwischen Mittelwert und Median beträgt 1,25 LE. Bezogen auf den Mittelwert ergibt sich damit eine Abweichung von 0,0002 Prozent. Die Linksschiefe ist somit schwach ausgeprägt bei einer annähernd symmetrischen Verteilung. Die Betrachtung aller Simulationsläufe zeigt auf, dass sich sowohl der Median als auch der Mittelwert der Tourenlänge, der Generierungszeit und der Generierungsfehler zwischen

den drei Simulationsläufen nicht signifikant voneinander unterscheiden. Außerdem ist erkennbar, dass eine Verzehnfachung der Replikationen sowohl zu einer Verzehnfachung der Gesamtgenerierungszeit als auch der Gesamtfehleranzahl führt.

Die Werte der Tourenlänge sind für die drei Simulationsläufe jeweils in ein Box-Plot-Diagramm eingetragen worden. Dieses ist in Abbildung 31 zu sehen. Der Mittelwert ist in dem Box-Plot durch \bar{x} gekennzeichnet und die Markierung des Medians erfolgt mit Hilfe einer horizontalen Linie innerhalb der Box. In der Abbildung kann auch der geringe Abstand des Mittelwerts vom Median beobachtet werden. Zudem ist auch die rechtsbeziehungsweise linksschiefe Verteilung der Tourenlänge daran erkennbar, dass der Mittelwert entweder oberhalb, im Falle der Rechtsschiefe, oder, im Falle der Linksschiefe, unterhalb des Medians eingezeichnet ist. Die Box des ersten Simulationslaufs mit 100 Replikationen, in der sich 50 Prozent der Werte der Tourenlänge befinden, sichtbar kleiner als die Boxen der beiden anderen Simulationsläufe. Mit Hilfe der Werte für den Interquartilsabstand kann dies bestätigt werden. Für den ersten Simulationslauf beträgt der Interquartilsabstand 3947,9 LE. Im zweiten Simulationslauf steigt dieser Abstand auf 4223,8 LE an und erreicht im dritten Simulationslauf mit 4264,48 LE sein Maximum. Damit sind die Interquartilsabstände der letzten beiden Simulationsläufe beinahe gleich groß. Der Abstand zwischen dem maximalen Interquartilsabstand und dem Interquartilsabstand aus dem ersten Simulationslauf beträgt 316,58 LE. Dies entspricht, bezogen auf den maximalen Interquartilsabstand, einer Abweichung von 7,4236 Prozent, sodass die Abweichungen der Interquartilsabstände zwischen dem ersten und zweiten Simulationslauf signifikant und zwischen dem zweiten und dritten Simulationslauf als gering einzustufen sind. Die untere Grenze der Box beträgt im Falle des ersten Simulationslaufs 446413,28 LE, im Falle des zweiten Simulationslaufs 446247,89 LE sowie 446502,65 LE im Falle des dritten Simulationslaufs. Anhand der geringen Abweichungen der Grenzen und der Interquartilsabstände des zweiten und dritten Simulationslaufs kann ausgesagt werden, dass sich die Grenzen dieser Simulationsläufe, in denen 50 Prozent der Werte liegen, auch für eine hohe Anzahl an Replikationen nicht signifikant verändern. Ein weiterer Unterschied, der in Abbildung 31 erkennbar ist, sind die unterschiedlichen Längen der Whisker. Die Grenzen der Whisker des ersten Simulationslaufs liegen für die Untere bei 440904,16 LE und die Obere bei 454867,97 LE. Für den zweiten Simulationslauf bei 440557,86 LE und 456737,9 LE sowie für den dritten Simulationslauf bei 440116,6 und 457126,57 LE.

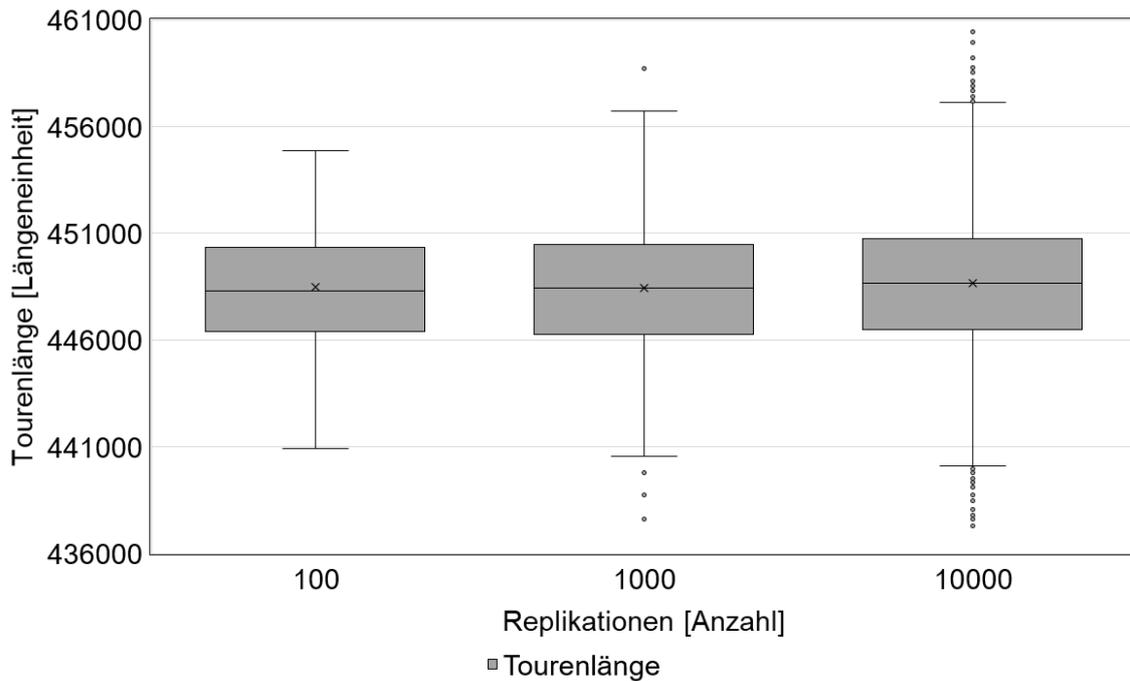


Abbildung 31: Normalverteilung Box-Plots Testset C110_1

Außerdem unterscheiden sich die drei Simulationsläufe hinsichtlich ihrer Ausreißer. Im ersten Simulationslauf werden keine Ausreißer erzeugt, im zweiten Simulationslauf existieren 3 untere und 1 oberer Ausreißer sowie im letzten Simulationslauf 31 untere und 39 obere Ausreißer. Daraus lässt sich schließen, dass sich eine Erhöhung der Replikationen von 100 auf 10000 in einer höheren Streubreite der Tourenlänge sowie einer höheren Anzahl an Ausreißern bemerkbar macht. Die Auswirkungen dieser beiden Faktoren und der signifikant unterschiedlichen Interquartilsabstände sind jedoch als gering einzuschätzen, da sich der für Ausreißer anfällige Mittelwert von 448497,12 LE bei 100 Replikationen auf 448649,69 LE bei 10000 Replikationen erhöht, was einem Abstand von 152,57 LE, beziehungsweise bezogen auf den Mittelwert der 10000 Replikationen einer Abweichung von 0,034 Prozent, entspricht.

Die approximierten 95% Konfidenzintervalle der Tourenlänge sind in Abbildung 32 für die drei Simulationsläufe aufgetragen. Der Abbildung kann entnommen werden, dass das Konfidenzintervall für eine höhere Anzahl an Replikationen kleiner wird. Für den ersten Simulationslauf beträgt die Breite des Konfidenzintervall 17704,05 LE. Im zweiten Simulationslauf sinkt diese Breite auf 1800,17 LE und erreicht im dritten Simulationslauf mit 215,37 LE ihr Minimum. Die Breite des Konfidenzintervalls nähert sich für eine steigende Anzahl an Replikationen dem jeweiligen Mittelwert der Tourenlänge an. Mit Hilfe des approximierten 95% Konfidenzintervalls der Tourenlänge des dritten

Simulationslaufs mit 10000 Replikationen kann ausgesagt werden, dass der wahre Wert mit einer Fehlerwahrscheinlichkeit von 5 Prozent durch das Intervall mit der Dimension (448542,01; 448757,38) überdeckt wird.

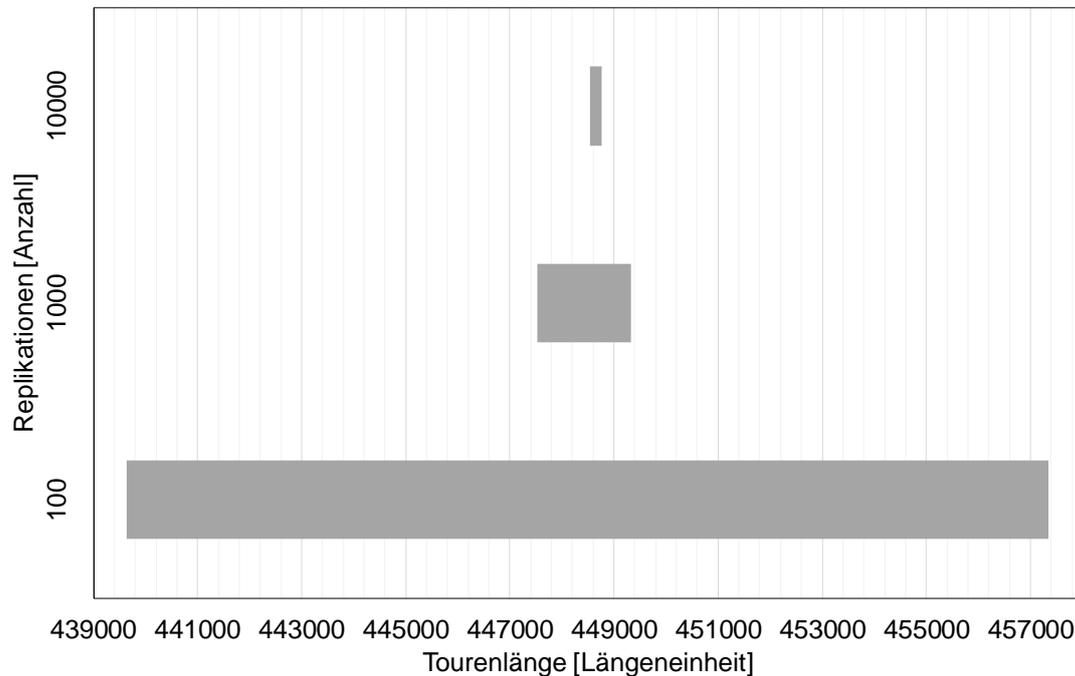


Abbildung 32: Normalverteilung 95% Konfidenzintervalle Testset C110_1

Die Auswertung der drei Simulationsläufe zeigt auch für eine hohe Anzahl Replikationen keine signifikante Veränderung sowohl der Mittelwerte als auch der Mediane der Tourenlänge, der Generierungsfehler und der Generierungszeit auf. Deswegen können die Ergebnisse jedes Simulationslaufs für den Vergleich der Generierungsverfahren in Abschnitt 7.1 genutzt werden. Jedoch ist mit dem approximierten 95% Konfidenzintervall des dritten Simulationslaufs, aufgrund der minimalen Breite des Konfidenzintervalls, die präziseste Abschätzung der Größenordnung der Tourenlänge möglich. Aus diesem Grund werden die Ergebnisse dieses Simulationslaufs in den Vergleich eingebracht.

6.2.4 Normalverteilung im Testset Antwerp1

Im letzten Testset dieses Abschnitts wird die Generierung der Kundenstandorte mit einer Normalverteilung im Testset Antwerp1 durchgeführt und ausgewertet. Die begrenzenden Koordinaten dieses Testsets betragen sowohl auf der x- als auch auf der y-Achse 2000 LE (vgl. Tabelle 5). Aus diesem Grund liegen die x- und y-Koordinaten der Kundenstandorte im Intervall (0; 2000). Damit die Kundenstandorte mit der

Normalverteilung im gesamten Stadtgebiet erzeugt werden, ergeben sich sowohl für die x- als auch für die y-Koordinate der Kundenstandorte ein Erwartungswert von 1000 LE und eine Varianz von 300 LE. In dem Testset sind 6001 Kunden vorhanden, weswegen in jeder Replikation 6001 Kundenstandorte erzeugt werden. Eine Auswahl an Ergebniswerten wird in Tabelle 16 aufgezeigt, wobei die Auswahl keine Bewertung der Ergebnisse darstellt. Im Anhang in Tabelle 40 sind alle Ergebniswerte zu finden.

Tabelle 16: Normalverteilung Testset Antwerp1

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	10708008,54	10710448,48	167694	7,063
1000	10707517,95	10707427,96	1674021	63,406
10000	10708670,00	10708441,89	16728024	627,657

Der Simulationslauf mit 100 Replikationen nimmt eine Gesamtgenerierungszeit von 7,063 Sekunden in Anspruch. Dabei werden für eine Replikation mindestens eine Generierungszeit von 0,046 und eine Maximale von 0,078 Sekunden benötigt. Für den Mittelwert ergibt sich dabei ein Wert von 0,062 Sekunden und für den Median ein Wert von 0,061 Sekunden. In diesem Testset findet, im Vergleich zum vorherigen Testset C110_1, eine Erhöhung von 1000 auf 6001 Kundenstandorten statt. Diese Erhöhung geht mit einer Erhöhung der Generierungszeit einher. Dieser Zusammenhang wurde bereits in der Auswertung des Testset C110_1 analysiert. Bei der Durchführung der 100 Replikationen entstehen insgesamt 167694 Fehler, wobei 1666 Fehler auf doppelte Standorte entfallen. Pro Replikation werden mindestens 6 Fehler und maximal 32 Fehler erzeugt. Bei beiden Werten handelt es sich um Extremwerte, da der Mittelwert bei 16,66 und der Median bei 17 Fehlern liegen. Außerhalb der Stadtgrenzen werden 1065 Standorte erzeugt, wobei in einer Replikation mindestens 5 Fehler und maximal 18 Fehler zu finden sind. Auch hier stellen, aufgrund des Medians von 11 Fehlern und des Mittelwerts von 10,65 Fehlern, der Minimal- und der Maximalwert Extremwerte dar. In der Generierung der Kundenstandorte entstehen 164963 Berechnungsfehler. Die minimale Fehleranzahl von 1527 Fehlern pro Replikation sowie die maximale Fehleranzahl von 1762 sind als Extremwerte zu bezeichnen, da der zugehörige Mittelwert 1649,63 und der Median 1647,5 Fehler betragen. In Abbildung 33 können die genannten

Minimal- und Maximalwerte unter anderem für den ersten Simulationslauf mit 100 Replikationen eingesehen werden. Für die Tourenlänge ergibt sich ein Mittelwert von 10708008,54 LE sowie ein Median von 10710448,48 LE. Mit diesen Werten ist der Median größer als der Mittelwert. Aufgrund dessen ist von einer linksschiefen Verteilung der Tourenplanung zu sprechen. Da der Abstand vom Mittelwert zum Median 2439,94 LE und damit einer Abweichung von 0,0227 Prozent, bezogen auf den Mittelwert, beträgt, ist von einer schwachen Ausprägung der Linksschiefe beziehungsweise einer beinahe symmetrischen Verteilung der Tourenlänge zu sprechen.

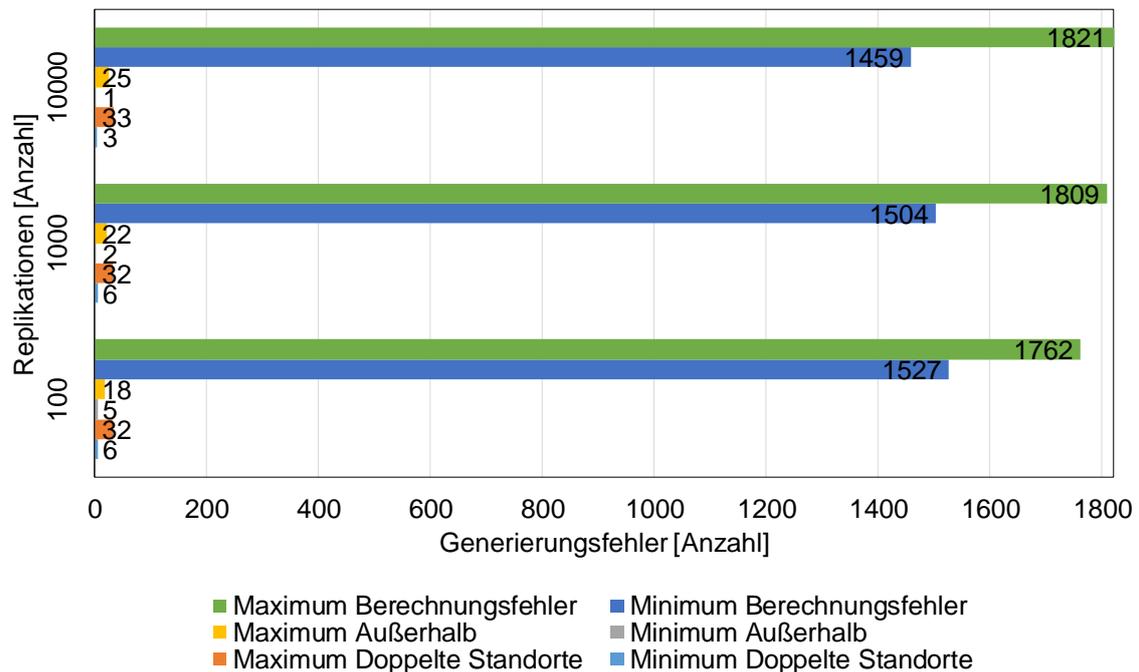


Abbildung 33: Normalverteilung Generierungsfehler pro Replikation Testset Antwerp1

Für den zweiten Simulationslauf mit 1000 Replikationen wird eine Gesamtgenerierungszeit von 63,406 Sekunden benötigt. Wie im vorherigen Simulationslauf, wird in einer Replikation eine minimale Generierungszeit von 0,046 Sekunden benötigt. Maximal beträgt die Generierungszeit 0,157 Sekunden. Sowohl der Median als auch der Mittelwert ergeben sich in diesem Simulationslauf zu 0,062 Sekunden und gleichen damit den Werten aus dem vorherigen Simulationslauf. Innerhalb dieses Simulationslaufs werden 1674021 Fehler erzeugt. 15894 Fehler entfallen dabei auf doppelte Standorte, wobei in einer Replikation minimal 6 Fehler und maximal 32 Fehler generiert werden. Bei beiden Werten handelt es sich um Extremwerte, weil der Median bei 16 Fehlern und der Mittelwert bei 15,89 Fehlern liegen. Insgesamt werden 10403 Standorte außerhalb der Stadtgrenzen erzeugt. Auf eine Replikation entfallen dabei

mindestens 2 Fehler und maximal 22 Fehler. Auch diese Werte verkörpern Extremwerte, da der Median bei 10 Fehlern und der Mittelwert bei 10,4 Fehlern liegen. Die Generierung der Standorte sorgt für 1647724 Berechnungsfehler. Mindestens werden 1504 Fehler und maximal 1809 Fehler in einer Replikation erzeugt. Aufgrund des Medians von 1648 Fehlern und des Mittelwerts von 1647,72 Fehlern, sind auch diese Werte als Extremwerte zu bezeichnen. Die angesprochenen Minimal- und Maximalwerte der Fehler sind in Abbildung 33 zu sehen. Für den Mittelwert der Tourenlänge ergibt sich ein Wert von 10707517,95 LE sowie ein Median von 10707427,96 LE, womit der Mittelwert größer als der Median ist. Aufgrund dessen liegt eine rechtsschiefe Verteilung der Tourenlänge vor. Als Abstand zwischen Median und Mittelwert ergeben sich 89,99 LE, die, bezogen auf den Mittelwert, einer Abweichung von 0,0008 Prozent entsprechen. Durch die geringe Abweichung liegt eine schwache Ausprägung der Rechtsschiefe vor und die Tourenlänge ist beinahe symmetrisch verteilt.

Die Generierung der Standorte im dritten Simulationslauf mit 10000 Replikationen benötigt eine Gesamtgenerierungszeit von 627,657 Sekunden, wobei für eine Replikation, wie im vorherigen Simulationslauf, mindestens 0,046 Sekunden und maximal 0,157 Sekunden benötigt werden. Als Mittelwert ergibt sich eine Zeit von 0,061 Sekunden und der Median liegt bei 0,062 Sekunden, beide Werte gleichen denen des vorherigen Simulationslaufes. Die Durchführung aller 10000 Replikationen führt zu insgesamt 16728024 Generierungsfehlern, wobei 159730 Fehler auf doppelte Standorte zurückzuführen sind. Pro Replikation werden mindestens 3 und maximal 33 dieser Fehler erzeugt. Bei beiden Werten handelt es sich um Extremwerte, da der Mittelwert bei 15,97 und der Median bei 16 Fehlern liegen. Insgesamt werden 103526 Standorte außerhalb des Stadtgebietes erzeugt. Dabei liegt die minimale Anzahl bei 1 Fehler und die Maximale bei 25 Fehlern. Die minimalen und maximalen Werte dieses Fehlers stellen Extremwerte dar, weil der Median bei 10 und der Mittelwert bei 10,35 liegt. Für die Berechnungsfehler ergibt sich insgesamt eine Anzahl von 16464768. Innerhalb einer Replikation werden mindestens 1459 und maximal 1821 Fehler erzeugt, bei denen es sich in Anbetracht des Medians mit 1646 und des Mittelwerts von 1646,48 Fehlern um Extremwerte handelt. Die Minimal- und Maximalwerte der Fehler können in Abbildung 33 gefunden werden. In diesem Simulationslauf wird für die Tourenlänge ein Mittelwert von 10708670 LE und ein Median von 10708441,89 LE erreicht. Wie im vorherigen Simulationslauf ist der Mittelwert damit weiterhin größer als der Median. Somit bleibt die rechtsschiefe Verteilung der Tourenlänge bestehen. Der Abstand zwischen Mittelwert und Median

beträgt 228,11 LE. Damit ergibt sich eine Abweichung von 0,0021 Prozent, bezogen auf den Mittelwert. Aus diesem Grund ist die rechtsschiefe Verteilung der Tourenlänge schwach ausgeprägt und kann auch als beinahe symmetrisch bezeichnet werden. Werden alle Simulationsläufe betrachtet, so ist erkennbar, dass eine Verzehnfachung der Replikationen sowohl zu einer Verzehnfachung der Generierungszeit als auch der Generierungsfehler führt. Außerdem verändern sich über alle Simulationsläufe sowohl der Mittelwert als auch der Median der Tourenlänge, der Generierungszeit und der Generierungsfehler nicht signifikant.

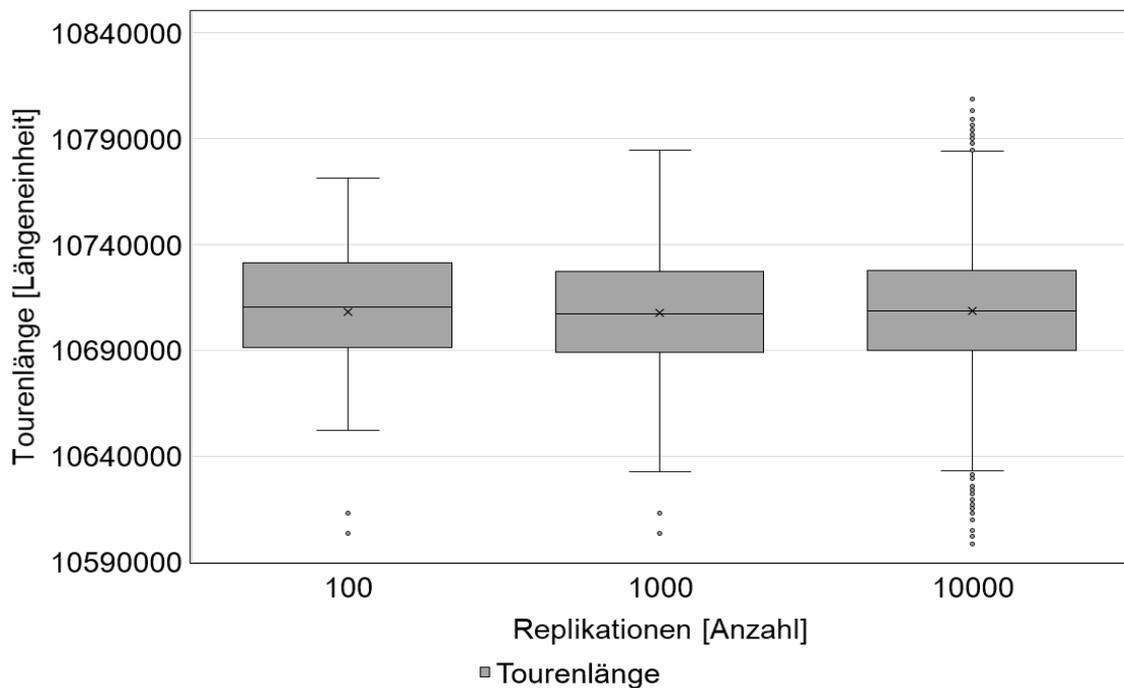


Abbildung 34: Normalverteilung Box-Plots Testset Antwerp1

Für die drei Simulationsläufe sind die zugehörigen Tourenlängen in Abbildung 34 jeweils in ein Box-Plot-Diagramm eingetragen worden. Innerhalb des Diagramms ist der Mittelwert durch x gekennzeichnet und der Median durch eine horizontale Linie in der Box. Die linksschiefe Verteilung in dem Simulationslauf mit 100 Replikationen ist in der Abbildung daran erkennbar, dass der Mittelwert unterhalb des Medians aufgetragen ist. Äquivalent ist die Rechtsschiefe der Tourenlänge im zweiten und dritten Simulationslauf an der Lage des Mittelwerts oberhalb des Medians erkennbar. Innerhalb des Box-Plot-Diagramms befinden sich 50 Prozent der Werte für die Tourenlänge. Aus der Abbildung wird ersichtlich, dass die Box des ersten Simulationslauf größer zu sein scheint als die Box des zweiten und dritten Simulationslaufs. Die Interquartilsabstände der drei Box-Plot-Diagramme bestätigen diese Beobachtung. Im ersten Simulationslauf beträgt der

Interquartilsabstand 39092,49 LE, im zweiten Simulationslauf sinkt der Wert auf 37998,92 LE und erreicht im dritten Simulationslauf mit 37886,49 sein Minimum. Die Interquartilsabstände der letzten beiden Simulationsläufe sind damit beinahe gleich groß. Der Abstand zwischen dem Interquartilsabstand des ersten und des dritten Simulationslaufs beträgt 1093,57 LE, was einer Abweichung von 2,877 Prozent entspricht, bezogen auf den maximalen Interquartilsabstand. Die Werte der unteren Grenzen der Boxen ähneln einander und betragen für den ersten Simulationslauf 10691295,16 LE, für den zweiten Simulationslauf 10689183,79 LE sowie 10689722,52 LE für den dritten Simulationslauf. Durch die geringe Abweichung der Grenzen sowie der Interquartilsabstände ist aussagbar, dass sich 50 Prozent der Werte für die Tourenlänge, im zweiten und dritten Simulationslauf, in ähnlichen Grenzen bewegen. Einen weiteren Unterschied stellen die unterschiedlich langen Whisker der Box-Plots dar. Im ersten Simulationslauf liegt die untere Grenze des Whiskers bei 10652109,1 LE und die Obere bei 10771254,42 LE. Für den zweiten Simulationslauf ergibt sich als untere Grenze ein Wert von 10632696,37 LE und für die Obere ein Wert von 10784711,7 LE sowie für den dritten Simulationslauf äquivalent ein Wert von 10633111,29 LE und ein Wert von 10784319,48 LE. Zusätzlich unterscheiden sich die drei Simulationsläufe in der Anzahl ihrer Ausreißer. Innerhalb des ersten Simulationslaufs sowie des zweiten Simulationslaufs existieren 2 untere Ausreißer. Im dritten Simulationslauf gibt es 37 untere sowie 37 obere Ausreißer. Die Auswirkungen der unterschiedlich langen Whisker, der Ausreißer und der signifikant voneinander abweichenden Interquartilsabstände können jedoch als gering eingeschätzt werden, da sich der Mittelwert, der für Ausreißer anfällig ist, von 10708008,54 LE bei 100 Replikationen auf 10708670 LE bei 10000 Replikationen erhöht. Dies entspricht einem Abstand von 661,46 LE zwischen beiden Mittelwerten und einer Abweichung von 0,0062 Prozent, bezogen auf den Mittelwert der 10000 Replikationen.

In Abbildung 35 sind die approximierten 95% Konfidenzintervalle der drei Simulationsläufe zu finden. Der Abbildung ist zu entnehmen, dass für eine höhere Anzahl an Replikationen die Breite des Konfidenzintervalls kleiner wird. Das Konfidenzintervall des ersten Simulationslaufs hat eine Breite von 422028,63 LE. Für das Konfidenzintervall des zweiten Simulationslaufs ergibt sich eine Breite von 42139,08 LE und erreicht im dritten Simulationslauf mit 4341,89 LE sein Minimum. Bei einer steigenden Anzahl an Replikationen nähert sich das Konfidenzintervall dem entsprechenden Mittelwert an. Aufgrund des approximierten 95% Konfidenzintervall des dritten Simulationslaufes kann

ausgesagt werden, dass der wahre Wert der Tourenlänge mit einer Fehlerwahrscheinlichkeit von 5 Prozent durch das Intervall der Dimension (10706499,05; 10710840,94) überdeckt wird.

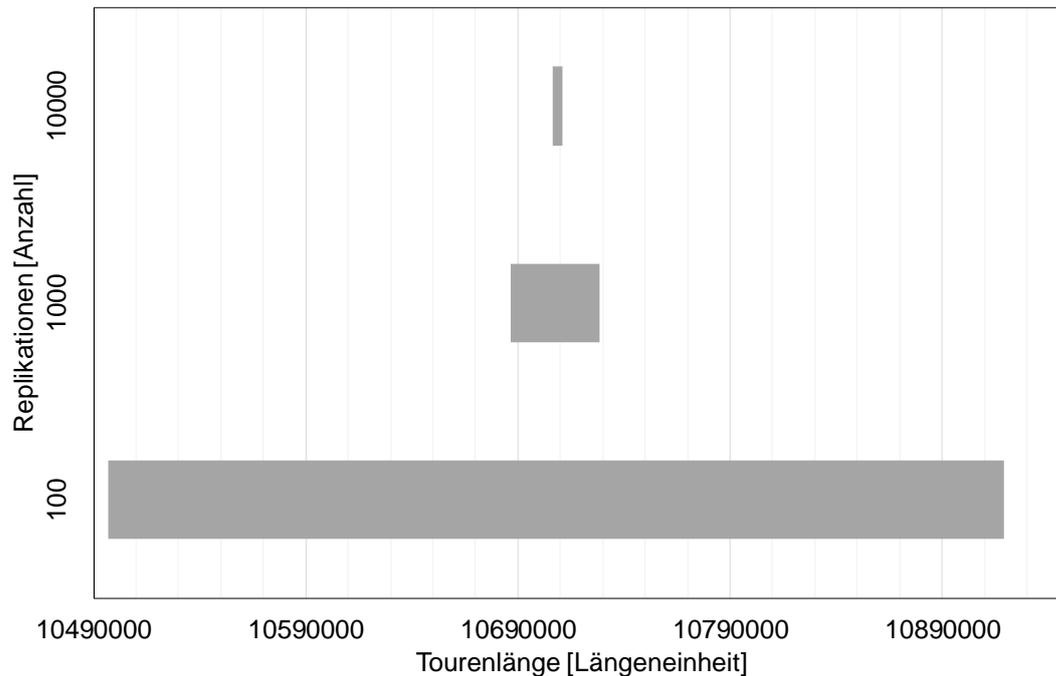


Abbildung 35: Normalverteilung 95% Konfidenzintervalle Testset Antwerp1

Aus der Auswertung der drei Simulationsläufe ist, auch für eine hohe Anzahl Replikationen, keine signifikante Veränderung der Lagemaße der Tourenlänge, der Generierungsfehler und der Generierungszeit erkennbar. Die Ergebnisse jedes Simulationslaufs können deswegen im Vergleich der Generierungsverfahren in Abschnitt 7.1 herangezogen werden. Mit dem approximierten 95% Konfidenzintervall des dritten Simulationslaufs ist, aufgrund dessen geringer Breite, die präziseste Aussage zur Größenordnung der Tourenlänge möglich- Aus diesem Grund werden die Ergebnisse dieses Simulationslaufs im Vergleich genutzt.

6.3 Simulation mit einer Dreiecksverteilung

In diesem Abschnitt des sechsten Kapitels werden die Standorte mit Hilfe einer Dreiecksverteilung generiert und im Anschluss daran ausgewertet. Dafür wurde der Abschnitt in vier Unterabschnitte, 6.3.1 bis 6.3.4, eingeteilt. Innerhalb jedes Unterabschnittes erfolgt die Generierung der Kundenstandorte sowie die Auswertung für ein Testset. Die eingesetzten Formeln (19) bis (21) benötigen als Parameter das Intervall

(a; b), auf dem die Standorte generiert werden, sowie den wahrscheinlichsten Wert m , den die Verteilung annehmen wird. Der wahrscheinlichste Wert wird in jedem Testset so festgelegt, dass er die Mitte des Stadtgebietes darstellt. Die Parameter werden für die Testsets so eingestellt, dass jeweilig auf dem gesamten Stadtgebiet Kundenstandorte erzeugt werden können. Der einzige Generierungsfehler, der in diesem Verfahren auftritt, ist der Fehler der doppelten Standorte. Damit die Entwicklung der Kennzahlen über eine steigende Anzahl Replikationen beobachtet werden kann, werden für jedes Testset drei Simulationsläufe mit 100, 1000 und 10000 Replikationen durchgeführt (vgl. Abschnitt 5.2). Im Anhang in Tabelle 41 und Tabelle 42 sind alle Kennzahlen sowie für die Grafiken benötigten Werte aufzufinden.

6.3.1 Dreiecksverteilung im Testset E-n22-k4

Als erstes erfolgt die Generierung der Kundenstandorte im Testset E-n22-k4. Für dieses Testset sind die begrenzenden Koordinaten auf der x -Achse 165 LE sowie auf der y -Achse 265 LE (vgl. Tabelle 5). In der Simulation werden somit die x -Koordinaten im Intervall (0; 165) und die y -Koordinaten im Intervall (0; 265). Für die wahrscheinlichsten Werte der x - und y -Koordinaten werden jeweils die Mitte der zugehörigen Intervalle angegeben. So werden die Standorte häufiger in der Mitte des Stadtgebiets als an dessen Rand erzeugt. Der wahrscheinlichste Wert für die x -Koordinate liegt damit bei 82,5 LE, der für die y -Koordinate bei 132,5 LE. Innerhalb des Testset existieren 22 Kundenstandorte, weswegen in jeder Replikation 22 Kundenstandorte erzeugt werden. In Tabelle 17 ist eine Auswahl an Ergebniswerten der drei Simulationsläufe zu finden, wobei diese Auswahl keine Bewertung der Ergebnisse darstellt. Sie dient lediglich einer ersten Übersicht. Alle berechneten Werte können im Anhang in Tabelle 41 gefunden werden.

Tabelle 17: Dreiecksverteilung Testset E-n22-k4

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	4377,94	4385,71	1	0,218
1000	4377,63	4381,02	9	2,281
10000	4387,25	4384,16	86	11,297

Für den ersten Simulationslauf dieses Testsets wird eine Gesamtgenerierungszeit von 0,218 Sekunden benötigt. Die minimale und maximale Generierungszeit sowie der Mittelwert und der Median ergeben jeweils 0 Sekunden. Die Generierung von 22 Standorten mit einer Dreiecksverteilung nimmt damit sehr wenig Zeit in Anspruch. In diesem Simulationslauf wird im Gesamten 1 doppelter Standorte erzeugt. Dadurch ergibt sich für das Minimum ein Wert von 0 Fehlern sowie für das Maximum ein Wert von 1 Fehler. Der Median liegt bei 0 Fehlern und der Mittelwert bei 0,01 Fehlern. Der Maximalwert wird in nur einer Replikation erreicht und stellt damit einen Extremwert dar. Dies wird auch aus Abbildung 36 deutlich, in der die Werte für das Minimum, das Maximum sowie den Mittelwert eingetragen sind. In dieser Abbildung ist der Mittelwert, der für Ausreißer anfällig ist, sichtbar kleiner als der Maximalwert, wodurch die Aussage unterstützt wird, dass es sich bei dem Maximalwert um einen Extremwert handelt. Der Mittelwert der Tourenlänge für 100 Replikationen beträgt 4377,94 LE und der Median 4385,71 LE, womit der Mittelwert kleiner als der Median ist. Dies legt die Vermutung einer linksschiefen Verteilung der Tourenlänge nahe. Als Abstand zwischen Mittelwert und Median ergeben sich 7,77 LE, die einer Abweichung von 0,1774 Prozent, bezogen auf den Mittelwert, entsprechen. Die Linksschiefe der Verteilung ist damit nur schwach ausgeprägt und die Tourenlänge ist beinahe symmetrisch verteilt.

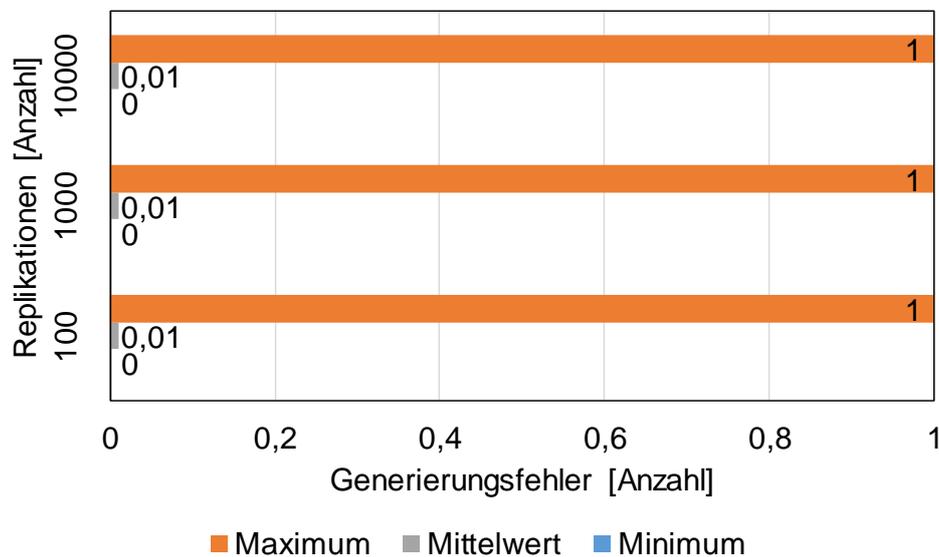


Abbildung 36: Dreiecksverteilung Generierungsfehler pro Replikation Testset E-n22-k4

Im zweiten Simulationslauf mit 1000 Replikationen wird eine Gesamtgenerierungszeit von 2,281 Sekunden benötigt. Die minimale Generierungszeit liegt bei 0 Sekunden, die Maximale bei 0,016 Sekunden. Wie im vorherigen Simulationslauf mit 100 Replikationen

betragen sowohl der Median als auch der Mittelwert 0 Sekunden. Somit bleibt die Aussage bestehen, dass die Generierung von 22 Standorten mit einer Dreiecksverteilung sehr wenig Zeit beansprucht. Für den Fehler der doppelten Standorte ergibt sich in diesem Simulationslauf ein Wert von 9 Fehlern. Das Minimum liegt bei 0 Fehlern und das Maximum bei 1 Fehler. Aufgrund des Medians von 0 Fehlern und des Mittelwerts von 0,01 Fehlern, ist der Maximalwert als Extremwert zu bezeichnen. In Abbildung 36 ist der Maximalwert auch als Extremwert zu erkennen, da der Mittelwert, der für Ausreißer sehr anfällig ist, sichtbar kleiner als der Maximalwert ist. Für den Mittelwert ergeben sich 4377,63 LE und für den Median 4381,02 LE, wodurch der Mittelwert, wie im vorherigen Simulationslauf, kleiner als der Median ist. Die Vermutung der linksschiefen Verteilung der Tourenlänge bleibt damit bestehen. Aufgrund des Abstands zwischen Mittelwert und Median von 3,39 LE und der sich daraus ergebenden Abweichung von 0,0774 Prozent, bezogen auf den Mittelwert, ist von einer schwachen Ausprägung der Linksschiefe zu sprechen. Durch die geringe Abweichung kann die Verteilung der Tourenlänge als beinahe symmetrisch bezeichnet werden.

Der dritte Simulationslauf mit 10000 Replikationen benötigt eine Gesamtgenerierungszeit von 11,297 Sekunden. Wie im vorherigen Simulationslauf ergeben sich eine minimale Generierungszeit von 0 Sekunden, eine Maximale von 0,016 und ein Median sowie Mittelwert von 0 Sekunden. Damit benötigt die Generierung von 22 Standorten mit einer Dreiecksverteilung weiterhin wenig Zeit. Insgesamt werden in den 10000 Replikationen 86 doppelte Standorte erzeugt. Die minimale Anzahl dieses Fehlers beträgt 0 Fehler und die maximale Anzahl 1 Fehler. Wie in den vorherigen Simulationsläufen liegt der Median bei 0 Fehlern und der Mittelwert bei 0,01 Fehlern. Damit handelt es sich bei dem Maximalwert der doppelten Standorte um einen Extremwert. Dies ist auch aus Abbildung 36 ersichtlich, denn in dieser ist der für Ausreißer anfällig Mittelwert deutlich kleiner als der Maximalwert. Für den Mittelwert der Tourenlänge ergibt sich ein Wert von 4387,25 LE und für den Median in diesem Simulationslauf ein Wert von 4384,16 LE. Mit diesen Werten ist der Mittelwert größer als der Median, wodurch die linksschiefe Verteilung der Tourenlänge in eine rechtsschiefe Verteilung übergeht. Aufgrund des Abstands zwischen Mittelwert und Median von 3,09 LE und der sich daraus ergebenden Abweichung von 0,0704 Prozent, bezogen auf den Mittelwert, ist die Rechtsschiefe als schwach ausgeprägt zu bezeichnen. Die geringe Abweichung der beiden Lagemaße lässt auch die Aussage einer beinahe symmetrischen Verteilung der Tourenlänge zu. Die Betrachtung aller Simulationsläufe

zeigt, dass sich sowohl der Median als auch der Mittelwert der Tourenlänge in jedem Simulationslauf nicht signifikant verändern. Auch der Mittelwert und der Median der Generierungsfehler sowie der Generierungszeit unterscheiden sich von Simulationslauf zu Simulationslauf in nicht signifikanter Weise voneinander. Somit führt eine Verhundertfachung der Replikationen nicht zu signifikant unterschiedlichen Werten der genannten Lagemaße.

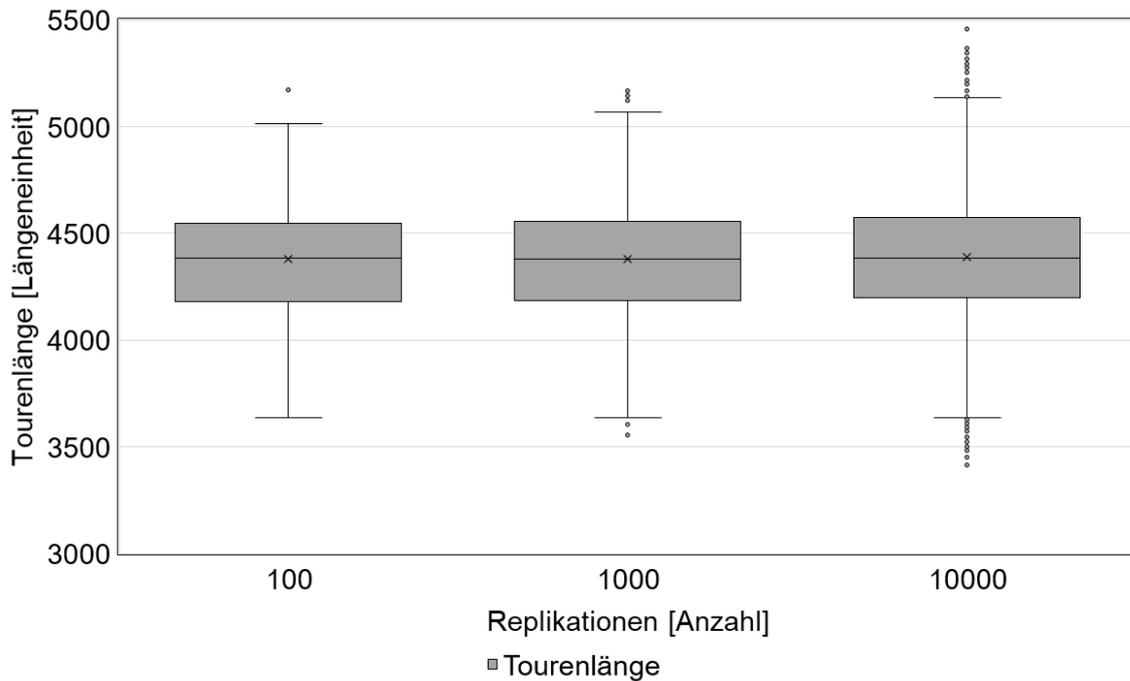


Abbildung 37: Dreiecksverteilung Box-Plots Testset E-n22-k4

Die Werte der Tourenlänge sind für jeden Simulationslauf in Abbildung 37 in ein eigenes Box-Plot-Diagramm eingetragen. In den Box-Plots ist der Mittelwert durch x und der Median durch eine horizontale Linie innerhalb der Box gekennzeichnet. Die sinkenden Abstände zwischen dem Mittelwert und dem Median der Tourenlänge können auch in den Box-Plot-Diagrammen beobachtet werden. Zudem ist auch die linksschiefe beziehungsweise rechtsschiefe Verteilung der Tourenlänge zu sehen, denn in den ersten beiden Simulationsläufen liegt der Mittelwert unterhalb des Medians und im dritten Simulationslauf ist dies der umgekehrte Fall. Die Boxen, in denen sich 50 Prozent aller Werte für die Tourenlänge befinden, besitzen im Diagramm eine vergleichbare Größe. Mit den Werten für die Interquartilsabstände der drei Simulationsläufe kann diese Beobachtung bestätigt werden. Für den ersten Simulationslauf liegt dieser bei 365,94 LE. Im zweiten Simulationslauf steigt der Interquartilsabstand auf 370,77 LE und erreicht im Dritten mit 375,95 LE sein Maximum. Außerdem weisen die unteren Grenzen der Box

eine wertmäßige Ähnlichkeit auf. Diese betragen für den ersten Simulationslauf 4179,38 LE, für den Zweiten 4186,53 LE sowie für den Dritten 4198,51 LE. An diesen Werten ist erkennbar, dass sich die Grenzen für 50 Prozent der Werte auch bei einer höheren Anzahl Replikationen nicht signifikant verändern. Ein erkennbarer Unterschied zwischen den drei Box-Plots besteht in der Länge der Whisker. Diese enden im Falle des ersten Simulationslauf bei 3636,93 LE für den unteren Whisker und 5012,62 LE für den Oberen. Im zweiten Simulationslauf mit 1000 Replikationen ist die Grenze des unteren Whisker 3636,63 LE und die des Oberen 5067,6 LE sowie für den Dritten äquivalent 3636,63 LE und 5137,28 LE. Weiterhin weisen die Box-Plot-Diagramme unterschiedlich viele Ausreißer auf. Im ersten Simulationslauf entsteht ein oberer Ausreißer. Für den zweiten Simulationslauf ergeben sich zwei untere und drei obere Ausreißer und im dritten Simulationslauf 35 untere sowie 31 obere Ausreißer. Anhand dessen ist erkennbar, dass mit einer Erhöhung der Replikationen von 100 auf 10000 eine Erhöhung der Streubreite der Tourenlänge sowie der Ausreißer einhergeht. Der Einfluss dieser beiden Faktoren kann jedoch als gering eingeschätzt werden, da der Mittelwert, der für Ausreißer anfällig ist, von 4377,94 LE im ersten Simulationslauf auf 4387,25 LE im dritten Simulationslauf ansteigt. Dies entspricht einem Abstand von 9,31 LE und bezogen auf den Mittelwert des dritten Simulationslaufs einer Abweichung von 0,2122 Prozent.

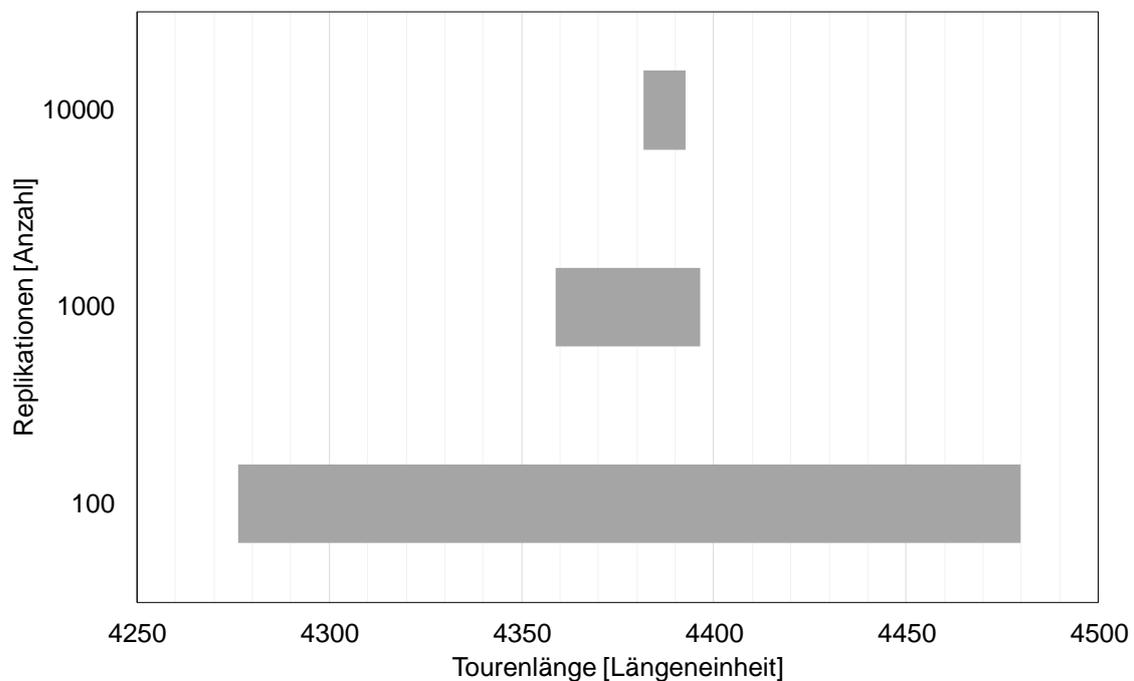


Abbildung 38: Dreiecksverteilung 95% Konfidenzintervalle Testset E-n22-k4

In Abbildung 38 sind die approximierten 95% Konfidenzintervalle für die drei Simulationsläufe abgebildet. Aus der Abbildung ist erkennbar, dass die Breite des Konfidenzintervalls für eine steigende Anzahl Replikationen sinkt. Für den ersten Simulationslauf ergibt sich eine Breite des Konfidenzintervalls von 203,47 LE. Im zweiten Simulationslauf sinkt diese auf 37,75 LE und erreicht im dritten Simulationslauf mit 10,94 LE ihr Minimum. Zudem nähert sich die Breite des Konfidenzintervalls mit steigender Anzahl Replikationen dem Mittelwert des zugehörigen Simulationslaufs an. Auf Basis des approximierten 95% Konfidenzintervalls des dritten Simulationslaufs mit 10000 und der Dimension (4381,78; 4392,72) kann ausgesagt werden, dass dieses mit einer Fehlerwahrscheinlichkeit von 5 Prozent den wahren Wert der Tourenlänge überdeckt. Die Auswertung zeigt, auch bei einer Erhöhung auf 10000 Replikationen verändern sich sowohl der Median als auch der Mittelwert der Tourenlänge, der Generierungszeit und der Generierungsfehler nicht signifikant. Deswegen können die Ergebnisse jedes Simulationslaufs für den Vergleich der Generierungsverfahren im Abschnitt 7.1 genutzt werden. Aufgrund der kleinsten Breite des approximierten 95% Konfidenzintervalls bei 10000 Replikationen, womit eine präzisere Vorhersage der Tourenlänge möglich ist, werden die Ergebnisse des dritten Simulationslaufs in den Vergleich einfließen.

6.3.2 Dreiecksverteilung im Testset CMT14

Im Folgenden wird die Generierung der Kundenstandorte mit einer Dreiecksverteilung im Stadtgebiet des Testsets CMT14 durchgeführt und anschließend ausgewertet. Die begrenzenden Koordinaten sind 95 LE auf der x- sowie 85 LE auf der y-Achse (vgl. Tabelle 5). In den Simulationsläufen werden die x-Koordinaten der Standorte deswegen auf dem Intervall (0; 95) und die y-Koordinaten auf dem Intervall (0; 85) erzeugt. Der wahrscheinlichste Wert für die x-Koordinate beträgt 47,5 LE und der für die y-Koordinate 42,5 LE. In den Simulationen werden in jeder Replikation 100 Kundenstandorte erzeugt, weil das Testset 100 Kunden enthält. Eine Auswahl der Ergebnisse ist in Tabelle 18 zu finden, wobei diese Auswahl keine Bewertung der Ergebnisse darstellt. Im Anhang in Tabelle 41 sind alle Kennzahlen und berechneten Werte der Simulationsläufe zu finden.

Table 18: Dreiecksverteilung Testset CMT14

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	8185,46	8193,06	106	0,313
1000	8190,05	8189,72	1074	2,234
10000	8192,56	8190,30	10735	20,250

Für die 100 Replikationen des ersten Simulationslaufs wird eine Gesamtgenerierungszeit von 0,313 Sekunden benötigt. Die minimale Generierungszeit einer Replikation liegt bei 0 Sekunden, die Maximale bei 0,016 Sekunden. Als Mittelwert und Median einer Replikation ergeben sich 0 Sekunden. Daran ist erkennbar, dass die Generierung von 100 Kundenstandorten wenig Zeit in Anspruch nimmt. In diesem Simulationslauf werden insgesamt 106 doppelte Standorte erzeugt, wobei die minimale Anzahl dieses Fehlers in einer Replikation bei 0 liegt. Maximal werden in einer Replikation 4 doppelte Standorte erzeugt. Bei diesem Maximalwert handelt es sich jedoch um einen Ausreißer, da der Mittelwert bei 1,06 Fehlern und der Median bei 1 Fehler liegen. Dies ist in Abbildung 39 nochmals verdeutlicht, da der Mittelwert, der für Ausreißer anfällig ist, sichtbar niedriger als der Maximalwert ist. Als Mittelwert der Tourenlänge ergibt sich ein Wert von 8185,46 LE und der Median liegt bei 8193,06 LE. Mit diesen Werten ist der Mittelwert kleiner als der Median, woraus auf eine linksschiefe Verteilung der Tourenlänge geschlossen werden kann. Aufgrund des Abstands beider Lagemaße von 7,6 LE, was einer Abweichung von 0,0928 Prozent, bezogen auf den Mittelwert, entspricht, ist diese Linksschiefe als schwach ausgeprägt zu beschreiben und kann auch als beinahe symmetrisch bezeichnet werden.

Im zweiten Simulationslauf wird für die 1000 Replikationen eine Gesamtgenerierungszeit von 2,234 Sekunden benötigt. Wie im vorherigen Simulationslauf beträgt die minimale Generierungszeit einer Replikation 0 Sekunden und die Maximale 0,016 Sekunden. Der Median beträgt 0 Sekunden und der Mittelwert 0,001 Sekunden. Damit bleibt die Annahme bestehen, dass die Generierung von 100 Kundenstandorten wenig Zeit in Anspruch nimmt. Insgesamt ergeben sich für den zweiten Simulationslauf 1074 doppelte Standorte, wovon in einer Replikation minimal 0 erzeugt werden. Als Maximum dieses Fehlers ergeben sich 6 doppelte Standorte in einer Replikation. Aufgrund des Medians von 1 Fehler und dem Mittelwert von 1,07 Fehlern, ist der Maximalwert als Extremwert

zu bezeichnen. In Abbildung 39 ist dies ebenfalls zu beobachten, denn der Mittelwert ist sichtbar kleiner als der Maximalwert. Für die Tourenlänge dieses Simulationslaufs ergibt sich ein Mittelwert von 8190,05 LE und ein Median von 8189,72 LE. Damit ist der Mittelwert größer als der Median und die linksschiefe Verteilung der Tourenlänge des vorherigen Simulationslaufs wechselt zu einer rechtsschiefen Verteilung. Diese ist jedoch nicht sonderlich ausgeprägt und kann auch als beinahe symmetrisch bezeichnet werden, da der Abstand zwischen Mittelwert und Median 0,33 LE beträgt und dies einer Abweichung von 0,004 Prozent, bezogen auf den Mittelwert, entspricht.

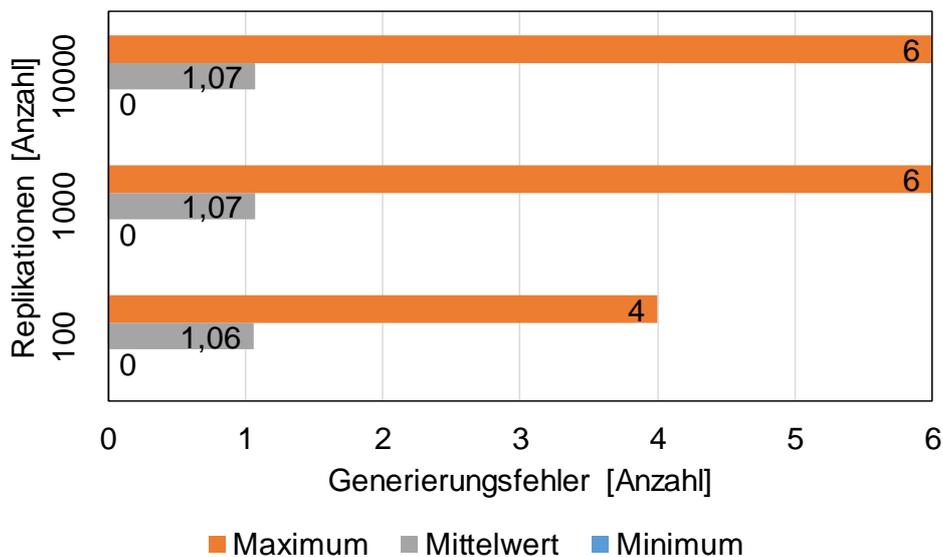


Abbildung 39: Dreiecksverteilung Generierungsfehler pro Replikation Testset CMT14

Der dritte Simulationslauf mit 10000 Replikationen benötigt eine Gesamtgenerierungszeit von 20,25 Sekunden. Für eine Replikation wird eine minimale Generierungszeit von 0 Sekunden sowie eine Maximale von 0,016 Sekunden benötigt. Wie im vorherigen Simulationslauf betragen der Median 0 Sekunden und der Mittelwert 0,001 Sekunden. Weiterhin bleibt die Annahme der geringen Generierungszeit für 100 Kundenstandorte bestehen. In diesem Simulationslauf ergeben sich insgesamt 10735 doppelte Standorte. In einer Replikation entstehen mindestens 0 Fehler und maximal 6. Die Maximalanzahl stellt einen Extremwert dar, da der Median 1 und der Mittelwert 1,07 betragen. Dies ist auch aus Abbildung 39 zu entnehmen, da der Mittelwert sichtbar kleiner als der Maximalwert ist. Als Mittelwert der Tourenlänge ergeben sich 8192,56 LE und für den Median ein Wert von 8190,3 LE. Der Mittelwert ist somit größer als der Median, die rechtsschiefe Verteilung aus dem vorherigen Simulationslauf bleibt bestehen. Für den Abstand zwischen Mittelwert und Median ergeben sich 2,26 LE, was einer Abweichung

von 0,0275 Prozent entspricht. Die Rechtsschiefe ist somit schwach ausgeprägt und die Verteilung der Tourenlänge kann als beinahe symmetrisch verteilt beschrieben werden. Werden alle Simulationsläufe zusammen betrachtet ist auffällig, dass sich der Median sowie der Mittelwert der Tourenlänge jedes Simulationslaufs nicht signifikant voneinander unterscheiden. Dieselbe Beobachtung gilt für den Median und Mittelwert der Generierungsfehler sowie für den Median und Mittelwert der Generierungszeit. Somit wird mit einer Verhundertfachung der Replikationen keine signifikante Veränderung der genannten Lagemaße herbeigeführt. Des Weiteren führt eine Verzehnfachung der Replikationen sowohl zu einer Verzehnfachung der Generierungszeit als auch der Generierungsfehler.

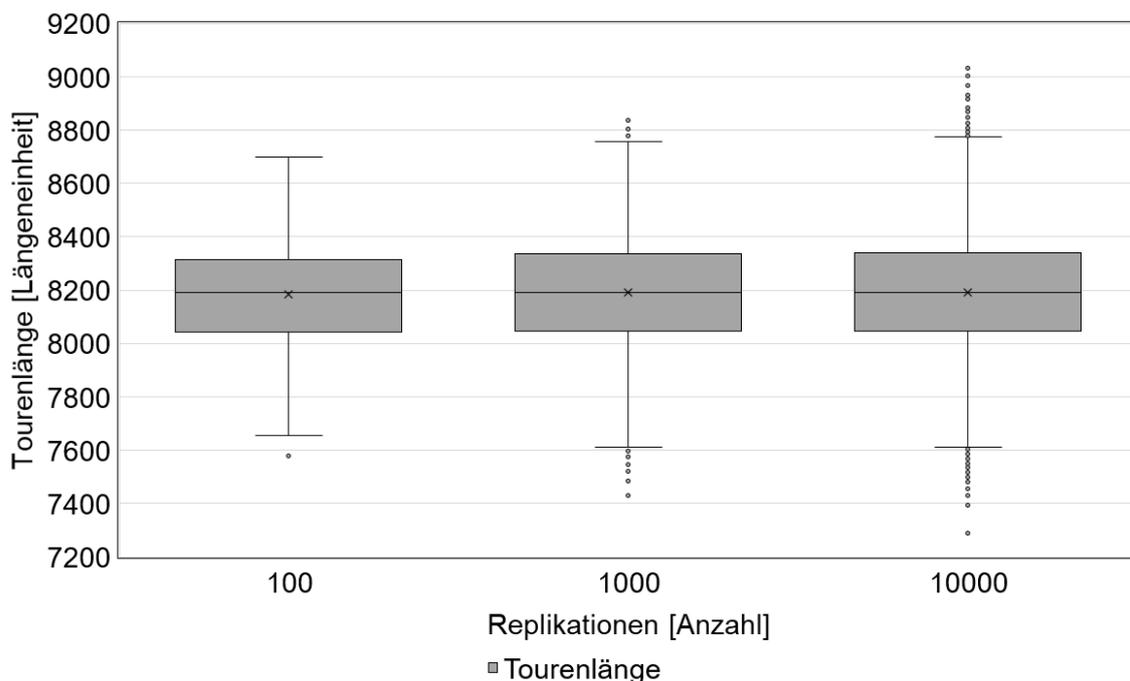


Abbildung 40: Dreiecksverteilung Box-Plots Testset CMT14

Für die Werte der Tourenlänge wurde für jeden Simulationslauf ein eigenes Box-Plot-Diagramm erstellt, welches in Abbildung 40 zu sehen ist. Der Mittelwert wird in den Box-Plots mit Hilfe des x markiert, der Median durch eine horizontale Linie innerhalb der Box. Die links- beziehungsweise rechtsschiefe Verteilung der Tourenlänge kann auch in der Abbildung erkannt werden. Im ersten Simulationslauf liegt der Mittelwert unterhalb des Medians und in den letzten beiden Simulationsläufen liegt der Mittelwert jeweils über dem Median. Die Box, in der sich 50 Prozent der Werte der Tourenlänge befinden, ist in der Abbildung sichtbar kleiner als die sich gleichenden Boxen des zweiten und dritten Simulationslaufs. Mit Hilfe der Werte der Interquartilsabstände kann diese Beobachtung

bestätigt werden. Für den Interquartilsabstand des ersten Simulationslaufs ergibt sich ein Wert von 273,32 LE. Im zweiten Simulationslauf steigt dieser Wert auf 291,31 LE und erreicht im dritten Simulationslauf bei 292,02 LE sein Maximum. Zwischen dem Interquartilsabstand des ersten Simulationslauf und dem Maximum liegen 18,7 LE, was bezogen auf das Maximum eine Abweichung von 6,4 Prozent darstellt. Die untere Grenze der Box liegt für den ersten Simulationslauf bei 8044,31 LE, im zweiten Simulationslauf bei 8045,7 LE sowie für den dritten Simulationslauf bei 8046,79 LE. Damit beginnen alle Boxen bei einem ähnlichen Wert. Zwischen den unteren Grenzen des zweiten und dritten Simulationslaufs gibt es keine signifikanten Unterschiede. Daran ist erkennbar, dass sich die Grenzen, in denen sich 50 Prozent der Tourenlänge befinden, vom ersten zum zweiten Simulationslauf verändern, sich jedoch bei einer weiteren Erhöhung auf 10000 Replikationen nicht weiter verschieben. Ein sichtbarer Unterschied zwischen den drei Box-Plots in Abbildung 40 ist die Länge der Whisker. Diese werden für den ersten Simulationslauf durch 7656,29 LE und 8700,14 LE begrenzt. Für den zweiten Simulationslauf verändern sich die Grenzen auf 7611,08 und 8758,45 LE und sind im dritten Simulationslauf mit 7609,98 LE sowie 8773,49 LE beinahe gleich. Einen weiteren Unterschied stellen die Ausreißer der drei Simulationsläufe dar. Im ersten Simulationslauf ergibt sich 1 unterer Ausreißer, im zweiten Simulationslauf 6 untere und 3 obere sowie für den dritten Simulationslauf 47 untere und 44 obere Ausreißer. Obwohl die Streubreite der Tourenlänge und die Anzahl der Ausreißer mit einer Erhöhung der Replikationen ansteigt und sich der Interquartilsabstand vom ersten Simulationslauf zu den anderen beiden signifikant unterscheidet, verändert sich der für Ausreißer anfällig Mittelwert kaum. Dieser steigt von 8185,46 LE bei 100 Replikationen auf 8192,56 LE bei 10000 Replikationen an, was einem Unterschied von 7,1 LE oder einer Abweichung von 0,0866 Prozent entspricht.

Die approximierten 95% Konfidenzintervalle sind in Abbildung 41 für die drei Simulationsläufe abgebildet. Daraus ist erkennbar, dass die Breite des Konfidenzintervalls für eine steigende Anzahl Replikationen sinkt. Im ersten Simulationslauf ergibt sich für die Breite des Konfidenzintervalls ein Wert von 335,76 LE. Die Breite sinkt im zweiten Simulationslauf auf 42,76 LE und erreicht im dritten Simulationslauf mit 10000 Replikationen mit 9,28 LE ihr Minimum. Je höher die Anzahl der Replikationen ist, umso mehr nähert sich die Breite des Konfidenzintervalls dem zugehörigen Mittelwert des Simulationslaufs an. Mit Hilfe des approximierten 95% Konfidenzintervalls des dritten Simulationslaufs kann ausgesagt werden, dass das

Konfidenzintervall mit der Dimension (8187,92; 8197,2) den wahren Wert der Tourenlänge mit einer Wahrscheinlichkeit von 95% überdeckt. In der Auswertung der drei Simulationsläufe ist, auch bei einer Erhöhung auf 10000 Replikationen, keine signifikante Veränderung in den Lagemaßen der Tourenlänge, der Generierungszeit sowie der Generierungsfehler erkennbar. Aus diesem Grund ist es möglich die Ergebnisse jedes Simulationslaufs für den Vergleich der Generierungsverfahren im Abschnitt 7.1 heranzuziehen. Weil die präziseste Aussage zur Größenordnung der Tourenlänge mit dem approximierten 95% Konfidenzintervall des dritten Simulationslaufs möglich ist, fließen die Ergebnisse dieses Simulationslaufs in den Vergleich ein.

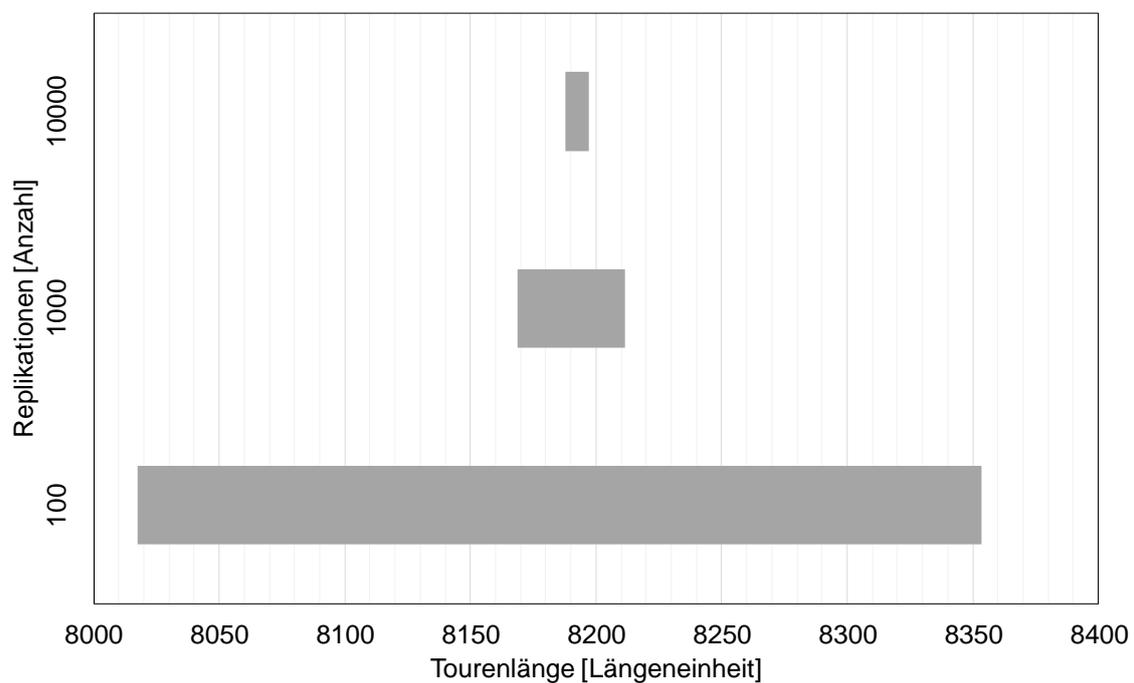


Abbildung 41: Dreiecksverteilung 95% Konfidenzintervalle Testset CMT14

6.3.3 Dreiecksverteilung im Testset C110_1

Nachfolgend werden die Kundenstandorte mit Hilfe der Dreiecksverteilung über das Stadtgebiet des Testsets C110_1 verteilt und anschließend ausgewertet. Aus dem Testset ergeben sich die begrenzenden Koordinaten sowohl auf der x- als auch auf der y-Achse zu 500 LE (vgl. Tabelle 5). Deswegen werden sowohl die x- als auch die y-Koordinaten der Kundenstandorte im Intervall (0; 500) erzeugt. Auch die wahrscheinlichsten Werte der x- und y-Koordinaten sind identisch und liegen bei 250 LE. Das vorliegende Testset enthält 1000 Kundenstandorte, weswegen in jeder Replikation jeweils 1000 Kundenstandorte erzeugt werden. Für die Testsets R110_1 sowie RC110_1 würde die

Dreiecksverteilung identisch parametrisiert. Weil eine identische Parametrisierung der Dreiecksverteilung identische Ergebnisse zur Folge hat, erfolgt die Generierung dreiecksverteilter Kundenstandorte nur im Testset C110_1. Äquivalent gelten die Ergebnisse dieser Generierung auch für die Testsets R110_1 und RC110_1. In Tabelle 19 sind ausgewählte Ergebniswerte zu finden, wobei die getroffene Auswahl keine Bewertung der Ergebnisse darstellt. Alle Kennzahlen und berechneten Werte der Simulationsläufe können in Tabelle 41 im Anhang gefunden werden.

Tabelle 19: Dreiecksverteilung Testset C110_1

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	454441,56	454569,25	335	0,765
1000	454652,35	454713,33	3635	7,407
10000	454653,88	454641,92	35928	67,063

Im ersten Simulationslauf werden für die 100 Replikationen insgesamt 0,765 Sekunden benötigt. Für eine Replikation werden mindestens 0 Sekunden und maximal 0,016 Sekunden benötigt, wobei der Median bei 0 Sekunden sowie der Mittelwert bei 0,005 Sekunden liegen. Somit nimmt die Generierung von 1000 Kundenstandorten wenig Zeit in Anspruch. Insgesamt werden in diesem Simulationslauf 335 doppelte Standorte erzeugt. In einer Replikation existieren mindestens 0 und maximal 8 doppelte Standorte. Als Mittelwert dieses Fehlers ergibt sich ein Wert von 3,35 Fehlern und als Median 3 Fehler. Deswegen ist der Maximalwert als Extremwert zu bezeichnen. Dies ist auch der Abbildung 42 zu entnehmen, weil der Mittelwert, der für Ausreißer anfällig ist, sichtbar kleiner als der Maximalwert ist. Für den Mittelwert der Tourenlänge ergibt sich ein Wert von 454441,56 LE und ein Median von 454569,25. Mit diesen Werten ist der Median größer als der Mittelwert, woran eine linksschiefe Verteilung der Tourenlänge zu erkennen ist. Der Abstand zwischen beiden Lagemaßen liegt bei 127,69 LE, was einer Abweichung von 0,028 Prozent, bezogen auf den Mittelwert, entspricht. Aus dieser geringen Abweichung kann geschlossen werden, dass die Linksschiefe schwach ausgeprägt ist und eine beinahe symmetrische Verteilung der Tourenlänge vorliegt.

Der zweite Simulationslauf mit 1000 Replikationen nimmt eine Gesamtgenerierungszeit von 7,407 Sekunden in Anspruch. Die minimale Generierungszeit einer Replikation

beträgt 0 Sekunden, der Maximalwert liegt bei 0,016 Sekunden. Als Median ergibt sich ein Wert von 0 Sekunden und als Mittelwert 0,005 Sekunden. Damit sind die letztgenannten vier Werte mit denen des ersten Simulationslaufs identisch, womit die Generierung von 1000 Standorten wenig Zeit in Anspruch nimmt. Innerhalb des zweiten Simulationslaufs werden 3635 doppelte Standorte erzeugt, wobei der minimale Wert dieses Fehlers für eine Replikation bei 0 liegt. Als Maximalwert ergeben sich in einer Replikation 16 Fehler, wobei der Median bei 4 Fehlern und der Mittelwert bei 3,64 Fehlern liegen. Aufgrund dieser Abweichung der Lagemaße vom Maximalwert, ist dieser als Extremwert zu bezeichnen. Diese Aussage wird von Abbildung 42 unterstützt, weil der Mittelwert sichtbar kleiner als der Maximalwert der Generierungsfehler ist. Als Mittelwert der Tourenlänge ergeben sich 454652,35 LE und der Median beträgt 454713,33 LE, womit der Median, wie im vorherigen Simulationslauf, größer als der Mittelwert ist und die linksschiefe Verteilung der Tourenlänge bestehen bleibt. Aufgrund des Abstands von 60,98 LE zwischen Mittelwert und Median, der, bezogen auf den Mittelwert, einer Abweichung von 0,0134 Prozent entspricht, ist die Linksschiefe als schwach ausgeprägt zu bezeichnen und es kann von einer beinahe symmetrischen Verteilung der Tourenlänge gesprochen werden.

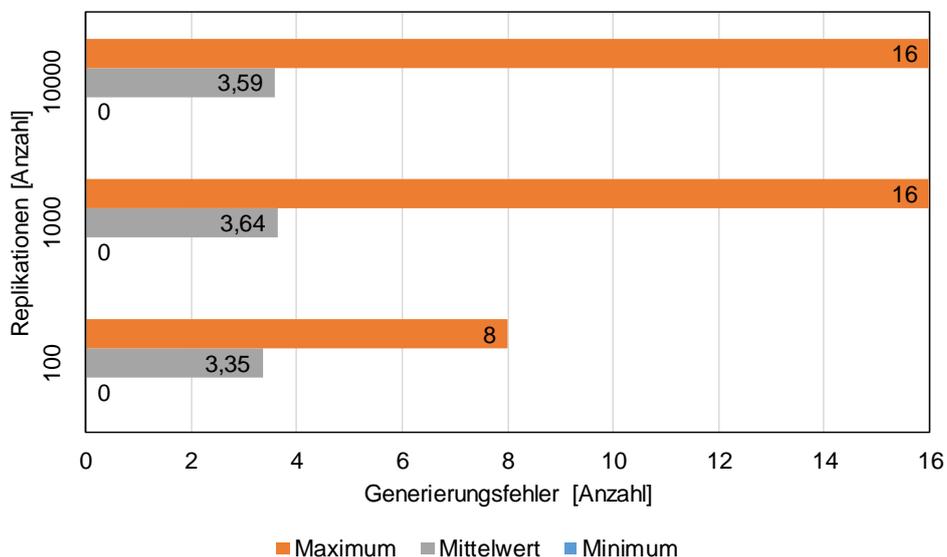


Abbildung 42: Dreiecksverteilung Generierungsfehler pro Replikation Testset C110_1

Für den dritten Simulationslauf wird in den 10000 Replikationen insgesamt eine Generierungszeit von 67,063 Sekunden benötigt. Die Werte für die minimale und maximale Generierungszeit sowie deren Median sind mit denen des vorherigen Simulationslaufs identisch. Abweichend zum vorherigen Simulationslauf ergibt sich ein

Mittelwert von 0,006 Sekunden. Weiterhin nimmt die Generierung von 10000 Standorten wenig Zeit in Anspruch. Innerhalb der 10000 Replikationen werden insgesamt 35928 doppelte Standorte erzeugt. Für eine Replikation ergibt sich für diesen Fehler ein minimaler Wert von 0 Fehlern und ein maximaler Wert von 16 Fehlern. Aufgrund des Medians von 3 Fehlern und des Mittelwerts von 3,59 Fehlern handelt es sich bei dem maximalen Wert um einen Extremwert. Die Abbildung 42 unterstützt diese Aussage, weil der Mittelwert sichtbar kleiner als der Maximalwert ist. Für die Tourenlänge ergibt sich ein Mittelwert von 454653,88 LE und ein Median von 454641,92 LE, womit der Median kleiner als der Mittelwert ist. Die linksschiefe Verteilung der Tourenlänge aus den vorherigen Simulationsläufen wechselt zu einer rechtsschiefen Verteilung. Da der Abstand zwischen diesen beiden Lagemaßen 11,96 LE beträgt, was einer Abweichung von 0,0026 Prozent, bezogen auf den Mittelwert, entspricht, ist die Rechtsschiefe als schwach ausgeprägt und die Verteilung der Tourenlänge als beinahe symmetrisch zu bezeichnen. Die Betrachtung aller Simulationsläufe führt zu der Beobachtung, dass sich der Median und der Mittelwert der Tourenlänge des zugehörigen Simulationslaufs für eine steigende Anzahl Replikationen nicht signifikant voneinander unterscheiden. Zudem gilt dies auch für den Median und den Mittelwert der Generierungszeit sowie für den Median und Mittelwert der Generierungsfehler. Bei einer Verhundertfachung der Replikationen verändern sich die Werte der genannten Lagemaße nicht signifikant. Außerdem führt eine Verzehnfachung der Anzahl der Replikationen sowohl zu einer Verzehnfachung der Generierungszeit als auch der Generierungsfehler.

In Abbildung 43 sind die Werte der Tourenlänge für jeden Simulationslauf jeweils in ein Box-Plot-Diagramm eingetragen. Innerhalb der Diagramme wird der Mittelwert mittels \bar{x} gekennzeichnet und der Median durch eine horizontale Linie in der Box markiert. Die linksschiefe Verteilung der Tourenlänge der ersten beiden Simulationsläufe ist in den Diagrammen daran zu erkennen, dass sich der Mittelwert unterhalb des Medians befindet. Analog dazu ist die rechtsschiefe Verteilung des dritten Simulationslaufs an der Lage des Mittelwerts oberhalb des Medians erkennbar. Die Größe der Box, in welcher sich 50 Prozent der Werte der Tourenlänge befinden, ist im ersten Simulationslauf sichtbar kleiner als im zweiten und dritten Simulationslauf. Daher ergeben sich für den Interquartilsabstand des ersten Simulationslaufs 4337,99 LE, im zweiten Simulationslaufs erhöht sich dieser auf 5305,67 LE und sinkt für den Dritten auf 5230,38 LE.

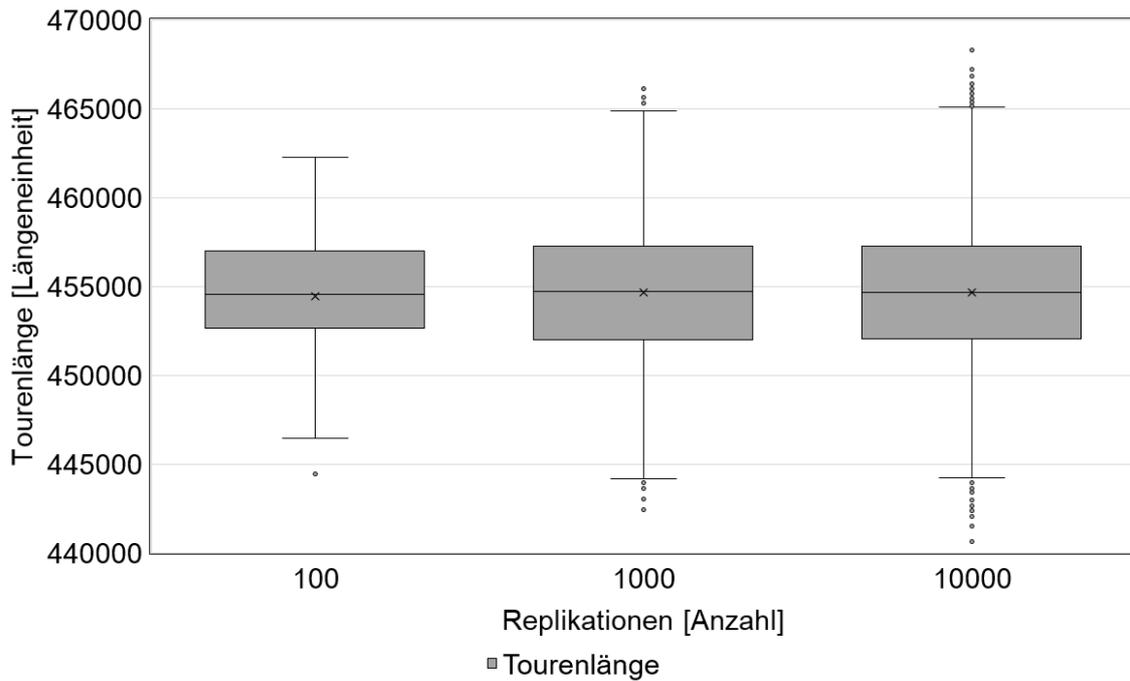


Abbildung 43: Dreiecksverteilung Box-Plots Testset C110_1

Zwischen dem Interquartilsabstand des ersten und zweiten Simulationslaufs liegen 967,68 LE. Bezogen auf den Interquartilsabstand des zweiten Simulationslaufs entspricht dies einer Abweichung von 18,2 Prozent. Somit wird die Streubreite, in denen die mittleren 50 Prozent der Werte der Tourenlänge liegen, für die Replikationserhöhung vom ersten auf den zweiten Simulationslauf signifikant größer. Eine weitere Erhöhung der Replikationen, vom zweiten auf den dritten Simulationslauf, ergibt für die Interquartilsabstände einen Abstand von 75,29 LE, was einer Abweichung von 1,44 Prozent, bezogen auf den Interquartilsabstand des dritten Simulationslaufs, entspricht. Die untere Grenze der Box des ersten Simulationslaufs befindet sich bei 452625,19 LE. Im zweiten Simulationslauf liegt diese Grenze bei 451979,5 LE sowie für den dritten Simulationslauf bei 452030,4 LE. Somit beginnt die Box des ersten Simulationslaufs bei einem höheren Wert als die Boxen des zweiten und dritten Simulationslaufs, welche bei einem ähnlichen Wert beginnen. Die Länge der Whisker für die drei Simulationsläufe unterscheiden sich in Abbildung 37 erkennbar voneinander. Im ersten Simulationslauf werden die Whisker durch 446429,575 LE und 462285,34 LE begrenzt. Für den zweiten Simulationslauf ergeben sich Grenzen von 444157,15 LE und 464899,08 LE sowie von 444222,1 LE und 465074,76 LE für den dritten Simulationslauf, womit sich die Grenzen der Whisker des zweiten und dritten Simulationslauf ähneln. Ein weiterer Unterschied zwischen den Box-Plot-Diagrammen ist die Anzahl der Ausreißer pro Simulationslauf. Im ersten Simulationslauf ergibt sich 1 unterer Ausreißer, wogegen im zweiten

Simulationslauf 4 untere und 3 obere Ausreißer erzeugt werden. Für den dritten Simulationslauf ergeben sich 34 untere sowie 43 obere Ausreißer. Eine Erhöhung der Replikationen führt somit zu einer Erhöhung der Ausreißer, die auch als Extremwerte bezeichnet werden können. Obwohl signifikante Unterschiede sowohl in der Streubreite der Tourenlänge für 50 Prozent der Werte als auch der Whisker existieren und mit einer Erhöhung der Replikationen auch die Anzahl der Extremwerte ansteigt, verändert sich der Mittelwert, der für Ausreißer anfällig ist, kaum. Im ersten Simulationslauf beträgt dieser 454441,56 LE, im zweiten Simulationslauf 454652,35 LE sowie 454653,88 LE für den dritten Simulationslauf. Zwischen dem ersten und dritten Simulationslauf unterscheiden sich die Mittelwerte um 212,32 LE, was einem Unterschied von 0,0467 Prozent, bezogen auf den Mittelwert des dritten Simulationslaufs, entspricht.

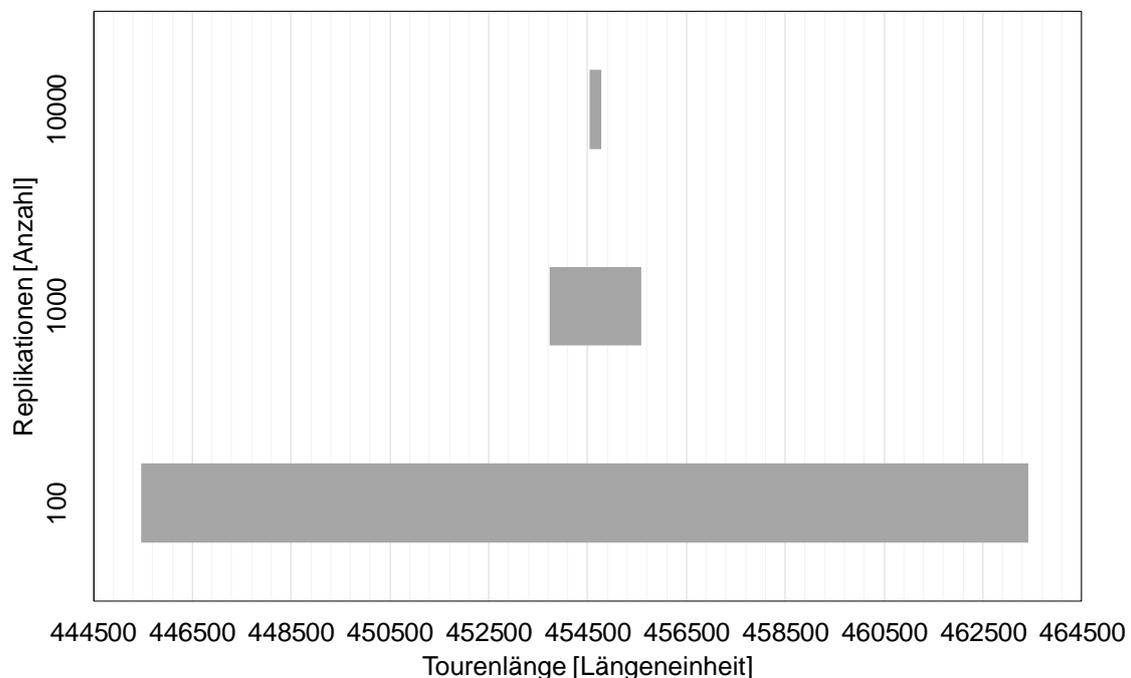


Abbildung 44: Dreiecksverteilung 95% Konfidenzintervalle Testset C110_1

Für die drei Simulationsläufe sind die approximierten 95% Konfidenzintervalle in Abbildung 44 zu sehen. Aus der Abbildung kann entnommen werden, dass die Breite der Konfidenzintervalle bei steigender Anzahl der Replikationen sinkt. Im ersten Simulationslauf beträgt die Breite des Konfidenzintervalls 17970,36 LE. Diese sinkt im zweiten Simulationslauf auf 1850,68 LE und erreicht im dritten Simulationslauf mit 234,46 LE ihr Minimum. Die Breite des Konfidenzintervalls nähert sich dem Mittelwert des jeweiligen Simulationslaufs an, je größer die Anzahl der Replikationen ist. Mit dem approximierten 95% Konfidenzintervall des dritten Simulationslaufs mit 10000

Replikationen kann ausgesagt werden, dass das Konfidenzintervall der Dimension (454536,65; 454771,11) den wahren Wert der Tourenlänge mit einer Fehlerwahrscheinlichkeit von 5 Prozent überdeckt. Aus der Auswertung der drei Simulationsläufe ist, auch bei einer Erhöhung auf 10000 Replikationen, keine signifikante Veränderung der Lagemaße für die Tourenlänge, die Generierungsfehler und der Generierungszeit erkennbar. Aus diesem Grund können für den Vergleich der Generierungsverfahren im Abschnitt 7.1 die Ergebnisse aus jedem Simulationslauf verwendet werden. Bei 10000 Replikationen stellt das approximierete 95% Konfidenzintervall jedoch ein Minimum dar, aufgrund dessen die präziseste Aussage zur Größenordnung der Tourenlänge möglich ist. Deswegen fließen die Ergebnisse dieses Simulationslaufs, als geeignetste Vertreter der drei Simulationsläufe, in den Vergleich ein.

6.3.4 Dreiecksverteilung im Testset Antwerp1

Als letzte Simulation dieses Abschnitts werden die Kundenstandorte mit einer Dreiecksverteilung im Testset Antwerp1 erzeugt und ausgewertet. Für die begrenzenden Koordinaten der x- und y-Koordinaten ergeben sich aus dem Testset 2000 LE (vgl. Tabelle 5). Aus diesem Grund werden sowohl die x- als auch die y-Koordinate im Intervall (0; 2000) erzeugt. Die wahrscheinlichsten Werte der x- und y-Koordinate betragen jeweils 1000 LE. Da das Testset 6001 Standorte enthält, werden in jeder Replikation 6001 Kundenstandorte erzeugt. Eine Auswahl an Ergebnissen ist in Tabelle 20 zu sehen. Diese Auswahl stellt keine Bewertung der dargestellten Ergebnisse dar. Im Anhang in Tabelle 42 sind alle Kennzahlen und berechneten Werte der Simulationsläufe aufgeführt.

Tabelle 20: Dreiecksverteilung Testset Antwerp1

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	10910983,00	10913771,64	844	3,390
1000	10912407,17	10912503,06	8030	34,906
10000	10912764,99	10912207,09	80164	333,703

Im ersten Simulationslauf mit 100 Replikationen wird eine Gesamtgenerierungszeit von 3,39 Sekunden benötigt, wobei eine Replikation in mindestens 0,015 Sekunden und maximal 0,047 Sekunden generiert wird. Der Median liegt bei 0,031 Sekunden und der Mittelwert bei 0,03 Sekunden. Die Erhöhung der Generierungszeit, im Vergleich zu den anderen Testsets, ist auf die höhere Anzahl der zu generierenden Kundenstandorte zurückzuführen. Über alle 100 Replikationen werden 844 doppelte Standorte erzeugt. Innerhalb einer Replikation sind mindestens 2 und maximal 15 dieser Fehler vorhanden. Für den Mittelwert ergeben sich 8,44 Fehler und als Median 8 Fehler. Anhand dessen sind sowohl der Minimal- als auch der Maximalwert der doppelten Standorte als Extremwert zu bezeichnen. In Abbildung 45 ist dieser Zusammenhang ebenso zu beobachten, da der Minimalwert sichtbar kleiner beziehungsweise der Maximalwert sichtbar größer als der Mittelwert ist, obwohl der Mittelwert für Ausreißer anfällig ist. Für den Mittelwert der Tourenlänge ergibt sich ein Wert von 10910983 LE und für den Median einer von 10913771,64 LE, womit der Median 2788,64 LE größer als der Mittelwert ist. Deswegen ist die Verteilung der Tourenlänge als linksschief zu bezeichnen. Aufgrund des Abstands beider Lagemaße von 2788,64 LE, was einer Abweichung von 0,0256 Prozent, bezogen auf den Mittelwert, entspricht, ist die Linksschiefe schwach ausgeprägt und die Verteilung der Tourenlänge kann als beinahe symmetrisch bezeichnet werden.

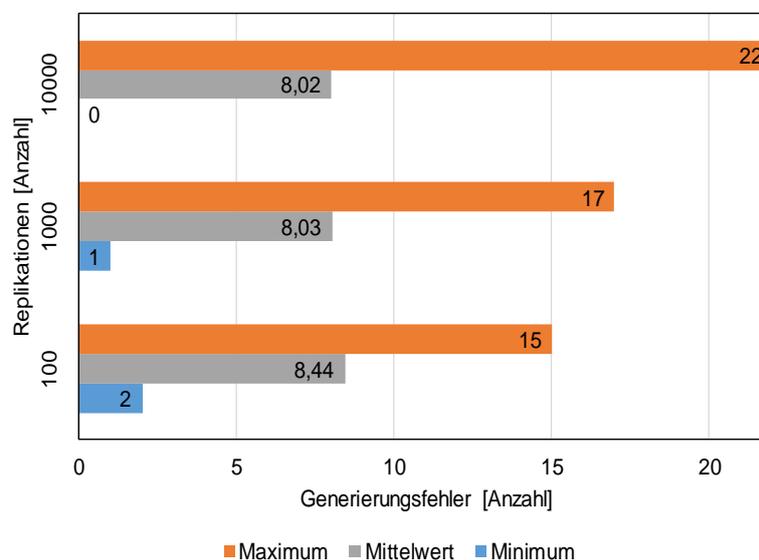


Abbildung 45: Dreiecksverteilung Generierungsfehler pro Replikation Testset Antwerp1

Für den zweiten Simulationslauf mit 1000 Replikationen wird eine Gesamtgenerierungszeit von 34,906 Sekunden benötigt. Eine Replikation wird in mindestens 0,015 Sekunden und maximal 0,047 Sekunden generiert. Der Median beträgt

0,031 Sekunden. Die Werte für die Mindest- und Maximalgenerierungszeit einer Replikation sowie der Median sind identisch mit denen aus dem ersten Simulationslauf. Der Mittelwert steigt hingegen auf 0,032 Sekunden an. Insgesamt werden über alle Replikationen 8030 doppelte Standorte erzeugt, wobei in einer Replikation mindestens 1 und maximal 17 doppelte Standorte generiert werden. Weil der Median dieses Fehlers bei 8 Fehlern und der Mittelwert bei 8,03 Fehlern liegen, handelt es sich bei den Minimal- und Maximalwerten um Extremwerte. Abbildung 45 unterstützt diese Aussage, denn in ihr ist der Mittelwert, der für Ausreißer anfällig ist, sichtbar größer als der Minimalwert und sichtbar kleiner als der Maximalwert. Als Mittelwert der Tourenlänge ergeben sich 10912407,17 LE und der Median beträgt 10912503,06 LE. Mit diesen Werten ist der Median weiterhin 95,89 LE größer als der Mittelwert und die Linksschiefe der Verteilung der Tourenlänge bleibt bestehen. Aufgrund des geringen Abstands beider Lagemaße und der daraus resultierenden Abweichung von 0,0009 Prozent, bezogen auf den Mittelwert, ist von einer schwachen Ausprägung der Linksschiefe zu sprechen beziehungsweise kann die Verteilung der Tourenlänge auch als beinahe symmetrisch bezeichnet werden.

Die 10000 Replikationen im dritten Simulationslauf benötigen eine Gesamtgenerierungszeit von 333,703 Sekunden. Für eine Replikation wird mindestens eine Generierungszeit von 0,015 Sekunden und maximal eine von 0,063 Sekunden benötigt. Sowohl für den Median als auch den Mittelwert ergibt sich ein Wert von 0,031 Sekunden. Im Simulationslauf werden insgesamt 80164 doppelte Standorte erzeugt. In einer Replikation treten mindestens 0 und maximal 22 dieser Fehler auf. Aufgrund des Medians von 8 Fehlern und des Mittelwerts von 8,02 sind sowohl der minimale- als auch der maximale Wert als Extremwert zu bezeichnen. Für den Mittelwert ergeben sich 10912764,99 LE sowie für den Median 10912207,09 LE, wodurch der Mittelwert größer als der Median ist und die Linksschiefe aus den vorherigen Simulationsläufen in eine rechtsschiefe Verteilung der Tourenlänge umschwenkt. Weil der Abstand zwischen beiden Lagemaßen 557,9 LE ergibt und dies, bezogen auf den Mittelwert, einer Abweichung von 0,0051 Prozent entspricht, ist die Rechtsschiefe als schwach ausgeprägt, beziehungsweise die Verteilung der Tourenlänge als beinahe symmetrisch zu bezeichnen. Eine gemeinsame Betrachtung aller Simulationsläufe ergibt, dass sich die zugehörigen Mittelwerte und Mediane der Tourenlänge über eine steigende Anzahl Replikationen nicht signifikant voneinander unterscheiden. Für die Mittelwerte und Mediane der Generierungszeit sowie der Generierungsfehler ist entsprechendes zu beobachten. Somit verändern sich auch bei einer Verhundertfachung der Replikationen, vom ersten auf den

dritten Simulationslauf, die Werte der genannten Lagemaße nicht signifikant. Eine Verzehnfachung der Replikationen führt zudem zu einer Verzehnfachung der Gesamtgenerierungsfehler und Gesamtgenerierungszeit.

Die Werte der Tourenlänge sind für den zugehörigen Simulationslauf in drei Box-Plot-Diagramme eingetragen worden, diese sind in Abbildung 46 zu sehen. Innerhalb der Diagramme wird der Mittelwert mit x und der Median mit einer horizontalen Linie in der Box gekennzeichnet. Die linksschiefe Verteilung der Tourenlänge in den ersten beiden Simulationsläufen kann in der Abbildung daran erkannt werden, dass sich die Markierung des Mittelwerts unterhalb des Medians befindet. Bei der rechtsschiefen Verteilung des dritten Simulationslaufs ist dies umgekehrt. Im ersten Simulationslauf ist die Box, in der sich 50 Prozent der Werte der Tourenlänge befinden, sichtbar kleiner als im zweiten und dritten Simulationslauf. Dies kann auch an den Werten des Interquartilsabstandes abgelesen werden. Für den ersten Simulationslauf beträgt dieser 43010,77 LE. Im zweiten Simulationslauf steigt der Interquartilsabstand auf 50168,58 LE und erreicht im Dritten mit 51190,56 LE sein Maximum. Zwischen dem Interquartilsabstand des ersten und zweiten Simulationslauf liegen 7157,81 LE, was bezogen auf den Interquartilsabstand des zweiten Simulationslaufs einer Abweichung von 14,26 Prozent entspricht. Der Unterschied zwischen dem Interquartilsabstand des zweiten und dritten Simulationslaufs beträgt 1021,98 LE. Dies entspricht, bezogen auf den Interquartilsabstand des dritten Simulationslauf, 1,99 Prozent. Die Box des ersten Simulationslaufs startet bei der unteren Grenze von 10891344,6 LE. Für den zweiten Simulationslauf liegt dieser Wert bei 10887753,44 LE und für den Dritten bei 10887003,97 LE. Damit beginnen die Boxen des zweiten sowie dritten Simulationslaufs bei ähnlichen Werten und die Box des ersten Simulationslauf bei einer höheren Grenze. Einen weiteren Unterschied stellt die Länge der Whisker der drei Simulationsläufe dar. Im ersten Simulationslauf beginnt der untere Whisker bei 10837991,95 LE und der obere endet bei 10989970,2 LE. Diese Werte liegen für den zweiten Simulationslauf bei 10815070,41 LE und 11005590,3 LE sowie für den dritten Simulationslauf bei 10810405,37 LE und 11014153,86 LE. Damit ähneln sich die Grenzen der Whisker des zweiten und dritten Simulationslaufs. Des Weiteren unterscheiden sich die drei Simulationsläufe in der Anzahl ihrer Ausreißer. Im ersten Simulationslauf werden 1 oberer und 1 unterer Ausreißer erzeugt. Für den zweiten Simulationslauf steigen diese Werte auf 2 untere und 3 obere Ausreißer und im dritten Simulationslauf werden 40 untere sowie 43 obere Ausreißer erzeugt. Damit führt eine Erhöhung der Replikationen zu einer Erhöhung der Ausreißer. Sowohl in der Streubreite

der Boxen als auch der Whisker existieren signifikante Unterschiede und die Anzahl der Ausreißer nimmt mit steigender Anzahl der Replikationen zu. Dabei verändert sich der Mittelwert, der für Ausreißer anfällig ist, hingegen bei einem Anstieg der Replikationen nicht signifikant. Im ersten Simulationslauf beträgt dieser 10910983 LE, für den zweiten Simulationslauf steigt er auf 10912407,17 LE und erreicht mit 10912764,99 LE im dritten Simulationslauf sein Maximum, womit zwischen dem Mittelwert des ersten und dritten Simulationslauf ein Abstand von 1781,99 LE liegt. Dieser Abstand entspricht, bezogen auf den Mittelwert des dritten Simulationslaufs, einer Abweichung von 0,0163 Prozent.

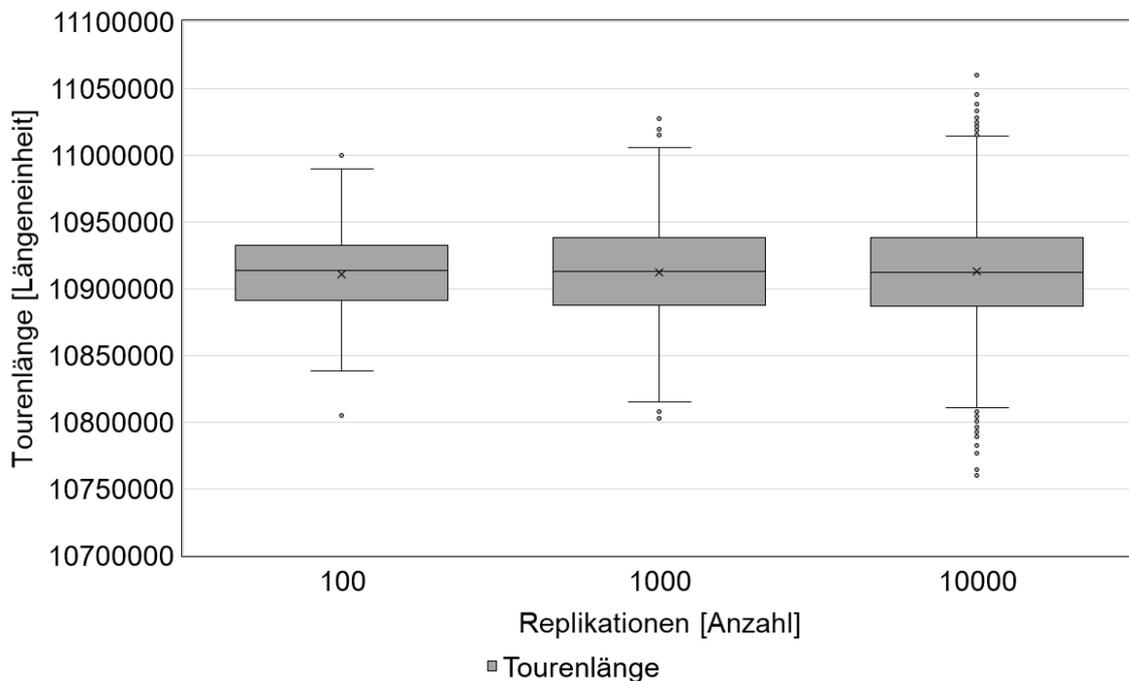


Abbildung 46: Dreiecksverteilung Box-Plots Testset Antwerp1

In Abbildung 47 sind die approximierten 95% Konfidenzintervalle der drei Simulationsläufe zu sehen. Aus der Abbildung wird ersichtlich, dass bei steigender Anzahl der Replikationen die Breite der Konfidenzintervalle sinkt. Diese beträgt im ersten Simulationslauf 430111,07 LE, sinkt im zweiten Simulationslauf auf 43059,44 LE und erreicht mit 4536,31 LE im dritten Simulationslauf ihr Minimum. Je größer die Anzahl der Replikationen ist, umso mehr nähert sich das Konfidenzintervall dem Mittelwert des jeweiligen Simulationslaufs an. Mit Hilfe des approximierten 95% Konfidenzintervall der Tourenlänge kann ausgesagt werden, dass der wahre Wert der Tourenlänge mit einer Wahrscheinlichkeit von 95 Prozent vom Konfidenzintervall der Dimension (10910496,84; 10915033,15) überdeckt wird. Anhand der Auswertung der drei Simulationsläufe lässt sich, auch bei einer Erhöhung auf 10000 Replikationen, keine

signifikante Veränderung der Mittelwerte sowie Mediane der Tourenlänge, der Generierungszeit und der Generierungsfehler ausmachen. Zum Vergleich der Generierungsverfahren in Abschnitt 7.1 können deswegen die Ergebnisse jedes Simulationslaufes herangezogen werden. Mit dem kleinsten approximierten 95% Konfidenzintervall, welches bei 10000 Replikationen ein Minimum erreicht, ist die präziseste Vorhersage zur Größenordnung der Tourenlänge möglich. Aus diesem Grund werden die Ergebnisse des Simulationslaufs mit 10000 Replikationen in den Vergleich in Abschnitt 7.1 einfließen.

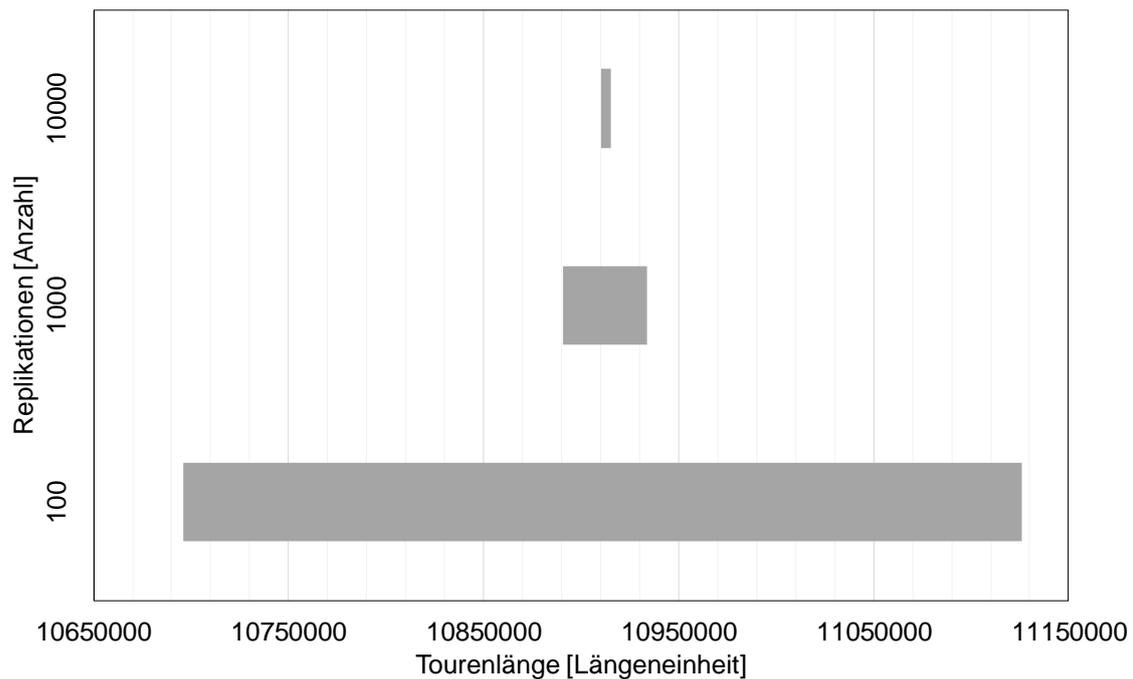


Abbildung 47: Dreiecksverteilung 95% Konfidenzintervalle Testset Antwerp1

6.4 Simulation unter Berücksichtigung der Bevölkerungsdichte

In diesem Abschnitt erfolgt die Generierung der Kundenstandorten unter Berücksichtigung der Bevölkerungsdichte im Stadtgebiet eines ausgewählten Testsets. Der Abschnitt ist dafür in drei Unterabschnitte, 6.4.1 bis 6.4.3, aufgeteilt. Die Generierung unter Berücksichtigung der Bevölkerungsdichte stellt dabei kein eigenes Verfahren zur Generierung von Kundenstandorten dar, denn es werden lediglich die Parameter der Generierungsverfahren an die Bevölkerungsdichte angepasst. Die Generierung der Kundenstandorte erfolgt mit der Gleich-, der Normal- und der

Dreiecksverteilung. Für die Simulation wird das Testset E-n22-k4 ausgewählt, da es aufgrund der enthaltenen 22 Kundenstandorte eine übersichtliche Vergleichbarkeit der Generierungsverfahren ermöglicht, denn die Generierungsverfahren müssten für die anderen Testsets (vgl. Tabelle 5) vermehrt angepasst werden. Zudem wird exemplarisch aufgezeigt, wie mit den Generierungsverfahren auf das Vorliegen von Daten, in diesem Fall die Bevölkerungsdichte, reagiert werden kann. Die Verteilung der Kundenstandorte des gewählten Testset ist im Anhang auf Seite 213 in Abbildung 72 zu sehen. Aus der Betrachtung dieser Abbildung wird erkennbar, dass sich die Kundenstandorte in der oberen rechten Ecke des Testsets befinden. Alle 22 Kundenstandorte befinden sich auf der x-Achse zwischen 128 und 165 sowie auf der y-Achse zwischen 182 und 264. Aus diesem Grund werden die x-Koordinaten der Kundenstandorte im Intervall (125; 165) und die y-Koordinaten im Intervall (180; 265) erzeugt. Um die Kennzahlen über eine ansteigende Anzahl Replikationen beobachten zu können, werden für jedes Generierungsverfahren drei Simulationsläufe mit 100, 1000 und 10000 Replikationen getätigt (vgl. Abschnitt 5.2). Sämtliche Ergebnisse der Simulationen sind im Anhang in Tabelle 43 zu finden.

6.4.1 Berücksichtigung der Bevölkerungsdichte Gleichverteilung

Die erste Generierung der Kundenstandorte in diesem Abschnitt erfolgt mit der Gleichverteilung. In der Simulation werden die x-Koordinaten im Intervall (125; 165) und die y-Koordinaten im Intervall (180; 265) generiert. Des Weiteren werden in jeder Replikation 22 Kundenstandorte erzeugt. In Tabelle 21 ist eine Auswahl der Ergebniswerte für die drei Simulationsläufe zu sehen. Diese Auswahl stellt keine Bewertung der abgebildeten Ergebniswerte dar, sondern ermöglicht lediglich eine erste Übersicht über die Ergebnisse. Die Ergebnisse sind vollständig im Anhang in Tabelle 43 zu finden.

Tabelle 21: Gleichverteilung Berücksichtigung Bevölkerungsdichte

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	7220,18	7228,23	6	0,219
1000	7219,20	7221,96	68	2,734
10000	7224,19	7223,62	672	16,578

Der erste Simulationslauf mit 100 Replikationen benötigt eine Gesamtgenerierungszeit von 0,219 Sekunden, wobei die minimale Generierungszeit einer Replikation bei 0 Sekunden und die Maximale bei 0,016 Sekunden liegt. Sowohl für den Median als auch den Mittelwert ergeben sich 0 Sekunden. Im ersten Simulationslauf werden insgesamt 19 doppelte Standorte erzeugt. Die minimale Anzahl dieses Fehlers für eine Replikation liegt bei 0 und die Maximale bei 1. Anhand des Mittelwerts von 0,06 und des Medians von 0 ist erkennbar, dass es sich bei dem Maximalwert um einen Extremwert handelt. Dies ist auch in Abbildung 48 erkennbar, da der Mittelwert sichtbar kleiner als der Maximalwert ist. Für den Mittelwert der Tourenlänge ergeben sich 7220,18 LE, während der Median 7228,23 LE beträgt. Damit ist der Mittelwert kleiner als der Median und die Verteilung der Tourenlänge ist als linksschief zu bezeichnen. Der Abstand beider Lagemaße beträgt 8,05 LE und entspricht damit, bezogen auf den Mittelwert, einer Abweichung von 0,1115 Prozent, wodurch die Linksschiefe als schwach ausgeprägt bezeichnet werden kann. Zusätzlich gilt aufgrund der geringen Abweichung, dass die Verteilung der Tourenlänge beinahe symmetrisch ist.

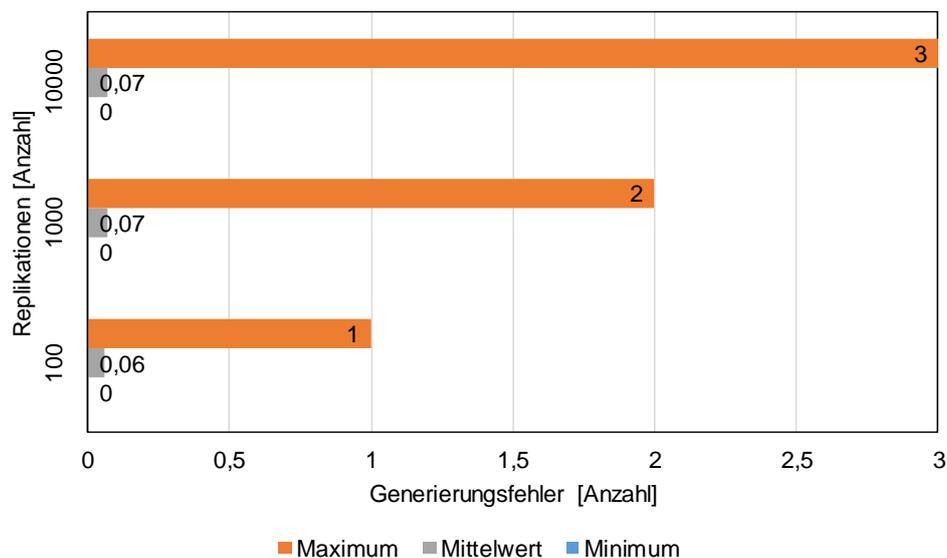


Abbildung 48: Gleichverteilung Generierungsfehler pro Replikation Berücksichtigung Bevölkerungsdichte

Im zweiten Simulationslauf steigt die Gesamtgenerierungszeit auf 2,734 Sekunden. Für eine Replikation wird eine minimale Generierungszeit von 0 Sekunden und eine Maximale von 0,016 Sekunden benötigt. Wie im vorherigen Simulationslauf, liegen der Median sowie der Mittelwert bei 0 Sekunden. In allen 1000 Replikationen werden insgesamt 68 doppelte Standorte erzeugt, davon in einer Replikation mindestens 0 und

maximal 2. Der Mittelwert beträgt 0,07 Fehler und der Median 0 Fehler. Aufgrund der geringen Größe der Lagemaße ist der maximale Fehlerwert als Extremwert zu bezeichnen. In Abbildung 48 ist dies dargestellt: Der Mittelwert ist sichtbar kleiner als der Maximalwert. Für den Mittelwert der Tourenlänge ergeben sich 7219,2 LE und der Median beträgt 7221,96 LE. Mit diesen Werten ist der Median weiterhin größer als der Mittelwert. Die Verteilung der Tourenlänge ist damit, wie im ersten Simulationslauf, linksschief. Als Abstand zwischen beiden Lagemaßen ergibt sich ein Wert von 2,76 LE, woraus eine Abweichung von 0,0382 Prozent, bezogen auf den Mittelwert, resultiert. Aufgrund dieser Abweichung ist die Linksschiefe schwach ausgeprägt und die Verteilung der Tourenlänge kann auch als beinahe symmetrisch bezeichnet werden.

Für den dritten Simulationslauf mit 10000 Replikationen wird eine Gesamtgenerierungszeit von 16,578 Sekunden benötigt. Die Generierung einer Replikation benötigt mindestens 0 Sekunden und maximal 0,016 Sekunden. Die Generierungszeit beträgt, wie in den vorherigen Simulationsläufen, im Mittelwert sowie Median 0 Sekunden. Im dritten Simulationslauf werden insgesamt 672 doppelte Standorte generiert. In einer Replikation werden mindestens 0 und maximal 3 dieser Fehler erzeugt. Weil der Median 0 Fehler und der Mittelwert 0,07 Fehler beträgt, ist der Maximalwert als Ausreißer zu bezeichnen. Abbildung 48 ermöglicht dies grafisch nachzuvollziehen, denn der Mittelwert, der für Ausreißer anfällig ist, ist sichtbar kleiner als der Maximalwert. Für den Mittelwert der Tourenlänge ergeben sich in diesem Simulationslauf 7224,19 LE und für den Median 7223,62 LE. Damit ist der Mittelwert, im Gegensatz zu den vorherigen Simulationsläufen, größer als der Median. Die linksschiefe wechselt in eine rechtsschiefe Verteilung der Tourenlänge. Zwischen beiden Lagemaßen liegt ein Abstand von 0,57 LE, was bezogen auf den Mittelwert einer Abweichung von 0,0078 Prozent entspricht. Aufgrund dieser Abweichung ist die rechtsschiefe Verteilung schwach ausgeprägt und die Verteilung der Tourenlänge kann als beinahe symmetrisch bezeichnet werden. Durch eine Betrachtung aller Simulationsläufe ist erkennbar, dass sich die Mittelwerte und Mediane der Tourenlänge auch bei einer Erhöhung der Anzahl der Replikationen nicht signifikant voneinander unterscheiden. Dieses gilt auch für die Mittelwerte und Mediane der Generierungsfehler sowie der Generierungszeit. Vom ersten auf den dritten Simulationslauf verändern sich die genannten Lagemaße auch bei einer entsprechenden Verhundertfachung der Anzahl der Replikationen nicht signifikant. Außerdem führt eine Verzehnfachung der Replikationen zu einer Verzehnfachung der Gesamtgenerierungsfehler.

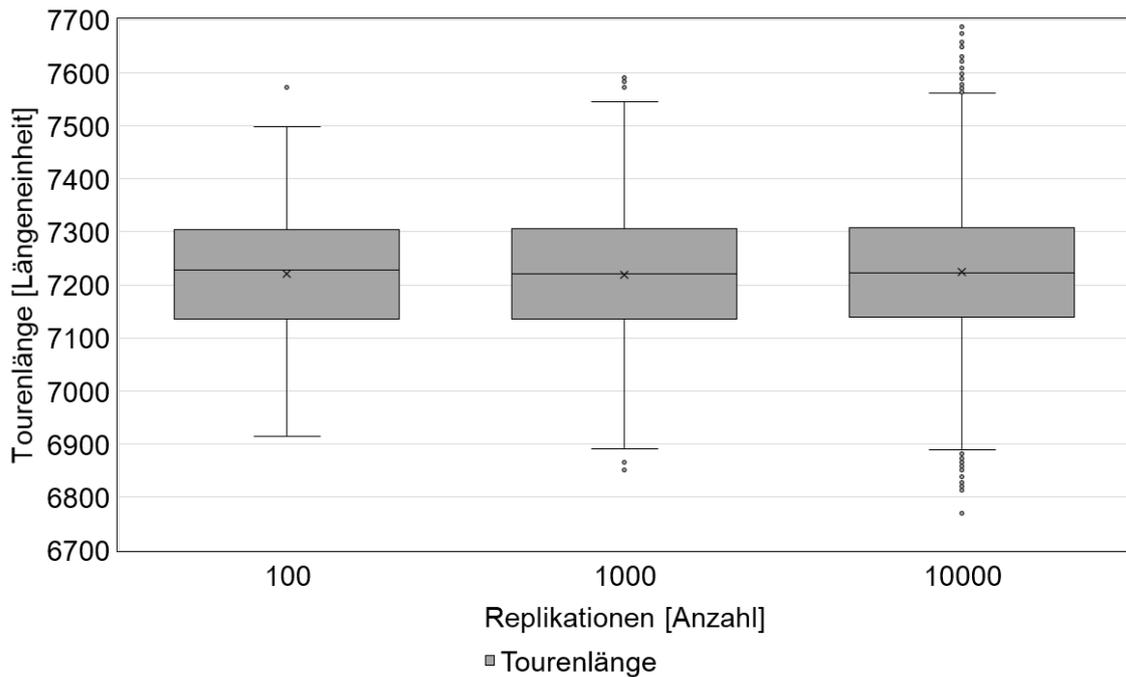


Abbildung 49: Gleichverteilung Box-Plots Berücksichtigung Bevölkerungsdichte

Für die Tourenlänge wurden alle Werte eines Simulationslaufs in ein Box-Plot-Diagramm eingetragen, welche in Abbildung 49 zu sehen sind. In den Box-Plot-Diagrammen wird der Mittelwert mit x und der Median mit einer horizontalen Linie innerhalb der Box gekennzeichnet. Die Abbildung zeigt die linksschiefe Verteilung der Tourenlänge in den ersten beiden Simulationsläufen, was daran erkennbar ist, dass der Mittelwert unterhalb des Medians liegt. Auch der Übergang in eine rechtsschiefe Verteilung im dritten Simulationslauf ist zu erkennen, weil sich der Mittelwert oberhalb des Medians befindet. Anhand der Abbildung ist außerdem ersichtlich, dass die Boxen, in denen sich die Mittleren 50 Prozent der Werte der Tourenlänge befinden, ähnlich groß sind. Die Werte für den Interquartilsabstand bestätigen diese Beobachtung, denn für den ersten Simulationslauf beträgt der Interquartilsabstand 167,87 LE, im Zweiten 170,94 LE und im Dritten 169,16 LE. Auch die unteren Grenzen der Boxen sind gleich. Sie betragen 7135,7 LE im ersten Simulationslauf, 7135,11 LE im Zweiten sowie 7139,63 LE im Dritten. Ein, in Abbildung 49 sichtbarer, Unterschied ist die Länge der Whisker der einzelnen Simulationsläufe. Im ersten Simulationslauf beginnt der untere Whisker bei 6914,5 LE und der obere Whisker endet bei 7499 LE. Für den zweiten Simulationslauf ergeben sich 6892,02 LE und 7546,28 LE sowie im dritten Simulationslauf 6888,58 LE und 7562,21 LE. Mit diesen Werten ähneln sich die Grenzen der Whisker im zweiten und dritten Simulationslauf. Zum ersten Simulationslauf hingegen weisen diese Abweichungen auf. Außerdem unterscheiden sich die Simulationsläufe hinsichtlich der

Anzahl ihrer Ausreißer. Für den ersten Simulationslauf ergibt sich 1 Ausreißer, im Zweiten existieren 2 untere und 3 obere Ausreißer sowie im Dritten 33 untere und 34 obere Ausreißer. Die Anzahl der Ausreißer nimmt damit bei einer Erhöhung der Replikationen zu. Außerdem werden durch eine Erhöhung der Replikationen die Boxen der einzelnen Simulationsläufe vergrößert, wodurch sich der Wertebereich erhöht, in dem 50 Prozent der Werte der Tourenlänge zu finden sind. Analoges gilt auch für die untere und obere Grenze der Whisker, sodass sich auch dieser Wertebereich mit steigender Anzahl der Replikationen vergrößert. Trotz dieser Beobachtungen verändert sich der Mittelwert, der für Ausreißer anfällig ist, über alle Simulationsläufe kaum. Für den ersten Simulationslauf beträgt der Mittelwert 7220,18 LE, steigt für den Zweiten auf 7221,96 LE und erreicht im Dritten mit 7224,19 LE sein Maximum. Der Abstand zwischen dem Mittelwert des ersten und dritten Simulationslaufs beträgt damit 4,01 LE, bezogen auf den Mittelwert des dritten Simulationslaufs, entspricht dies einer Abweichung von 0,0056 Prozent und ist damit nicht signifikant.

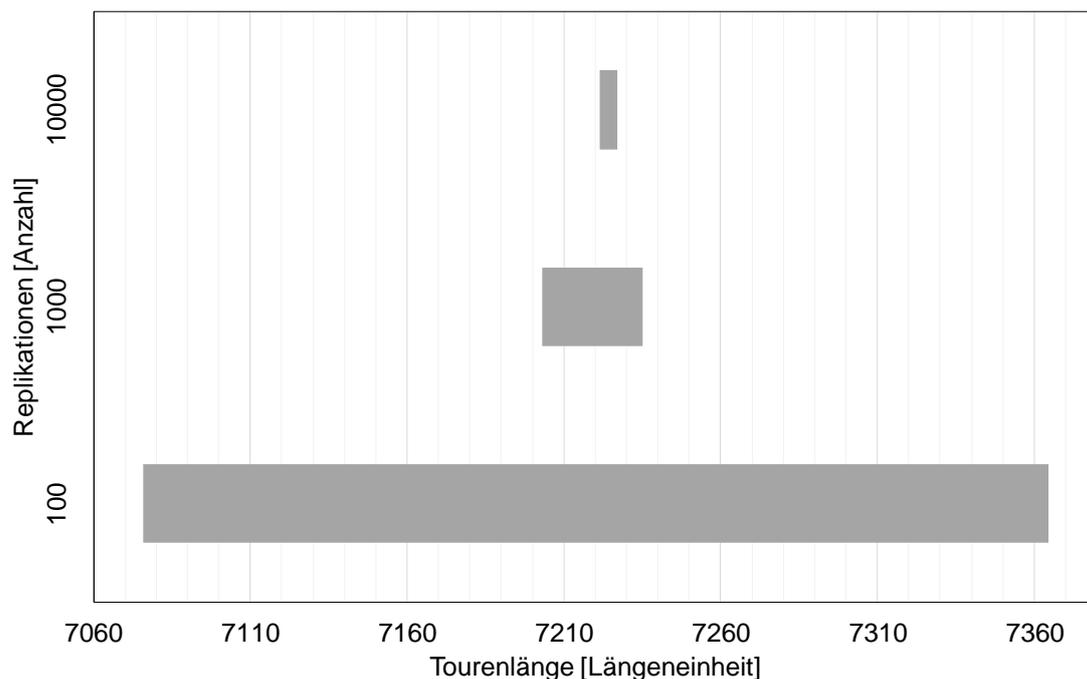


Abbildung 50: Gleichverteilung 95% Konfidenzintervall Berücksichtigung Bevölkerungsdichte

Die approximierten 95% Konfidenzintervalle für die drei Simulationsläufe sind in Abbildung 50 zu sehen. Die Betrachtung der Abbildung zeigt, dass die Breite der Konfidenzintervall bei steigender Anzahl der Replikationen sinkt. Für den ersten Simulationslauf beträgt die Breite des Konfidenzintervalls 288,75 LE, im Zweiten sinkt

diese auf 32,16 LE und erreicht im Dritten mit 5,66 LE ihr Minimum. Außerdem nähert sich das Konfidenzintervall mit steigender Anzahl der Replikationen dem Mittelwert des Simulationslaufs an. Mit dem approximierten 95% Konfidenzintervalls des dritten Simulationslaufs kann ausgesagt werden, dass der wahre Wert der Tourenlänge mit einer Fehlerwahrscheinlichkeit von 5 Prozent vom Konfidenzintervall der Dimension (7221,36; 7227,02) überdeckt wird. Die Auswertung der drei Simulationsläufe zeigt, dass auch bei einer Erhöhung der Replikationen auf 10000 keine signifikante Veränderung der Mittelwerte sowie Mediane der Tourenlänge, der Generierungszeit und der Generierungsfehler stattfindet. Aus diesem Grund können die Werte jedes Simulationslaufes zum Vergleich der Generierungsverfahren in Abschnitt 7.1 herangezogen werden. Die präziseste Vorhersage zur Größenordnung der Tourenlänge ist mit dem kleinsten approximierten 95% Konfidenzintervall möglich, welches bei 10000 Replikationen der Fall ist, weil dieses das Minimum darstellt. Deswegen wird der Simulationslauf mit 10000 Replikationen als geeignetster Vertreter dieser Simulationsläufe ausgewählt, um in den Vergleich in Abschnitt 7.1 einzufließen.

6.4.2 Berücksichtigung der Bevölkerungsdichte Normalverteilung

Die zweite Generierung der Kundenstandorte in diesem Abschnitt erfolgt mittels Normalverteilung. Alle x-Koordinaten, die außerhalb des Intervalls (125; 165), sowie alle y-Koordinaten, die außerhalb des Intervalls (180; 265) liegen, werden verworfen und als Fehler aufgenommen. Im Anschluss erfolgt die Generierung neuer Koordinaten für den betroffenen Standort. Um die Koordinaten auf dem eingegrenzten Stadtgebiet zu erzeugen, liegen für die x-Koordinaten die Varianz bei 10 LE und der Erwartungswert bei 145 LE. Für die y-Koordinaten ergeben sich für die Varianz 20 LE und 222,5 LE für den Erwartungswert. Zudem werden in jeder Replikation 22 Kundenstandorte erzeugt. In Tabelle 22 ist eine Auswahl an Ergebniswerten zu finden. Diese Auswahl stellt keine Bewertung der dargestellten Ergebniswerte dar. Alle Ergebniswerte können im Anhang in Tabelle 43 eingesehen werden.

Tabelle 22: Normalverteilung Berücksichtigung Bevölkerungsdichte

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	7216,01	7217,63	907	0,063
1000	7218,93	7219,21	8950	0,656
10000	7218,04	7217,61	87924	6,015

Im ersten Simulationslauf wird für die 100 Replikationen eine Generierungszeit von 0,063 Sekunden benötigt. Die minimale Generierungszeit einer Replikation liegt bei 0 Sekunden und die Maximale bei 0,016 Sekunden. Sowohl der Median als auch der Mittelwert ergeben 0 Sekunden. Insgesamt werden bei der Durchführung der 100 Replikationen 907 Fehler erzeugt. Auf die doppelten Standorte entfallen 13 Fehler, wobei in einer Replikation mindestens 0 und maximal 2 dieser Fehler entstehen. Aufgrund des Medians von 0 Fehlern und dem Mittelwert von 0,13 Fehlern ist der Maximalwert als Ausreißer zu bezeichnen. In diesem Simulationslauf werden 216 Standorte außerhalb der Stadtgrenzen erzeugt, mindestens 0 und maximal 7 in einer Replikation. Anhand des Medians von 2 Fehlern und des Mittelwerts von 2,16 Fehlern ist der Maximalwert als Ausreißer zu bezeichnen. Bei der Generierung der Standorte treten 678 Berechnungsfehler auf, mindestens 1 und maximal 17 pro Replikation. Der Maximalwert der Berechnungsfehler ist als Ausreißer zu bezeichnen, da der Median bei 6 Fehlern und der Mittelwert bei 6,78 Fehlern liegen. Die besprochenen Minimal- und Maximalwerte der Fehler können auch in Abbildung 51 eingesehen werden. Für die Tourenlänge ergeben sich in diesem Simulationslauf ein Mittelwert von 7216,01 LE sowie ein Median von 7217,63 LE. Mit diesen Werten ist der Median größer als der Mittelwert, womit die Verteilung der Tourenlänge linksschief ist. Der Abstand beider Lagemaße zueinander beträgt 1,62 LE und entspricht somit, bezogen auf den Mittelwert, einer Abweichung von 0,0225 Prozent. Aufgrund dieser Abweichung ist die Linksschiefe schwach ausgeprägt und die Verteilung der Tourenlänge beinahe symmetrisch.

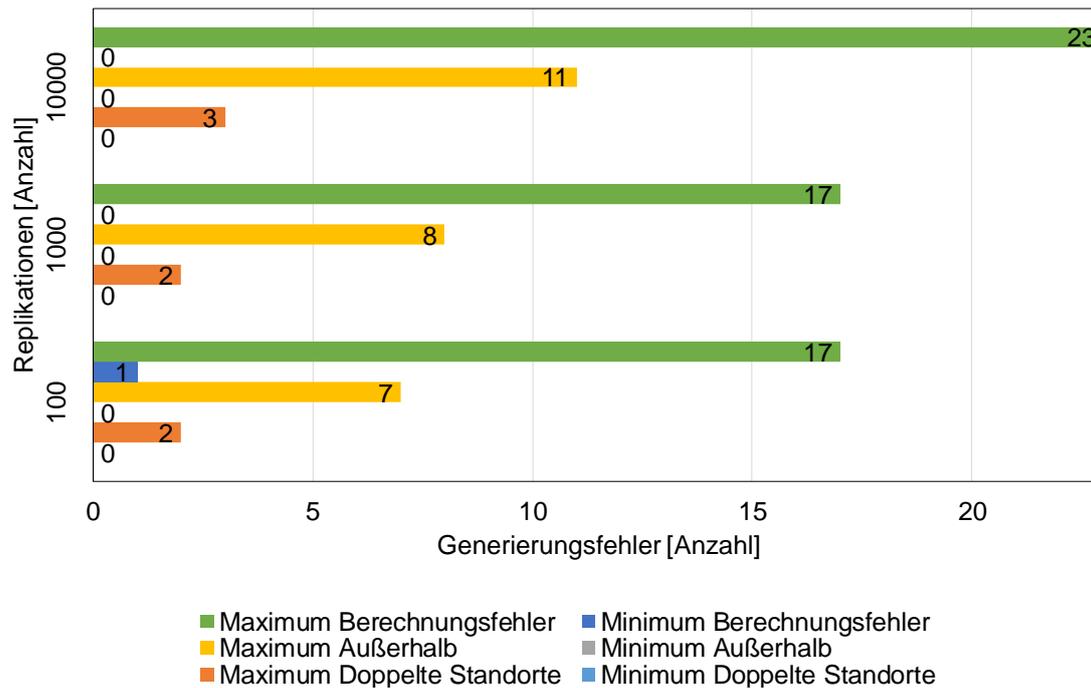


Abbildung 51: Normalverteilung Generierungsfehler pro Replikation Berücksichtigung Bevölkerungsdichte

Der zweite Simulationslauf mit 1000 Replikationen nimmt eine Gesamtgenerierungszeit von 0,656 Sekunden in Anspruch. Für eine Replikation wird eine minimale Generierungszeit von 0 und eine Maximale von 0,016 Sekunden benötigt. Wie im vorherigen Simulationslauf, betragen sowohl der Median als auch der Mittelwert 0 Sekunden. Im Simulationslauf werden insgesamt 8950 Generierungsfehler erzeugt, wovon 119 auf die doppelten Standorte entfallen. Mindestens 0 und maximal 2 doppelte Standorte werden in einer Replikation erzeugt. Aufgrund des Medians von 0 Fehlern und des Mittelwerts von 0,12 Fehlern ist die maximale Anzahl der Fehler als Ausreißer zu bezeichnen. Außerhalb der Stadtgrenzen werden in diesem Simulationslauf 2132 Standorte erzeugt, wobei in einer Replikation mindestens 0 und maximal 8 Standorte außerhalb der Stadtgrenzen liegen. Der Maximalwert dieses Fehlers ist wiederum als Ausreißer zu beschreiben, da der Median bei 2 Fehlern und der Mittelwert bei 2,13 Fehlern liegen. Bei der Generierung der Kundenstandorte treten insgesamt 6699 Berechnungsfehler auf. In einer Replikation sind dies mindestens 0 und maximal 17 Berechnungsfehler. Auch hier ist der Maximalwert als Ausreißer zu bezeichnen, weil der Median bei 6 Fehlern und der Mittelwert bei 6,7 Fehlern liegen. Für die genannten Lagemaße sind die Minimal- und Maximalwerte in Abbildung 51 zu sehen. Als Mittelwert der Tourenlänge ergeben sich 7218,93 LE und der Median beträgt 7219,21 LE. Die linksschiefe Verteilung aus dem vorherigen Simulationslauf bleibt damit

bestehen. Zwischen Mittelwert und Median ergibt sich ein Abstand von 0,28 LE, was einer Abweichung von 0,0039 Prozent entspricht, bezogen auf den Mittelwert. Aufgrund dieser Abweichung ist die Linksschiefe als schwach ausgeprägt zu bezeichnen und die Verteilung der Tourenlänge beinahe symmetrisch.

Im dritten Simulationslauf mit 10000 Replikationen wird eine Gesamtgenerierungszeit von 6,015 Sekunden benötigt. Eine Replikation nimmt dabei mindestens 0 und maximal 0,016 Sekunden in Anspruch. Sowohl der Median als auch der Mittelwert betragen, wie in den vorherigen Simulationsläufen, 0 Sekunden. Insgesamt werden in diesem Simulationslauf 87924 Generierungsfehler erzeugt, wovon 1150 auf doppelte Standorte entfallen. Dabei werden in einer Replikation mindestens 0 und maximal 3 doppelte Standorte generiert. Aufgrund des Mittelwerts von 0,12 und des Medians von 0 für diesen Fehler ist der Maximalwert als Ausreißer zu bezeichnen. In diesem Simulationslauf werden 20631 Standorte außerhalb der Stadtgrenzen erzeugt, wobei mindestens eine Anzahl von 0 und maximal eine von 11 dieser Fehler in einer Replikation entstehen. Der Maximalwert ist als Ausreißer zu bezeichnen, da der Median bei 2 und der Mittelwert bei 2,06 Fehlern liegen. Bei der Generierung der Standorte entstehen 66143 Berechnungsfehler. Innerhalb einer Replikation ergeben sich mindestens 0 und maximal 23 Berechnungsfehler, wobei der Maximalwert einen Ausreißer darstellt, weil der Median bei 6 und der Mittelwert bei 6,61 liegen. Die Mindest- und Maximalwerte der genannten Lagemaße sind grafisch aufbereitet in Abbildung 51 zu finden. Für den Mittelwert der Tourenlänge ergeben sich 7218,04 LE und für den Median 7217,61 LE. Mit diesen Werten ist der Median, im Gegensatz zu den vorherigen Simulationsläufen, kleiner als der Mittelwert und die Verteilung der Tourenlänge ist nun rechtsschief. Zwischen beiden Lagemaßen liegt ein Abstand von 0,43 LE, was bezogen auf den Mittelwert eine Abweichung von 0,0059 Prozent ist. Wegen dieser Abweichung ist die Rechtsschiefe schwach ausgeprägt und die Verteilung der Tourenlänge beinahe symmetrisch. Für alle Simulationsläufe ist aussagbar, dass sich die Mittelwerte und Mediane der Tourenlänge für eine steigenden Anzahl an Replikationen nicht signifikant voneinander unterscheiden. Zudem ist sowohl für die Mittelwerte und Mediane der Generierungszeit als auch der Generierungsfehler Entsprechendes zu beobachten. Eine Verhundertfachung der Replikationen vom ersten zum dritten Simulationslauf führt somit nicht zu signifikant unterschiedlichen Werten für die genannten Lagemaße. Zudem verzehnfachen sich die Werte der Gesamtgenerierungszeit und der Gesamtgenerierungsfehler, wenn die Replikationen um das Zehnfache erhöht werden.

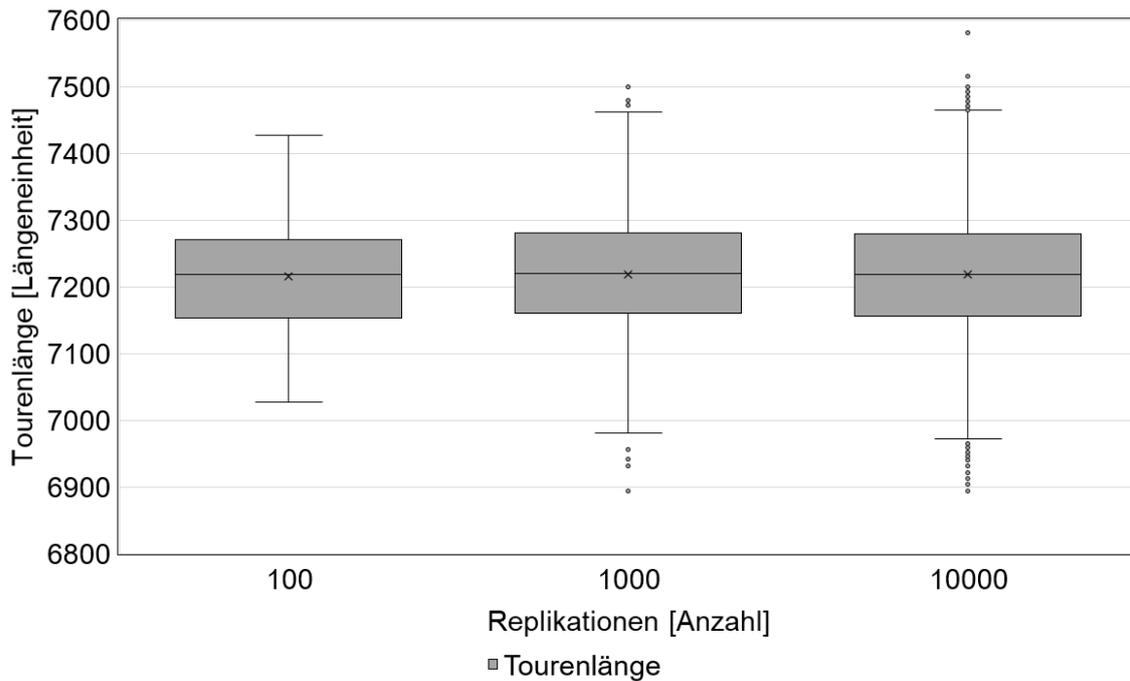


Abbildung 52: Normalverteilung Box-Plots Berücksichtigung Bevölkerungsdichte

Für die drei Simulationsläufe sind die zugehörigen Tourenlängen jeweils in ein Box-Plot-Diagramm eingetragen worden, welches in Abbildung 52 zu sehen ist. Die Markierung des Medians erfolgt durch eine horizontale Linie innerhalb der Box, die des Mittelwerts mit x. In der Abbildung ist sowohl die linksschiefe, wobei der Mittelwert unterhalb des Medians liegt, als auch die rechtsschiefe Verteilung, bei der der Mittelwert oberhalb des Medians liegt, zu erkennen. Die Boxen, in denen sich 50 Prozent der Werte der Tourenlänge befinden, scheinen in der Abbildung für alle drei Simulationsläufe ähnlich groß, die Lage hingegen unterschiedlich zu sein. Durch die Werte des Interquartilsabstandes lässt sich bestätigen, dass die Boxen der drei Simulationsläufe ähnliche Größen aufweisen. Für den ersten Simulationslauf beträgt dieser 121,02 LE, für den Zweiten 121,27 LE sowie für den Dritten 123,25 LE. Die unterschiedliche Lage der Box lässt sich mit den Werten für das 25% Quartil überprüfen, welche die unteren Grenzen der jeweiligen Boxen darstellen. Im ersten Simulationslauf liegt diese Grenze bei 7148,9 LE, steigt im zweiten Simulationslauf auf 7159,7 LE und sinkt im letzten Simulationslauf auf 7156,18 LE. Einen Unterschied der in Abbildung 52 sichtbar ist, stellt die Länge der Whisker dar. Im ersten Simulationslauf beginnt der untere Whisker bei 7027,49 LE und der obere Whisker endet bei 7427,32 LE. Für den zweiten Simulationslauf ergeben sich äquivalent 6981,21 LE und 7462,51 LE sowie 6972,42 LE und 7464,13 LE im dritten Simulationslauf. Zudem unterscheiden sich die drei Simulationsläufe bezüglich ihrer Ausreißer. Im ersten Simulationslauf existiert kein

Ausreißer. Hingegen existieren im zweiten Simulationslauf 4 untere und 3 obere Ausreißer sowie im dritten Simulationslauf 37 untere und 33 obere Ausreißer. Somit führt eine Erhöhung der Replikationen zu einer höheren Anzahl an Ausreißern. Zudem ergeben sich durch die unterschiedlichen Werte des 25% Quartils für die Boxen unterschiedliche Wertebereiche. Außerdem erhöht sich mit steigender Anzahl an Replikationen auch der Wertebereich, der zwischen den Whiskern liegt. Trotz dieser Ergebnisse verändert sich der für Ausreißer anfällig Mittelwert über alle drei Simulationsläufe kaum. Im ersten Simulationslauf beträgt dieser 7216,01 LE, erreicht sein Maximum im Zweiten 7218,93 LE und sinkt auf 7218,04 LE im dritten Simulationslauf. Zwischen dem kleinsten und größten Mittelwert liegt ein Abstand von 2,92 LE, was einer maximalen Abweichung von 0,0405 Prozent, bezogen auf den maximalen Wert, entspricht. Mit dieser maximalen Abweichung sind die Veränderungen des Mittelwerts nicht signifikant.

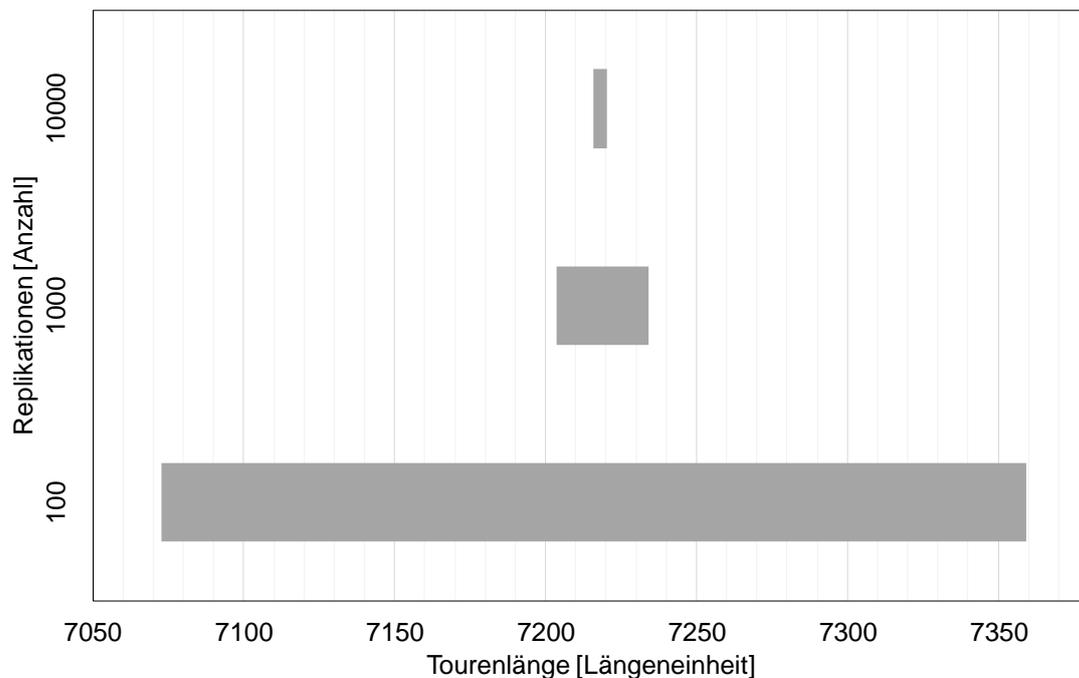


Abbildung 53: Normalverteilung 95% Konfidenzintervall Berücksichtigung Bevölkerungsdichte

Für die drei Simulationsläufe sind die zugehörigen approximierten 95% Konfidenzintervalle der Tourenlänge in Abbildung 53 zu finden. In der Abbildung ist zu erkennen, dass für eine steigende Anzahl an Replikationen die Breite des Konfidenzintervalls sinkt. Diese Breite beträgt für den ersten Simulationslauf 286,36 LE, sinkt für den Zweiten auf 30,43 LE und erreicht im Dritten mit 4,56 LE ihr Minimum. Je größer die Anzahl der Replikationen ist, umso mehr nähert sich das Konfidenzintervall

dem Mittelwert der Tourenlänge des zugehörigen Simulationslaufs. Mit Hilfe des approximierten 95% Konfidenzintervalls des dritten Simulationslaufs lässt sich die Aussage treffen, dass mit einer Wahrscheinlichkeit von 95 Prozent der wahre Wert der Tourenlänge vom Konfidenzintervall der Dimension (7215,76; 7220,32) überdeckt wird. In den drei Simulationsläufen führt auch eine Erhöhung auf 10000 Replikationen nicht zu einer signifikanten Veränderung der Mediane sowie Mittelwerte der Tourenlänge, der Generierungsfehler sowie der Generierungszeit. Für den Vergleich der Generierungsverfahren in Abschnitt 7.1 können deswegen die Werte jedes Simulationslaufs genutzt werden. Bei 10000 Replikationen stellt jedoch die Breite des approximierten 95% Konfidenzintervall ein Minimum dar und die Tourenlänge kann somit präziser geschätzt werden. Aufgrund dessen werden die Ergebnisse des dritten Simulationslaufs mit 10000 Replikationen in den Vergleich in Abschnitt 7.1 einfließen.

6.4.3 Berücksichtigung der Bevölkerungsdichte Dreiecksverteilung

In der dritten und letzten Simulation in diesem Abschnitt erfolgt die Generierung der Kundenstandorte mit einer Dreiecksverteilung. Die x-Koordinaten der Kundenstandorte werden im Intervall (125; 165) sowie die y-Koordinaten im Intervall (180; 265) erzeugt. Der wahrscheinlichste Wert für die x-Koordinaten beträgt 145 LE und der für die y-Koordinaten 222,5 LE. In jeder Replikation werden 22 Kundenstandorte erzeugt. Eine Auswahl an Ergebniswerten ist in Tabelle 23 zu finden, wobei diese Auswahl keine Bewertung der dargestellten Ergebnisse vornimmt. Im Anhang in Tabelle 43 sind sämtliche Ergebniswerte zu finden.

Tabelle 23: Dreiecksverteilung Berücksichtigung Bevölkerungsdichte

Replikationen [Anzahl]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
100	7212,94	7216,56	19	0,375
1000	7213,27	7214,03	128	2,453
10000	7216,48	7216,25	1230	11,219

Für den ersten Simulationslauf mit 100 Replikationen werden 0,375 Sekunden benötigt. Als minimale Generierungszeit einer Replikation ergeben sich 0 Sekunden und als

Maximale 0,016 Sekunden. Sowohl der Median als auch der Mittelwert liegen bei 0 Sekunden. Bei der Generierung der Standorte werden insgesamt 19 doppelte Standorte erzeugt, wobei in einer Replikation mindestens 0 und maximal 2 doppelte Standorte vorliegen. Aufgrund des Medians von 0 Fehlern und des Mittelwerts von 0,19 Fehlern, ist der Maximalwert als Ausreißer zu bezeichnen. In Abbildung 54 ist dies grafisch aufgearbeitet. Aus der Abbildung ist zu entnehmen, dass der Mittelwert sichtbar kleiner als der Maximalwert ist. Für die Tourenlänge ergibt sich ein Mittelwert von 7212,94 LE und ein Median von 7216,56 LE, womit der Median größer als der Mittelwert ist. Aufgrund dessen ist die Verteilung der Tourenlänge linksschief. Da zwischen beiden Lagemaßen ein Abstand von 3,62 LE liegt und dies einer Abweichung von 0,0502 Prozent, bezogen auf den Mittelwert, entspricht, ist die Linksschiefe schwach ausgeprägt und kann auch als beinahe symmetrisch bezeichnet werden.

Im zweiten Simulationslauf wird eine Gesamtgenerierungszeit von 2,453 Sekunden benötigt. Für eine Replikation ergeben sich mindestens 0 und maximal 0,016 Sekunden. Wie im vorherigen Simulationslauf betragen sowohl der Median als auch der Mittelwert 0 Sekunden. Insgesamt ergeben sich aus den 1000 Replikationen 128 doppelte Standorte, wobei in einer Replikation mindestens 0 und maximal 2 doppelte Standorte erzeugt werden. Aufgrund des Medians von 0 und des Mittelwerts von 0,13, handelt es sich bei dem Maximalwert der doppelten Standorte um einen Ausreißer. Dies ist auch in Abbildung 54 verdeutlicht, weil der Mittelwert sichtbar kleiner als der Maximalwert ist. Als Mittelwert der Tourenlänge ergeben sich 7213,27 LE und als Median 7214,03 LE. Somit ist der Median weiterhin größer als der Mittelwert und die Linksschiefe bleibt bestehen. Der Abstand zwischen beiden Lagemaßen beträgt 0,76 LE, was bezogen auf den Mittelwert eine Abweichung von 0,0105 Prozent darstellt. Aufgrund dieser Abweichung ist die Linksschiefe schwach ausgeprägt und die Verteilung der Tourenlänge ist beinahe symmetrisch. Der dritte Simulationslauf mit 10000 Replikationen benötigt eine Gesamtgenerierungszeit von 11,219 Sekunden, wobei der Minimalwert für eine Replikation bei 0 Sekunden und der Maximalwert bei 0,016 Sekunden liegen. Sowohl der Median als auch der Mittelwert ergeben sich, wie in den vorherigen Simulationsläufen, zu 0 Sekunden. In allen 10000 Replikationen dieses Simulationslaufs werden insgesamt 1230 doppelte Standorte erzeugt. Dabei liegen in einer Replikation mindestens 0 und maximal 3 doppelte Standorte vor. Aufgrund des Medians von 0 und des Mittelwerts von 0,12 handelt es sich bei dem Maximalwert dieses Fehlers um einen Ausreißer. Abbildung 54 unterstützt diese Aussage, da der Maximalwert sichtbar größer als der für Ausreißer

anfällige Mittelwert ist. Der Mittelwert der Tourenlänge beträgt 7216,48 LE und der Median 7216,25 LE, womit der Mittelwert größer als der Median ist und die Linksschiefe zu einer Rechtsschiefe wird. Aufgrund des Abstands von 0,23 LE zwischen beiden Lagemaßen, der einer Abweichung von 0,0032 Prozent entspricht, ist die Rechtsschiefe schwach ausgeprägt und die Verteilung der Tourenlänge beinahe symmetrisch.

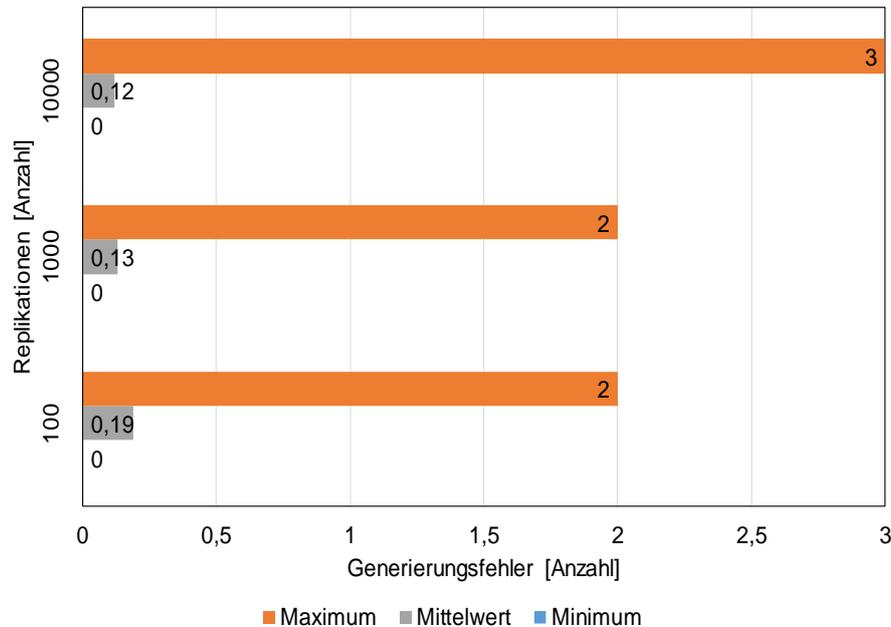


Abbildung 54: Dreiecksverteilung Generierungsfehler pro Replikation Berücksichtigung Bevölkerungsdichte

Eine Betrachtung aller Simulationsläufe zeigt auf, dass sich der Median und der Mittelwert von Simulationslauf zu Simulationslauf nicht signifikant voneinander unterscheiden. Analog ist dies auch für die Mittelwerte und Mediane der Generierungsfehler und der Gesamtgenerierungszeit zu beobachten. Daran ist erkennbar, dass sich die genannten Lagemaße bei einer Verhundertfachung der Replikationen nicht signifikant verändern.

In Abbildung 55 sind die Tourenlängen der drei Simulationsläufe in ein zugehöriges Box-Plot-Diagramm eingetragen worden. Der Mittelwert ist in den Diagrammen mit \bar{x} gekennzeichnet, der Median durch eine horizontale Linie innerhalb der Box. Die linksschiefe Verteilung der Tourenlänge in den ersten zwei Simulationsläufen kann in der Abbildung daran erkannt werden, dass in den zugehörigen Box-Plot-Diagrammen die Mittelwerte niedriger als die Mediane sind. Äquivalent ist die Rechtsschiefe im dritten Simulationslauf an der Lage des Mittelwerts oberhalb des Medians erkennbar. Die Box des dritten Simulationslaufs erscheint in Abbildung 55 kleiner zu sein als die Boxen des

ersten und zweiten Simulationslaufs, welche ähnlich groß erscheinen. Mit Hilfe der Werte des Interquartilsabstandes kann diese Beobachtung bestätigt werden. Dieser beträgt im ersten Simulationslauf 121,89 LE, im Zweiten 120,42 LE sowie 117,71 LE im Dritten. Zudem scheint sich die Lage der Box des dritten Simulationslaufs von den anderen beiden Simulationsläufen in der Abbildung zu unterscheiden. Dies kann mit den Werten des 25% Quartils, welche die unteren Grenzen der Boxen darstellen, bestätigt werden. Für den ersten Simulationslauf liegt diese bei 7151,86 LE, im Zweiten bei 7152,36 sowie im Dritten bei 7157,62 LE.

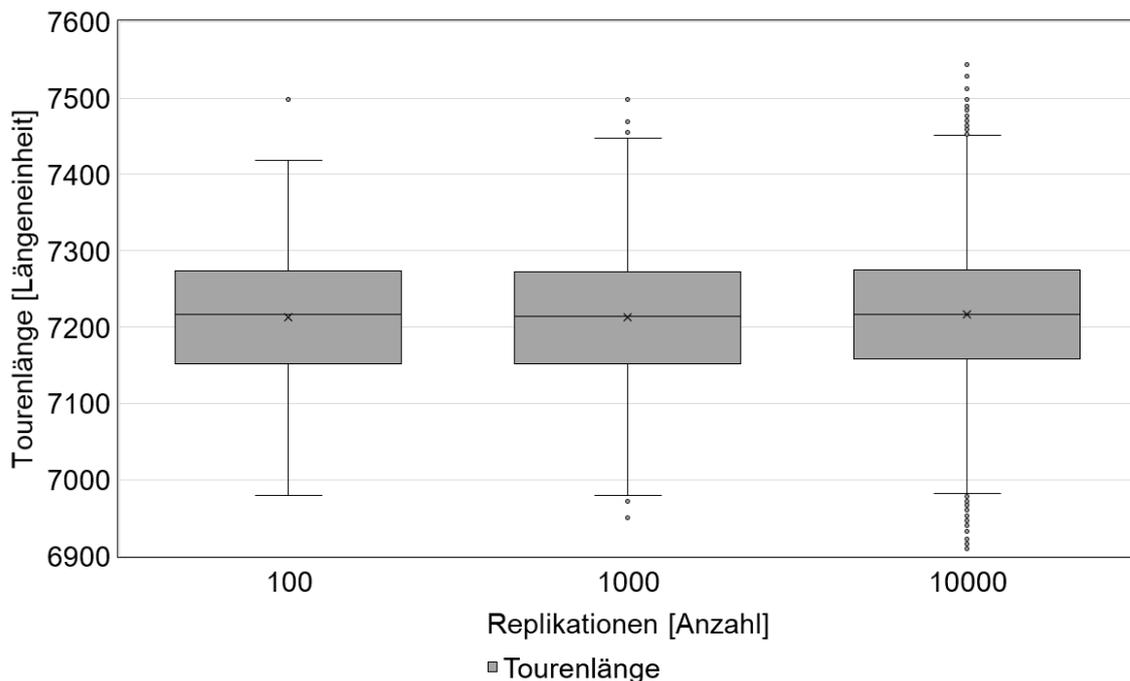


Abbildung 55: Dreiecksverteilung Box-Plots Berücksichtigung Bevölkerungsdichte

Des Weiteren ist in der Abbildung zu beobachten, dass die Whisker eine unterschiedliche Länge besitzen. Im ersten Simulationslauf startet der untere Whisker bei 6978,89 LE und der obere Whisker endet bei 7418,65 LE, im Zweiten ergeben sich Werte von 6978,89 LE und 7447,49 sowie im Dritten 6981,3 LE und 7451,51 LE. Außerdem existieren in den drei Simulationsläufen unterschiedlich viele Ausreißer. Für den ersten Simulationslauf ergibt sich 1 oberer Ausreißer, für den zweiten Simulationslauf 2 untere und 3 obere Ausreißer sowie für den dritten Simulationslauf 34 untere und 37 obere Ausreißer. Damit nimmt die Anzahl der Ausreißer bei einer Erhöhung der Replikationen zu. Anhand der unterschiedlichen Werte für die Interquartilsabstände, der 25% Quartile und der Whisker ist zudem erkennbar, dass sich die Wertebereiche, in denen sich die Tourenlängen befinden, bei einer steigenden Anzahl der Replikationen vergrößern. Trotz

dieser Beobachtungen verändert sich der Mittelwert, der für Ausreißer anfällig ist, über alle Simulationsläufe kaum. Im ersten Simulationslauf beträgt der Mittelwert der Tourenlänge 7212,94 LE und steigt im dritten Simulationslauf auf 7216,48 LE an, womit der Abstand zwischen beiden 3,54 LE beträgt. Bezogen auf den Mittelwert des dritten Simulationslaufs entspricht dieser Abstand einer Abweichung von 0,049 Prozent und ist damit nicht signifikant.

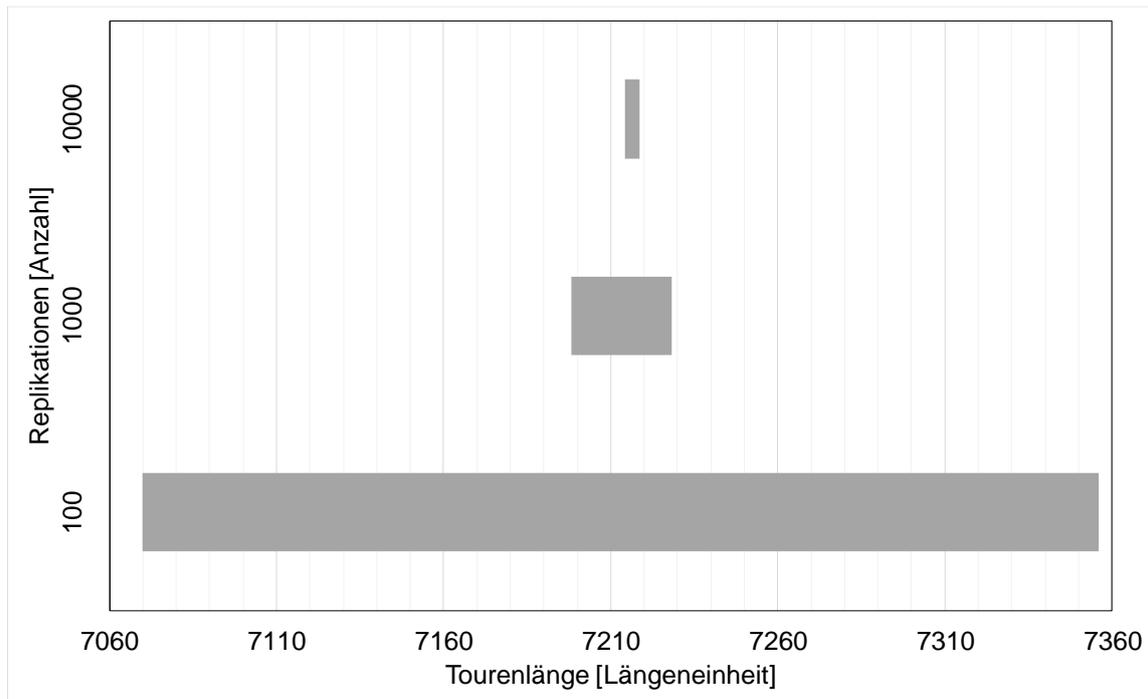


Abbildung 56: Dreiecksverteilung 95% Konfidenzintervall Berücksichtigung Bevölkerungsdichte

Die approximierten 95% Konfidenzintervalle der Tourenlänge sind in Abbildung 56 für jeden Simulationslauf zu sehen. Aus der Abbildung kann entnommen werden, dass die Breite der Konfidenzintervalle mit einer steigenden Anzahl Replikationen sinkt. Die Breite des Konfidenzintervalls beträgt im ersten Simulationslauf 286,33 LE, sinkt im zweiten Simulationslauf auf 30,26 LE und erreicht mit 4,43 LE im dritten Simulationslauf ihr Minimum. Mit steigender Anzahl der Replikationen nähert sich das Konfidenzintervall dem Mittelwert des zugehörigen Simulationslaufs an. Das approximierte 95% Konfidenzintervall des dritten Simulationslaufs führt zu der Aussage, dass der wahre Wert der Tourenlänge mit einer Fehlerwahrscheinlichkeit von 5 Prozent mit dem Konfidenzintervall der Dimension (7214,27; 7218,7) überdeckt wird. Die Auswertung der drei Simulationsläufe zeigt, dass eine Erhöhung der Replikationen auf 10000 zu keinen signifikanten Veränderungen der Mittelwerte sowie Mediane der

Tourenlänge, der Generierungszeit und der Generierungsfehler führt. Deswegen können die Werte eines jeden Simulationslaufs im Abschnitt 7.1 herangezogen werden, um die Generierungsverfahren miteinander zu vergleichen. Die präziseste Vorhersage zur Größenordnung der Tourenlänge ist mit dem kleinsten approximierten 95% Konfidenzintervall möglich, welches bei 10000 Replikationen ein Minimum darstellt, weswegen dieser Simulationslauf als geeignetster Vertreter der drei Simulationsläufe ausgewählt wird, um in den Vergleich in Abschnitt 7.1 einzufließen.

7. Vergleich und Bewertung der simulierten Generierungsverfahren

Basierend auf den Auswertungen des vorherigen Kapitels wird im ersten Abschnitt dieses Kapitels ein Vergleich aller Generierungsverfahren durchgeführt. Dafür ist der erste Abschnitt in fünf Unterkapitel aufgeteilt, wobei in einem Unterkapitel jeweils die Ergebnisse der Generierungsverfahren bezüglich eines Testsets verglichen werden. Darauf aufbauend erfolgt zum Abschluss dieses Kapitels im zweiten Abschnitt die Bewertung der einzelnen Generierungsverfahren.

7.1 Vergleich der Generierungsverfahren

In diesem Abschnitt werden die Generierungsverfahren für jedes Testset anhand der Ergebnisse aus Kapitel 6 miteinander verglichen. Der Vergleich wird für jedes Testset separat in einem eigenen Unterabschnitt, 7.1.1 bis 7.1.5, durchgeführt. Dabei werden die Verfahren den jeweiligen Testsets zugeordnet und so einander gegenübergestellt. Wie in den Abschnitten des vorherigen Kapitels beschrieben, werden für jedes Testset und jedes Verfahren die Simulationsergebnisse des dritten Simulationslaufs mit 10000 Replikationen für den jeweiligen Vergleich im Testset herangezogen. Für die einzelnen Vergleiche werden jeweils zu Beginn die Mittelwerte und Mediane der Tourenlänge, die Gesamtfehler und die Generierungszeit tabellarisch dargestellt. Diese Auswahl ermöglicht einen Überblick über die Ergebnisse und stellt keine Bewertung dieser dar. Sämtliche Ergebniswerte können im Anhang in Tabelle 37 bis Tabelle 43 gefunden werden.

7.1.1 Vergleich Testset E-n22-k4

Der erste Vergleich in diesem Abschnitt wird für das Testset E-n22-k4 durchgeführt. Die dazugehörigen Simulationen sind in den Abschnitten 6.1.1, 6.2.1 sowie 6.3.1 zu finden. Aus Tabelle 24 kann entnommen werden, dass sich für die Normal- und die Dreiecksverteilung vergleichbare Lagemaße der Tourenlängen ergeben. Der Mittelwert der Normalverteilung beträgt 4362,06 LE und der entsprechende Median 4359,94 LE sowie der Mittelwert der Dreiecksverteilung 4387,25 LE und der zugehörige Median

4384,16 LE. Abweichend davon beträgt der Mittelwert der Gleichverteilung 4542,45 LE und der Median 4539,18 LE. Damit führt die Generierung der Kundenstandorte mit einer Normalverteilung sowohl zu dem geringsten Mittelwert als auch dem geringsten Median der Tourenlänge, dicht gefolgt von der Dreiecksverteilung. Das Schlusslicht bildet somit die Gleichverteilung.

Table 24: Vergleich Testset E-n22-k4 Ergebnisse

Generierungsverfahren [Name]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
Gleichverteilung	4542,45	4539,18	44	12,234
Normalverteilung	4362,06	4359,94	75016	5,453
Dreiecksverteilung	4387,25	4384,16	86	11,297

Bei der Gesamtanzahl der Fehler ergibt sich ein anderes Bild. Hier erzeugt die Gleichverteilung 44 Fehler und die Dreiecksverteilung 86 Fehler. Bei der Normalverteilung entstehen hingegen 75016 Fehler. Die Generierung der Kundenstandorte mit einer Gleich- und Dreiecksverteilung führt nur zu dem Generierungsfehler „doppelte Standorte“. In die Gesamtanzahl der Fehler für die Normalverteilung fließen neben der Anzahl der Generierungsfehler „doppelte Standorte“ auch die Anzahl der Generierungsfehler „außerhalb der Stadtgrenzen“ und „Berechnungsfehler“ ein. Werden die letztgenannten außer Acht gelassen, ergeben sich für die Normalverteilung 122 doppelte Standorte. Auch mit dieser so verringerten Fehleranzahl bleibt diese bei der Normalverteilung am höchsten.

Bei der minimalen und maximalen Anzahl sowie beim Mittelwert des Fehlers „doppelte Standorte“ pro Replikation, liegen die drei Generierungsverfahren gleich auf (vgl. Abbildung 57). Für alle Generierungsverfahren ergeben sich 0 Fehler für die minimale und 1 Fehler für die maximale Anzahl. Der Mittelwert der Gleichverteilung beträgt 0 Fehler und sowohl für die Normal- als auch die Dreiecksverteilung ergibt sich ein Mittelwert von 0,01 Fehler. Die Gesamtgenerierungszeit ist für die Normalverteilung mit 5,453 Sekunden am niedrigsten. Hingegen unterscheidet sich die Gesamtgenerierungszeit der Dreiecksverteilung mit 11,297 Sekunden kaum von der Gesamtgenerierungszeit der Gleichverteilung mit 12,234 Sekunden.

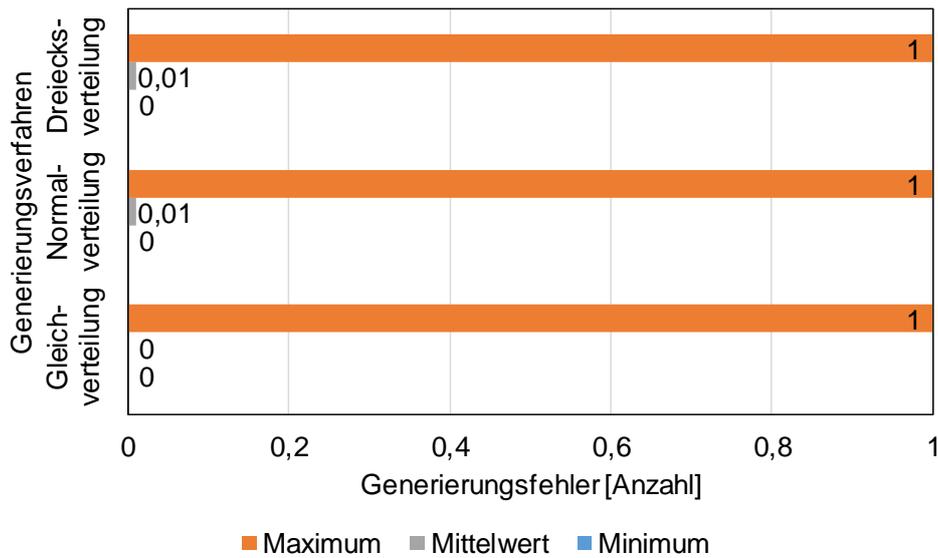


Abbildung 57: Vergleich Generierungsfehler pro Replikation Testset E-n22-k4

In Abbildung 58 sind die Box-Plot-Diagramme der Tourenlänge für die drei Generierungsverfahren dargestellt. Die Tourenlänge der Gleichverteilung besitzt aufgrund der weit gestreuten Ausreißer und der Länge der beiden Whisker die größte Streuung für die Werte der Tourenlänge. Insgesamt entstehen bei der Gleichverteilung 61 Ausreißer, welche sich in 36 untere und 25 obere Ausreißer aufteilen.

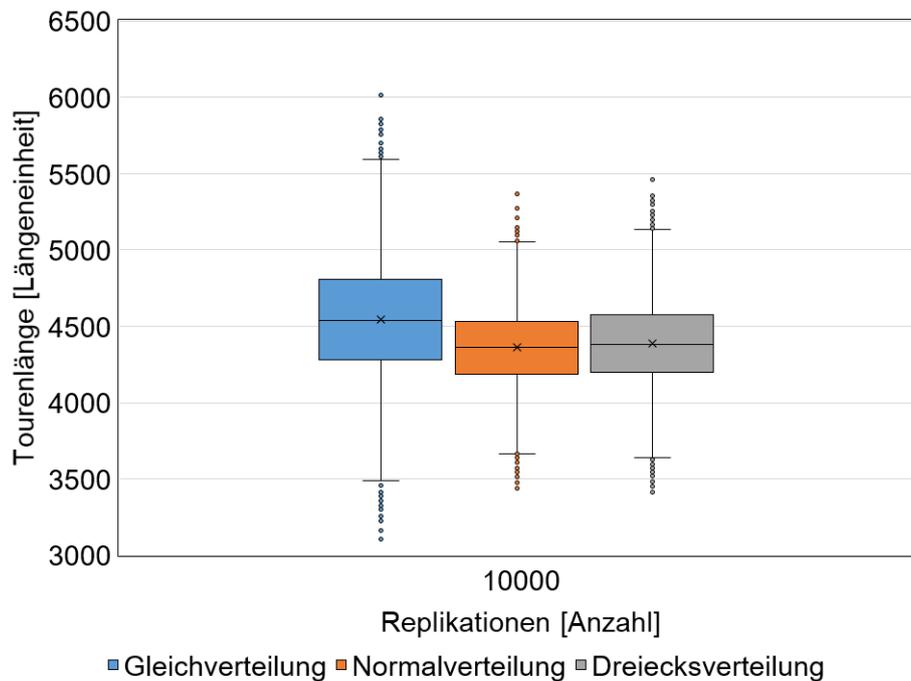


Abbildung 58: Vergleich Box-Plots Testset E-n22-k4

Die Normalverteilung erzeugt im Vergleich mit den anderen beiden Generierungsverfahren die geringste Streuung der Tourenlänge, denn die Länge beider Whisker ist am kleinsten und auch die Ausreißer liegen dichter zusammen. In der Normalverteilung entstehen insgesamt 58 Ausreißer, 32 untere und 26 obere Ausreißer. Sowohl die Länge der Whisker als auch die Verteilung der Ausreißer unterscheiden sich bei der Dreiecksverteilung nur im geringen Maße von der Normalverteilung und die Dreiecksverteilung besitzt damit eine ähnliche Streuung der Tourenlänge. Die Anzahl der Ausreißer ist mit insgesamt 66, 35 unteren und 31 oberen Ausreißern, jedoch größer als die der beiden anderen Generierungsverfahren.

Die approximierten 95% Konfidenzintervalle der drei Generierungsverfahren, die in Abbildung 59 dargestellt sind, unterscheiden sich sichtbar hinsichtlich ihrer Breite und Lage. Für das Konfidenzintervall der Gleichverteilung ergibt sich eine Breite von 15,34 LE, für die Normalverteilung von 10,12 LE sowie für die Dreiecksverteilung von 10,94 LE. Damit ist die Breite des Konfidenzintervalls der Gleichverteilung am größten und die der Normalverteilung am geringsten, obwohl zwischen der Breite des Konfidenzintervalls der Normalverteilung und der Dreiecksverteilung lediglich ein geringer Unterschied besteht. Aufgrund dessen ist mit dem Konfidenzintervall der Normalverteilung die präziseste Aussage hinsichtlich der Größenordnung der Tourenlänge möglich.

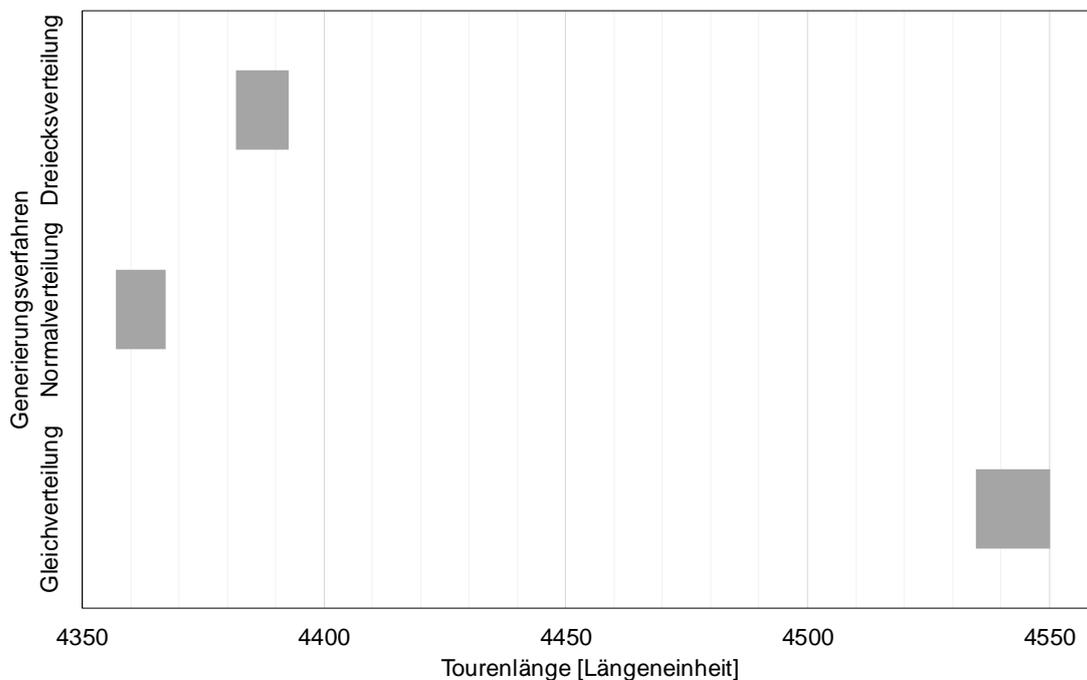


Abbildung 59: Vergleich 95% Konfidenzintervalle Testset E-n22-k4

Der weitaus sichtbarere Unterschied besteht in der Lage der Konfidenzintervalle. Für die Gleichverteilung liegt das Konfidenzintervall zwischen 4580 und 4550 LE. Im Unterschied dazu befindet sich das Konfidenzintervall der Dreiecksverteilung zwischen 4380 und 4395 LE sowie das der Normalverteilung zwischen 4350 und 4360 LE. Mit diesen Werten führen Aussagen bezüglich der Größenordnung der Tourenlänge, die mit dem approximierten Konfidenzintervall der Normalverteilung getroffen werden, zu einer Tourenlänge mit den wenigsten LE. Aufgrund der Ergebnisse dieses Vergleichs wird für dieses Testset die Normalverteilung an erster Stelle, die Dreiecksverteilung an Zweiter und die Gleichverteilung an dritter Stelle platziert.

7.1.2 Vergleich Testset E-n22-k4 unter Berücksichtigung der Bevölkerungsdichte

Der zweite Vergleich dieses Abschnittes wird wiederum für das Testset E-n22-k4 durchgeführt, in diesem Fall aber unter Berücksichtigung der Bevölkerungsdichte. Die Simulationen für dieses Testset werden in Abschnitt 6.4 beschrieben. Die Betrachtung von Tabelle 25 zeigt auf, dass sich die Ergebnisse von denen in Tabelle 24 aus Abschnitt 7.1.1 unterscheiden, obwohl bei beiden dasselbe Testset zugrunde liegt. Dieser Unterschied ist darauf zurückzuführen, dass bei den Simulationen, die in Abschnitt 7.1.1 ausgewertet werden, Kundenstandorte auf dem gesamten Gebiet des Testsets und unter Berücksichtigung der Bevölkerungsdichte Kundenstandorte nur in der oberen rechten Ecke erzeugt werden (vgl. Abschnitt 6.4).

Tabelle 25: Vergleich Testset E-n22-k4 mit Bevölkerungsdichte Ergebnisse

Generierungsverfahren [Name]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
Gleichverteilung	7224,19	7223,62	672	16,578
Normalverteilung	7218,04	7217,61	87924	6,015
Dreiecksverteilung	7216,48	7216,25	1230	11,219

Des Weiteren ähneln sich die Ergebnisse hinsichtlich des Medians und des Mittelwerts der Tourenlänge. Für die Gleichverteilung beträgt der Mittelwert 7224,19 LE und der Median 7223,62 LE, für die Normalverteilung entsprechend 7218,04 LE und 7217,61 LE

sowie entsprechend für die Dreiecksverteilung 7216,48 LE und 7216,25 LE. Mit diesen Werten führt die Dreiecksverteilung sowohl im Mittelwert als auch im Median zu der niedrigsten Tourenlänge, gefolgt von der Normalverteilung. Die Gleichverteilung folgt mit geringem Abstand.

In der Gesamtanzahl der Generierungsfehler stellt sich dies für die drei Generierungsverfahren anders dar. Die Generierung der Kundenstandorte mit einer Gleichverteilung führt hier zu 672 Fehlern, was die geringste Anzahl darstellt. Für die Dreiecksverteilung ergeben sich mit 1230 beinahe doppelt so viele erzeugte Generierungsfehler und die Generierung mit einer Normalverteilung ergibt 87924 Generierungsfehler. Hierbei gilt es jedoch zu beachten, dass bei der Normalverteilung in die Gesamtanzahl der Generierungsfehler zusätzlich zu dem Fehler „doppelte Standorte“, der in der Gleich- und Dreiecksverteilung den einzigen Generierungsfehler darstellt, die Fehler „außerhalb der Stadtgrenzen“ sowie „Berechnungsfehler“ eingehen. Werden in der Normalverteilung nur die doppelten Standorte betrachtet, so ergibt sich eine Anzahl von 1150 Fehlern. Diese Fehleranzahl ist niedriger als die der Dreiecksverteilung.

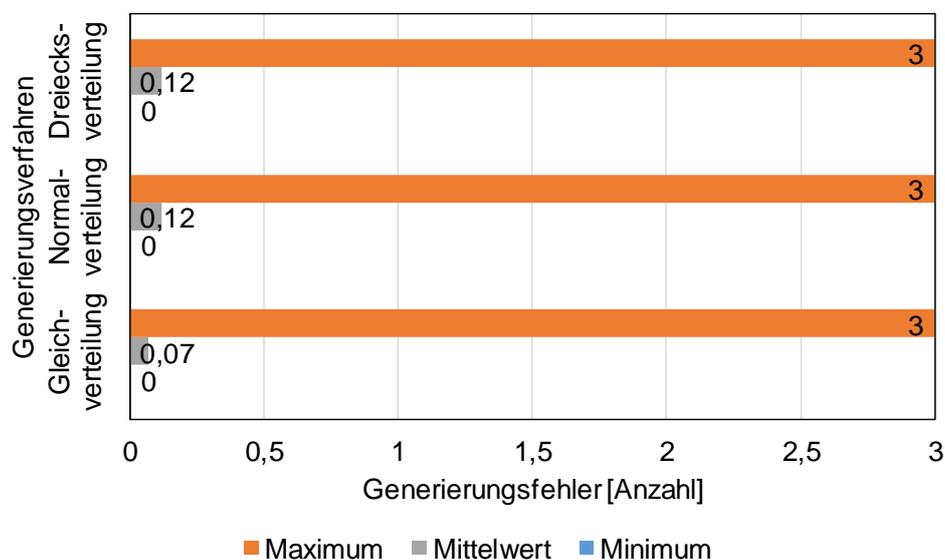


Abbildung 60: Vergleich Generierungsfehler pro Replikation Testset E-n22-k4 mit Bevölkerungsdichte

Sowohl die minimale als auch die maximale Anzahl pro Replikation des Generierungsfehlers „doppelte Standorte“ ist für die drei betrachteten Generierungsverfahren gleich (vgl. Abbildung 60). Die minimale Anzahl beträgt 0 und die maximale Anzahl 3 Fehler. Zudem betragen die Mediane der drei Generierungsverfahren jeweils 0 Fehler. Geringe Unterschiede in den drei

Generierungsverfahren existieren hinsichtlich der Mittelwerte, welcher für die Gleichverteilung 0,07 Fehler sowie sowohl für die Normal- als auch die Dreiecksverteilung 0,12 Fehler beträgt. Die geringste Gesamtgenerierungszeit ist mit 6,015 Sekunden der Normalverteilung zuzuordnen (vgl. Tabelle 25). Für die Dreiecksverteilung wird eine Generierungszeit von 11,219 Sekunden und für die Gleichverteilung eine von 16,578 Sekunden benötigt, was der maximalen Generierungszeit der drei Generierungsverfahren entspricht.

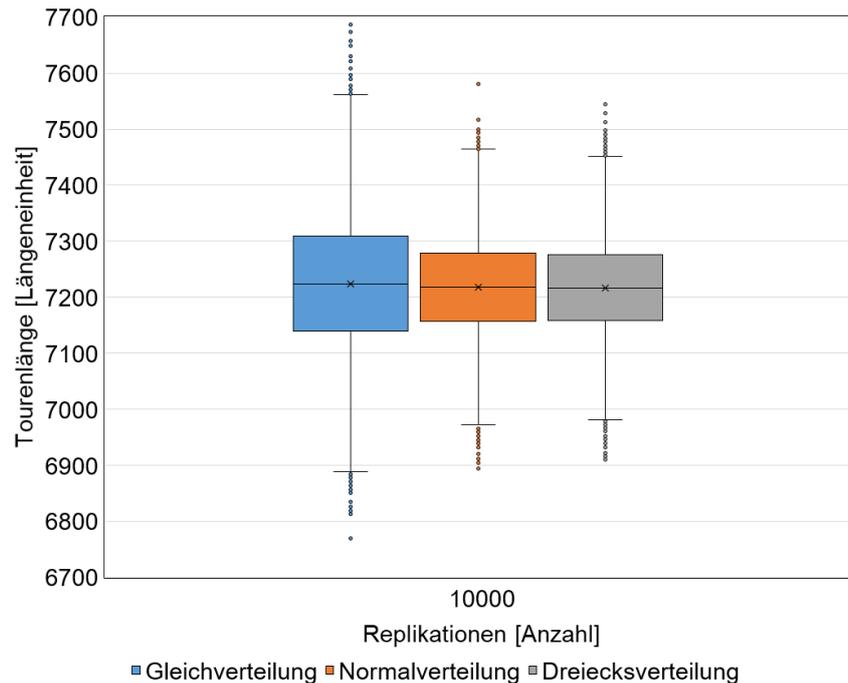


Abbildung 61: Vergleich Box-Plots Testset E-n22-k4 mit Bevölkerungsdichte

Die Box-Plot-Diagramme der Tourenlängen der drei Generierungsverfahren sind in Abbildung 60 dargestellt. Aus den Diagrammen können auch die ähnlichen Werte für den Mittelwert und die Mediane entnommen werden. Für die Tourenlänge der Gleichverteilung ergibt sich aufgrund der weit gestreuten Ausreißer und der Länge beider Whisker die größte Streuung. In diesem Generierungsverfahren entstehen insgesamt 67 Ausreißer, 33 untere und 34 obere Ausreißer. Die Streuung der Tourenlänge ist beim Generierungsverfahren der Normalverteilung geringer, denn die Ausreißer liegen dichter beieinander und die Whisker sind kürzer als die der Gleichverteilung. Für die Ausreißer ergibt sich eine Gesamtanzahl von 70 Ausreißer, 37 untere sowie 33 obere Ausreißer. Die Unterschiede der Dreiecksverteilung zur Normalverteilung sind gering, jedoch in den Box-Plot-Diagrammen sichtbar. Sowohl die Länge der Whisker als auch die dichte Lage der Ausreißer verringern die Streuung der Tourenlänge abermals, sodass diese ein

Minimum darstellt. Insgesamt werden mit der Dreiecksverteilung 71 Ausreißer erzeugt, davon 34 untere sowie 37 obere. Wie im vorherigen Abschnitt ist die Gesamtanzahl der Ausreißer für die Gleichverteilung am geringsten, die der Dreiecks- und Normalverteilung unterscheiden sich um 1 voneinander.

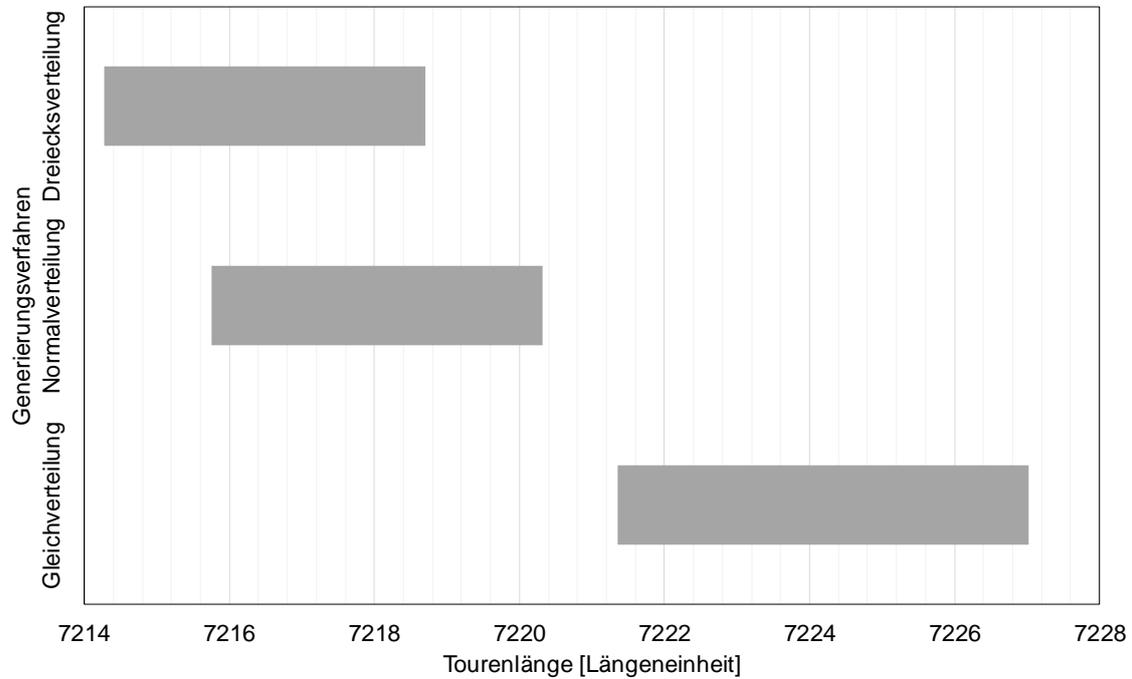


Abbildung 62: Vergleich 95% Konfidenzintervalle Testset E-n22-k4 mit Bevölkerungsdichte

Sowohl hinsichtlich der Breite als auch hinsichtlich der Lage unterscheiden sich die approximierten 95% Konfidenzintervalle der drei Generierungsverfahren, die in Abbildung 62 zu sehen sind, voneinander. Die Breite des Konfidenzintervalls beträgt für die Gleichverteilung 5,66 LE und stellt damit das Maximum dar. Für die Normalverteilung ergibt sich eine Breite von 4,56 LE sowie für die Dreiecksverteilung eine Breite von 4,43 LE. Obwohl zwischen der Breite des Konfidenzintervalls der Normal- und der Dreiecksverteilung eine geringfügige Abweichung besteht, ist der Wert der Dreiecksverteilung das Minimum für dieses Testset. Für die Gleichverteilung liegt das Konfidenzintervall zwischen 7221 und 7227 LE. Damit liegt es in der Nähe des Konfidenzintervalls der Normalverteilung, welches sich zwischen 7215 und 7221 LE befindet. Das Konfidenzintervall der Dreiecksverteilung, das zwischen 7214 und 7219 LE liegt, überschneidet sich mit dem Konfidenzintervall der Normalverteilung. Für das Eintreten einer Überschneidung der Konfidenzintervalle wurde in der Literatur die Verwendung eines Paired-t-Konfidenzintervall empfohlen (vgl. Abschnitt 3.4). Dieses

Konfidenzintervall beträgt, bei einem Konfidenzniveau von 95 Prozent, für die Normal- und Dreiecksverteilung $(-0,032; 3,425)$ LE. Da in dem berechneten Konfidenzintervall die 0 enthalten ist, kann nur die Aussage getroffen werden, dass kein Generierungsverfahren dem anderen bezüglich der Frage, welches Generierungsverfahren hinsichtlich der Größenordnung der Tourenlänge zu den niedrigsten LE führt, überlegen ist. Deswegen werden sowohl die Dreiecksverteilung als auch die Normalverteilung an erster Stelle und die Gleichverteilung an dritter Stelle platziert.

7.1.3 Vergleich Testset CMT14

Im dritten Vergleich dieses Abschnitts werden die Ergebnisse der drei Generierungsverfahren für das Testset CMT14 miteinander verglichen. Die entsprechenden Simulationen hierzu sind in den Abschnitten 6.1.2, 6.2.2 sowie 6.3.2 durchgeführt worden.

Tabelle 26: Vergleich Testset CMT14 Ergebnisse

Generierungsverfahren [Name]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
Gleichverteilung	8519,63	8517,40	6015	33,469
Normalverteilung	8306,28	8308,85	725900	16,313
Dreiecksverteilung	8192,56	8190,30	10735	20,250

In Tabelle 26 sind die Ergebnisse der drei Generierungsverfahren für dieses Testset abgebildet. Bei Betrachtung dieser ist auffällig, dass sich sowohl Mittelwert als auch Median voneinander unterscheiden. Der Mittelwert der Gleichverteilung beträgt 8519,63 LE und der Median 8517,4 LE. Damit stellen diese Werte das Maximum sowohl für den Mittelwert als auch den Median dar. Für die Normalverteilung ergibt sich ein Mittelwert von 8306,28 LE sowie ein Median von 8308,85 LE. Das Minimum dieser Werte wird mit der Dreiecksverteilung erreicht. In dieser beträgt der Mittelwert der Tourenlänge 8192,56 LE und der Median 8190,3 LE. Bei der Betrachtung der Gesamtanzahl der Generierungsfehler ergibt sich ein anderes Bild. Mit der Gleichverteilung werden insgesamt 6015 Generierungsfehler und mit der Dreiecksverteilung 10735

Generierungsfehler erzeugt. Die Generierung der Kundenstandorte mit der Normalverteilung führt zu 725900 Generierungsfehlern. Bei dieser Betrachtung muss jedoch wieder beachtet werden, dass die Gesamtanzahl der Generierungsfehler für die Gleich- und Dreiecksverteilung nur die Anzahl des Fehlers „doppelte Standorte“ enthält. Bei der Normalverteilung enthält die Gesamtanzahl der Generierungsfehler zusätzlich noch die Fehler „außerhalb der Stadtgrenzen“ sowie „Berechnungsfehler“. Werden die beiden letztgenannten Fehler herausgerechnet, ergeben sich für die doppelten Standorte in der Normalverteilung 7558 Fehler. Damit ist die Anzahl dieses Fehlers für die Normalverteilung niedriger als bei der Dreiecksverteilung.

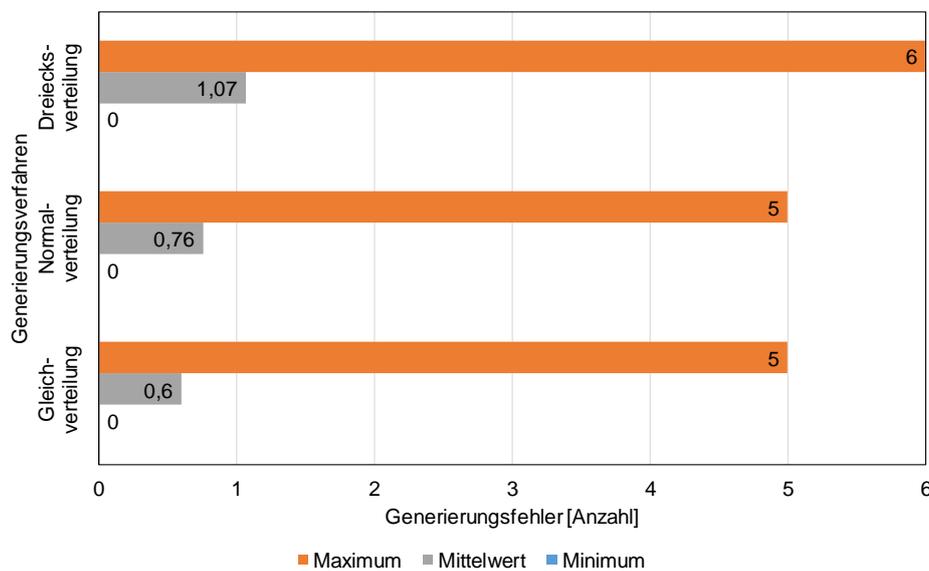


Abbildung 63: Vergleich Generierungsfehler pro Replikation Testset CMT14

Die minimale Anzahl pro Replikation für den Fehler „doppelte Standorte“ ist für alle drei Generierungsverfahren gleich (vgl. Abbildung 63). Zudem ist auch die Maximalanzahl von 5 Fehlern für die Gleich- und Normalverteilung dieselbe. Für die Dreiecksverteilung hingegen beträgt diese Maximalanzahl 6. Der Mittelwert des Fehlers liegt für die Gleichverteilung bei 0,6 Fehlern, für die Normalverteilung bei 0,76 Fehlern sowie für die Dreiecksverteilung bei 1,07 Fehlern. Im Median von 1 Fehler gleichen sich die Dreiecks- und Normalverteilung, wohingegen der Median der Gleichverteilung 0 Fehler beträgt. Die minimale Gesamtgenerierungszeit, in diesem Fall die von der Normalverteilung benötigte Zeit, liegt bei 16,313 Sekunden. Mit 20,25 Sekunden Gesamtgenerierungszeit liegt die Dreiecksverteilung im Mittelfeld, denn die maximale Generierungszeit von 33,469 Sekunden erfordert die Simulation mit der Gleichverteilung (vgl. Tabelle 26).

In Abbildung 64 befinden sich die Box-Plot-Diagramme für die Tourenlängen der drei Generierungsverfahren. Aus den Diagrammen sind die unterschiedlichen Werte für den Mittelwert und den Median ersichtlich. Zudem ist sichtbar, dass die Gleichverteilung, aufgrund der Länge der Whisker und der weit gestreuten Ausreißer, die größte Streuung hinsichtlich der Tourenlänge besitzt. Insgesamt erzeugt die Gleichverteilung 90 Ausreißer, von denen 55 untere und 35 obere Ausreißer sind. Diese Werte sind für die Normalverteilung sichtbar kleiner, da die Whisker kürzer sind und die Streuung der Ausreißer enger ist. Durch die Generierung der Kundenstandorte mit der Normalverteilung fallen insgesamt 86 Ausreißer an, die sich in 47 untere und 39 obere Ausreißer aufteilen. Für die Dreiecksverteilung wird durch die weitere Verkleinerung der Whisker und der Dichte der Ausreißer, ein Minimum dieses Testset hinsichtlich der Streuung der Tourenlänge erreicht. Insgesamt entstehen in diesem Generierungsverfahren 91 Ausreißer, wovon 47 obere und 44 untere Ausreißer sind. Damit erzeugt die Normalverteilung für dieses Testset in der Tourenlänge die wenigsten Ausreißer.

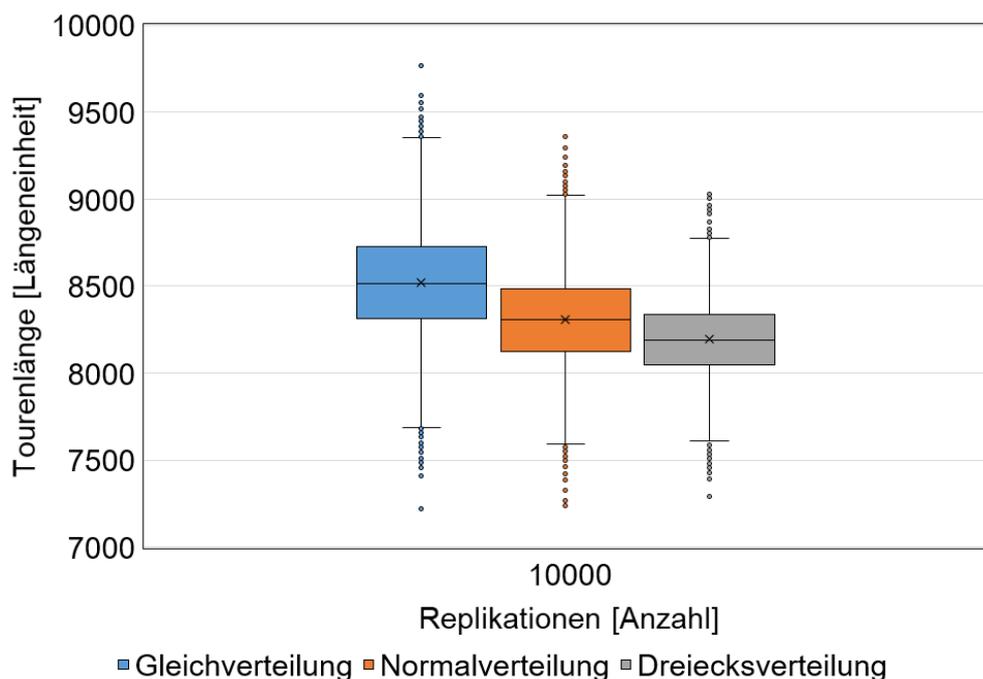


Abbildung 64: Vergleich Testset CMT14 Box-Plots

Die approximierten 95% Konfidenzintervalle der drei Simulationsläufe unterscheiden sich sichtbar in ihrer Breite und Lage (vgl. Abbildung 65). Die maximale Breite stellt das Konfidenzintervall der Gleichverteilung mit 12,92 LE dar. Für die Normalverteilung ergibt sich ein Wert von 11,09 LE und die Breite des Konfidenzintervalls der Dreiecksverteilung ist mit 9,28 LE das Minimum dieses Testsets. Auf die Breite bezogen

liegen die drei Konfidenzintervalle sehr weit auseinander, denn das Konfidenzintervall der Gleichverteilung liegt zwischen 8510 und 8530 LE, das der Normalverteilung zwischen 8300 und 8320 sowie das der Dreiecksverteilung zwischen 8180 und 8200 LE. Damit liegen zwischen allen Konfidenzintervallen mindestens 100 LE, wobei der Abstand zwischen Dreiecks- und Normalverteilung geringer als der zwischen Normal- und Gleichverteilung ist.

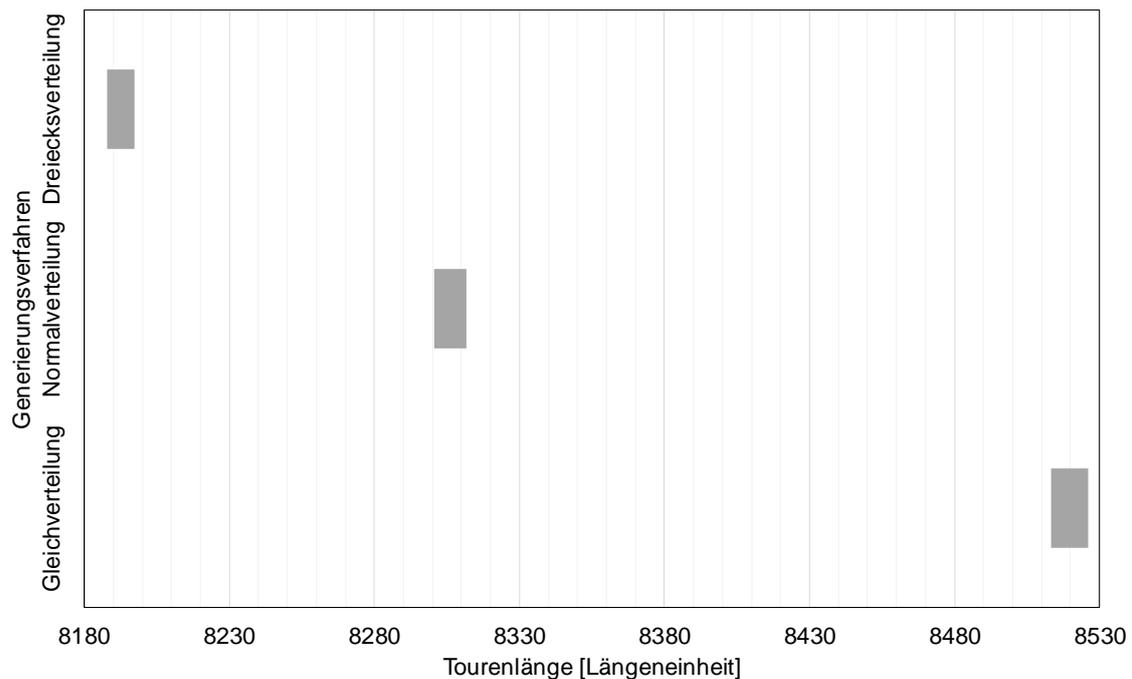


Abbildung 65: Vergleich 95% Konfidenzintervalle Testset CMT14

Aufgrund der geringen Breite des Konfidenzintervalls der Dreiecksverteilung und der Lage kann die Größenordnung der Tourenlänge mit diesem am präzisesten abgeschätzt werden, weswegen es an erster Stelle platziert wird. Die Normalverteilung wird aufgrund der niedrigen Gesamtgenerierungszeit und der zweitniedrigsten Breite des Konfidenzintervalls an zweiter Stelle platziert, wodurch sich die dritte Platzierung für die Gleichverteilung ergibt.

7.1.4 Vergleich Testset C110_1

Der vierte Vergleich in diesem Abschnitt wird mit den Ergebnissen der drei Generierungsverfahren für das Testset C110_1 durchgeführt. Die Simulationen für dieses Testset wurden in den Abschnitten 6.1.3, 6.2.3 sowie 6.3.3 beschrieben. Die Ergebnisse der Simulationen sind in Tabelle 27 dargestellt. Bei näherer Betrachtung ist auffällig, dass

sich sowohl die Werte des Medians als auch des Mittelwerts für die drei Generierungsverfahren voneinander unterscheiden. Für den Mittelwert der Gleichverteilung ergibt sich ein Wert von 472313,07 LE sowie für den Median ein Wert von 472246,2 LE. Diese Werte stellen die Maximalwerte beider Lagemaße dar. Der Mittelwert der Dreiecksverteilung sinkt auf 454653,88 LE sowie der Median auf 454641,92. Auf ihr Minimum sinken die Lagemaße mit der Normalverteilung, für die der Mittelwert 448649,69 LE und der Median 448650,94 LE erreicht.

Tabelle 27: Vergleich Testset C110_1 Ergebnisse

Generierungsverfahren [Name]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
Gleichverteilung	472313,07	472246,20	20079	148,109
Normalverteilung	448649,69	448650,94	2885845	118,984
Dreiecksverteilung	454653,88	454641,92	35928	67,063

In der Gesamtzahl der Generierungsfehler stellen die Generierungsfehler der Gleichverteilung mit 20079 Fehlern das Minimum für dieses Testset dar. Weiterhin erzeugt die Dreiecksverteilung 35928 Fehler. Für die Normalverteilung ergeben sich 2885845 Generierungsfehler, wobei hier wieder zu beachten ist, dass die Gesamtanzahl der Generierungsfehler für die Normalverteilung zusätzlich zu dem Fehler „doppelte Standorte“ auch die Fehler „außerhalb der Stadtgrenzen“ sowie „Berechnungsfehler“ beinhalten. Sowohl die Gleich- als auch die Dreiecksverteilung beinhalten nur den Fehler „doppelte Standorte“. Werden die anderen Fehler für die Normalverteilung aus der Gesamtanzahl der Generierungsfehler herausgerechnet, so ergeben sich 55943 doppelte Standorte, womit dieser Wert das Maximum für den Fehler „doppelte Standorte“ darstellt.

Für alle drei Generierungsverfahren beträgt die Minimale Anzahl der Generierungsfehler in einer Replikation 0 Fehler (vgl. Abbildung 66). Als Maximalwert für die Gleichverteilung ergeben sich in einer Replikation 8 Fehler, im Unterschied dazu liegt dieser Wert sowohl für die Normal- als auch die Dreiecksverteilung bei 16 Fehlern. Weitere Unterschiede zwischen den Generierungsverfahren existieren hinsichtlich der Mediane und Mittelwerte für die Fehleranzahl in einer Replikation. Für die Gleichverteilung ergeben sich als Mittelwert 2,01 Fehler sowie als Median 2 Fehler. Der Mittelwert der Normalverteilung liegt bei 5,59 Fehlern und der Median bei 5 Fehlern.

Diese Werte sinken bei der Dreiecksverteilung auf 3,59 Fehler für den Mittelwert sowie auf 3 Fehler für den Median. Als Generierungszeit benötigt die Gleichverteilung 148,109 Sekunden. Dieser Wert sinkt bei der Generierung mit einer Normalverteilung auf 118,984 Sekunden und erreicht in der Generierung mit einer Dreiecksverteilung sein Minimum von 67,063 Sekunden (vgl. Tabelle 27).

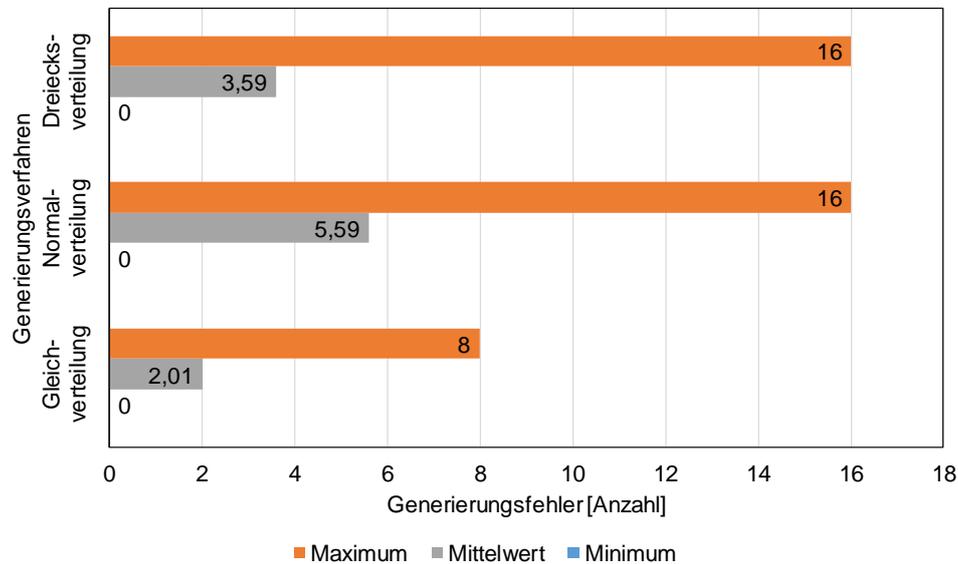


Abbildung 66: Vergleich Generierungsfehler pro Replikation Testset C110_1

Die Box-Plot-Diagramme, die die Tourenlängen der drei Generierungsverfahren enthalten, sind in Abbildung 68 dargestellt. Die unterschiedlichen Werte für die Tourenlängen sind in den Diagrammen deutlich sichtbar. Durch die Länge der Whisker und die weit gestreuten Ausreißer ist die Streuung der Tourenlänge für die Gleichverteilung im Vergleich am größten, zudem sind die Werte für die Tourenlänge größer. Bei der Generierung mit der Gleichverteilung entstehen insgesamt 74 Ausreißer, sowohl 37 obere als auch 37 untere Ausreißer. Bei Verwendung der Dreiecksverteilung sind diese Werte deutlich kleiner, denn die sowohl die Länge der Whisker als auch die Dichte der Ausreißer ist sichtbar geringer. Aus diesem Grund sinkt die Streuung der Tourenlänge und die Werte für die Tourenlänge fallen kleiner aus. In der Dreiecksverteilung fallen insgesamt 77 Ausreißer an, die sich in 34 untere und 43 obere Ausreißer aufteilen. In der Normalverteilung verkleinern sich die Whisker abermals und die Ausreißer liegen dichter beieinander. Damit erreicht die Streuung der Tourenlänge ihr Minimum und die Werte für die Tourenlänge sind nochmals sichtbar kleiner. Für die Ausreißer ergeben sich 31 Untere und 39 Obere, was sich zu insgesamt 70 Ausreißern addiert. Mit diesen Werten erzeugt die Normalverteilung die wenigsten Ausreißer.

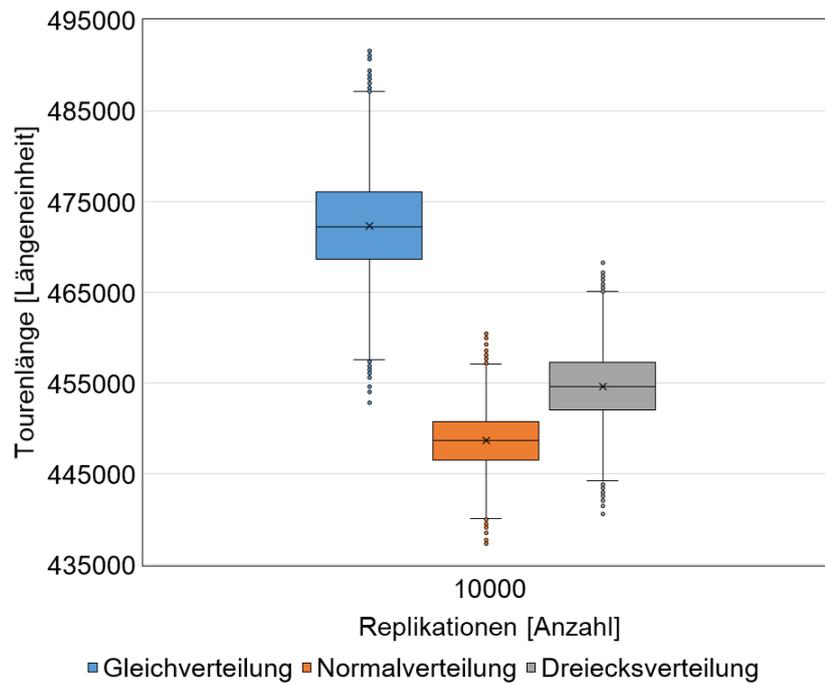


Abbildung 67: Vergleich Testset C110_1 Box-Plots

Die approximierten 95% Konfidenzintervalle unterscheiden sich bezüglich ihrer Breite und Lage deutlich voneinander (vgl. Abbildung 68). Für die Gleichverteilung beträgt die Breite des Konfidenzintervalls 284,63 LE und das Konfidenzintervall befindet sich zwischen 472000 und 473000 LE. Das Konfidenzintervall der Dreiecksverteilung besitzt eine Breite von 234,46 LE und liegt zwischen 454000 und 455000 LE sowie für die Normalverteilung eine Breite von 215,37 LE und eine Lage zwischen 448000 und 449000 LE. Mit diesen Werten gehört die maximale Breite zum Konfidenzintervall der Gleichverteilung und die Breite des Konfidenzintervalls der Normalverteilung stellt ein Minimum dar. Zwischen dem Konfidenzintervall der Normal- und Dreiecksverteilung liegen mindestens 5000 LE. Dieser Abstand ist damit geringer als der zwischen Dreiecksverteilung und Gleichverteilung, der mindestens 17000 LE beträgt. Durch die geringe Breite des Konfidenzintervalls kann die Größenordnung der Tourenlänge mit dem Konfidenzintervall der Normalverteilung am präzisesten abgeschätzt werden und durch die Lage ist der abgeschätzte Wert der Tourenlänge am niedrigsten. Deswegen wird die Normalverteilung an erster Stelle platziert. Aufgrund der geringen Gesamtgenerierungszeit und dem zweitniedrigsten Konfidenzintervall, wird die Dreiecksverteilung an zweiter Stelle platziert. Daraus folgt die dritte Platzierung für die Gleichverteilung.

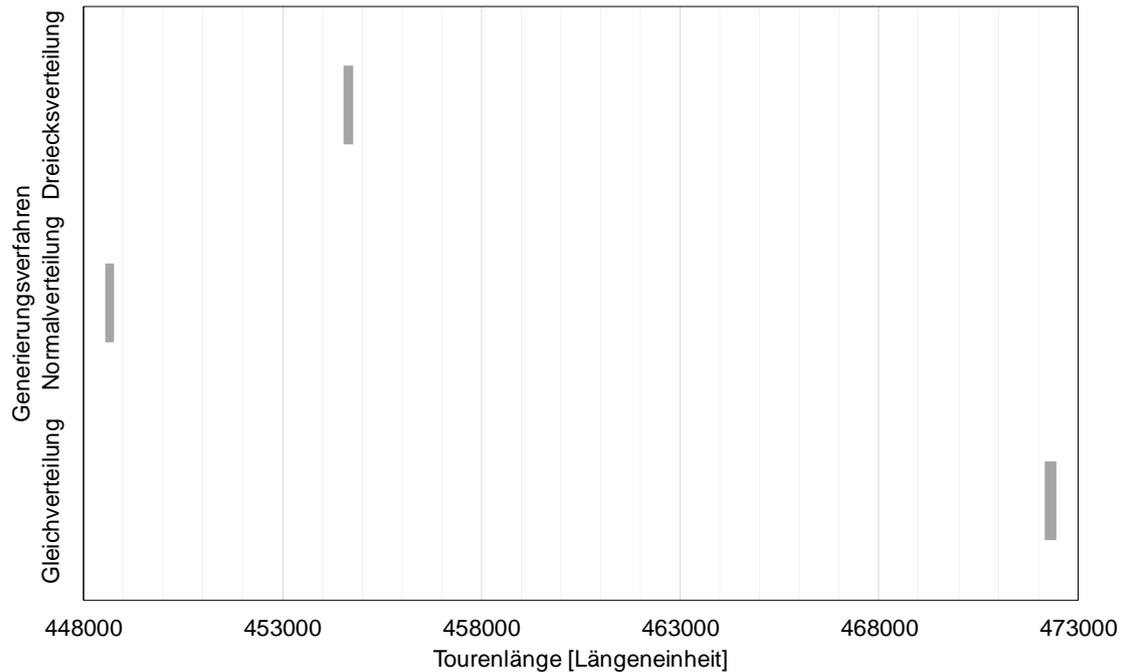


Abbildung 68: Vergleich Testset C110_1 Konfidenzintervalle

7.1.5 Vergleich Testset Antwerp1

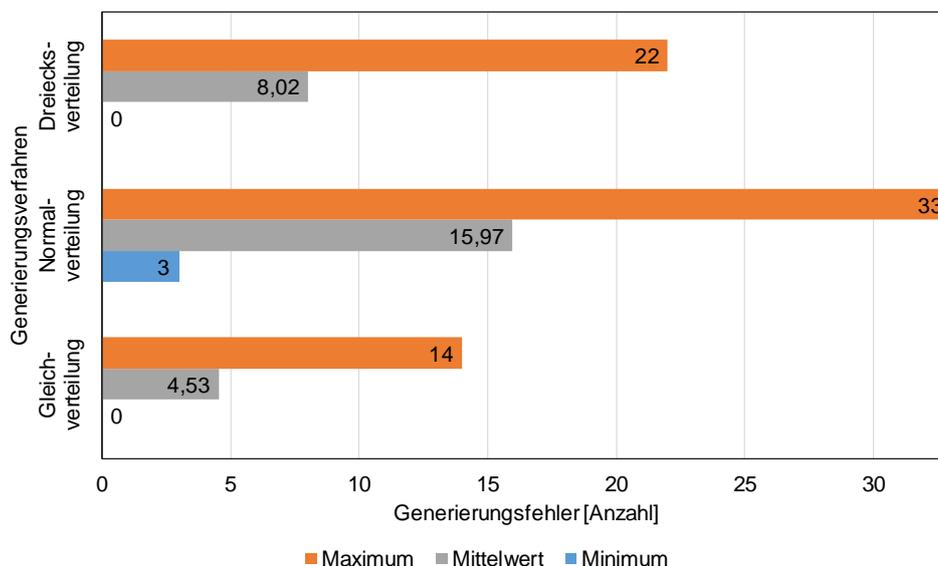
Für den letzten Vergleich in diesem Abschnitt werden die drei Generierungsverfahren für das Testset Antwerp1 hinsichtlich ihrer Ergebnisse ausgewertet. In den Abschnitten 6.1.4, 6.2.4 sowie 6.3.4 sind die zugehörigen Simulationen zu finden. In Tabelle 28 sind die Ergebnisse dieser Simulationen zu finden. Die Betrachtung dieser Werte zeigt auf, dass zwischen den Mittelwerten und Medianen der drei Generierungsverfahren große Unterschiede existieren. Als Mittelwert der Gleichverteilung ergeben sich 11334315,97 LE und für den Median 11333328,41 LE. Diese Werte stellen das Maximum der Lagemaße für dieses Testset. Für die Dreiecksverteilung sinken sowohl der der Mittelwert auf 10912764,99 LE als auch der Median auf 10912207,09 LE. Das Minimum der Lagemaße wird mit der Normalverteilung und einem Mittelwert von 10708670 LE sowie Median von 10708441,89 LE erreicht.

Die großen Unterschiede zwischen den Generierungsverfahren setzen sich auch in den Generierungsfehlern weiter fort. Für die Gleichverteilung werden in der Generierung der Kundenstandorte 45300 Generierungsfehler sowie für die Dreiecksverteilung 80164 Generierungsfehler erzeugt.

Table 28: Vergleich Testset Antwerp1 Ergebnisse

Generierungsverfahren [Name]	Tourenlänge [LE]		Fehler [Anzahl]	Generierungszeit [s]
	Mittelwert	Median	Gesamt	Gesamt
Gleichverteilung	11334315,97	11333328,41	45300	788,047
Normalverteilung	10708670,00	10708441,89	16728024	627,657
Dreiecksverteilung	10912764,99	10912207,09	80164	333,703

Bei der Generierung mit einer Normalverteilung entstehen 16728024 Generierungsfehler, dabei gilt es zu beachten, dass neben dem Fehler „doppelte Standorte“, zusätzlich die Fehler „außerhalb der Stadtgrenzen“ sowie „Berechnungsfehler“ in der Anzahl der Generierungsfehler enthalten sind. In der Dreiecks- und der Gleichverteilung stellt der Fehler „doppelte Standorte“ den einzigen Generierungsfehler dar. Nach der Herausrechnung der anderen Generierungsfehler, ergeben sich für die doppelten Standorte in der Normalverteilung 159730 Fehler, welche die maximale Anzahl der Generierungsfehler sind. Die Minimalwerte in einer Replikation für den Fehler „doppelte Standorte“ liegen für die Gleich- und die Dreiecksverteilung bei 0 Fehlern (vgl. Abbildung 69).

**Abbildung 69:** Vergleich Testset Antwerp1 Fehler „doppelte Standorte“

Für die Normalverteilung beträgt der Minimalwert einer Replikation 3 Fehler. Als Maximalwert der Gleichverteilung werden in einer Replikation 14 Fehler generiert, für

die Normalverteilung sind dies 33 Fehler und für die Dreiecksverteilung maximal 22 Fehler. Bezüglich der Mittelwerte und Mediane existieren zwischen den Generierungsverfahren sichtbare Unterschiede. Der Mittelwert der Gleichverteilung beträgt 4,53 LE und der Median 4 LE. Diese Werte steigen für die Dreiecksverteilung auf 8,02 im Mittelwert und 8 im Median an und erreichen bei der Normalverteilung mit sowohl einem Mittelwert von 15,97 LE als auch einem Median von 16 LE ihr Maximum. Für die Generierungszeit der Gleichverteilung ergeben sich 788,047 Sekunden, die gleichzeitig den Maximalwert der Generierungszeit darstellen. Bei der Generierung der Kundenstandorte mit einer Normalverteilung werden 627,657 Sekunden benötigt. Das Minimum der Generierungszeit ergibt sich für die Dreiecksverteilung und beträgt 333,703 Sekunden (vgl. Tabelle 28).

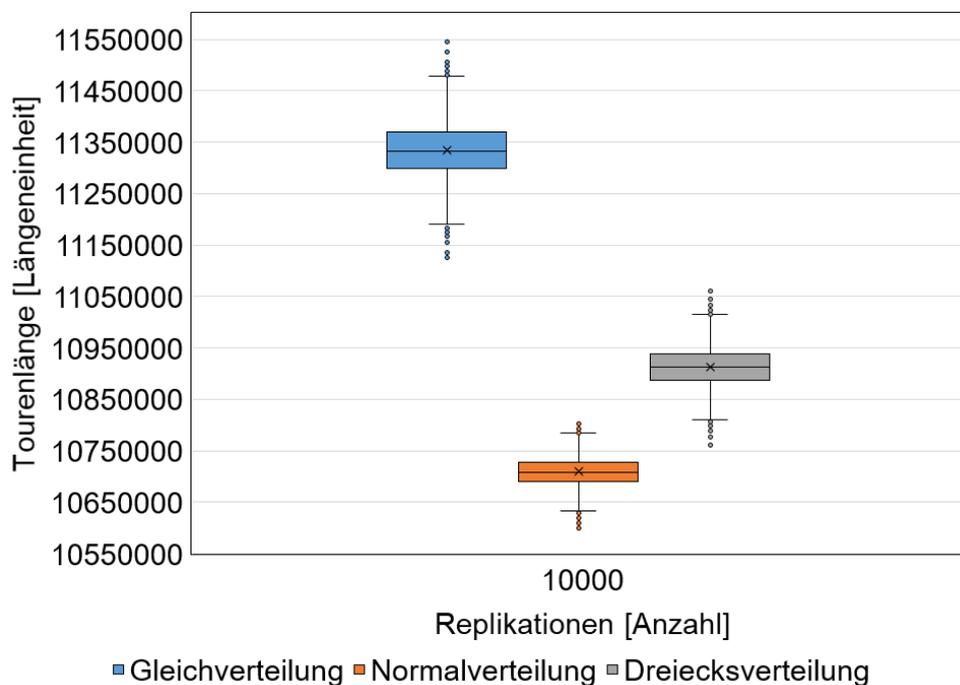


Abbildung 70: Vergleich Testset Antwerp1 Box-Plots

Für die drei Generierungsverfahren sind die zugehörigen Tourenlängen in Abbildung 70 in Box-Plot-Diagramme eingetragen. Aus der Abbildung werden die unterschiedlichen Werte der Tourenlänge ersichtlich. Aufgrund der Länge der Whisker und der weit gestreuten Ausreißer unterliegen die Werte der Tourenlänge in der Gleichverteilung einer großen Streuung. Insgesamt werden in der Generierung der Kundenstandorte mit einer Gleichverteilung 74 Ausreißer erzeugt, von denen 35 obere und 39 untere Ausreißer sind. Der Abbildung ist zu entnehmen, dass die Werte der Tourenlänge bei der Dreiecksverteilung weniger streuen, da die Whisker kleiner und die Ausreißer dichter

zusammen sind. Bei der Generierung mit einer Dreiecksverteilung entstehen insgesamt 83 Ausreißer, wobei diese sich in 40 untere und 43 obere Ausreißer aufteilen. Weiter verringert sich die Streuung der Tourenlänge bei der Normalverteilung, da bei dieser nochmals die Whisker kleiner werden und die Ausreißer dichter zusammen liegen. Für die Normalverteilung ergeben sich insgesamt 74 Ausreißer, sowohl 37 für die Unteren als auch für die Oberen. Es ist ersichtlich, dass durch die Gleich- und Normalverteilung die wenigsten Ausreißer erzeugt werden. Zudem ist in Abbildung 70 aufgrund der unterschiedlichen Lage der Boxplots sichtbar, dass sich für die Werte der Tourenlängen der drei Generierungsverfahren unterschiedliche Werte ergeben.

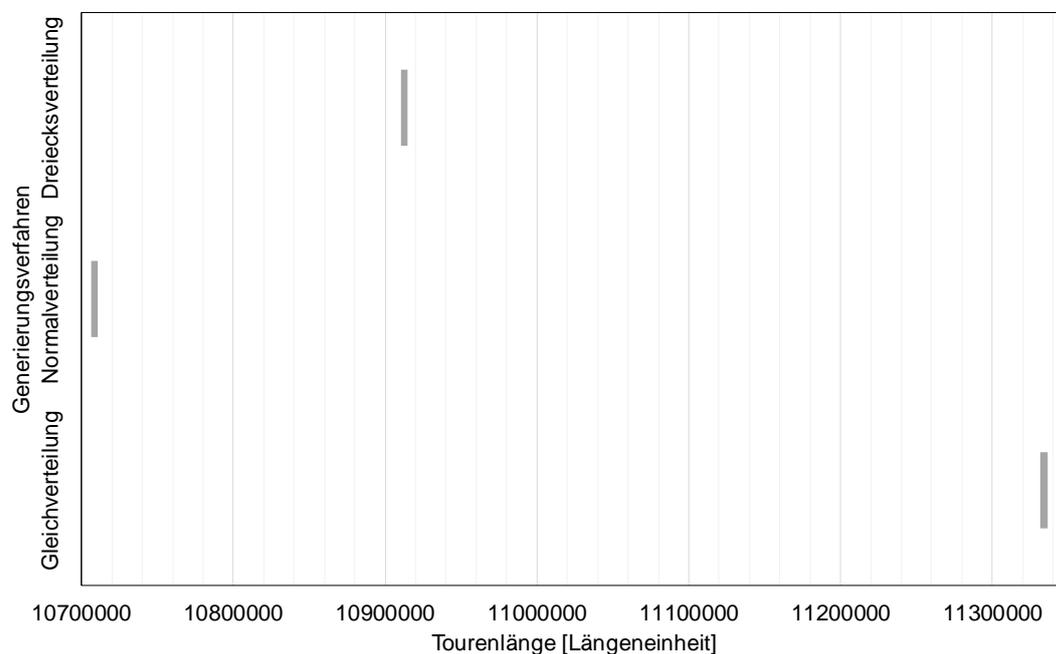


Abbildung 71: Vergleich Testset Antwerp1 Konfidenzintervalle

Die approximierten 95% Konfidenzintervalle weisen hinsichtlich ihrer Breite und Lage sichtbare Unterschiede auf (vgl. Abbildung 71). Für die Gleichverteilung beträgt die Breite des Konfidenzintervalls 4929,35 LE und das Konfidenzintervall befindet sich zwischen 11320000 und 11340000 LE. Bei der Dreiecksverteilung sinken diese Werte auf 4536,31 LE für die Breite und 10900000 bis 10920000 LE für die Lage des Konfidenzintervalls. Diese Werte nehmen mit der Normalverteilung weiter ab, für die Breite ergeben sich 4341,89 LE und für die Lage des Konfidenzintervalls 10700000 bis 10720000 LE. Aufgrund dieser Werte ist die Breite des Konfidenzintervalls der Gleichverteilung ein Maximum und die der Normalverteilung ein Minimum. Für die Lage der Konfidenzintervalle der Gleich- und Dreiecksverteilung ergibt sich ein Mindestabstand von 400000 LE sowie für die der Dreiecks- und Normalverteilung einer

von 180000 LE, welcher aufgrund dieses Wertes geringer ist. Weil die Breite des Konfidenzintervalls der Normalverteilung ein Minimum darstellt wird mit Hilfe dieses Konfidenzintervalls die Größenordnung der Tourenlänge am präzisesten abgeschätzt. Zusätzlich wird die Tourenlänge, aufgrund der Lage des Konfidenzintervalls, den geringsten Wert innerhalb der drei Generierungsverfahren annehmen. Deswegen wird die Normalverteilung an erster Stelle platziert. Die Dreiecksverteilung wird wegen der geringsten Gesamtgenerierungszeit und dem zweitkleinsten Konfidenzintervall an zweiter Stelle platziert. Aus dieser Platzierung ergibt sich die dritte Platzierung für die Gleichverteilung.

7.2 Bewertung der Generierungsverfahren

Als Abschluss dieses Kapitels werden die Generierungsverfahren auf der Basis des Vergleichs aus Abschnitt 7.1 bewertet. Diese Auswertung erfolgt mit den in Abschnitt 5.3 erarbeiteten Kriterien, die in Tabelle 8 als Kriterienkatalog zusammengefasst wurden. Mit Hilfe des Unterkatalogs aus Tabelle 7 werden die Ergebnisse der Generierungsverfahren einzeln für jedes Testset bewertet. Die abschließende Bewertung wird mit dem Kriterienkatalog durchgeführt. Diese Bewertung des Testsets E-n22-k4 unter Berücksichtigung der Bevölkerungsdichte fließt in die Gesamtbewertung der Generierungsverfahren nicht ein, da nur für dieses eine Testset Simulationen durchgeführt wurden, welche die Bevölkerungsdichte berücksichtigen. Damit das sechste Kriterium, die Frage ob die Konfidenzintervalle die wahre Tourenlänge der Testsets überdecken, bewertet werden kann, wurden für jedes Testset die Tourenlängen berechnet. Diese Werte sind in Tabelle 44 im Anhang zu finden.

Die Formel, mit der die Gleichverteilung der Zufallsvariablen berechnet wird, benötigt als Parameter die Intervallgrenzen a und b , zwischen denen die Standorte erzeugt werden (vgl. Abschnitt 6.1). Damit werden für die Gleichverteilung zwei Parameter benötigt. In der Formel für die Normalverteilung der Zufallsvariablen werden als Parameter der Erwartungswert und die Varianz benötigt. Zusätzlich dazu werden die Intervallgrenzen a und b benötigt, um einer Erzeugung der Kundenstandorte außerhalb der Intervallgrenzen vorzubeugen (vgl. Abschnitt 6.2). Daraus ergeben sich für die Normalverteilung insgesamt vier Parameter. Die Formel, die für die Berechnung der dreiecksverteilten Standorte eingesetzt wird, benötigt als Parameter die Intervallgrenzen a und b sowie den wahrscheinlichsten Wert m der Zufallsvariable (vgl. Abschnitt 6.3). Somit ergeben sich

für die Dreiecksverteilung drei Parameter. Aufgrund dieser Werte wird die Gleichverteilung in diesem Kriterium als bestes bewertet und bekommt drei Punkte. Die Dreiecksverteilung als zweitbestes erhält zwei Punkte und die Normalverteilung, die bezüglich dieses Kriteriums das Schlusslicht bildet, erhält einen Punkt.

Alle drei Generierungsverfahren können genutzt werden, um stetige Zufallsvariablen zu erzeugen (vgl. Abschnitt 4.4), weswegen alle Generierungsverfahren für den Einsatz in Simulationsprogrammen geeignet sind. Aufgrund dieses Gleichstands erhalten die Generierungsverfahren hinsichtlich des zweiten Kriteriums jeweils zwei Punkte. Auch die Anzahl der durchgeführten Replikationen ist für alle Generierungsverfahren in allen Testsets identisch. Aus diesem Grund wird das dritte Kriterium aus dem Unterkatalog für die einzelnen Testsets, Tabelle 7, entfernt und nur noch im Kriterienkatalog verwendet. Des Weiteren erhalten wegen dieses Gleichstandes alle Generierungsverfahren erneut zwei Punkte.

Nachfolgend werden die Generierungsverfahren für das Testset E-n22-k4 bewertet. Der Vergleich dazu wurde im Unterabschnitt 7.1.1 durchgeführt. Im vierten Kriterium, der Generierungszeit, erhält die Normalverteilung die beste Bewertung und entsprechend drei Punkte. Obwohl zwischen den Generierungszeiten der Dreiecks- und der Gleichverteilung lediglich ein geringer Unterschied besteht, erhält die Dreiecksverteilung aufgrund der kürzeren Gesamtgenerierungszeit zwei Punkte und die Gleichverteilung einen Punkt. Für das fünfte Kriterium, der Breite der Konfidenzintervalle, ergibt sich ein identisches Bild. Die Normalverteilung bekommt, aufgrund der geringsten Breite, drei Punkte. Dahinter folgen die Dreiecksverteilung mit zwei und die Gleichverteilung mit einem Punkt.

Der wahre Wert der Tourenlänge von 7273,99 LE für das Testset E-n22-k4 (vgl. Tabelle 44) wird von keinem der Konfidenzintervalle überdeckt. Für die Gleichverteilung beträgt das Intervall (4537,78; 4550,12) LE, für die Normalverteilung (4357; 4367,12) LE sowie für die Dreiecksverteilung (4381,78; 4392,72) LE. Mit diesen Werten hat die obere Grenze des Konfidenzintervalls der Gleichverteilung den geringsten Abstand zur wahren Tourenlänge und erhält somit drei Punkte. Die Dreiecksverteilung erhält zwei Punkte und die Normalverteilung, aufgrund der größten Abstand zur wahren Tourenlänge, einen Punkt. Im Kriterium der Gesamtanzahl der Fehler erhält die Gleichverteilung wegen der geringsten Anzahl an Fehlern drei Punkte, die Dreiecksverteilung zwei Punkte sowie die Normalverteilung einen Punkt. Für die Normalverteilung ergeben sich im achten

Kriterium, der Anzahl erzeugter Ausreißer, drei Punkte, für die Gleichverteilung zwei Punkte und für die Dreiecksverteilung ein Punkt. Aufgrund der Platzierung, die im Unterabschnitt 7.1.1 getätigt wurde, erhält die Normalverteilung im achten Kriterium drei Punkte, die Dreiecksverteilung zwei Punkte sowie die Gleichverteilung einen Punkt. Die vollständige Bewertung dieses Testset ist in Tabelle 29 eingetragen. Nach dieser Auflistung ist die Normalverteilung für die Generierung von Kundenstandorten mit dem Kriterienkatalog aus Abschnitt 5.3 am besten bewertet. Für die Gleich- und die Normalverteilung ergibt sich dieselbe Punktzahl.

Tabelle 29: Bewertung Testset E-n22-k4

Kriterium	Gleichverteilung [Punktzahl]	Normalverteilung [Punktzahl]	Dreiecksverteilung [Punktzahl]
Generierungszeit	1	3	2
Breite der Konfidenzintervalle	1	3	2
Konfidenzintervall überdeckt die Tourenlänge des Testsets	3	1	2
Gesamtanzahl der Fehler	3	1	2
Anzahl erzeugter Ausreißer	2	3	1
Platzierung des Verfahrens	1	3	2
Gesamtpunktzahl	11	14	11

Im Folgenden werden Generierungsverfahren anhand ihrer Ergebnisse für das Testset E-n22-k4 unter Berücksichtigung der Bevölkerungsdichte bewertet. Der Vergleich dieser Ergebnisse erfolgte in Unterabschnitt 7.1.2. Die Generierungszeit ist für die Normalverteilung am geringsten. Deswegen erhält diese im vierten Kriterium, der Bewertung der Generierungszeit, drei Punkte. Des Weiteren erhält die Dreiecksverteilung zwei Punkte und die Gleichverteilung einen Punkt. Im fünften Kriterium, der Breite der Konfidenzintervalle, wird die Dreiecksverteilung am besten bewertet und erhält damit drei Punkte. Entsprechend erhält die Normalverteilung zwei Punkte und die Gleichverteilung, wegen des größten Intervalls am schlechtesten bewertet, einen Punkt.

Als wahre Tourenlänge ergibt sich ein Wert von 7273,99 LE (vgl. Tabelle 44). Dieser Wert ist nicht im Konfidenzintervall der Gleichverteilung mit (7221,36;7227,02) LE, nicht im Konfidenzintervall der Normalverteilung mit (7215,76;7220,32) LE sowie nicht im Konfidenzintervall der Dreiecksverteilung mit (7214,27;7218,7) LE enthalten.

Aufgrund dieser Werte wird der geringste Abstand der oberen Grenze zur wahren Tourenlänge mit dem Konfidenzintervall der Gleichverteilung erreicht. Deswegen erhält die Gleichverteilung drei Punkte, die Normalverteilung entsprechend zwei Punkte und die Dreiecksverteilung einen Punkt. Bezüglich des Kriteriums der Gesamtanzahl der Fehler wird die Gleichverteilung am besten bewertet und bekommt damit drei Punkte, die Dreiecksverteilung zwei Punkte, sowie die Normalverteilung einen Punkt. Dieses Bild wird im achten Kriterium, der Anzahl erzeugter Ausreißer, fortgesetzt, denn die Gleichverteilung wird wiederum als Bestes bewertet und erhält damit drei Punkte. Die Normalverteilung bekommt zwei Punkte sowie die Dreiecksverteilung einen Punkt. Im Vergleich aus Unterabschnitt 7.1.2 werden die Normal- und die Dreiecksverteilung gleich platziert, weswegen beide Generierungsverfahren im neunten Kriterium jeweils 2,5 Punkte erhalten. Die Gleichverteilung bekommt aufgrund des dritten Platzes einen Punkt. In Tabelle 30 ist die gesamte Bewertung aufgelistet. Im Unterschied zur Bewertung ohne Berücksichtigung der Bevölkerungsdichte ergibt sich lediglich, dass die Gleich- und die Dreiecksverteilung nicht gleichermaßen zu bewerten sind, sondern die Gleichverteilung eine bessere Bewertung als die Dreiecksverteilung erfährt. Zudem ist anzumerken, dass die Konfidenzintervalle der drei Generierungsverfahren für 10000 Replikationen die wahre Tourenlänge nicht enthalten. Demgegenüber ist dies hinsichtlich der Box-Plots für 10000 Replikationen, siehe Abbildung 61, für alle drei Generierungsverfahren der Fall.

Tabelle 30: Bewertung Testset E-n22-k4 mit Berücksichtigung der Bevölkerungsdichte

Kriterium	Gleichverteilung [Punktzahl]	Normalverteilung [Punktzahl]	Dreiecksverteilung [Punktzahl]
Generierungszeit	1	3	2
Breite der Konfidenzintervalle	1	2	3
Konfidenzintervall überdeckt die Tourenlänge des Testsets	3	2	1
Gesamtanzahl der Fehler	3	2	1
Anzahl erzeugter Ausreißer	3	2	1
Platzierung des Verfahrens	1	2,5	2,5
Gesamtpunktzahl	12	13,5	10,5

Die Bewertung der Generierungsverfahren im Hinblick auf das Testset CMT14 wird nachfolgend durchgeführt, wobei der zugehörige Vergleich im Unterabschnitt 7.1.3 durchgeführt wurde. Im Kriterium der Generierungszeit wird die Normalverteilung am

besten bewertet, da sie die geringste Gesamtgenerierungszeit benötigt. Deswegen bekommt die Normalverteilung drei Punkte, die Dreiecksverteilung 2 Punkte sowie die Gleichverteilung einen Punkt. Hinsichtlich des fünften Kriteriums erhält die Dreiecksverteilung drei Punkte, da diese das Konfidenzintervall mit der geringsten Breite hervorbringt. Die Normalverteilung erreicht zwei Punkte und die Gleichverteilung wiederum einen Punkt. Für die wahre Tourenlänge des Testsets CMT14 ergeben sich 8424,84 LE. Dieser Wert wird vom Konfidenzintervall der Gleichverteilung mit der Dimension (8513,17; 8526,09) LE übertroffen. Sowohl das Konfidenzintervall der Normalverteilung mit (8300,74; 8311,83) LE als auch das Konfidenzintervall der Dreiecksverteilung mit (8187,92; 8197,2) LE liegen unterhalb der wahren Tourenlänge. Weil die obere Grenze des Konfidenzintervalls den kürzesten Abstand zur wahren Tourenlänge hat, erhält die Gleichverteilung drei Punkte. Aufgrund des zweit niedrigsten Abstands erhält die Normalverteilung zwei Punkte, womit die Dreiecksverteilung mit dem weitesten Abstand einen Punkt bekommt.

Hinsichtlich des Kriteriums der Gesamtanzahl der Fehler wird die Gleichverteilung aufgrund der niedrigsten Gesamtanzahl mit drei Punkten bewertet. Darauf folgt die Dreiecksverteilung mit zwei Punkten sowie die Normalverteilung mit einem Punkt. Aufgrund der niedrigsten Anzahl an Ausreißern wird die Normalverteilung mit drei Punkten bezüglich des achten Kriteriums bewertet. Weiterhin ergeben sich für die Gleichverteilung zwei Punkte sowie ein Punkt für die Dreiecksverteilung. Aus der Platzierung der Generierungsverfahren, die im Unterabschnitt 7.1.3 vorgenommen wurde, ergeben sich für die Dreiecksverteilung drei Punkte, für die Normalverteilung zwei Punkte sowie die Gleichverteilung ein Punkt. Wie in der Auflistung in Tabelle 31 sichtbar ist, wird die Normalverteilung mit dem Unterkatalog aus Abschnitt 5.3 am besten bewertet, wobei die Ergebnisse jeweils einen Punkt auseinander liegen. Für die Dreiecksverteilung ergibt sich damit ein zweiter Platz sowie für die Gleichverteilung der dritte Platz. Obwohl die wahre Tourenlänge nicht von einem der drei Konfidenzintervalle überdeckt wird, ist der wahre Wert dennoch in den Box-Plots für 10000 Replikationen enthalten. In den Box-Plots der Gleich- sowie Normalverteilung liegt die wahre Tourenlänge zudem innerhalb der Box.

Table 31: Bewertung Testset CMT14

Kriterium [Name]	Gleichverteilung [Punktzahl]	Normalverteilung [Punktzahl]	Dreiecksverteilung [Punktzahl]
Generierungszeit	1	3	2
Breite der Konfidenzintervalle	1	2	3
Konfidenzintervall überdeckt die Tourenlänge des Testsets	3	2	1
Gesamtanzahl der Fehler	3	1	2
Anzahl erzeugter Ausreißer	2	3	1
Platzierung des Verfahrens	1	2	3
Gesamtpunktzahl	11	13	12

Die Generierungsverfahren werden im Folgenden bezüglich des Testsets C110_1 ausgewertet. Zudem erfolgt auch die Bewertung hinsichtlich der Testsets R110_1 und RC110_1. Für diese Testsets ergeben sich aufgrund der identischen Parametrisierung keine unterschiedlichen Werte bezüglich der Kriterien vier, fünf sowie sieben bis neun. In Kriterium sechs ändert sich hingegen der Wert der wahren Tourenlänge, da die Kundenstandorte innerhalb der drei Testsets variieren. Deswegen wird das sechste Kriterium für jedes genannte Testset bewertet. Es werden somit im sechsten Kriterium dreimal sechs, insgesamt also 18, Punkte vergeben. Der durchgeführte Vergleich für dieses Testset ist im Unterabschnitt 7.1.4 zu finden. Im Kriterium der Generierungszeit erhält die Dreiecksverteilung drei Punkte, weil diese die kürzeste Gesamtgenerierungszeit beansprucht. Für die Normalverteilung ergeben sich zwei Punkte sowie die Gleichverteilung ein Punkt. Die niedrigste Breite eines Konfidenzintervalls erreicht die Normalverteilung, weswegen sie im fünften Kriterium drei Punkte erhält. Dementsprechend wird die Dreiecksverteilung mit zwei Punkten sowie die Gleichverteilung mit einem Punkt bewertet.

Die wahre Tourenlänge beträgt für das Testset C110_1 466386,6 LE, für das Testset R110_1 470940,64 LE sowie für das Testset RC110_1 488349,06 LE. Keines der Konfidenzintervalle der drei Generierungsverfahren überdeckt eine der wahren Tourenlängen. Auch wenn die Werte des Konfidenzintervalls der Gleichverteilung mit der Dimension (472170,75; 472455,38) LE in der Nähe der wahren Tourenlängen liegen, übersteigt dieses Konfidenzintervall bezüglich der Werte die wahre Tourenlänge des Testsets C110_1 und unterbietet die Werte der Testsets R110_1 und RC110_1. Sowohl

das Konfidenzintervall der Normalverteilung mit der Dimension (448542,01; 448757,38) LE als auch das Konfidenzintervall der Dreiecksverteilung mit der Dimension (454536,65; 454771,11) LE sind kleiner als die wahren Tourenlängen der drei Testsets. Die obere Grenze des Konfidenzintervalls der Gleichverteilung besitzt zu den drei Tourenlängen den geringsten Abstand, weswegen die Gleichverteilung insgesamt neun Punkte erhält. Für die Dreiecksverteilung ergeben sich sechs Punkte sowie für die Normalverteilung drei Punkte.

Aufgrund der geringen Gesamtanzahl der Fehler erreicht die Gleichverteilung im siebten Kriterium drei Punkte. Somit erhält die Dreiecksverteilung zwei Punkte und die Normalverteilung einen Punkt. Im achten Kriterium wird die Normalverteilung wegen der geringen Anzahl an Ausreißern am besten bewertet und bekommt damit drei Punkte. An die Gleichverteilung werden zwei Punkte sowie an die Dreiecksverteilung ein Punkt vergeben. Durch die Platzierung aus dem Vergleich in Unterabschnitt 7.1.4 erhält die Normalverteilung, aufgrund ihres ersten Platzes, drei Punkte, die Dreiecksverteilung zwei Punkte und die Gleichverteilung einen Punkt. Aufgrund dieser Bewertung wird die Gleichverteilung insgesamt am besten bewertet, gefolgt von der Dreiecksverteilung (vgl. Tabelle 32). Das Schlusslicht bildet die Normalverteilung. Im Ganzen betrachtet liegen die einzelnen Generierungsverfahren jeweils einen Punkt auseinander. Die wahre Tourenlänge wird von keinem der Konfidenzintervalle überdeckt, jedoch liegt die wahre Tourenlänge des Testsets R110_1 innerhalb der Box des Box-Plots der Gleichverteilung (vgl. Abbildung 19). Mit Hilfe des unteren Whiskers des gleichen Generierungsverfahrens wird auch die wahre Tourenlänge des Testsets C110_1 abgedeckt. Das Testset RC110_1 liegt jedoch nur in den oberen Ausreißern des Box-Plots der Gleichverteilung. Dies ist auch für das Testset C110_1 und die oberen Ausreißer des Box-Plots der Dreiecksverteilung der Fall (vgl. Abbildung 43).

Tabelle 32: *Bewertung Testset C110_1*

Kriterium [Name]	Gleichverteilung [Punktzahl]	Normalverteilung [Punktzahl]	Dreiecksverteilung [Punktzahl]
Generierungszeit	1	2	3
Breite der Konfidenzintervalle	1	3	2
Konfidenzintervall überdeckt die Tourenlänge des Testsets	9	3	6
Gesamtanzahl der Fehler	3	1	2
Anzahl erzeugter Ausreißer	2	3	1
Platzierung des Verfahrens	1	3	2
Gesamtpunktzahl	17	15	16

Abschließend erfolgt die Bewertung der Generierungsverfahren bezüglich des Testsets Antwerp1, dessen Vergleich im Unterabschnitt 7.1.5 zu finden ist. Im vierten Kriterium, der Generierungszeit, wird die Dreiecksverteilung am besten bewertet und erhält damit drei Punkte. Die Normalverteilung bekommt zwei Punkte sowie die Gleichverteilung einen Punkt. Aufgrund der niedrigsten Breite für das Konfidenzintervall wird die Normalverteilung im fünften Kriterium mit drei Punkten bewertet. Dementsprechend erhält die Dreiecksverteilung zwei Punkte und die Gleichverteilung einen Punkt. Die wahre Tourenlänge des Testsets Antwerp1 beträgt 10967935,9 LE (vgl. Tabelle 44), wobei dieser Wert von keinem der drei Konfidenzintervalle überdeckt wird. Mit dem Konfidenzintervall der Gleichverteilung mit der Dimension (11331851,3; 11336780,65) LE wird die wahre Tourenlänge übertroffen, die Konfidenzintervalle sowohl der Normalverteilung mit der Dimension (10706499,05; 10710840,94) LE als auch der Dreiecksverteilung mit der Dimension (10910496,84; 10915033,15) LE liegen unterhalb dieses Wertes. Die geringste Abweichung der oberen Grenze des Konfidenzintervalls zur wahren Tourenlänge hat die Dreiecksverteilung, weswegen diese mit drei Punkten bewertet wird. Für die Normalverteilung ergeben sich zwei Punkte sowie für die Gleichverteilung ein Punkt.

Aufgrund der geringsten Gesamtanzahl an Generierungsfehlern erhält die Gleichverteilung drei Punkte. Die Dreiecksverteilung bekommt zwei Punkte sowie die Normalverteilung einen Punkt. Sowohl für die Normal- als auch die Gleichverteilung werden im achten Kriterium 2,5 Punkte vergeben, da beide Generierungsverfahren die identische Anzahl an Ausreißern erzeugen. Aufgrund dessen erhält die

Dreiecksverteilung einen Punkt. Aus der in Unterabschnitt 7.1.5 vorgenommenen Platzierung wird die Normalverteilung im neunten Kriterium mit drei Punkten, die Dreiecksverteilung zwei Punkten sowie die Gleichverteilung mit einem Punkt bewertet. Anhand der Auflistung der Bewertung in Tabelle 33 ist erkennbar, dass die Normalverteilung mit 13,5 Punkten knapp als Bestes bewertet wird, dahinter folgt die Dreiecksverteilung mit 13 Punkten. Die Gleichverteilung bildet mit 9,5 Punkten das Schlusslicht. Keines der Konfidenzintervalle überdeckt die wahre Tourenlänge dieses Testsets, jedoch ist die wahre Tourenlänge im oberen Whisker der Dreiecksverteilung enthalten (vgl. Abbildung 70).

Tabelle 33: Bewertung Testset Antwerp1

Kriterium [Name]	Gleichverteilung [Punktzahl]	Normalverteilung [Punktzahl]	Dreiecksverteilung [Punktzahl]
Generierungszeit	1	2	3
Breite der Konfidenzintervalle	1	3	2
Konfidenzintervall überdeckt die Tourenlänge des Testsets	1	2	3
Gesamtanzahl der Fehler	3	1	2
Anzahl erzeugter Ausreißer	2,5	2,5	1
Platzierung des Verfahrens	1	3	2
Gesamtpunktzahl	9,5	13,5	13

Die Gesamtbewertung der Generierungsverfahren in Tabelle 34 zeigt, dass die Normalverteilung insgesamt am besten bewertet wird. Darauf folgt die Dreiecksverteilung mit 0,5 Punkten Abstand sowie die Gleichverteilung mit 3,5 Punkten Abstand von der Dreiecksverteilung. Die Ergebnisse der Bewertung stellen insofern eine Überraschung dar, dass die Normalverteilung sich lediglich im Testset CMT14 mit dem zugehörigen Box-Plot der wahren Tourenlänge annähert. Dasselbe gilt für die Dreiecksverteilung, welche die wahre Tourenlänge in den Testsets CMT14, C110_1 und Antwerp1 nur durch ihre Whisker überdeckt. Die Gleichverteilung hingegen überdeckt mit der zugehörigen Box der Box-Plot-Diagramme die wahre Tourenlänge in den Testsets CMT14 und R110_1 sowie mit dem unteren Whisker im Falle des Testsets C110_1 und mit dem oberen Whisker im Falle des Testsets RC110_1.

Table 34: Gesamtbewertung der Generierungsverfahren

Kriterium [Name]	Gleichverteilung [Punktzahl]	Normalverteilung [Punktzahl]	Dreiecksverteilung [Punktzahl]
Benötigte Parameter	3	1	2
Nutzung in Simulationsprogrammen	2	2	2
Anzahl Replikationen	2	2	2
Generierungszeit	4	10	10
Breite der Konfidenzintervalle	4	11	9
Konfidenzintervall überdeckt die Tourenlänge des Testsets	16	8	12
Gesamtanzahl der Fehler	12	4	8
Anzahl erzeugter Ausreißer	8,5	11,5	4
Platzierung des Verfahrens	4	11	9
Gesamtpunktzahl	55,5	60,5	58

Da das Testset CMT14 gruppierte Kundenstandorte und im Unterschied dazu das Testset R110_1 gleichverteilte Kundenstandorte enthält, ist festzuhalten, dass mit dem Generierungsverfahren der Gleichverteilung neben der Abbildung gleichverteilter Standorte auch gruppierte Standorte abgebildet werden können. Dies wird dadurch unterstützt, dass das Testset C110_1 gruppierte und das Testset RC110_1 sowohl gruppierte als auch gleichverteilte Kundenstandorte enthält. Alle für die Gleichverteilung angesprochenen Testsets besitzen zudem eine hohe Dichte der Kundenstandorte. Darin könnte ein zusätzlicher Grund für die gute Abbildung der angesprochenen Testsets liegen. Das Testset Antwerp1 wird von der Gleichverteilung hinsichtlich der wahren Tourenlänge schlechter abgebildet als durch die Dreiecks- oder Normalverteilung. Dieses Testset besitzt jedoch eine kleinere Dichte an Kundenstandorten als die vorher angesprochenen Testsets. Aus diesem Grund könnte die Generierung mit einer Gleichverteilung von einer Mindestdichte abhängig sein, wenn die wahre Tourenlänge angenähert werden soll. Die Dreiecksverteilung nähert die wahre Tourenlänge ausschließlich für Testsets an, die nur gruppierte Kundenstandorte enthalten. Durch die Normalverteilung wird die wahre Tourenlänge nur in dem Testset angenähert, welches die größte Dichte an Kundenstandorten besitzt. Daher ist es möglich, dass die Dreiecksverteilung geeignet ist, um Stadtgebiete mit gruppierten Kundenstandorten abzubilden, und die Normalverteilung für Stadtgebiete mit einer hohen Dichte an

Kundenstandorten geeignet ist. Die Gleichverteilung hingegen scheint sowohl für die Abbildung von gleichverteilten als auch gruppierten Standorten geeignet zu sein, sofern diese eine Minstdichte besitzen.

Die wahre Tourenlänge wird im Testset E-n22-k4 von keinem der Generierungsverfahren angenähert. Dies könnte darin begründet liegen, dass sich die Kundenstandorte nicht auf der gesamten Stadtfläche befinden, sondern nur in der oberen rechten Ecke. Findet dahingehend eine Anpassung der Parametrisierung der Generierungsverfahren statt, so ist der wahre Wert der Tourenlänge in allen Box-Plots enthalten. Durch diese Anpassung konnte die Qualität der drei Generierungsverfahren hinsichtlich der Abbildung der wahren Tourenlänge erhöht werden.

Bei dieser Bewertung gilt es jedoch einschränkend zu beachten, dass die verwendeten Testsets nur einen kleinen Ausschnitt betragen und nicht alle in der Realität verfügbaren Testszenarien abdecken. Damit gilt die getroffene Bewertung nur für die genutzten Testsets und kann nicht ohne weiteres auf andere Szenarien übertragen werden.

Erfolgt die Bewertung der Generierungsverfahren, ohne dass die wahre Tourenlänge Berücksichtigung findet, müssen das fünfte, sechste und neunte Kriterium aus dem Kriterienkatalog entfernt werden. Durch diesen Schritt ergibt sich eine neue Rangfolge: Die Gleichverteilung ist mit 31,5 Punkten das bestbewertete Generierungsverfahren. Dahinter folgt die Dreiecksverteilung mit 30 Punkten und die Normalverteilung mit 28 Punkten. Daraus ergibt sich, dass die Gleichverteilung zur Generierung von Kundenstandorten eingesetzt werden sollte, wenn keine Informationen über die Kundenstandorte existieren und die Anforderungen, denen dies in der vorliegenden Arbeit erstellten und genutzten Kriterienkatalogs entsprechen.

8. Zusammenfassung und Ausblick

Die zunehmende Urbanisierung führt unter anderem zu einem weiteren Anstieg des Verkehrs in den Innenstädten. Weiterhin ist die Politik darin bestrebt die Bürger in den Innenstädten geringeren CO₂-Emissionen auszusetzen. Um das Verkehrsaufkommen und die CO₂-Emissionen zu reduzieren werden neuartige Versorgungskonzepten entwickelt, welche mit Hilfe von Simulationsmodellen bezüglich ihrer Effektivität und Umweltfreundlichkeit untersucht werden. Grundlage für diese Simulationsmodelle sind quantitativ und qualitativ hochwertige Eingangsdaten, zu denen auch solch sensible Daten wie die Kundenstandorte zählen. Die vorliegende Arbeit befasste sich mit der Problematik des Fehlens derartiger Datensätze und der Entwicklung geeigneter Verfahren, um stattdessen alternative Kundenstandorte zu generieren.

Basierend auf dem bisherigen Stand der Forschung wurde deutlich, dass in der wissenschaftlichen Literatur die zufällige Generierung von Standorten zwar in vielseitiger Hinsicht Verwendung findet, eine Überprüfung dahingehend, ob mit Hilfe dieser generierten Standorte reale Standorte realitätsnah abgebildet werden können, bisher jedoch nicht erfolgt ist.

Aus diesem Grund stellte das Ziel dieser Arbeit die Aufarbeitung dieses Versäumnisses dar. Zu diesem Zweck erfolgte die Entwicklung eines Testszenarios, innerhalb dessen ein fiktiver KEP-Dienstleister in Unkenntnis der Standorte seiner Kunden die Erzeugung von Kundenstandorten durch geeignete Generierungsverfahren beabsichtigte. Insbesondere aufgrund des großen wirtschaftlichen und ökologischen Einflusses der Fahrstrecke auf die Effizienz des Unternehmens, war das Augenmerk auf die Auswirkungen der Generierungsverfahren auf die Gesamtfahrstrecke – hier als Tourenlänge bezeichnet – zu richten. Auf Grundlage des Testszenarios erfolgte der Aufbau eines Simulationsmodells, in welchem sowohl die x- als auch die y-Koordinaten der Kundenstandorte als Zufallsvariablen entweder anhand einer Gleich-, einer Normal- oder einer Dreiecksverteilung erzeugt werden. Zur Bewertung der Generierungsverfahren wurde ein geeigneter Kriterienkatalog erstellt und im Anschluss daran mit dem Simulationsmodell Simulationen für verschiedene Stadtgebiete unterschiedlicher Größe und mit wechselnden Anzahlen sowie Verteilungen der Kundenstandorte durchgeführt. Um die Ergebnisse der Simulation mit Realdaten vergleichen zu können, wurden bereits bestehende Testsets aus der Tourenplanung entliehen.

Durch die Auswertung, Bewertung und vergleichende Betrachtung der Simulationsergebnisse ergab sich, dass die Generierung mit einer Normalverteilung oder Dreiecksverteilung deutlich geringere Werte für die Tourenlänge erzeugte als die Gleichverteilung. Jedoch zeigte sich dabei, dass keines der genutzten Generierungsverfahren die Tourenlänge der Testsets präzise abbilden konnte. Zudem stellte sich bei der Gleichverteilung die Vorgabe einer Minstdichte der Kundenstandorte als Notwendigkeit heraus, um solche Werte für die Tourenlängen zu generieren, die sich denen aus den Testsets annähern. Dabei fiel auf, dass sich die Werte aus der Generierung mit einer Normalverteilung nur innerhalb des Testsets mit der höchsten Standortdichte der Tourenlänge des Testsets annäherten. Daneben scheint sich die Dreiecksverteilung hingegen ausschließlich für die Generierung von gruppierten Standorten zu eignen, da sich deren Werte nur in den entsprechenden Testsets der zugehörigen Tourenlänge annäherten. Mit der Tourenlänge des Testsets vergleichbare Ergebnisse ließen sich nur unter Anpassung der drei Generierungsverfahren im Hinblick auf die Dichte der Kundenstandorte erzielen.

Hieran zeigt sich weiterer Forschungsbedarf. Da die sechs Testsets nicht sämtliche realen Testszenarien abdecken, lassen sich die oben getroffenen Aussagen nicht verallgemeinern. Es gilt deswegen zu untersuchen, ob die beschriebenen Einflüsse der Standortdichte sowie der Anordnung der Standorte auf die Generierungsverfahren auch hinsichtlich anderer Testsets zutreffen. Des Weiteren sind neben den drei hier vorgestellten Generierungsverfahren weitere Verfahren auf ihre Eignung zur Kundenstandortgenerierung hin zu untersuchen. Alle in der vorliegenden Arbeit behandelten Generierungsverfahren können zudem zur Generierung von Kundenstandorten in den Simulationsprogrammen AnyLogic und SimChain genutzt werden. Bei diesem Einsatz besteht jedoch die Gefahr, dass die Standorte beispielsweise in Flüssen oder unzugänglichen, nicht erschlossenen, Waldgebieten generiert werden. Diesbezüglich bedarf es der Untersuchung, ob eine entsprechende Modifikation der besprochenen Generierungsverfahren durchführbar ist. Des Weiteren sollte zukünftig untersucht werden, ob ein Austausch der Programmiersprache VBA gegen eine Hochsprache wie C++ zweckmäßig ist, um von deren höheren Ausführungsgeschwindigkeit zu profitieren und nicht weiter auf Sekundärquellen, die beispielsweise die Zufallszahlengeneratoren in VBA übersetzen, angewiesen zu sein.

Erfolgt eine Bewertung der Generierungsverfahren ohne Berücksichtigung der hier ausgewählten Testsets und werden die entsprechenden Kriterien daraufhin aus dem Kriterienkatalog entfernt, so wird die Generierung mit einer Gleichverteilung am besten bewertet. Aus diesem Grund lässt sich die abschließende Aussage treffen, dass der Gleichverteilung zur Generierung von Kundenstandorten der Vorzug zu geben ist, sofern keine Informationen über die Kundenstandorte vorliegen und die eigenen Anforderungen an das Generierungsverfahren mit dem hiesigen Kriterienkatalog übereinstimmen.

Literaturverzeichnis

- Agatz, N.; Campbell, A. M.; Fleischmann, M.; van Nunen, J.; Savelsbergh, M.: Revenue Management Opportunities For Internet Retailers. *Journal of Revenue and Pricing Management* 12 (2013) 2, S. 128–138.
- Aized, T.; Srari, J. S.: Hierarchical Modelling Of Last Mile Logistics Distribution System. *International Journal Of Advanced Manufacturing Technology* 70 (2014) 5–8, S. 1053–1061.
- Amazon: Prime Air, 2016, <https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011>, (zuletzt abgerufen am 13.06.2019).
- AnyLogic: Placing Agents In GIS Space, 2019, <https://help.anylogic.com/index.jsp?topic=%2Fcom.anylogic.help%2Fhtml%2Fgis%2FAgents+Placement.html>, (zuletzt abgerufen am 15.08.2019).
- Aoun, B.; Vatcheva, S.: Shortening The Last Mile: Winning Logistics Strategies In The Race To The Urban Consumer. A Custom Report Compiled By Euromonitor International For Deutsche Post DHL Group, 2018, <https://www.dpdhl.com/content/dam/dpdhl/en/media-center/media-relations/documents/2018/dhl-whitepaper-shortening-the-last-mile.pdf>, (zuletzt abgerufen am 13.06.2019).
- Apparicio, P.; Shearmur, R.; Brochu, M.; Dussault, G.: The Measure Of Distance In A Social Science Policy Context: Advantages And Costs Of Using Network Distances In Eight Canadian Metropolitan Areas. *Journal of Geographic Information and Decision Analysis* 7 (2003) 2, S. 105–131.
- ArcGIS, 2019, <https://www.esri.de/produkte/arcgis>, (zuletzt abgerufen am 06.07.2019).
- Arena, 2019, <https://www.arenasimulation.com/>, (zuletzt abgerufen am 16.07.2019).
- Arnold, F.; Cardenas, I.; Sörensen, K.; Dewulf, W.: Simulation Of B2C E-Commerce Distribution In Antwerp Using Cargo Bikes And Delivery Points. *European Transport Research Review* 10 (2018) 1, Seite 1–13.
- Arnold, F.; Gendreau, M.; Sörensen, K.: Efficiently Solving Very Large-Scale Routing Problems. *Computers & Operations Research* 107 (2019), S. 32–42.
- Arroyo, J. L.; Felipe, À.; Ortuno, M. T.; Tirado, G.: Effectiveness Of Carbon Pricing Policies For Promoting Urban Freight Electrification: Analysis Of Last Mile Delivery In Madrid. *Central European Journal of Operations Research* (2019), S. 1–24.

- Auffermann, C.; Lange, V.: Handelslogistik. In: Arnold, D.; Isermann, H.; Kuhn, A.; Tempelmeier, H.; Furmans, K. (Hrsg.): Handbuch Logistik. 3. Auflage. Berlin Heidelberg: Springer, 2008, S. 525–533.
- Ballou, R. H.: Business Logistics, Supply Chain Management. Planning, Organizing, And Controlling The Supply Chain. 5. Auflage. Upper Saddle River: Pearson Prentice Hall, 2004.
- Banks, J.; Carson II, J. S.; Nelson, B. L.; Nicol, D. M.: Discrete-Event System Simulation. 5. Auflage. Essex: Pearson Education Limited, 2014a.
- Banks, J.; Carson II, J. S.; Nelson, B. L.; Nicol, D. M.: Software. Source Code, 2014b, www.bcn.net, (zuletzt abgerufen am 16.07.2019).
- Bates, O.; Friday, A.; Allen, J.; McLeod, F.; Cherrett, T.; Wise, S.; Piecyk, M.; Piotrowska, M.; Bektas, T.; Nguyen, T.: ICT For Sustainable Last-Mile Logistics: Data, People and Parcels. In: Penzenstadler, B.; Easterbrook, S.; Venters, C.; Ahmed, S. I. (Hrsg.): ICT4S2018. 5th International Conference on Information And Communication Technology For Sustainability 54 (2018), S. 49–67.
- Becker, J.; Winkelmann, A.: Handelscontrolling. Optimale Informationsversorgung mit Kennzahlen. 4. Auflage. Berlin Heidelberg: Springer Gabler, 2019.
- Bertazzi, L.; Maggioni, F.: Solution Approaches For The Stochastic Capacitated Traveling Salesman Location Problem With Recourse. *Journal of Optimization Theory and Applications* 166 (2015) 1, S. 321–342.
- Bertsimas, D. J.: Traveling Salesman Facility Location Problems. *Transportation Science* 23 (1989) 3, S. 184–191.
- Bertsimas, D.; Doan, X. V.; Natarajan, K.; Teo, C.-P.: Models For Minimax Stochastic Linear Optimization Problems With Risk Aversion. *Mathematics Of Operations Research* 35 (2010) 3, S. 580–602.
- Bettstetter, C.; Hartenstein, H.; Pèrez-Costa, X.: Stochastic Properties Of The Random Waypoint Mobility Model. *Wireless Networks* 10 (2004) 5, S. 555–567.
- Bichler, K.; Krohn, R.; Philippi, P.; Schneidereit, F. (Hrsg.): Kompakt - Lexikon Logistik. 2250 Begriffe nachschlagen, verstehen, anwenden. 3. Auflage. Wiesbaden: Springer Gabler, 2017.
- Bierhorst, P.; Knill, E.; Glancy, S.; Zhang, Y.; Mink, A.; Jordan, S.; Rommal, A.; Liu, Y.-K.; Christensen, B.; Nam, S. W.; Stevens, M. J.; Shalm, L. K.: Experimentally Generated Randomness Certified By The Impossibility Of Superluminal Signals. *Nature* 556 (2018) 7700, S. 223–226.

- Blanco, V.; Ben Ali, S. E. H.; Puerto, J.: Minimizing Ordered Weighted Averaging Of Rational Functions With Applications To Continuous Location. *Computers & Operations Research* 40 (2013) 5, S. 1448–1460.
- Bogdanski, R.: Innovationen auf der letzten Meile. Bewertung der Chancen für die nachhaltige Stadtlogistik von morgen. Nachhaltigkeitsstudie 2017 im Auftrag des Bundesverbandes Paket und Expresslogistik e.V. (BIEK), 2017, <https://www.biek.de/download.html?getfile=508>, (zuletzt abgerufen am 06.06.2019).
- Bokrantz, J.; Skoogh, A.; Lämkuill, D.; Hanna, A.; Perera, T.: Data Quality Problems In Discrete Event Simulation Of Manufacturing Operations. *Simulation: Transactions of the Society for Modeling and Simulation International* 94 (2018) 11, S. 1009 - 1025.
- Botchkarev, A.: Assessing Excel VBA Suitability For Monte Carlo Simulation. *Spreadsheets in Education* 8 (2015) 2, S. 1–30.
- Boysen, N.; Briskorn, D.; Fedtke, S.; Schwerdfeger, S.: Drone Delivery From Trucks: Drone Scheduling For Given Truck Routes. *Networks* 72 (2018) 4, S. 506 - 527.
- Box, G. E. P.; Muller, M. E.: A Note On The Generation Of Random Normal Deviates. *The Annals of Mathematical Statistics* 29 (1958) 2, S. 610–611.
- Brandimarte, P.: *Handbook In Monte Carlo Simulation. Applications In Financial Engineering, Risk Management, And Economics*. Hoboken: John Wiley & Sons, 2014.
- Bretzke, W.-R.: *Logistische Netzwerke*. 3. Auflage. Berlin: Springer Vieweg, 2015.
- Brown, J. R.; Guiffrida, A. L.: Carbon Emissions Comparison Of Last Miley Delivery Versus Customer Pickup. *International Journal of Logistics: Research and Applications* 17 (2014) 6, S. 503–521.
- C++: Mersenne Twister. 2019, https://en.cppreference.com/w/cpp/numeric/random/mersenne_twister_engine, (zuletzt abgerufen am 16.07.2019).
- Cappiello, C.; Francalanci, C.; Pernici, B.: Data Quality Assessment From The User's Perspective. In: Naumann, F.; Scannapieco, M.: *Proceedings of the 2004 International Workshop on Information Quality in Information Systems, Paris (Frankreich), 18. Juni 2004*, S. 68–73.
- Cardenas, I.; Beckers, J.; Vanelslander, T.; Verhetsel, A.; Dewulf, W.: Spatial Characteristics Of Failed And Successful E-Commerce Deliveries In Belgian Cities. *Proceedings of the 6th International Conference on Information Systems, Logistics and Supply Chain, Bordeaux (France), 1. - 4. Juni 2016*, S. 1–10.
- Cardenas, I.; Borbon-Galvez, Y.; Verlinden, T.; Van de Voorde, E.; Vanelslander, T.; Dewulf, W.: *City Logistics, Urban Goods Distribution And Last Mile Delivery And*

- Collection. *Competition and Regulation in Network Industries* 18 (2017a) 1 - 2, S. 22–43.
- Cardenas, I.; Beckers, J.; Vanelslander, T.: *E-Commerce Last-Mile In Belgium: Developing An External Cost Delivery Index*. *Research in Transportation Business & Management* 24 (2017b), S. 123–129.
- Carbone, R.; Mehrez, A.: *The Single Facility Minimax Distance Problem Under Stochastic Location of Demand* 26 (1980) 1, S. 113–115.
- Cardeneo, A.: *Kurier-, Express- und Paketdienste*. In: Arnold, D.; Isermann, H.; Kuhn, A.; Tempelmeier, H; Furmans, K. (Hrsg.): *Handbuch Logistik*. 3. Auflage. Berlin Heidelberg: Springer, 2008, S. 782–788.
- Castillo, V. E.; Bell, J. E.; Rose, W. J.; Rodrigues, A. M.: *Crowdsourcing Last Mile Delivery: Strategic Implications And Future Research Directions*. *Journal of Business Logistics* 39 (2018) 1, S. 7–25.
- Chhetri, P. K.: *Predicting Upslope Expansion Of Sub-Alpine Forest In The Makalu Barun National Park, Eastern Nepal, With A Hybrid Cartographic Model*. *Journal of Forestry Research* 29 (2018) 1, S. 129–137.
- Choi, B. K.; Kang, D.: *Modeling And Simulation Of Discrete-Event Systems*. Hoboken: John Wiley & Sons, 2013.
- Christofides, N.; Eilon, S.: *An Algorithm For The Vehicle-Dispatching Problem*. *Operational Research Society* 20 (1969) 3, S. 309–318.
- Christofides, N.; Mingozzi, A.; Toth, P.: *The Vehicle Routing Problem*. In: Christofides, N.; Mingozzi, A.; Toth, P.; Sandi, C. (Hrsg.): *Combinatorial Optimization*. Wiley: Chichester, 1979, S. 315–338.
- Clarke, G.; Thompson, C.; Birkin, M.: *The Emerging Geography Of E-Commerce In British Retailing*. *Regional Studies, Regional Science* 2 (2015) 1, S. 371–391.
- Cleophas, C.; Ehmke, J. F.: *Wann sind Lieferaufträge profitabel? Berücksichtigung des Auftragswertes und der Transportkapazität in der Tourenplanung für die „letzte Meile“ in Ballungsräumen*. *Wirtschaftsinformatik* 54 (2014) 3, S. 173–184.
- Crainic, T. G.; Perboli, G.; Mancini, S.; Tadei, R.: *Two-Echelon Vehicle Routing Problem: A Satellite Location Analysis*. *Procedia - Social and Behavioral Sciences* 2 (2010) 3, S. 5944–5955.
- Dablanc, L.; Rakotonarivo, D.: *The Impact Of Logistics Sprawl: How Does The Location Of Parcel Transport Terminals Affect The Energy Efficiency Of Goods' Movements*

- In Paris And What Can We Do About It?. *Procedia Social and Behavioral Sciences* 2 (2010) 3, S. 6087–6096.
- Davoodi, M.; Mohades, A.: Solving The Constrained Coverage Problem. *Applied Soft Computing* 11 (2011) 1, S. 963–969.
- Davoodi, M.; Mohades, A.; Rezaei, J.: A Genetic Algorithm For The Constrained Coverage Problem. In: Mehnen, J.; Köppen, M.; Saad, A.; Tiwari, A. (Hrsg.): *Applications of Soft Computing. From Theory To Praxis*. In: Kapcprzyk, J. (Hrsg.) *Advances in Intelligent and Soft Computing* 58. Berlin Heidelberg: Springer, 2009.
- Davoodi, M.; Mohades, A.; Rezaei, J.: Solving The Constrained P-Center Problem Using Heuristic Algorithms. *Applied Soft Computing* 11 (2011) 4, S. 3321–3328.
- Deb, S.; Debnath, M. K.; Chakraborty, S.; Weindorf, D. C.; Kumar, D.; Deb, D.; Choudhury, A.: Anthropogenic Impacts On Forest Land Use And Land Cover Change: Modelling Future Possibilities In The Himalayan Terai. *Anthropocene* 21 (2018), S. 32–41.
- Deutsche Post AG: Delivering Tomorrow. *Zukunftstrend Nachhaltige Logistik*. Wie Innovation und „grüne“ Nachfrage eine CO₂-effiziente Branche schaffen, 2010, https://www.post-und-telekommunikation.de/PuT/1Fundus/Dokumente/Studien/Deutsche_Post/Zukunftstrend_nachhaltige_Logistik.pdf, (zuletzt abgerufen am 14.06.2019).
- DHL: Self-Driving Vehicles In Logistics. A DHL Perspective On Implications And Use Cases For The Logistics Industry, 2014, <https://www.logistics.dhl/content/dam/dhl/global/core/documents/pdf/glo-logistics-insights-selfdriving-vehicles.pdf>, (zuletzt abgerufen am 13.06.2019).
- DHL: DHL's Parcelcopter: Changing Shipping Forever, 2018, <https://discover.dhl.com/business/business-ethics/parcelcopter-drone-technology>, (zuletzt abgerufen am 13.06.2019).
- Domschke, W.; Drexl, A.; Klein, R.; Scholl, A.: *Einführung in Operations Research*. 9. Auflage. Berlin Heidelberg: Springer Gabler, 2015.
- DPDHL: Geschäftsbericht 2018. Deutsche Post DHL Group, 2018, <https://www.dpdhl.com/content/dam/dpdhl/de/media-center/investors/documents/geschaeftsberichte/DPDHL-Geschaeftsbericht-2018.pdf>, (zuletzt abgerufen am 07.06.2019).
- Edwards, J.; McKinnon, A.; Cherrett, T.; McLeod, F.; Song, L.: The Impact Of Failed Home Deliveries On Carbon Emissions: Are Collection/ Delivery Points

- Environmentally-Friendly Alternatives?. Proceedings of the 14th Annual Logistics Research Network Conference: Volatile and Fragile Supply Chains, 9. bis 11. September 2009, Cardiff, S. 102–109.
- Edwards, J. B.; McKinnon, A. C.; Cullinane, S. L.: Comparative Analysis Of The carbon Footprints Of Conventional And Online Retailing. *International Journal of Physical Distribution & Logistics Management* 40 (2010) 1/2, S. 103–123.
- Ehmke, J. F.; Mattfeld, D. C.: Vehicle Routing For Attended Home Delivery In City Logistics. *Procedia – Social and Behavioral Sciences* 39 (2012), S. 622–632.
- Esser, K.; Kurte, J.: Digitaler, Effizienter. KEP-Studie 2018 - Analyse des Marktes in Deutschland. Eine Untersuchung im Auftrag des Bundesverbands Paket und Expresslogistik e.V. (BIEK), 2018, <https://www.biek.de/download.html?getfile=1928>, (zuletzt abgerufen am 07.06.2019).
- EU: Weißbuch zum Verkehr. Fahrplan zu einem einheitlichen Europäischen Verkehrsraum - Hin zu einem wettbewerbsorientierten und ressourcenschonenden Verkehrssystem. Luxemburg: Amt für Veröffentlichungen der Europäischen Union, 2011.
- Europäische Kommission: Does Consent Given Before 25 May 2018 Continue To Be Valid Once The GDPR Starts To Apply On 25 May 2018?, 2019, https://ec.europa.eu/info/law/law-topic/data-protection/reform/rules-business-and-organisations/legal-grounds-processing-data/grounds-processing/does-consent-given-25-may-2018-continue-be-valid-once-gdpr-starts-apply-25-may-2018_en, (zuletzt abgerufen am 30.07.2019).
- Eurotransport: Parcellock startet 2019 durch. Erste Paketstationen stehen, 2019, <https://www.eurotransport.de/artikel/parcellock-startet-2019-durch-erste-paketstationen-stehen-10711099.html>, (zuletzt abgerufen am 14.06.2019).
- Fleischmann, B.: Begriffliche Grundlagen. In: Arnold, D.; Isermann, H.; Kuhn, A.; Tempelmeier, H.; Furmans, K. (Hrsg.): *Handbuch Logistik*. 3. Auflage. Berlin Heidelberg: Springer, 2008a, S. 3–12.
- Fleischmann, B.: Systeme der Transportlogistik. In: Arnold, D.; Isermann, H.; Kuhn, A.; Tempelmeier, H.; Furmans, K. (Hrsg.): *Handbuch Logistik*. 3. Auflage. Berlin Heidelberg: Springer, 2008b, S. 12–18.
- Fleischmann, B.; Kopfer, H.; Sürle, C.: Transport Planning For Procurement And Distribution. In: Stadler, H.; Kilger, C.; Meyr, H.: *Supply Chain Management And*

- Advanced Planning. Concepts, Models, Software, And Case Studies. 5. Auflage. Berlin Heidelberg: Springer, 2015, S. 225–240.
- Foley, S. M.; Price, S. J.; Dorcas, M. E.: Nest-Site Selection And Nest Depredation Of Semi-Aquatic Turtles On Golf Courses. *Urban Ecosystems* 15 (2012) 2, S. 489–497.
- Foul, A.: A 1-Center Problem In The Plane With Uniformly Distributed Demand Points. *Operations Research Letters* 34 (2006) 3, S. 264–268.
- Foul, A.: Determining Optimal Straight Line Route Of A Moving Facility On The Plane With Random Demand Points. *Journal of King Saud University – Science* 24 (2012) 1, S. 19–23.
- Franz, C.: Simultane Standort- und Tourenplanung am Beispiel des Lebensmittel-Online-Handels. Dissertation, Universität Duisburg-Essen, 2018.
- Free Pascal: Random. 2019, <https://www.freepascal.org/docs-html/rtl/system/random.html>, (zuletzt abgerufen am 16.07.2019).
- Gabriel, C.; Wittmann, C.; Sych, D.; Dong, R.; Mauerer, W.; Andersen, U. L.; Marquardt, C.; Leuchs, G.: A Generator For Unique Quantum Random Numbers Based On Vacuum States. *Nature Photonics* 4 (2010) 10, S. 711–715.
- Gentle, J. E.: Random Number Generation And Monte Carlo Methods. 2. Auflage. New York: Science+Business Media, 2005.
- Gerdes, J.; Heinemann, G.: Urbane Logistik der Zukunft - ganzheitlich, nachhaltig und effizient. In: Heinemann, G.; Gehrckens, H. M.; Täuber, T.; Accenture GmbH (Hrsg.): *Handel mit Mehrwert. Digitaler Wandel in Märkten, Geschäftsmodellen und Geschäftssystemen*. Wiesbaden: Springer Gabler, 2019, S. 397–422.
- Gerlach, S.: *Computerphysik. Einführung, Beispiele und Anwendungen*. 2. Auflage. Berlin: Springer Spektrum, 2019.
- Gevaers, R.; Van de Voorde, E.; Vanelslander, T.: Characteristics And Typology Of Last-Mile Logistics From An Innovation Perspective In An Urban Context. In: Macharis, C.; Melo, S. (Eds.): *City Distribution And Urban Freight Transport: Multiple Perspectives*, Cheltenham: Edward Elgar Publishing, 2011, S. 56–72.
- Gevaers, R.; Van de Voorde, E.; Vanelslander, T.: Cost Modelling And Simulation Of Last-mile Characteristics In An Innovative B2C Supply Chain Environment With Implications On Urban Areas And Cities. *Procedia – Social and Behavioral Sciences* 125 (2014), S. 398–411.

- Gietz, M.: Tourenplanung. In: Arnold, D.; Isermann, H.; Kuhn, A.; Tempelmeier, H.; Furmans, K. (Hrsg.): Handbuch Logistik. 3. Auflage. Berlin Heidelberg: Springer, 2008, S. 137–152.
- Gißrau, M.; Gereke, T.: Transformation Of Real-Time Reporting Event Data To Long-Term Simulation Models. In: Wenzel, S.; Peter, T. (Hrsg.): Simulation in Produktion und Logistik. Kassel: kassel university press, 2017, S. 393–402.
- Golem: DHL liefert in China per Drohne aus, 2019, <https://www.golem.de/news/ehang-dhl-liefert-in-china-per-drohne-aus-1905-141315.html>, (zuletzt abgerufen am 14.06.2019).
- Gonzalez-Martin, S.; Juan, A. A.; Riera, D.; Elizondo, M. G.; Ramos, J. J.: A Simheuristic Algorithm For Solving The Arc Routing Problem With Stochastic Demands. *Journal of Simulation* 12 (2018) 1, S. 53–66.
- Göpfer, I.: Ein Zukunftsmodell für Handelslogistik im Jahr 2036. In: Göpfert, I. (Hrsg.): Logistik der Zukunft - Logistics For The Future. 8. Auflage. Wiesbaden: Springer Gabler, 2019, S. 233–252.
- Goodman, R. W.: Whatever You Call It, Just Don't Think Of Last-Mile Logistics, *Last. Global Logistics & Supply Chain Strategies* 9 (2005) 12, S. 46–51.
- Groß, J.: Grundlegende Statistik mit R. Eine anwendungsorientierte Einführung in die Verwendung der Statistik Software R. Wiesbaden: Springer Fachmedien, 2010.
- Gudehus, T.: Logistik. Grundlagen - Strategien - Anwendungen. 4. Auflage. Berlin: Springer, 2010.
- Gutenschwager, K.; Rabe, M.; Spieckermann, S.; Wenzel, S.: Simulation in Produktion und Logistik. Grundlagen und Anwendungen. Berlin: Springer Vieweg, 2017.
- Handelsblatt: Die Paketlieferung bis zur Haustür könnte zum Luxus werden, 2019, <https://www.handelsblatt.com/unternehmen/handel-konsumgueter/e-commerce-die-paketlieferung-bis-zur-haustuer-koennte-zum-luxus-werden/24444676.html?ticket=ST-1488445-LsYnfnZ6tvGpuaLUlies-ap2>, (zuletzt abgerufen am 14.06.2019).
- Hofstadler, C.; Kummer, M.: Chancen- und Risikomanagement in der Bauwirtschaft. Für Auftraggeber und Auftragnehmer in Projektmanagement, Baubetrieb und Bauwirtschaft. Berlin: Springer Vieweg, 2017.
- Homberger, J.; Gehring, H.: A Two-Phase Hybrid Metaheuristic For The Vehicle Routing Problem With Time Windows. *European Journal of Operational Research* 162 (2004) 1, S. 220–238.

- Irnich, S.; Toth, P.; Vigo, D.: Chapter 1. The Family Of Vehicle Routing Problems. In: Vigo, D.; Toth, P. (Hrsg.): Vehicle Routing: Problems, Methods, And Applications. 2. Auflage. Philadelphia: Society for Industrial and Applied Mathematics, 2014, S. 1–34.
- Jaller, M.; Pineda, L.; Ambrose, H.: Evaluating The Use Of Zero-Emission Vehicles In Las Mile Deliveries. ITS Reports, 2018, <https://escholarship.org/uc/item/7kr753nm>, (zuletzt abgerufen am 05.06.2019).
- Jaller, M.; Pineda, L.; Phong, D.: Spatial Analysis Of Warehouses And Distribution Centers In Southern California. Transportation Research Record: Journal of the Transportation Research Board 2610 (2017) 1, S. 44–53.
- Janjevic, M.; Ndiaye, A. B.: Inland Waterways Transport For City Logistics: A Review Of Experiences And The Role Of Local Public Authorities. Urban Transport XX WIT Transactions on the Built Environment 138 (2014), S. 279–290.
- Joerss, M.; Schröder, J.; Neuhaus, F.; Klink, C.; Mann, F.: Parcel Delivery. The Future Of Last Mile. Travel, Transport And Logistics. McKinsey & Company, 2016, https://www.mckinsey.com/~media/mckinsey/industries/travel%20transport%20and%20logistics/our%20insights/how%20customer%20demands%20are%20reshaping%20last%20mile%20delivery/parcel_delivery_the_future_of_last_mile.ashx (zuletzt abgerufen am 06.06.2019).
- Julia: Random Numbers. 2019, <https://docs.julialang.org/en/v1/stdlib/Random/index.html>, (zuletzt abgerufen am 16.07.2019).
- Kalantari, H.; Badiie, A.; Ghazanfari, M.: A New Heuristic Model For Fuzzy Transfer Point Location Problem. Journal of Uncertain Systems 8 (2014) 1, S. 31–43.
- Kellner, F.: Analyse logistischer Systeme mittels Online-Navigationsdiensten - Bessere Planung auf Grundlage besserer Daten. HMD Praxis der Wirtschaftsinformatik 53 (2016) 6, S. 894–905.
- Kelton, W. D.; Sadowski, R. P.; Zupick, N. B.: Simulation With Arena. 6. Auflage. New York: McGraw-Hill Education, 2015.
- Klein, M.: Monte-Carlo Simulationen und Due Diligence: Ein methodischer Ansatz zur computergestützten Aggregation von Wahrscheinlichkeitsverteilungen aus Expertenbefragungen. Working Papers in Accounting Valuation Auditing No. 2010-5, 2010, <http://hdl.handle.net/10419/32770>, (zuletzt abgerufen am 26.07.2019), S. 1–24.
- Ko, S. Y.; Cho, S. W.; Lee, C.: Pricing And Collaboration In Last Mile Delivery Services. Sustainability 10 (2018) 12, S. 1–20.

- Koether, R.: Distributionslogistik. Effiziente Absicherung der Lieferfähigkeit. Wiesbaden: Springer Gabler, 2012.
- Köhler, C.; Haferkamp, J.: Evaluation Of Delivery Cost Approximation For Attended Home Deliveries. *Transportation Research Procedia* 37 (2019), S. 67–74.
- Kolonko, M.: Stochastische Simulation. Grundlagen, Algorithmen und Anwendungen. Wiesbaden: Vieweg Teubner, 2008.
- Kopfer, H. W.; Schönberger, J.; Kopfer, H.: Reducing Greenhouse Gas Emissions Of A Heterogeneous Vehicle Fleet. *Flexible Services and Manufacturing Journal* 26 (2014) 1-2, S. 221–248.
- Kosfeld, R.; Eckey, H. F.; Türck, M.: Deskriptive Statistik. Grundlagen – Methoden – Beispiele – Aufgaben. 6. Auflage. Wiesbaden: Springer Fachmedien, 2016.
- Kuwok, F.; Asdecker, B.: Grüne Logistik-Lösungen auf der letzten Meile: Sind Packstationen ökologisch nachhaltig?. In: Sucky, E.; Werner, J.; Kolke, R.; Biethahn, N. (Hrsg.): *Mobility In A Globalised World 2014*. Bamberg: University of Bamberg Press, 2015, S. 139–156.
- Lange, T.; Mosler, K.: Statistik kompakt. Basiswissen für Ökonomen und Ingenieure. Berlin Heidelberg: Springer Gabler, 2017.
- Law, A. M.: *Simulation Modeling And Analysis*. 5. Auflage. New York: McGraw-Hill Education, 2015.
- L'Ecuyer, P.: Good Parameters And Implementations For Combined Multiple Recursive Random Number Generators. *Operation Research* 47 (1999) 1, S. 159 - 164.
- L'Ecuyer, P.: Random Number Generation and Quasi-Monte Carlo, *Wiley StatsRef: Statistics Reference Online*, 2015, <https://onlinelibrary.wiley.com/doi/full/10.1002/9781118445112.stat04386.pub2#references-section>, (zuletzt abgerufen am 11.09.2019), S. 1–12.
- L'Ecuyer, P.; Simard, R.; Chen, E. J.; Kelton, W. D.: An Object-Oriented Random-Number Package With Many Long Streams And Substreams. *Operations Research* 50 (2002) 6, S. 1073–1075.
- Legge, J. T.; Doran, P. J.; Herbert, M. E.; Asher, J.; O'Neil, G.; Mysorekar, S.; Sowa, S.; Hall, K. R.: From Model Outputs To Conservation Action: Prioritizing Locations For Implementing Agricultural Best Management Practices In A Midwestern Watershed. *Journal of Soil and Water Conservation* 68 (2013) 1, S. 22–33.
- Lehmacher, W.: *Logistik im Zeichen der Urbanisierung. Versorgung von Stadt und Land im digitalen und mobilen Zeitalter*. Wiesbaden: Springer Gabler, 2015.

- Louf, R.; Barthelemy, M.: A Typology Of Street Patterns. *Journal of the Royal Society Interface* 11 (2014) 101, S. 1–7.
- Löhr, M.; Schäfer, A.: Same Day Delivery - durch digitale Prozesse den lokalen Standort skalierbar als Wettbewerbsvorteil nutzen - Praxisbeispiele. In: Knoppe, M.; Wild, M. (Hrsg.): *Digitalisierung im Handel. Geschäftsmodelle, Trends und Best Practice*. Wiesbaden: Springer Gabler, 2018, S. 197–234.
- Maes, J.; Vanelander, T.: The Use Of Bicycle Messengers In The Logistics Chain, Concepts Further Revised. *Procedia – Social and Behavioral Sciences* 39 (2012), S.409–423.
- Manner-Romberg, H.; Miller, J.; Müller-Steinfahrt, U.; Symanczyk, W.: Marktuntersuchung und Entwicklungstrends von Kurier-, Express- und Paketdienstleistungen 2017, 2017, https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/Post/Unternehmen_Institutionen/Marktbeobachtung/Briefdienstleistungen/MarktuntersuchungKEP2017.pdf?__blob=publicationFile&v=2 , (zuletzt abgerufen am 19.07.2019).
- Marsaglia, G.; Bray, T. A.: A Convenient Method For Generating Normal Variables. *SIAM Review* 6 (1964) 3, S. 260–264.
- MATLAB: RandStream.list, 2019, <https://uk.mathworks.com/help/matlab/ref/randstream.list.html>, (zuletzt abgerufen am 16.07.2019).
- Mutsumoto, M.; Kurita, Y.: Strong Deviations From Randomness In M-Sequences Based On Trinomials. *ACM Transactions on Modeling and Computer Simulation* 6 (1996) 2, S. 99–106.
- Matsumoto, M.; Nishimura, T.: Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator. *ACM Transactions on Modeling and Computer Simulation* 8 (1998) 1, S. 3–30.
- Mattfeld, D.; Vahrenkamp, R.: *Logistiknetzwerke. Modelle für Standortwahl und Tourenplanung*. 2. Auflage. Wiesbaden: Springer Gabler, 2014.
- Melo, S.; Baptista, P.: Evaluating The Impacts Of Using Cargo Cycles On Urban Logistics: Integrating Traffic, Environmental And Operational Boundaries. *European Transport Research Review* 9 (2017) 2, S. 1–10.
- Metzler, U.: Anwendungsbereiche der Transportplanung. In: Clausen, U.; Geiger, C.: *Verkehrs- und Transportlogistik*. 2. Auflage. Berlin Heidelberg: Springer Vieweg, 2013, S. 277–290.

- Microsoft, 2019a, <https://support.office.com/de-de/article/zufallsbereich-funktion-4cc7f0d1-87dc-4eb7-987f-a469ab381685>, (zuletzt abgerufen am 11.07.2019).
- Microsoft, 2019b, <https://support.office.com/de-de/article/zufallszahl-funktion-4cbfa695-8869-4788-8d90-021ea9f5be73>, (zuletzt abgerufen am 11.07.2019).
- Mundform, D. J.; Schaffer, J.; Kim, M.-J.; Shaw, D.; Thongteeraparp, A.; Preecha, C.: Number Of Replications Required In Monte Carlo Simulation Studies: A Synthesis Of Four Studies. *Journal of Modern Applied Statistical Methods* 10 (2011) 1, S. 19–28.
- Networking and Emerging Optimization Research Group: Vehicle Routing Problem, 2013, <http://neo.lcc.uma.es/vrp/vrp-instances/>, (zuletzt abgerufen am 06.08.2019)
- OECD: The Metropolitan Century. Understanding Urbanisation And Its Consequences. Paris: OECD Publishing, 2015.
- Palisade: @Risk, 2019, <https://www.palisade.com/risk/de/>, (zuletzt abgerufen am 25.07.2019).
- Panneton, F.; L'Ecuyer, P.; Matsumoto, M.: Improved Long-Period Generators Based On Linear Recurrences Modulo 2. *ACM Transactions on Mathematical Software* 32 (2006) 1, S. 1–16.
- Pfeilsticker, L.; Juan, A. A.; Rabe, M.: Development Of A Simheuristic Approach For Solving Realistic Inventory Routing Problems. In: Rabe, M.; Clausen, U.: *Simulation In Production And Logistics 2015*. Stuttgart: Fraunhofer Verlag, 2015, S. 59–68.
- Pfohl, H.-C.: *Logistiksysteme. Betriebswirtschaftliche Grundlagen*. 9. Auflage. Berlin: Springer Vieweg, 2018.
- Pisinger, D.; Ropke, S.: A General Heuristic For Vehicle Routing Problems. *Computers & Operational Research* 34 (2007) 8, S. 2403–2435.
- Poeting, M.; Schaudt, S.; Clausen, U.: Simulation Of An Optimized Last-Mile Parcel Delivery Network Involving Delivery Robots. In: Clausen, U.; Langkau, S.; Kreuz, F.: *Advances In Production, Logistics and Traffic. Proceedings of the 4th Interdisciplinary Conference on Production Logistics and Traffic 2019, Dortmund (Deutschland)*, 27.-28. März 2019, S. 1–19.
- Post, R. M.; Buijs, P.; uit het Broek; M. A. J.; Lopez Alvarez, J. A.; Szirbik, N. B.; Vis, I. F. A.: A Solution Approach For Deriving Alternative Fuel Station Infrastructure Requirements. *Flexible Services and Manufacturing Journal* 30 (2018) 3, S. 592–607.
- Puerto, J.; Rodríguez-Chìa, A. M.; Tamir, A.: On The Planar Piecewise Quadratic 1-Center Problem. *Algorithmica* 57 (2010) 2, S. 252–283.

- Punakivi, M.; Yrjölä, H.; Holmström, J.: Solving The Last Mile Issue: Reception Box Or Delivery Box?. *International Journal of Physical Distribution & Logistics Management* 31 (2001) 6, S. 427–439.
- Rabe, M.; Klüter, A.; Tietze, S.: Comparing Different Distance Metrics For Calculating Distances In Urban Areas With A Supply Chain Simulation Tool. In: Wenzel, S.; Peter, T.: *Simulation in Produktion und Logistik 2017*, Kassel: kassel university press, 2017, S. 119–128.
- Rabe, M.; Klüter, A.; Wuttke, A.: Evaluating The Consolidation Of Distribution Flows Using A Discrete Event Supply Chain Simulation Tool: Application To A Case Study In Greece. In: Rabe, M.; Juan, A. A.; Mustafee, N.; Skoogh, A.; Jain, S.; Johannson, B. (Hrsg.): *Proceedings of the 2018 Winter Simulation Conference (WSC)*, Gothenburg (Schweden), 09.-12. Dezember 2018, S. 2815–2826.
- Rabe, M.; Spieckermann, S.; Wenzel, S.: *Verifikation und Validierung für die Simulation in Produktion und Logistik. Vorgehensmodelle und Techniken*. Berlin Heidelberg: Springer, 2008.
- Rabta, B.; Wankmüller, C.; Reiner, G.: A Drone Fleet Model For Last-Mile Distribution In Disaster Relief Operations. *International Journal of Disaster Risk Reduction* 28 (2018), S. 107–112.
- Rall, B.: *Distributionslogistik*. In: Furmans, K.; Kilger, C. (Hrsg.): *Betrieb von Logistiksystemen*. Berlin Heidelberg: Springer Vieweg, 2019, S. 133–160.
- Raps, J.: *Bewertung von Distanzmetriken für ein Supply-Chain-Simulationsmodell*. Masterarbeit TU Dortmund, 2017, http://www.itpl.mb.tu-dortmund.de/publikationen/files/MA_2017_Raps.pdf (zuletzt abgerufen am 06.06.2019).
- Richard, J. H.; Côté, S. D.: Space Use Analyses Suggest Avoidance Of A Ski Area By Mountain Goats. *The Journal of Wildlife Management* 80 (2016) 3, S. 387–395.
- Rieck, J.; Zimmermann, J.; Glagow, M.: Tourenplanung mittelständischer Speditionsunternehmen in Stückgutkooperationen: Modellierung und heuristische Lösungsverfahren. *Zeitschrift für Planung & Unternehmenssteuerung* 17 (2007), S. 365–388.
- Robinson, S.: *Simulation. The Practice Of Model Development And Use*. 2. Auflage. Basingstoke: Palgrave Macmillan, 2014.
- Rocher, L.; Hendrickx, J. M.; de Montjoye, Y.-A.: Estimating The Success Of Re-Identifications In Incomplete Datasets Using Generative Models. *Nature Communications* 10 (2019) 3069, S. 1–10.

- Rubinstein, R. Y.; Kroese, D. P.: *Simulation And The Monte Carlo Method*. 3. Auflage. Hoboken: John Wiley & Sons, 2017.
- Samarasinghe, T.; Inaltekin, H.; Evans, J. E.: Outage Capacity Of Opportunistic Beamforming With Random User Locations. *IEEE Global Communications Conference (GLOBECOM)*, Atlanta (USA), 9. - 13. Dezember 2013, S. 1944–1949.
- Saito, M.; Matsumoto, M.: SIMD-Oriented Fast Mersenne Twister: A 128-Bit Pseudorandom Number Generator. In: Keller, A.; Heinrich, S.; Niederreiter, H.: *Monte Carlo and Quasi-Monte Carlo Methods 2006*. Berlin Heidelberg: Springer, 2008, S. 607–622.
- Sawilowsky, S. S.: You Think You've Got Trivials?. *Journal of Modern Applied Statistical Methods* 2 (2003) 1, S. 218–225.
- Sboui, L.; Ghazzai, H.; Rezki, Z.; Alouini, M.-S.: On Green Cognitive Radio Cellular Networks: Dynamic Spectrum and Operation Management. *IEEE Access* 4 (2016), S. 4046–4057.
- Schulte, C.: *Logistik. Wege zur Optimierung der Supply Chain*. 7. Auflage. München: Verlag Franz Vahlen, 2017.
- SimPlanAG: SimChain, 2019, <https://www.simchain.net>, (zuletzt abgerufen am 15.08.2019).
- SimChain: Tutorial, 2014, https://www.simchain.net/wp-content/uploads/SimChain_Tutorial.pdf, (zuletzt abgerufen am 15.08.2019).
- SIMUL8: Random Numbers. 2019, https://www.simul8.com/support/help/doku.php?id=features:random_numbers, (zuletzt abgerufen am 16.07.2019).
- Sonntag, F.; Thulesius, M.: E-Commerce, Multi-Channel und Logistik - Zum Einfluss sich verändernder Handelsstrukturen auf Distributionsnetzwerke. In: Voß, P. H. (Hrsg.): *Logistik - eine Industrie, die (sich) bewegt. Strategien und Lösungen entlang der Supply Chain 4.0*. Wiesbaden: Springer Gabler, 2015, S.87–104.
- Sörensen, K.: Investigation Of Practical, Robust And Flexible Decisions For Facility Location Problems Using Tabu Search And Simulation. *The Journal of the Operational Research Society* 59 (2008) 5, S. 624–636.
- Stefanovic, D.; Stefanovic, N.; Radenkovic, B.: Supply Network Modelling And Simulation Methodology. *Simulation Modelling Practice and Theory* 17 (2009) 4, S. 743–766.
- Stocker, T. C.; Steinke, I.: *Statistik. Grundlagen und Methodik*. Berlin Boston: Walter de Gruyter, 2017.

- Sun, P.; Samaan, N.: Random Node Failures And Wireless Networks Connectivity: Theoretical Analysis. *IEEE Wireless Communications Letters* 4 (2015) 5, S. 461–464.
- Süddeutsche Zeitung: Nie mehr Schlangestehen fürs Päckchen, 2014, <https://www.sueddeutsche.de/geld/paketkasten-der-deutsche-post-mein-haus-mein-garten-meine-box-1.1960733>, (zuletzt abgerufen am 14.06.2019).
- Tagesschau: Amazon will bald per Drohne liefern, 2019, <https://www.tagesschau.de/wirtschaft/amazon-usa-drohnen-101.html>, (zuletzt abgerufen am 13.06.2019).
- Tausworthe, R. C.: Random Numbers Generated By Linear Recurrence Modulo Two. *Mathematics of Computation* 19 (1965) 90, S. 201–209.
- ten Hompel, M.; Heidenblut, V. (Hrsg.): Taschenlexikon Logistik. Abkürzungen, Definitionen und Erläuterungen der wichtigsten Begriffe aus Materialfluss und Logistik. Berlin: Springer, 2011.
- Thaller, C.; Telake, M.; Clausen, U.; Dahmen, B.;Leerkamp, B.: KEP-Verkehr in urbanen Räumen. In: Proff, H.; Fojcik, T. M. (Hrsg.): Innovative Produkte und Dienstleistungen in der Mobilität. Technische und betriebswirtschaftliche Aspekte. Wiesbaden: Springer Gabler, 2017, S. 443–458.
- The AnyLogic Company: AnyLogic, 2019, <https://www.anylogic.de>, (zuletzt abgerufen am 15.08.19).
- Thonemann, U.: Operations Management. Konzepte, Methoden und Anwendungen. 3. Auflage. Halbergmoos: Pearson Deutschland GmbH, 2015.
- Tripp, C.: Distributions- und Handelslogistik. Netzwerke und Strategien der Omnichannel-Distribution im Handel. Wiesbaden: Springer Gabler, 2019.
- Umundum, P.: Paradigmenwechsel auf der letzten Meile. In: Voß, P. H. (Hrsg.): Logistik - eine Industrie, die (sich) bewegt. Strategien und Lösungen entlang der Supply Chain 4.0. Wiesbaden: Springer Gabler, 2015, S.120–134.
- UN: World Urbanization Prospects: The 2018 Revision, 2019, <https://population.un.org/wup/Publications/Files/WUP2018-KeyFacts.pdf>, (zuletzt abgerufen am 01.08.2019).
- Vahrenkamp, R.; Kotzab, H.: Logistik. Management und Strategien. 7. Auflage. München:Oldenbourg Wissenschaftsverlag, 2012.
- van Bonn, B.: Basisdaten der Logistikplanung. In: Clausen, U.; Geiger, C.: Verkehrs- und Transportlogistik. 2. Auflage. Berlin Heidelberg: Springer Vieweg, 2013, S. 291–298.

- van Duin, J. H. R.; de Goffau, W.; Wiegmans, B.; Tavasszy, L. A.; Saes, M.: Improving Home Delivery Efficiency By Using Principle Of Address Intelligence For B2C Deliveries. *Transportation Research Procedia* 12 (2016), S. 14–25.
- van Duin, J. H. R.; Kortmann, R.; van den Boogard, S. L.: City Logistics Trough The Canals? A Simulation Study On Freight Waterborne Transport In The Inner-City Of Amsterdam. *International Journal of Urban Sciences* 18 (2014) 2, S. 186–200.
- van Duin, J. H. R.; Kortmann, L. J.; van de Kamp, M.: Toward Sustainable Urban Distribution Using City Canals: The Case Of Amsterdam. In: Taniguchi, E.; Thompson, R. G. (Hrsg.): *City Logistics 1. New Opportunities And Challenges*. London: ISTE, 2018, S. 65–84.
- VDI-Gesellschaft Produktion und Logistik (Hrsg.): *VDI - Richtlinie 3633 Blatt 1. Simulation von Logistik-, Materialfluss- und Produktionssystemen – Grundlagen*. Berlin: Beuth, 2014.
- Verordnung EU 2016/679 Des Europäischen Parlaments und des Rates vom 27. April 2016 zum Schutz natürlicher Personen bei der Verarbeitung personenbezogener Daten, zum freien Datenverkehr und zur Aufhebung der Richtlinie 95/46/EG (Datenschutz Grundverordnung), ABl 2016 L119/1.
- Vinel, A.; Silva, D. F.: Probability Distribution Of The Length Of The Shortest Tour Between A Few Random Points: A Simulation Study. In: Rabe, M.; Juan, A. A.; Mustafee, N.; Skoogh, A.; Jain, S.; Johannson, B. (Hrsg.): *Proceedings of the 2018 Winter Simulation Conference (WSC)*, Gothenburg (Schweden), 09.-12. Dezember 2018, S. 3156–3167.
- Waldmann, K.-H.; Helm, W. E.: *Simulation stochastischer Systeme. Eine anwendungsorientierte Einführung*. Berlin Heidelberg: Springer Gabler, 2016.
- Wang: Excel VBA Version of Mersenne Twister. 2015, <http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/VERSIONS/BASIC/basic.html>, (zuletzt abgerufen am 16.07.2019).
- Wannenwetsch, H.: *Integrierte Materialwirtschaft, Logistik und Beschaffung*. 5. Auflage. Berlin Heidelberg: Springer Vieweg, 2014.
- Wegner, U.; Wegner, K.: *Einführung in des Logistikmanagement. Prozesse - Strukturen - Anwendungen*. 3. Auflage. Wiesbaden: Springer Gabler, 2017.
- Wenzel, S.; Bernhard, J.: Definition und Modellierung von Systemlastdaten für die Simulation logistischer Systeme. In: Nyhuis, P. (Hrsg.): *Beiträge zu einer Theorie der Logistik*. Berlin Heidelberg: Springer, 2008, S. 487–509.

- Wenzel, S.; Weiß, M.; Collisi-Böhmer, S.; Pitsch, H.; Rose, O.: Qualitätskriterien für die Simulation in Produktion und Logistik. Planung und Durchführung von Simulationsstudien. Berlin, Heidelberg: Springer, 2008.
- Wenzel, S.: Simulation logistischer Systeme. In: Tempelmeier, H. (Hrsg.): Modellierung logistischer Systeme. Berlin: Springer, 2018.
- Williams, E. J.; Ülgen, O. M.: Pitfalls In Managing A Simulation Project. In: Laroque, C.; Himmelpach, J.; Pasupathy, R.; Rose, O.; Uhrmacher, A. M. (Hrsg.): Proceedings of the 2012 Winter Simulation Conference (WSC), Berlin (Deutschland), 09.-12. Dezember 2012, S. 3667–3674.
- WinFuture: Die Post rüstet ab: Paketkästen werden eingestellt, PostBot kommt nicht, 2019, <https://winfuture.de/news,107238.html>, (zuletzt abgerufen am 14.06.2019).
- Wirtschaftswoche: Wohin mit unseren Paketen?, 2015, <https://www.wiwo.de/unternehmen/dienstleister/hermes-dpd-und-gls-stellen-paketbox-vor-wohin-mit-unsere-paketen/12412740.html>, (zuletzt abgerufen am 14.06.2019).
- WirtschaftsWoche: Google-Mutterkonzern liefert erstmals per Drohne Waren aus, 2019, <https://www.wiwo.de/technologie/digitale-welt/alphabet-google-mutterkonzern-liefert-erstmal-per-drohne-waren-aus/24201280.html>, (zuletzt abgerufen am 13.06.2019).
- Uchoa, E.; Pecin, D.; Pessoa, A.; Poggi, M.; Subramanian, A.; Thibaut, V.; Oliveira, D.: Capacitated Vehicle Routing Problem Library, 2014, <http://vrp.atd-lab.inf.puc-rio.br/index.php/en/>, (zuletzt abgerufen am 01.08.2019).
- Yilmaz, E.; Elbasi, S.; Ferhatosmanoglu, H.: Predicting Optimal Facility Locations Without Customer Locations. In: Matwin, S.; Yu, S.; Farooq, F. (Hrsg.): KDD'17 Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax (Canada), 13. - 17. August 2017, S. 2121–2130.
- Yousefli, A.; Kalantari, H.; Ghazanfari, M.: Stochastic Transfer Point Location Problem: A Probabilistic Rule-Based Approach. *Uncertain Supply Chain Management* 6 (2018) 1, S. 65–74.
- Zarimpas, V.; Honary, B.; Darnell, M.: Channel Adaptive Multiple I/O Protocol For Mixed Wireless And Wired Channels. *IET Communications* 1 (2007) 5, S. 982–989.
- Zentes, J.; Schramm-Klein, H.: Neue Anforderungen an die Handelslogistik - Implikationen aus Theorie und Praxis mit besonderem Fokus auf Multi-Channel-Systeme des Handels. In: Nyhuis, P. (Hrsg.): Beiträge zu einer Theorie der Logistik. Berlin Heidelberg: Springer, 2008, S. 409–438.

Abkürzungsverzeichnis

BIEK	Bundesverbands Paket & Express Logistik
B2B	Business-to-Business
B2C	Business-to-Consumer
C2B	Consumer-to-Business
C2C	Consumer-to-Consumer
GIS	Geografisches Informationssystem
KEP	Kurier-Express-Paket
LE	Längeneinheit
OSM	OpenStreetMaps
QLE	Quadratlängeneinheit
UCC	Urban Consolidation Center
VRP	Vehicle Routing Problem
V&V	Verifikation und Validierung

Tabellenverzeichnis

Tabelle 1: Verfahren zur Standortgenerierung in Kommunikationsnetzwerken	41
Tabelle 2: Verfahren zur Standortgenerierung in Flora und Fauna	42
Tabelle 3: Verfahren zur Standortgenerierung in der Standortplanung.....	42
Tabelle 4: Verfahren zur Standortgenerierung in der Last Mile.....	43
Tabelle 5: Ausgewählte Benchmark-Testsets.....	61
Tabelle 6: Parameter des genutzten Kompositionsgenerators.....	64
Tabelle 7: Unterkatalog für die Bewertung der Generierungsverfahren eines Testsets	68
Tabelle 8: Kriterienkatalog für die Bewertung der Generierungsverfahren	68
Tabelle 9: Gleichverteilung Testset E-n22-k4	70
Tabelle 10: Gleichverteilung Testset CMT14	75
Tabelle 11: Gleichverteilung Testset C110_1	80
Tabelle 12: Gleichverteilung Testset Antwerp1	86
Tabelle 13: Normalverteilung Testset E-n22-k4	92
Tabelle 14: Normalverteilung Testset CMT14.....	98
Tabelle 15: Normalverteilung Testset C110_1	104
Tabelle 16: Normalverteilung Testset Antwerp1	110
Tabelle 17: Dreiecksverteilung Testset E-n22-k4	116
Tabelle 18: Dreiecksverteilung Testset CMT14.....	122
Tabelle 19: Dreiecksverteilung Testset C110_1	127
Tabelle 20: Dreiecksverteilung Testset Antwerp1	132
Tabelle 21: Gleichverteilung Berücksichtigung Bevölkerungsdichte	138
Tabelle 22: Normalverteilung Berücksichtigung Bevölkerungsdichte.....	144
Tabelle 23: Dreiecksverteilung Berücksichtigung Bevölkerungsdichte.....	149
Tabelle 24: Vergleich Testset E-n22-k4 Ergebnisse	156
Tabelle 25: Vergleich Testset E-n22-k4 mit Bevölkerungsdichte Ergebnisse	159
Tabelle 26: Vergleich Testset CMT14 Ergebnisse	163
Tabelle 27: Vergleich Testset C110_1 Ergebnisse	167
Tabelle 28: Vergleich Testset Antwerp1 Ergebnisse.....	171
Tabelle 29: Bewertung Testset E-n22-k4	176
Tabelle 30: Bewertung Testset E-n22-k4 mit Berücksichtigung der Bevölkerungsdichte	177
Tabelle 31: Bewertung Testset CMT14.....	179
Tabelle 32: Bewertung Testset C110_1.....	181
Tabelle 33: Bewertung Testset Antwerp1	182
Tabelle 34: Gesamtbewertung der Generierungsverfahren	183
Tabelle 35: Weitere Verfahren zur Standortgenerierung in der Standortplanung	212
Tabelle 36: Weitere Verfahren zur Standortgenerierung in der Last Mile.....	212
Tabelle 37: Ergebnisse der Gleichverteilung	218
Tabelle 38: Ergebnisse der Gleichverteilung (Fortsetzung)	219

Tabelle 39: Ergebnisse der Normalverteilung	220
Tabelle 40: Ergebnisse der Normalverteilung (Fortsetzung)	221
Tabelle 41: Ergebnisse der Dreiecksverteilung	222
Tabelle 42: Ergebnisse der Dreiecksverteilung (Fortsetzung).....	223
Tabelle 43: Ergebnisse der Generierung auf Basis der Bevölkerungsdichte	224
Tabelle 44: Wahre Werte für die Tourenlänge der Testsets	225

Abbildungsverzeichnis

Abbildung 1: Ein- und zweistufige Logistiknetzwerke (nach Gudehus 2010, S. 18 f.) .	6
Abbildung 2: Drei- und mehrstufige Logistiknetzwerke (nach Gudehus 2010, S. 19 f.)	6
Abbildung 3: Ein beispielhaftes Depotnetzwerk.....	9
Abbildung 4: Ein beispielhaftes Hub-and-Spoke-Netzwerk.....	10
Abbildung 5: Hub-and-Spoke-Netzwerk mit Regionalhubs	11
Abbildung 6: Struktur einer Monte-Carlo-Simulation (nach Brandimarte 2014, S. 254 und Klein 2010)	27
Abbildung 7: Vorgehensmodell für Simulationsstudien (nach Rabe et al. 2008, S. 5)	28
Abbildung 8: Dichtefunktionen von zwei stetigen Gleichverteilungen	52
Abbildung 9: Dichtefunktionen von drei Normalverteilungen	53
Abbildung 10: Dichtefunktionen von drei Dreiecksverteilungen	54
Abbildung 11: Programmablaufplan.....	63
Abbildung 12: Gleichverteilung Generierungsfehler pro Replikation Testset E-n22-k4	71
Abbildung 13: Gleichverteilung Box-Plots Testset E-n22-k4	72
Abbildung 14: Gleichverteilung 95% Konfidenzintervalle Testset E-n22-k4.....	74
Abbildung 15: Gleichverteilung Generierungsfehler pro Replikation Testset CMT14	76
Abbildung 16: Gleichverteilung Box-Plots Testset CMT14.....	78
Abbildung 17: Gleichverteilung 95% Konfidenzintervalle Testset CMT14	79
Abbildung 18: Gleichverteilung Generierungsfehler pro Replikation Testset C110_1	82
Abbildung 19: Gleichverteilung Box-Plots Testset C110_1.....	83
Abbildung 20: Gleichverteilung 95% Konfidenzintervalle Testset C110_1	85
Abbildung 21: Gleichverteilung Generierungsfehler pro Replikation Testset Antwerp1	87
Abbildung 22: Gleichverteilung Box-Plots Testset Antwerp1	89
Abbildung 23: Gleichverteilung 95% Konfidenzintervalle Testset Antwerp1	90
Abbildung 24: Normalverteilung Generierungsfehler pro Replikation Testset E-n22-k4	93
Abbildung 25: Normalverteilung Box-Plots Testset E-n22-k4.....	95
Abbildung 26: Normalverteilung 95% Konfidenzintervalle Testset E-n22-k4	96
Abbildung 27: Normalverteilung Generierungsfehler pro Replikation Testset CMT14	99
Abbildung 28: Normalverteilung Box-Plots Testset CMT14	101
Abbildung 29: Normalverteilung 95% Konfidenzintervalle Testset CMT14.....	103
Abbildung 30: Normalverteilung Generierungsfehler pro Replikation Testset C110_1	105
Abbildung 31: Normalverteilung Box-Plots Testset C110_1	108
Abbildung 32: Normalverteilung 95% Konfidenzintervalle Testset C110_1.....	109

Abbildung 33: Normalverteilung Generierungsfehler pro Replikation Testset Antwerp1	111
Abbildung 34: Normalverteilung Box-Plots Testset Antwerp1	113
Abbildung 35: Normalverteilung 95% Konfidenzintervalle Testset Antwerp1	115
Abbildung 36: Dreiecksverteilung Generierungsfehler pro Replikation Testset E-n22-k4.....	117
Abbildung 37: Dreiecksverteilung Box-Plots Testset E-n22-k4.....	119
Abbildung 38: Dreiecksverteilung 95% Konfidenzintervalle Testset E-n22-k4	120
Abbildung 39: Dreiecksverteilung Generierungsfehler pro Replikation Testset CMT14	123
Abbildung 40: Dreiecksverteilung Box-Plots Testset CMT14	124
Abbildung 41: Dreiecksverteilung 95% Konfidenzintervalle Testset CMT14.....	126
Abbildung 42: Dreiecksverteilung Generierungsfehler pro Replikation Testset C110_1	128
Abbildung 43: Dreiecksverteilung Box-Plots Testset C110_1	130
Abbildung 44: Dreiecksverteilung 95% Konfidenzintervalle Testset C110_1	131
Abbildung 45: Dreiecksverteilung Generierungsfehler pro Replikation Testset Antwerp1.....	133
Abbildung 46: Dreiecksverteilung Box-Plots Testset Antwerp1	136
Abbildung 47: Dreiecksverteilung 95% Konfidenzintervalle Testset Antwerp1.....	137
Abbildung 48: Gleichverteilung Generierungsfehler pro Replikation Berücksichtigung	Bevölkerungsdichte
.....	139
Abbildung 49: Gleichverteilung Box-Plots Berücksichtigung Bevölkerungsdichte...	141
Abbildung 50: Gleichverteilung 95% Konfidenzintervall Berücksichtigung	Bevölkerungsdichte
.....	142
Abbildung 51: Normalverteilung Generierungsfehler pro Replikation Berücksichtigung	Bevölkerungsdichte
.....	145
Abbildung 52: Normalverteilung Box-Plots Berücksichtigung Bevölkerungsdichte.	147
Abbildung 53: Normalverteilung 95% Konfidenzintervall Berücksichtigung	Bevölkerungsdichte
.....	148
Abbildung 54: Dreiecksverteilung Generierungsfehler pro Replikation Berücksichtigung	Bevölkerungsdichte
.....	151
Abbildung 55: Dreiecksverteilung Box-Plots Berücksichtigung Bevölkerungsdichte	152
Abbildung 56: Dreiecksverteilung 95% Konfidenzintervall Berücksichtigung	Bevölkerungsdichte
.....	153
Abbildung 57: Vergleich Generierungsfehler pro Replikation Testset E-n22-k4.....	157
Abbildung 58: Vergleich Box-Plots Testset E-n22-k4	157

Abbildung 59: Vergleich 95% Konfidenzintervalle Testset E-n22-k4	158
Abbildung 60: Vergleich Generierungsfehler pro Replikation Testset E-n22-k4 mit Bevölkerungsdichte	B 1
Abbildung 61: Vergleich Box-Plots Testset E-n22-k4 mit Bevölkerungsdichte	161
Abbildung 62: Vergleich 95% Konfidenzintervalle Testset E-n22-k4 mit Bevölkerungsdichte	Bevölkerun 162
Abbildung 63: Vergleich Generierungsfehler pro Replikation Testset CMT14	164
Abbildung 64: Vergleich Testset CMT14 Box-Plots	165
Abbildung 65: Vergleich 95% Konfidenzintervalle Testset CMT14	166
Abbildung 66: Vergleich Generierungsfehler pro Replikation Testset C110_1	168
Abbildung 67: Vergleich Testset C110_1 Box-Plots	169
Abbildung 68: Vergleich Testset C110_1 Konfidenzintervalle	170
Abbildung 69: Vergleich Testset Antwerp1 Fehler „doppelte Standorte“	171
Abbildung 70: Vergleich Testset Antwerp1 Box-Plots	172
Abbildung 71: Vergleich Testset Antwerp1 Konfidenzintervalle	173
Abbildung 72: Standortverteilung im Testset E-n22-k4	213
Abbildung 73: Standortverteilung im Testset CMT14	213
Abbildung 74: Standortverteilung im Testset C110_1	214
Abbildung 75: Standortverteilung im Testset R110_1	215
Abbildung 76: Standortverteilung im Testset RC110_1	216
Abbildung 77: Standortverteilung im Testset Antwerp1	217
Abbildung 78: Standortverteilung im Testset Antwerp1 Replikation Nr. 17 Gleichverteilung	226
Abbildung 79: Standortverteilung im Testset Antwerp1 Replikation Nr. 17 Normalverteilung	227
Abbildung 80: Standortverteilung im Testset Antwerp1 Replikation Nr. 17 Dreiecksverteilung	228

Formelverzeichnis

Formel (1): Arithmetischer Mittelwert.....	35
Formel (2): Berechnung des Konfidenzintervalls.....	36
Formel (3): Korrigierte empirische Varianz.....	36
Formel (4): Quartilsberechnung	38
Formel (5): Interquartilsabstand	38
Formel (6): Median.....	38
Formel (7): Lineare Kongruenzgenerator.....	47
Formel (8): Kompositionsgenerator I.....	48
Formel (9): Kompositionsgenerator II.....	48
Formel (10): Kompositionsgenerator III	48
Formel (11): Kompositionsgenerator IV	48
Formel (12): Stetige gleichverteilte Zufallsvariablenberechnung	51
Formel (13): Diskrete gleichverteilte Zufallsvariablenberechnung.....	51
Formel (14): Transformation standartnormalverteilter Zufallszahlen.....	52
Formel (15): Methode von Box und Muller I.....	52
Formel (16): Methode von Box und Muller II	52
Formel (17): Polarmethode I	53
Formel (18): Polarmethode II.....	53
Formel (19): Dreiecksverteilung Inversion I.....	54
Formel (20): Dreiecksverteilung Inversion II.....	54
Formel (21): Dreiecksverteilung Inversion III	54

Anhang A: Weitere Verfahren Standortgenerierung

Tabelle 35: Weitere Verfahren zur Standortgenerierung in der Standortplanung

Autoren	Verfahren	Hinweise
Bertsimas 1989	Zufällig, unabhängig	-
Bertsimas et al. 2010	Zufällig	Einheitsquadrat
Blanco et al. 2013	Zufällig	Verteilt im Einheitsquadrat und Einheitswürfel
Carbone und Mehrez 1980	Mehrdimensionale Normalverteilung	Koordinatensystem (x; y), Erwartungswert $\mu = 0$, Varianz = 1
Davoodi und Mohades 2011	Zufällig	-
Davoodi et al. 2009	Zufällig	-
Foul 2006	Bivariate Gleichverteilung	Verteilung auf (a; b) oder auf (a; b) und (c; d) oder für jeden Standort wechselndes Intervall (a; b)
Foul 2012	Bivariate Gleichverteilung Bivariate Exponentialverteilung Bivariate Normalverteilung	-
Puerto et al. 2010	Stetige Gleichverteilung	-
Yilmaz et al. 2017	Voronoi Diagramm	Anzahl Kundenstandorte

Tabelle 36: Weitere Verfahren zur Standortgenerierung in der Last Mile

Autor(en)	Verfahren	Hinweise
Ko et al. 2018	Zufällig	Einheitsquadrat
Vinel & Silva 2018	Stetige Gleichverteilung	-
Kopfer et al. 2014	Zufällig	Koordinatensystem (x; y), Verteilt auf (0; 300)
Brown und Guiffrida 2014	Stetige Gleichverteilung Kreisförmige Region	-
Crainic et al. 2010	Zufällig	-

Anhang B: Testsets und Simulationsergebnisse

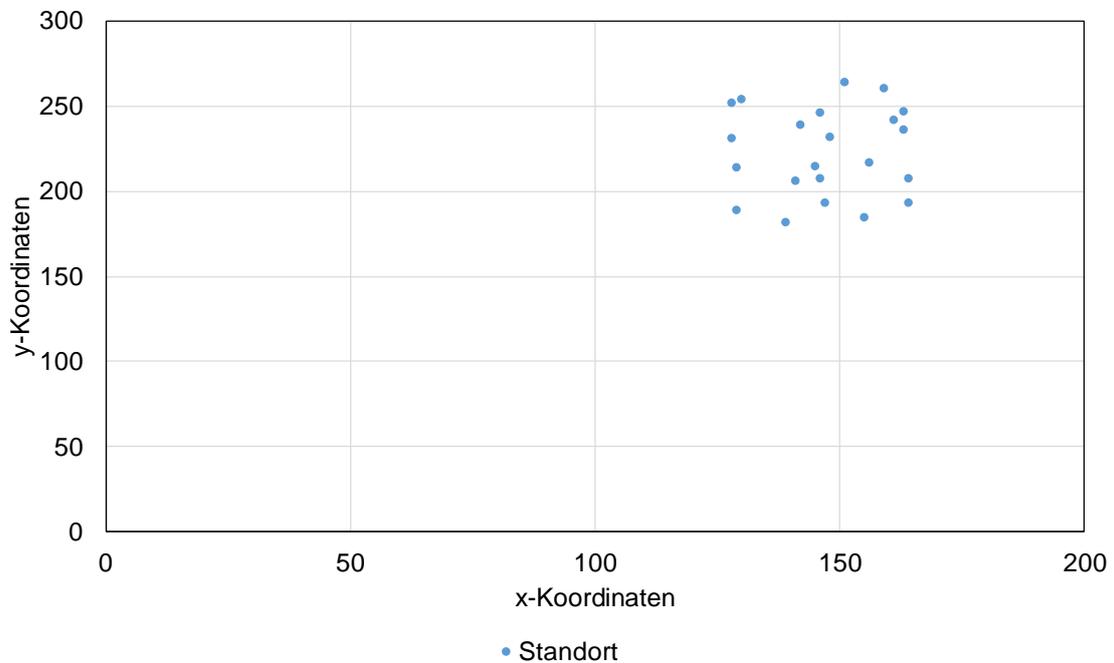


Abbildung 72: Standortverteilung im Testset E-n22-k4

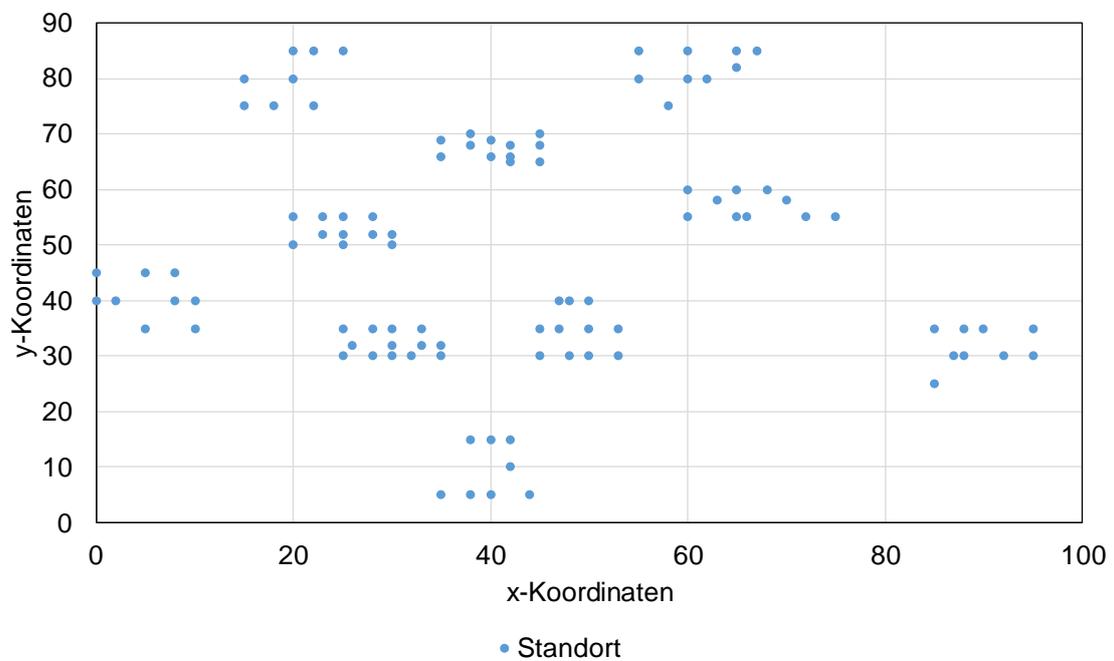


Abbildung 73: Standortverteilung im Testset CMT14

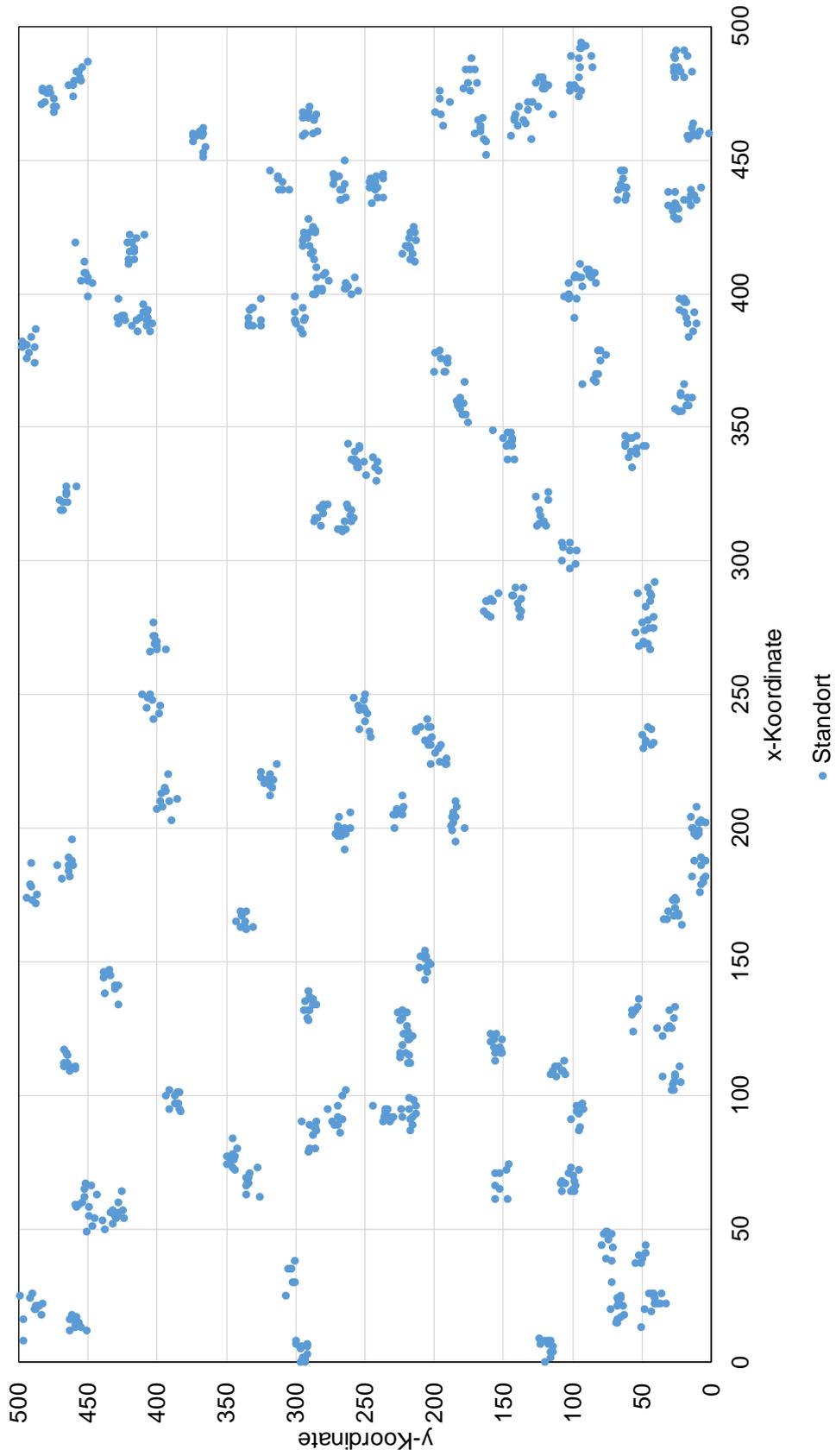


Abbildung 74: Standortverteilung im Testset C110_1

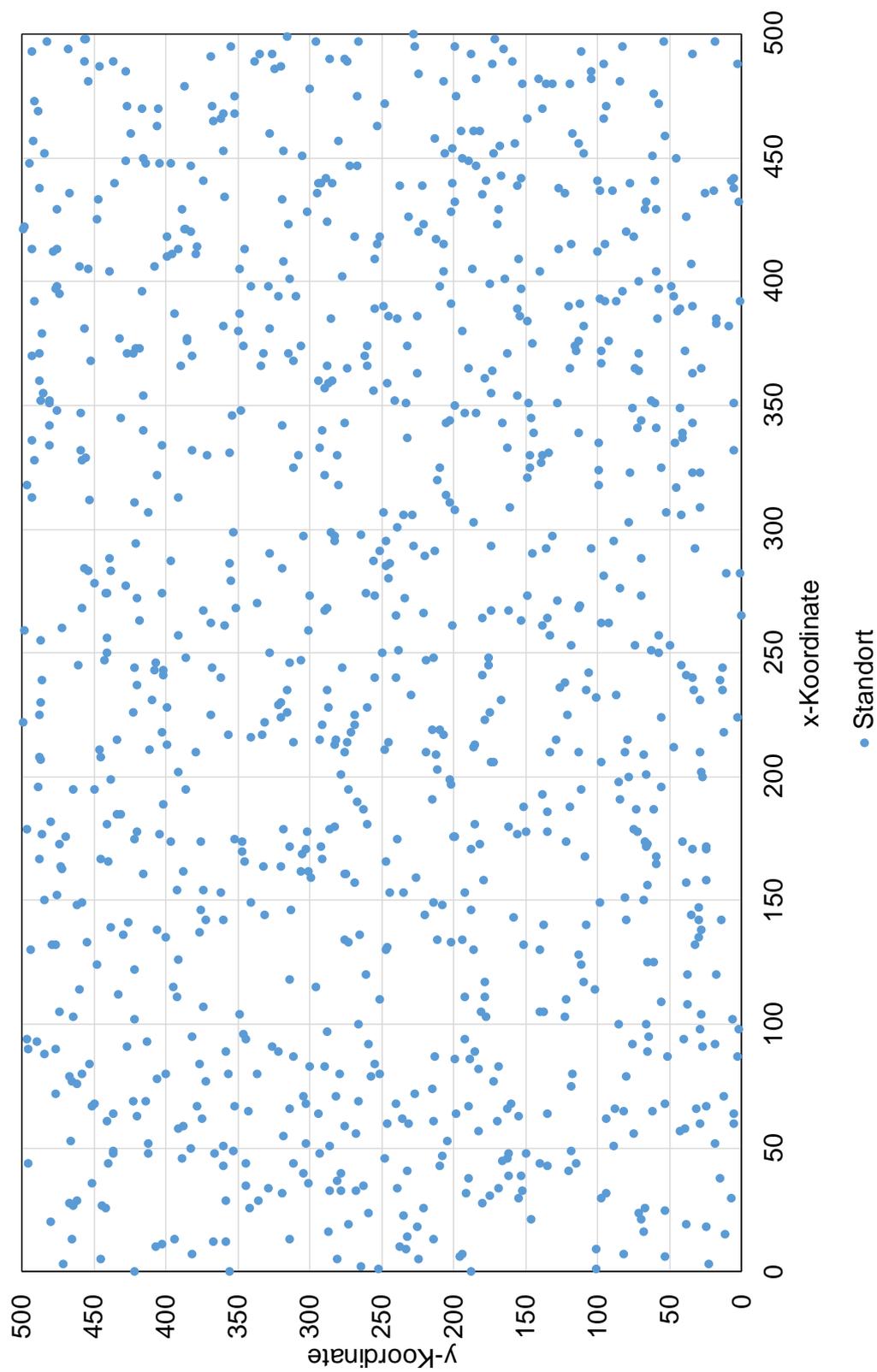


Abbildung 75: Standortverteilung im Testset R110_1

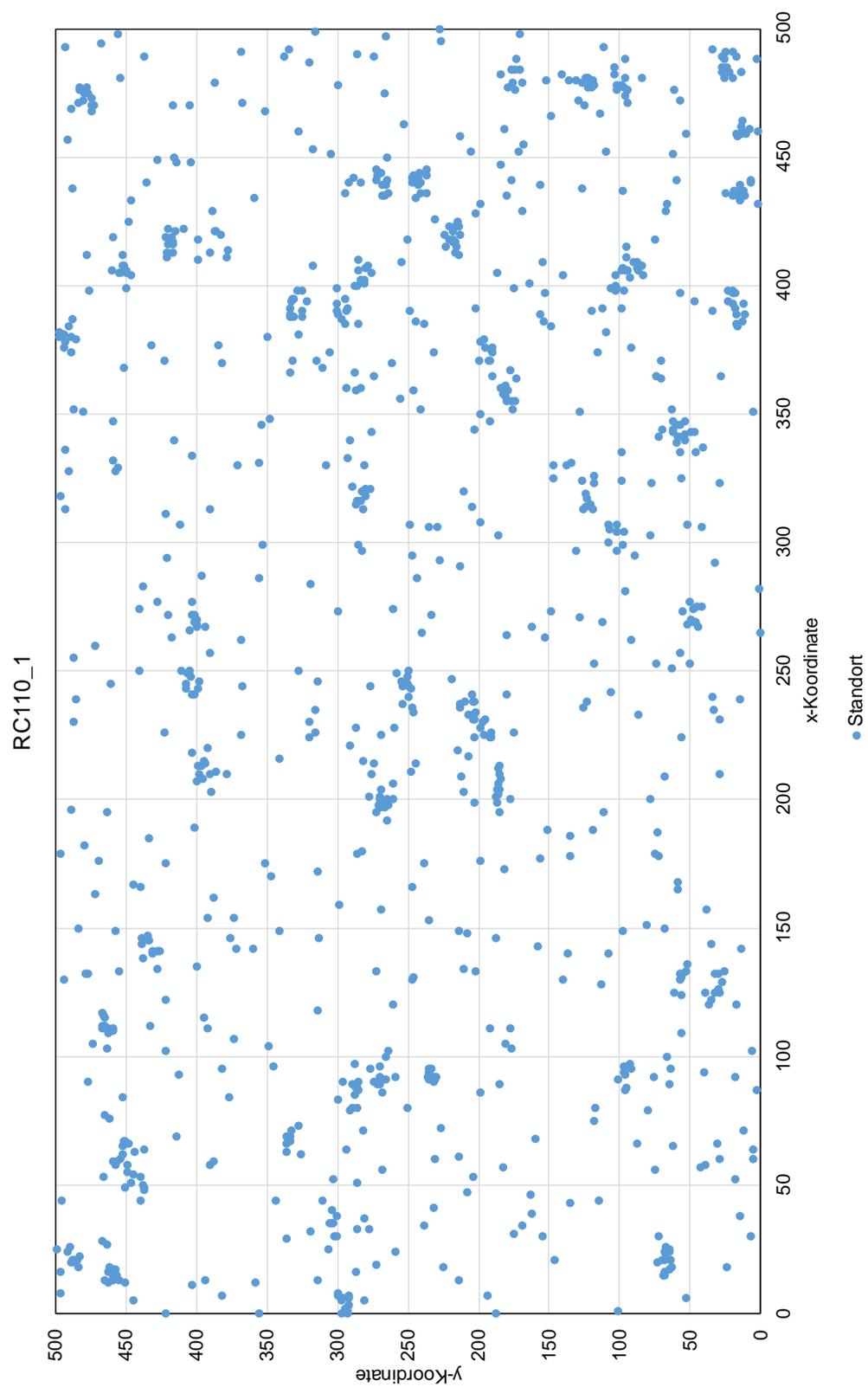


Abbildung 76: Standortverteilung im Testset RC110_1

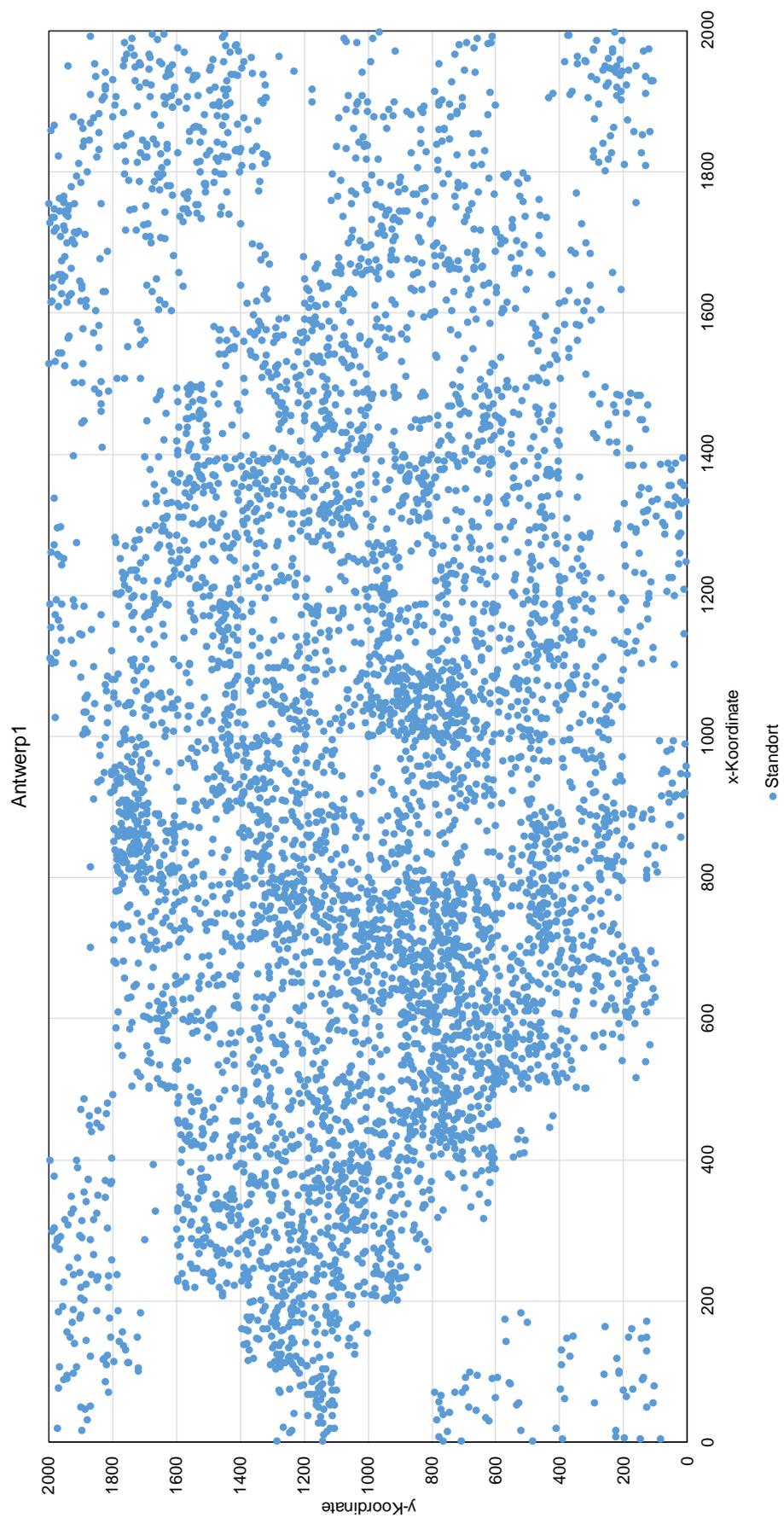


Abbildung 77: Standortverteilung im Testset Antwort1

Tabelle 37: Ergebnisse der Gleichverteilung

Bezeichnung [Name]	Replikation [Anzahl]	Tourenlänge [LE]			Generierungszeit [s]				Fehler Gesamt/Min/Max /Median/Mittelwert [Anzahl]		
		Mittelwert	Median	Konfidenzintervall	Gesamt	Min/ Max	Median/ Mittelwert	Doppelte Standorte	Außerhalb Stadtgrenze	Berechnungs- fehler	
E-n22-k4	100	4520,69	4554,69	(4409,03; 4632,34)	0,265	0/0,016	0/0	0/0/0/0	-	-	
	1000	4547,7	4545,3	(4521,82; 4573,57)	2	0/0,016	0/0	6/0/1/0/ 0,01	-	-	
	10000	4541,54	4544,9	(4533,89; 4549,2)	16,703	0/0,016	0/0	63/0/2/0/ 0,01	-	-	
CMT14	100	8518,27	8530,07	(8339,68; 8696,87)	0,422	0/0,016	0/0,002	53/0/3/0/ 0,53	-	-	
	1000	8519,94	8519,62	(8493,94; 8545,94)	2,938	0/0,016	0/0,001	619/0/4/0/ 0,62	-	-	
	10000	8525,97	8522,81	(8519,55; 8532,38)	27,312	0/0,016	0/0,001	6176/0/5/0/ 0,62	-	-	
C110_1	100	471675,8	471791,52	(462316,73; 481034,86)	1,657	0/0,016	0,016/0,013	211/0/5/2/ 2,11	-	-	
	1000	472175,48	472325,86	(471188,54; 473162,42)	15,016	0/0,016	0,016/0,013	2000/0/7/2/2	-	-	
	10000	472226,13	472385,62	(472056,85; 472395,4)	141,297	0/0,032	0,016/0,014	19911/0/9/2/ 1,99	-	-	

Table 38: Ergebnisse der Gleichverteilung (Fortsetzung)

Bezeichnung [Name]	Replikation [Anzahl]	Tourenlänge [LE]			Generierungszeit [s]				Fehler Gesamt/Min/Max /Median/Mittelwert [Anzahl]		
		Mittelwert	Median	Konfidenz- intervall	Gesamt	Min/ Max	Median/ Mittelwert	Doppelte Standrote	Außerhalb Stadtgrenze	Berechnungs- fehler	
Antwerp1	100	11330733,09	11331845,17	(11107316,66; 11554149,51)	8,109	0,078/ 0,094	0,078/ 0,079	449/0/10/4/ 4,49	-	-	
	1000	11332987,82	11331687,15	(11310515,87; 11355459,77)	82,015	0,078/ 0,125	0,078/ 0,081	4561/0/12/4/ 4,56	-	-	
	10000	11334486,71	11334488,61	(11332022,67; 11336950,75)	970,734	0,062/ 0,297	0,078/ 0,095	44728/0/ 14/4/4,47	-	-	

Tabelle 39: Ergebnisse der Normalverteilung

Bezeichnung [Name]	Replikation [Anzahl]	Tourenlänge [LE]			Generierungszeit [s]				Fehler Gesamt/Min/Max /Median/Mittelwert [Anzahl]			
		Mittelwert	Median	Konfidenz- intervall	Gesamt	Min/ Max	Median/ Mittelwert	Doppelte Standorte	Außerhalb Stadtgrenze	Berechnungs- fehler		
E-n22-k4	100	4360	4320,03	(4262,36; 4457,65)	0,093	0/0,015	0/0	1/0/1/0/0,01	132/0/6/1/ 1,32	656/0/16/6/ 6,56		
	1000	4368,16	4366,18	(4350,38; 4385,93)	0,531	0/0,156	0/0	17/0/1/0/ 0,02	1205/0/7/1/ 1,2	6397/0/17/6/ 6,4		
	10000	4362,06	4359,94	(4357; 4367,12)	5,453	0/0,016	0/0	122/0/1/0/0,0	11580/0/7/1/ 1,16	63314/0/21/6/6,3 3		
CMT14	100	8320,97	8328,61	(8149,37; 8492,58)	0,187	0/0,016	0/0,001	61/0/2/1/ 0,61	3424/19/53/34/ 34,24	3712/23/61/36/3 7,12		
	1000	8314,95	8330,16	(8291,76; 8338,15)	1,453	0/0,016	0/0,001	730/0/4/1/0,7 3	34982/17/59/3 4,5/34,98	36709/13/61/36/ 36,71		
	10000	8306,28	8308,85	(8300,74; 8311,83)	16,313	0/0,016	0/0,001	7558/0/5/1/0, 76	348903/12/ 64/34/34,89	369439/13/72/37 /36,94		
C110_1	100	448497,12	448295,57	(439645,09; 457349,14)	1,078	0/0,016	0,015/0,01	550/0/16/5/5, 5	702/2/13/7/ 7,02	27788/226/ 324/280/277,8		
	1000	448420,17	448424,79	(447520,08; 449320,25)	10,296	0/0,032	0,015/0,01	5625/0/16/5/5 ,62	6834/1/16/7/ 6,83	275966/226/ 334/276/275,9		
	10000	448649,69	448650,94	(448542,01; 448757,38)	118,984	0/0,032	0,015/ 0,012	55943/0/ 16/5/5,59	67430/0/17/7/ 6,74	2762472/210/ 348/276/276,2		

Tabelle 40: Ergebnisse der Normalverteilung (Fortsetzung)

Bezeichnung [Name]	Replikation [Anzahl]	Tourenlänge [LE]			Generierungszeit [s]				Fehler Gesamt/Min/Max /Median/Mittelwert [Anzahl]		
		Mittelwert	Median	Konfidenz- intervall	Gesamt	Min/ Max	Median/ Mittelwert	Doppelte Standrote	Außerhalb Stadtgrenze	Berechnungsfe- hler	
Antwerp1	100	10708008,54	10710448,48	(10496994,23; 10919022,86)	7,063	0,046/ 0,078	0,062/0,061	1666/6/32/17/ 16,66	1055/5/18/ 11/10,65	164963/1527/ 1762/1647,5/ 1649,63	
	1000	10707517,95	10707427,96	(10686448,41; 10728587,49)	63,406	0,046/ 0,157	0,062/0,062	15894/6/32/ 16/15,89	10403/2/22/10/ 10,4	1647724/1504/ 1809/1648/ 1647,72	
	10000	10708670	10708441,89	(10706499,05; 10710840,94)	627,657	0,046/ 0,157	0,062/0,061	159739/3/33/ 16/15,97	103526/1/25/ 10/10,35	16464768/1459 /1821/1646/ 1646,48	

Tabelle 41: Ergebnisse der Dreiecksverteilung

Bezeichnung [Name]	Replikation [Anzahl]	Tourenlänge [LE]			Generierungszeit [s]				Fehler Gesamt/Min/Max /Median/Mittelwert [Anzahl]		
		Mittelwert	Median	Konfidenz- intervall	Gesamt	Min/ Max	Median/ Mittelwert	Doppelte Standrote	Außerhalb Stadtgrenze	Berechnungs- fehler	
E-n22-k4	100	4377,94	4385,71	(4276,21; 4479,68)	0,218	0/00	0/0	1/0/1/0/ 0,01	-	-	
	1000	4377,63	4381,02	(4358,76; 4396,51)	2,281	0/0,016	0/0	9/0/1/0/ 0,01	-	-	
	10000	4387,25	4384,16	(4381,78; 4392,72)	11,297	0/0,016	0/0	86/0/1/0/ 0,01	-	-	
CMT14	100	8185,46	8193,06	(8017,58;8353,34)	0,313	0/0,016	0/0	106/0/4/1/ 1,06	-	-	
	1000	8190,05	8189,72	(8168,67;8211,43)	2,234	0/0,016	0/0,001	1074/0/6/1/ 1,07	-	-	
	10000	8192,56	8190,3	(8187,92;8197,2)	20,25	0/0,016	0/0,001	10735/0/6/1/ 1,07	-	-	
C110_1	100	454441,56	454569,25	(445456,38;46342 6,74)	0,765	0/0,016	0/0,005	335/0/8/3/ 3,35	-	-	
	1000	454652,35	454713,33	(453727,01;45557 7,69)	7,407	0/0,016	0/0,005	3635/0/16/4/ 3,64	-	-	
	10000	454653,88	454641,92	(454536,65;45477 1,11)	67,063	0/0,016	0/0,006	35928/0/16/3/ 3,59	-	-	

Tabelle 42: Ergebnisse der Dreiecksverteilung (Fortsetzung)

Bezeichnung [Name]	Replikation [Anzahl]	Tourenlänge [LE]			Generierungszeit [s]				Fehler Gesamt/Min/Max /Median/Mittelwert [Anzahl]		
		Mittelwert	Median	Konfidenz- intervall	Gesamt	Min/ Max	Median/ Mittelwert	Doppelte Standrote	Außerhalb Stadtgrenze	Berechnungs- fehler	
Antwerp1	100	10910983	10913771,64	(10695927,46; 11126038,53)	3,39	0,015/ 0,047	0,031/ 0,03	844/2/15/8/ 8,44	-	-	
	1000	10912407,17	10912503,06	(10890877,45; 10933936,89)	34,906	0,015/ 0,047	0,0,1/ 0,032	8030/1/17/8/ 8,03	-	-	
	10000	10912764,99	10912207,09	(10910496,84; 10915033,15)	6333,703	0,015/ 0,063	0,031/ 0,031	80164/0/22/8/ 8,02	-	-	

Tabelle 43: Ergebnisse der Generierung auf Basis der Bevölkerungsichte

Generierungsverfahren [Namen]	Replikation [Anzahl]	Tourenlänge [LE]			Generierungszeit [s]				Fehler Gesamt/Min/Max /Median/Mittelwert [Anzahl]		
		Mittelwert	Median	Konfidenzintervall	Gesamt	Min/ Max	Median/Mittelwert	Doppelte Standorte	Außerhalb Stadtgrenze	Berechnungsfehler	
Gleichverteilung	100	7220,18	7228,23	(7075,81; 7364,56)	0,219	0/0,016	0/0	6/0/1/0/ 0,06	-	-	
	1000	7219,2	7221,96	(7203,12; 7235,28)	2,734	0/0,016	0/0	68/0/2/0/ 0,07	-	-	
	10000	7224,19	7223,62	(7221,36; 7227,02)	16,578	0/0,016	0/0	672/0/3/0/ 0,07	-	-	
Normalverteilung	100	7216,01	7217,63	(7072,83; 7359,19)	0,063	0/0,016	0/0	13/0/2/0/ 0,13	216/0/7/2/ 2,16	678/1/17/6/6,78	
	1000	7218,93	7219,21	(7203,72; 7234,15)	0,656	0/0,016	0/0	119/0/2/0/ 0,12	2132/0/17/6/ 6,7	6699/0/17/6/6,7	
	10000	7218,04	7217,61	(7215,76; 7220,32)	6,015	0/0,016	0/0	1150/0/3/0/0, 12	20631/0/11/2/ 2,06	66143/0/23/6/ 6,61	
Dreiecksverteilung	100	7212,94	7216,56	(7069,77; 7356,1)	0,375	0/0,016	0/0	19/0/2/0/ 0,19	-	-	
	1000	7213,27	7214,03	(7198,14; 7228,4)	2,453	0/0,016	0/0	128/0/2/0/ 0,13	-	-	
	10000	7216,48	7216,25	(7214,27; 7218,7)	11,219	0/0,016	0/0	1230/0/3/0/ 0,12	-	-	

Tabelle 44: Wahre Werte für die Tourenlänge der Testsets

Testset	Wahre Tourenlänge [LE]
E-n22-k4	7273,99
CMT14	8424,84
C110_1	466386,60
R110_1	470940,64
RC110_1	488349,06
Antwerp1	10967935,9

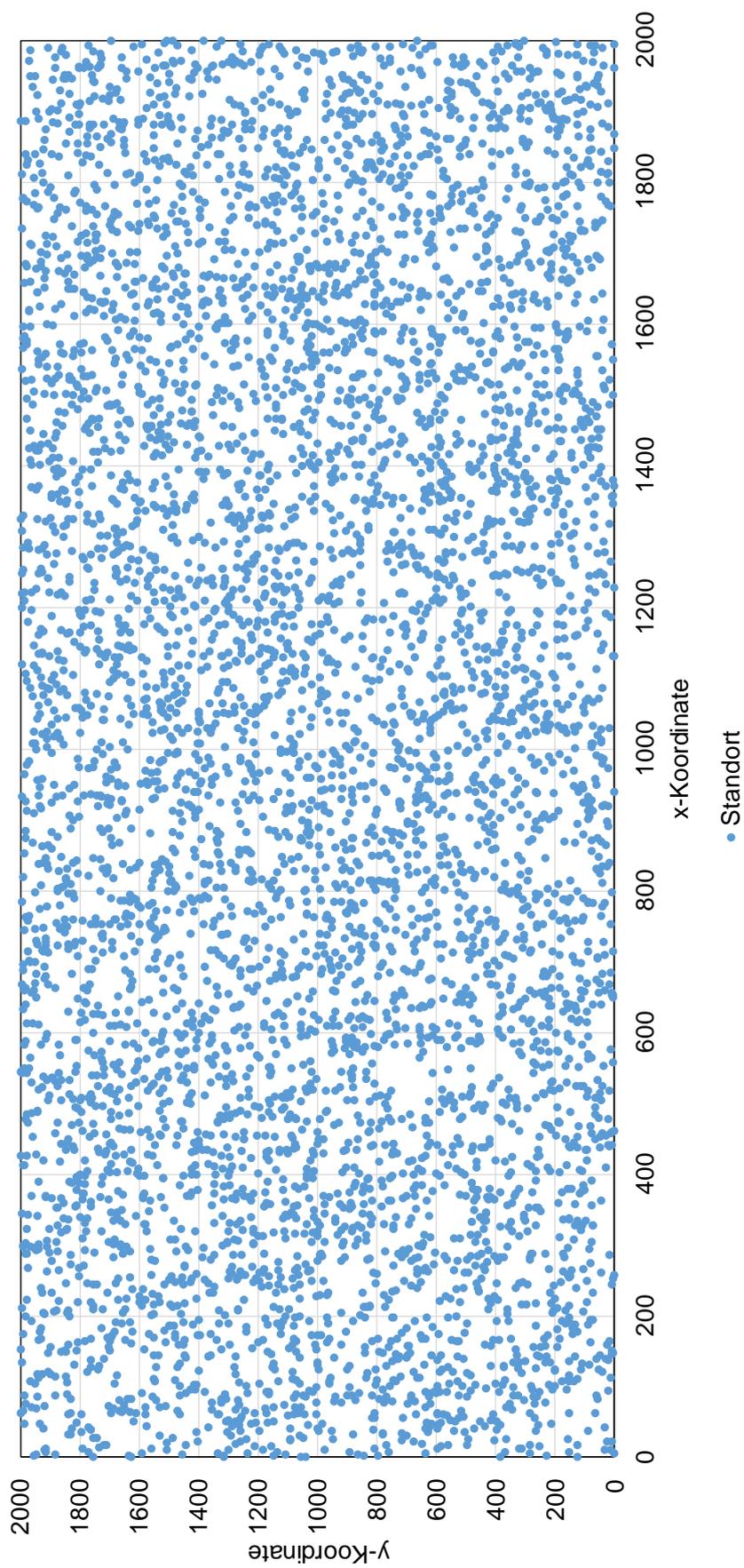


Abbildung 78: Standortverteilung im Testset Antwort1 Replikation Nr. 17 Gleichverteilung

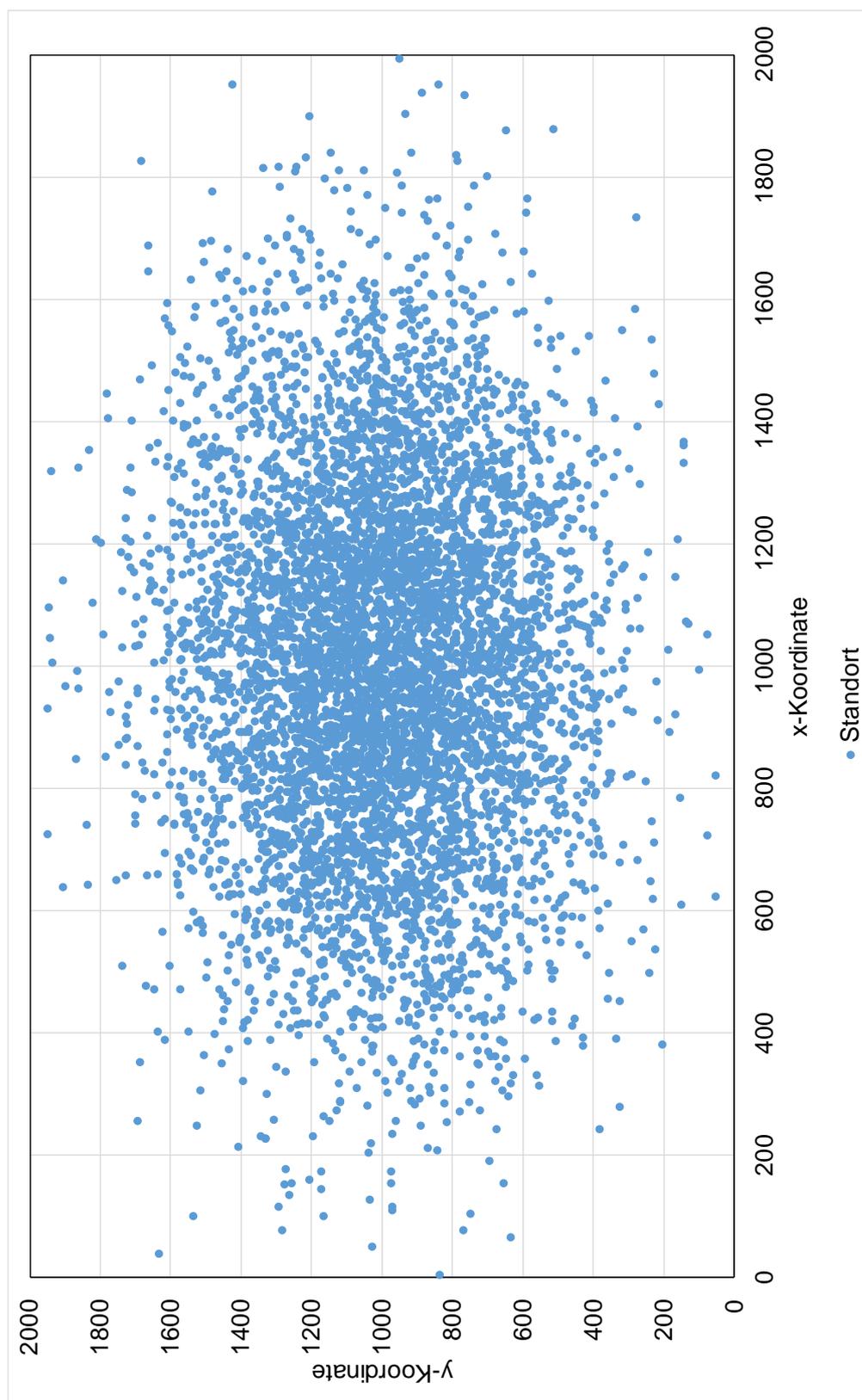


Abbildung 79: Standortverteilung im Testset Antwerp1 Replikation Nr. 17 Normalverteilung

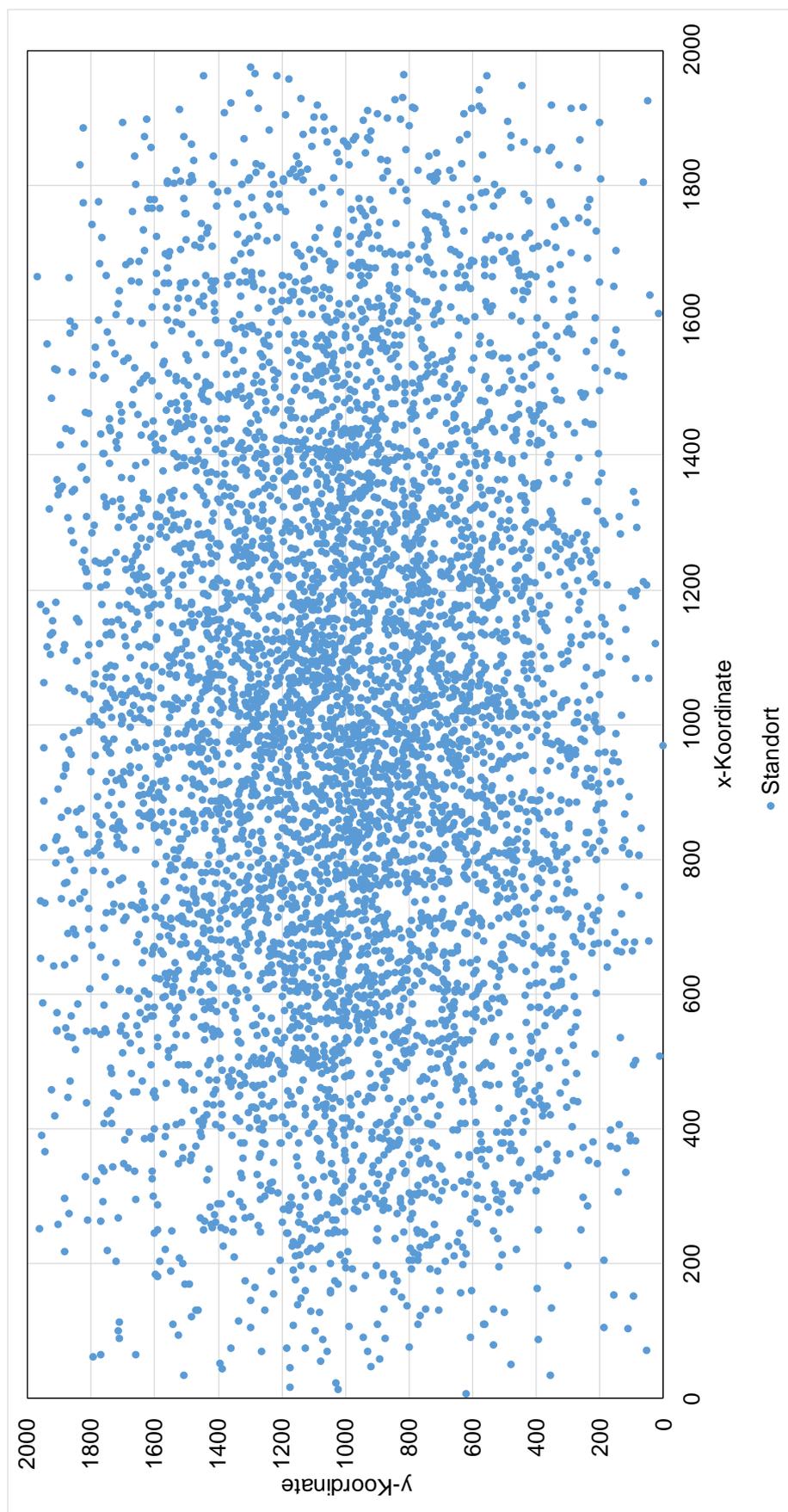


Abbildung 80: Standortverteilung im Testset Antwerp1 Replikation Nr. 17 Dreiecksverteilung

Anhang C: Code

Modul 1

Option Explicit

```
'Quelle: Banks et al. 2014b
' This module contains a (very) long period pseudorandom number generator
(MRG32k3a)
' and functions for most of the random-variate generation examples in Chapter 8 of
' Banks, Carson, Nelson and Nicol, Discrete-Event System Simulation, 5th edition.
'
' The pseudo-random generator MRG32k3a requires 6 initial seeds which is done by the
sub InitializeRNSeed
' This sub must be called prior to the first call to MRG32k3a
```

```
Const norm = 2.32830654929573E-10
Const m1 = 4294967087#
Const m2 = 4294944443#
Const a12 = 1403580#
Const a13n = 810728#
Const a21 = 527612#
Const a23n = 1370589#
```

```
Dim s10 As Double
Dim s11 As Double
Dim s12 As Double
Dim s20 As Double
Dim s21 As Double
Dim s22 As Double
```

```
Dim s10_2 As Double
Dim s11_2 As Double
Dim s12_2 As Double
Dim s20_2 As Double
Dim s21_2 As Double
Dim s22_2 As Double
Private Declare PtrSafe Function GetTickCount Lib "kernel32.dll" () As Long
```

```
Public Sub InitializeRNSeed(replication As Integer)
```

```
'Quelle: Banks et al. 2014b
' this function must be called before using MRG32k3a or any of the
' variate generation functions. The first three seeds must be
' between 0 and 4294967087 but not all 0; the second 3 seeds must be
' between 0 and 4294944443 but not all 0.
```

'and self adapted

```
Dim wb As Workbook
Set wb = ThisWorkbook
Dim ws_Seeds As Worksheet
Set ws_Seeds = wb.Sheets("Seeds")
```

```
s10 = ws_Seeds.Cells(replication + 1, 2).value
s11 = ws_Seeds.Cells(replication + 1, 3).value
s12 = ws_Seeds.Cells(replication + 1, 4).value
s20 = ws_Seeds.Cells(replication + 1, 6).value
s21 = ws_Seeds.Cells(replication + 1, 7).value
s22 = ws_Seeds.Cells(replication + 1, 8).value
```

End Sub

Public Function MRG32k3a() As Double

'Quelle: Banks et al. 2014b

' The following pseudo random number generator is a VBA translation of the C program found in

' P. L'Ecuyer, ``Good Parameter Sets for Combined Multiple Recursive Random Number Generators",

' Operations Research, 47, 1 (1999), 159--164 and also in

' P. L'Ecuyer, R. Simard, E. J. Chen, W. D. Kelton, "An Object-oriented Random-number Package with

' Many Long Streams and Substreams," Operations Research, 50, 6 (2002), 1073--1075.

,

```
Dim k As Long
Dim P1 As Double
Dim P2 As Double
'Component 1
P1 = a12 * s11 - a13n * s10
k = P1 / m1
P1 = P1 - k * m1
If (P1 < 0#) Then P1 = P1 + m1
s10 = s11
s11 = s12
s12 = P1
```

```
'Component 2
P2 = a21 * s22 - a23n * s20
k = P2 / m2
P2 = P2 - k * m2
If (P2 < 0#) Then P2 = P2 + m2
s20 = s21
s21 = s22
s22 = P2
```

'Combination

```

If (P1 <= P2) Then
    MRG32k3a = ((P1 - P2 + m1) * norm)
Else
    MRG32k3a = ((P1 - P2) * norm)
End If
End Function

Public Function Uniform(ByVal a As Double, ByVal b As Double) As Double
'Quelle: Banks et al. 2014b
' function to generate U(a, b) variates via the inverse cdf method

    Uniform = a + (b - a) * MRG32k3a()

End Function

Public Function DiscreteUniform(ByVal a As Double, ByVal b As Double) As Double
'Quelle: Law 2015 S. 464
' function to generate a discrete uniform on 1, 2, ..., k via the inverse cdf method

    DiscreteUniform = (a + WorksheetFunction.Floor(MRG32k3a * (b - a + 1), 1))

End Function

Public Sub generate_discrete_uniform()

    Dim lTime As Long
    Dim evalTime As Long
    Dim totalTime As Long
    Dim avg_time As Long
    With Application
        .EnableEvents = False
        .Calculation = xlCalculationManual
        .ScreenUpdating = False
    End With

    Dim wb As Workbook
    Set wb = ThisWorkbook
    Dim wsG As Worksheet
    Set wsG = wb.Sheets("One_Generator")
    Dim wsP As Worksheet
    Set wsP = wb.Sheets("Parameter")
    wsG.Range("A3:F100000").ClearContents
    wsG.Range("H3:N5").ClearContents
    wsG.Range("I7:I12").ClearContents
    wsG.Range("I15:I18").ClearContents
    wsG.Range("I22:M24").ClearContents
    wsG.Range("O3:P100000").ClearContents

    Dim x_min As Double
    Dim x_max As Double
    Dim y_min As Double

```

```
Dim y_max As Double

Dim f1_error, f2_error, f3_error As Long
f1_error = 0
f2_error = 0
f3_error = 0

Dim replication As Integer
Dim replication_nr As Integer
Dim i As Integer
Dim p As Integer
Dim customer As Integer
Dim x As Integer
Dim y As Integer
Dim dc_x, dc_y As Integer
Dim norm_error As Long
Dim outofbounce_error As Long

Dim distance As Double
Dim fulldistance As Double
Dim disav As Double

Dim already_used_coordinates As Object
Set already_used_coordinates = CreateObject("Scripting.Dictionary")
Dim coordinate_string As String
Dim found_valid As Boolean
Dim already_used_error As Integer

x_min = wsP.Range("E11").value
x_max = wsP.Range("E12").value
y_min = wsP.Range("E13").value
y_max = wsP.Range("E14").value

replication_nr = wsP.Range("B7").value
customer = wsP.Range("B8").value
dc_x = wsP.Range("B11").value
dc_y = wsP.Range("B12").value
replication = 1
disav = 0
totalTime = GetTickCount
i = 3

Do
    lTime = GetTickCount
    fulldistance = 0
    InitializeRNSeed (replication)
    already_used_error = 0
    p = 1
```

```

Do
  found_valid = False

Do
  x = DiscreteUniform(x_min, x_max)
  y = DiscreteUniform(y_min, y_max)

  coordinate_string = CStr(x) & "/" & CStr(y)

  If Not already_used_coordinates.exists(coordinate_string) Then
    found_valid = True
    already_used_coordinates.Add coordinate_string, coordinate_string
    distance = 1.234 * Sqr((x - dc_x) ^ 2 + (y - dc_y) ^ 2)
    fulldistance = fulldistance + distance
    wsG.Cells(p, 20).value = x
    wsG.Cells(p, 21).value = y
    p = p + 1
  Else
    already_used_error = already_used_error + 1

  End If
Loop Until found_valid = True

Loop Until customer = already_used_coordinates.Count

f1_error = already_used_error + f1_error
f2_error = outofbounce_error + f2_error
f3_error = norm_error + f3_error

disav = disav + fulldistance
wsG.Cells(i, 1).value = replication
wsG.Cells(i, 2).value = fulldistance
lTime = GetTickCount - lTime
avg_time = avg_time + lTime
wsG.Cells(i, 3).value = (lTime / 1000)
wsG.Cells(i, 4).value = already_used_error
already_used_coordinates.RemoveAll
lTime = 0
replication = replication + 1
i = i + 1
Loop Until replication = replication_nr + 1

evalTime = GetTickCount
Call create_boxplot(replication_nr, disav, avg_time, f1_error, f2_error, f3_error)
evalTime = GetTickCount - evalTime
totalTime = GetTickCount - totalTime
wsG.Range("I8").value = (evalTime / 1000)

```

```
wsG.Range("I7").value = (totalTime / 1000)
wsG.Range("T:U").ClearContents
With Application
    .EnableEvents = True
    .Calculation = xlCalculationAutomatic
    .ScreenUpdating = True
End With
End Sub
Public Sub construct_distance_kpi(ByVal replication_nr As Integer)

    Dim wb As Workbook
    Set wb = ThisWorkbook
    Dim wsG As Worksheet
    Set wsG = wb.Sheets("One_Generator")
    Dim wsP As Worksheet
    Set wsP = wb.Sheets("Parameter")
    Dim construct_kpi_laufp As Integer
    Dim distance_kpi As Double
    Dim item As Variant

    Dim sort_distance As Object
    Set sort_distance = CreateObject("Scripting.Dictionary")
    Dim dict_i As Long
    sort_distance.RemoveAll

    For construct_kpi_laufp = 1 To replication_nr

        distance_kpi = wsG.Cells(construct_kpi_laufp + 2, 2).value

        sort_distance.Add distance_kpi, distance_kpi

    Next
    construct_kpi_laufp = 3
    Set sort_distance = SortDictionaryByValue(sort_distance)

    For Each item In sort_distance
        wsG.Cells(construct_kpi_laufp, 7).value = sort_distance(item)
        construct_kpi_laufp = construct_kpi_laufp + 1
    Next item

    sort_distance.RemoveAll
End Sub

Public Function SortDictionaryByValue(dict As Object _
    , Optional sortorder As XlSortOrder = xlAscending) As Object
' From https://excelmacromastery.com/vba-dictionary/#Sorting_the_Dictionary

    On Error GoTo eh

    Dim arrayList As Object
```

```
Set arrayList = CreateObject("System.Collections.ArrayList")

Dim dictTemp As Object
Set dictTemp = CreateObject("Scripting.Dictionary")

' Put values in ArrayList and sort
' Store values in tempDict with their keys as a collection
Dim key As Variant, value As Variant, coll As Collection
For Each key In dict

    value = dict(key)

    ' if the value doesn't exist in dict then add
    If dictTemp.exists(value) = False Then
        ' create collection to hold keys
        ' - needed for duplicate values
        Set coll = New Collection
        dictTemp.Add value, coll

        ' Add the value
        arrayList.Add value

    End If

    ' Add the current key to the collection
    dictTemp(value).Add key

Next key

' Sort the value
arrayList.Sort

' Reverse if descending
If sortorder = xlDescending Then
    arrayList.Reverse
End If

dict.RemoveAll

' Read through the ArrayList and add the values and corresponding
' keys from the dictTemp
Dim item As Variant
For Each value In arrayList
    Set coll = dictTemp(value)
    For Each item In coll
        dict.Add item, value
    Next item
Next value

Set arrayList = Nothing
```

```

' Return the new dictionary
Set SortDictionaryByValue = dict

Done:
Exit Function
eh:
If Err.Number = 450 Then
    Err.Raise vbObjectError + 100, "SortDictionaryByValue" _
        , "Cannot sort the dictionary if the value is an object"
End If
End Function

Public Sub create_boxplot(ByVal replication_nr As Integer, ByVal disav As Double,
ByVal avg_time As Double, ByVal f1_error As Long, ByVal f2_error As Long, ByVal
f3_error As Long)

Dim wb As Workbook
Set wb = ThisWorkbook
Dim wsG As Worksheet
Set wsG = wb.Sheets("One_Generator")
'Dim wsP As Worksheet
'Set wsP = wb.Sheets("Parameter")

Dim lquant As Integer
Dim uquant As Integer
Dim lquantval As Double
Dim uquantval As Double
Dim lquantd As Double
Dim uquantd As Double
Dim dq As Double
Dim median As Integer
Dim median_dummy As Integer
Dim median_dummy2 As Integer
Dim median_bool As Boolean
median_bool = False

Dim average As Double
'Dim disav As Double
Dim laufp_b As Integer
Dim laufp_b2 As Integer
laufp_b2 = 3
Dim laufp_b3 As Integer
laufp_b3 = 3

Dim replication_nr_dummy As Integer

Dim s_var As Double
s_var = 0
Dim s_abw As Double
s_abw = 0
Dim laufp_k, laufp_k2 As Integer

```

```

Dim x_i As Double
x_i = 0
Dim konf_left, konf_right As Double
konf_left = 0
konf_right = 0
'Dim replication_nr As Integer
'replication_nr = wsP.Range("B7").value
'disav = wsG.Range("V1").value
'WorksheetFunction.Ceiling

Call construct_distance_kpi(replication_nr)

average = (disav / replication_nr)
wsG.Range("H3").value = Round(average, 2)

If replication_nr Mod 2 = 0 Then
median_bool = True
Else
median_bool = False
End If

    If median_bool = True Then

        median_dummy = (replication_nr / 2)
        median_dummy2 = (replication_nr / 2) + 1
        wsG.Range("I3").value = Round(1 / 2 * (wsG.Cells(median_dummy + 2, 7).value +
wsG.Cells(median_dummy2 + 2, 7)), 2)

    Else

        median = ((replication_nr + 1) / 2)
        wsG.Range("I3").value = Round(wsG.Cells(median + 2, 7).value, 2)

    End If
'boxplot
lquant = Round(0.25 * replication_nr, 0)
uquant = Round(0.75 * replication_nr, 0)
lquantval = wsG.Cells(lquant + 2, 7).value
uquantval = wsG.Cells(uquant + 2, 7).value
wsG.Range("K3").value = Round(lquantval, 2)
wsG.Range("L3").value = Round(uquantval, 2)
dq = (uquantval - lquantval)
wsG.Range("N3").value = Round(dq, 2)
lquantd = lquantval - (1.5 * dq)
uquantd = uquantval + (1.5 * dq)
wsG.Range("J3").value = Round(lquantd, 2)
wsG.Range("M3").value = Round(uquantd, 2)

For laufp_b = 3 To replication_nr + 2

```

```

If wsG.Cells(laufp_b, 7).value < lquantd Then

    wsG.Cells(laufp_b2, 15).value = Round(wsG.Cells(laufp_b, 7).value, 2)
    laufp_b2 = laufp_b2 + 1
Else
    If wsG.Cells(laufp_b, 7).value > uquantd Then
        wsG.Cells(laufp_b3, 16).value = Round(wsG.Cells(laufp_b, 7).value, 2)
        laufp_b3 = laufp_b3 + 1
    End If
End If
Next

's_var
laufp_k = 3
laufp_k2 = 1

For laufp_k2 = 1 To replication_nr + 1

    x_i = wsG.Cells(laufp_k, 7).value

    s_abw = s_abw + ((x_i - average) ^ 2)
    laufp_k = laufp_k + 1
Next

s_var = Sqr((1 / (replication_nr - 1)) * s_abw)
konf_left = Round(average - (1.96 * (s_var / Sqr(replication_nr))), 2)
konf_right = Round(average + (1.96 * (s_var / Sqr(replication_nr))), 2)

wsG.Range("I15").value = "(" & CStr(konf_left) & ";" & CStr(konf_right) & ")"
wsG.Range("I16").value = konf_left
wsG.Range("I17").value = konf_right
wsG.Range("I18").value = konf_right - konf_left
wsG.Range("G:G").ClearContents
'time KPI
laufp_k = 3
laufp_k2 = 1

For laufp_k2 = 1 To replication_nr + 1

    wsG.Cells(laufp_k, 7).value = wsG.Cells(laufp_k, 3).value
    laufp_k = laufp_k + 1
Next

replication_nr_dummy = replication_nr + 2

wsG.Range("G3:G" & replication_nr_dummy).Sort Key1:=wsG.Range("G3"),
Order1:=xlAscending

average = (avg_time / replication_nr / 1000)
wsG.Range("I12").value = Round(average, 3)

```

```

'median = Round((replication_nr / 2), 0)
'wsG.Range("I11").value = Round(wsG.Cells(median + 2, 7).value, 3)

    If median_bool = True Then

        median_dummy = (replication_nr / 2)
        median_dummy2 = (replication_nr / 2) + 1
        wsG.Range("I11").value = Round(1 / 2 * (wsG.Cells(median_dummy + 2, 7).value +
wsG.Cells(median_dummy2 + 2, 7)), 3)

    Else

        median = ((replication_nr + 1) / 2)
        wsG.Range("I11").value = Round(wsG.Cells(median + 2, 7).value, 3)

    End If
wsG.Range("I9").value = wsG.Range("G3").value
wsG.Range("I10").value = wsG.Range("G" & replication_nr_dummy).value
wsG.Range("G:G").ClearContents

'double location KPI
laufp_k = 3
laufp_k2 = 1

For laufp_k2 = 1 To replication_nr + 1

wsG.Cells(laufp_k, 7).value = wsG.Cells(laufp_k, 4).value
laufp_k = laufp_k + 1
Next

wsG.Range("G3:G" & replication_nr_dummy).Sort Key1:=wsG.Range("G3"),
Order1:=xlAscending

wsG.Range("I22").value = f1_error
average = (f1_error / replication_nr)
wsG.Range("M22").value = Round(average, 2)

    If median_bool = True Then

        median_dummy = (replication_nr / 2)
        median_dummy2 = (replication_nr / 2) + 1
        wsG.Range("L22").value = Round(1 / 2 * (wsG.Cells(median_dummy + 2, 7).value +
wsG.Cells(median_dummy2 + 2, 7)), 3)

    Else

        median = ((replication_nr + 1) / 2)
        wsG.Range("L22").value = Round(wsG.Cells(median + 2, 7).value, 3)

    End If
wsG.Range("J22").value = wsG.Range("G3").value

```

```
wsG.Range("K22").value = wsG.Range("G" & replication_nr_dummy).value
wsG.Range("G:G").ClearContents

'outofbounce KPI
laufp_k = 3
laufp_k2 = 1

For laufp_k2 = 1 To replication_nr + 1

wsG.Cells(laufp_k, 7).value = wsG.Cells(laufp_k, 5).value
laufp_k = laufp_k + 1
Next

wsG.Range("G3:G" & replication_nr_dummy).Sort Key1:=wsG.Range("G3"),
Order1:=xlAscending

wsG.Range("I23").value = f2_error
average = (f2_error / replication_nr)
wsG.Range("M23").value = Round(average, 2)

    If median_bool = True Then

        median_dummy = (replication_nr / 2)
        median_dummy2 = (replication_nr / 2) + 1
        wsG.Range("L23").value = Round(1 / 2 * (wsG.Cells(median_dummy + 2, 7).value +
wsG.Cells(median_dummy2 + 2, 7)), 3)

    Else

        median = ((replication_nr + 1) / 2)
        wsG.Range("L23").value = Round(wsG.Cells(median + 2, 7).value, 3)

    End If
wsG.Range("J23").value = wsG.Range("G3").value
wsG.Range("K23").value = wsG.Range("G" & replication_nr_dummy).value
wsG.Range("G:G").ClearContents

'norm error KPI
laufp_k = 3
laufp_k2 = 1

For laufp_k2 = 1 To replication_nr + 1

wsG.Cells(laufp_k, 7).value = wsG.Cells(laufp_k, 6).value
laufp_k = laufp_k + 1
Next

wsG.Range("G3:G" & replication_nr_dummy).Sort Key1:=wsG.Range("G3"),
Order1:=xlAscending

wsG.Range("I24").value = f3_error
```

```
average = (f3_error / replication_nr)
wsG.Range("M24").value = Round(average, 2)

If median_bool = True Then

    median_dummy = (replication_nr / 2)
    median_dummy2 = (replication_nr / 2) + 1
    wsG.Range("L24").value = Round(1 / 2 * (wsG.Cells(median_dummy + 2, 7).value +
wsG.Cells(median_dummy2 + 2, 7)), 3)

Else

    median = ((replication_nr + 1) / 2)
    wsG.Range("L24").value = Round(wsG.Cells(median + 2, 7).value, 3)

End If
wsG.Range("J24").value = wsG.Range("G3").value
wsG.Range("K24").value = wsG.Range("G" & replication_nr_dummy).value
wsG.Range("G:G").ClearContents

End Sub
```

Modul 2

Option Explicit

```
Private Declare PtrSafe Function GetTickCount Lib "kernel32.dll" () As Long
```

```
Public Sub generate_normal()
```

```
    Dim lTime As Long
```

```
    Dim evalTime As Long
```

```
    Dim totalTime As Long
```

```
    Dim avg_time As Long
```

```
    With Application
```

```
        .EnableEvents = False
```

```
        .Calculation = xlCalculationManual
```

```
.ScreenUpdating = False

End With

Dim wb As Workbook

Set wb = ThisWorkbook

Dim wsG As Worksheet

Set wsG = wb.Sheets("One_Generator")

Dim wsP As Worksheet

Set wsP = wb.Sheets("Parameter")

wsG.Range("A3:F100000").ClearContents

wsG.Range("H3:N5").ClearContents

wsG.Range("I7:I12").ClearContents

wsG.Range("I15:I18").ClearContents

wsG.Range("I22:M24").ClearContents

wsG.Range("O3:P100000").ClearContents

Dim x_min As Double

Dim x_max As Double

Dim y_min As Double

Dim y_max As Double

Dim f1_error, f2_error, f3_error As Long

f1_error = 0

f2_error = 0
```

```
f3_error = 0
```

```
Dim replication As Integer
```

```
Dim replication_nr As Integer
```

```
Dim customer As Integer
```

```
Dim x As Integer
```

```
Dim y As Integer
```

```
Dim dc_x, dc_y As Integer
```

```
Dim lauf_x As Integer
```

```
Dim laufp_x As Integer
```

```
Dim lauf_y As Integer
```

```
Dim laufp_y As Integer
```

```
Dim mu_x As Double
```

```
Dim sig_sq_x As Double
```

```
Dim mu_y As Double
```

```
Dim sig_sq_y As Double
```

```
Dim norm_error As Long
```

```
Dim outofbounce_error As Long
```

```
Dim distance As Double
```

```
Dim fulldistance As Double
```

```
Dim disav As Double
```

```
Dim i As Integer

Dim norm_coordx As Double

Dim norm_coordy As Double

Dim already_used_coordinates As Object

Set already_used_coordinates = CreateObject("Scripting.Dictionary")

Dim coordinate_string As String

Dim found_valid As Boolean

Dim already_used_error As Long

x_min = wsP.Range("H11").value
x_max = wsP.Range("H12").value
y_min = wsP.Range("H13").value
y_max = wsP.Range("H14").value
mu_x = wsP.Range("H17").value
sig_sq_x = wsP.Range("H18").value
mu_y = wsP.Range("H20").value
sig_sq_y = wsP.Range("H21").value

replication = 1

replication_nr = wsP.Range("B7").value

customer = wsP.Range("B8").value

dc_x = wsP.Range("B11").value
```

```
dc_y = wsP.Range("B12").value
```

```
disav = 0
```

```
totalTime = GetTickCount
```

```
Do
```

```
    lTime = GetTickCount
```

```
    fulldistance = 0
```

```
    InitializeRNSeed (replication)
```

```
    already_used_error = 0
```

```
    norm_error = 0
```

```
    outofbounce_error = 0
```

```
    i = 1
```

```
        Do
```

```
            found_valid = False
```

```
                Do
```

```
                    Modul3.normal mu_x, mu_y, sig_sq_x, sig_sq_y, norm_error,  
norm_coordx, norm_coordy
```

```
                        x = Round(norm_coordx, 0)
```

```
                        y = Round(norm_coordy, 0)
```

```
If x > x_min And x < x_max And y > y_min And y < y_max Then
```

```
coordinate_string = CStr(x) & "/" & CStr(y)
```

```
If Not already_used_coordinates.exists(coordinate_string) Then
```

```
found_valid = True
```

```
already_used_coordinates.Add coordinate_string, coordinate_string
```

```
distance = 1.234 * Sqr((x - dc_x) ^ 2 + (y - dc_y) ^ 2)
```

```
fulldistance = fulldistance + distance
```

```
' wsG.Cells(i, 20).value = x
```

```
' wsG.Cells(i, 21).value = y
```

```
' i = i + 1
```

```
Else
```

```
already_used_error = already_used_error + 1
```

```
End If
```

```
Else
```

```
outofbounce_error = outofbounce_error + 1
```

```
End If
```

```
Loop Until found_valid = True
```

```
Loop Until customer = already_used_coordinates.Count
```

```
f1_error = already_used_error + f1_error
```

```
f2_error = outofbounce_error + f2_error

f3_error = norm_error + f3_error

replication = replication + 1

disav = disav + fulldistance

wsG.Cells(replication + 1, 1).value = replication - 1

wsG.Cells(replication + 1, 2).value = fulldistance

lTime = GetTickCount - lTime

avg_time = avg_time + lTime

wsG.Cells(replication + 1, 3).value = (lTime / 1000)

wsG.Cells(replication + 1, 4).value = already_used_error

wsG.Cells(replication + 1, 5).value = outofbounce_error

wsG.Cells(replication + 1, 6).value = norm_error

already_used_coordinates.RemoveAll

lTime = 0

Loop Until replication = replication_nr + 1

' MsgBox ("Generated 100 coordinates, with " & CStr(already_used_error) & " new
generations")

' MsgBox (CStr(already_used_coordinates.Count))

' MsgBox (CStr(already_used_coordinates.Count))

evalTime = GetTickCount
```

```
Call create_boxplot(replication_nr, disav, avg_time, f1_error, f2_error, f3_error)
```

```
evalTime = GetTickCount - evalTime
```

```
totalTime = GetTickCount - totalTime
```

```
wsG.Range("I8").value = (evalTime / 1000)
```

```
wsG.Range("I7").value = (totalTime / 1000)
```

```
wsG.Range("G:G").ClearContents
```

```
wsG.Range("T:U").ClearContents
```

```
With Application
```

```
    .EnableEvents = True
```

```
    .Calculation = xlCalculationAutomatic
```

```
    .ScreenUpdating = True
```

```
End With
```

```
End Sub
```

Modul 3

```
Option Explicit
```

```
Public Sub normal(ByVal mu_x As Double, ByVal mu_y As Double, ByVal sig_sq_x  
As Double, ByVal sig_sq_y As Double, ByRef norm_error As Long, ByRef result_x As  
Double, ByRef result_y As Double)
```

```
Dim g1 As Double
```

```
Dim g2 As Double
```

```
Dim s1 As Double
```

```
Dim s2 As Double
```

```
Dim w_dec As Double
```

```
Dim dummy_x As Double
```

```
Dim dummy_y As Double
```

```
Dim dummy_ln As Double
```

```
Dim norm_coordx As Double
```

```
Dim norm_coordy As Double
```

```
Do
```

```
    g1 = MRG32k3a()
```

```
    g2 = MRG32k3a()
```

```
    s1 = 2 * g1 - 1
```

```
    s2 = 2 * g2 - 1
```

```
    w_dec = (s1) ^ 2 + (s2) ^ 2
```

```
    If w_dec >= 1 Then
```

```
        norm_error = norm_error + 1
```

```
    End If
```

```
Loop Until w_dec <= 1
```

```
dummy_ln = Sqr((-2 * Application.WorksheetFunction.Ln(w_dec)) / w_dec)
```

```
dummy_x = dummy_ln * s1
```

```
dummy_y = dummy_ln * s2
```

```
result_x = mu_x + dummy_x * sig_sq_x
```

```
result_y = mu_y + dummy_y * sig_sq_y
```

```
End Sub
```

Modul 4

```
Option Explicit
```

```
Private Declare PtrSafe Function GetTickCount Lib "kernel32.dll" () As Long
```

```
Public Sub generate_discrete_uniform_repeat()
```

```
    Dim lTime As Long
```

```
    Dim evalTime As Long
```

```
    Dim totalTime As Long
```

```
    Dim avg_time As Long
```

```
    With Application
```

```
        .EnableEvents = False
```

```
        .Calculation = xlCalculationManual
```

```
        .ScreenUpdating = False
```

```
    End With
```

```
Dim wb As Workbook

Set wb = ThisWorkbook

Dim wsRP As Worksheet

Set wsRP = wb.Sheets("Repeat_Replication")

wsRP.Range("J3:Q100000").ClearContents

Dim x_min As Double

Dim x_max As Double

Dim y_min As Double

Dim y_max As Double

Dim f1_error, f2_error, f3_error As Long

f1_error = 0

f2_error = 0

f3_error = 0

Dim replication As Integer

Dim replication_nr As Integer

Dim p As Integer

Dim customer As Integer

Dim x As Integer

Dim y As Integer

Dim dc_x, dc_y As Integer
```

```
Dim norm_error As Long
```

```
Dim outofbounce_error As Long
```

```
Dim distance As Double
```

```
Dim fulldistance As Double
```

```
Dim disav As Double
```

```
Dim already_used_coordinates As Object
```

```
Set already_used_coordinates = CreateObject("Scripting.Dictionary")
```

```
Dim coordinate_string As String
```

```
Dim found_valid As Boolean
```

```
Dim already_used_error As Integer
```

```
x_min = wsRP.Range("E8").value
```

```
x_max = wsRP.Range("E9").value
```

```
y_min = wsRP.Range("E10").value
```

```
y_max = wsRP.Range("E11").value
```

```
replication_nr = wsRP.Range("B4").value
```

```
customer = wsRP.Range("B5").value
```

```
dc_x = wsRP.Range("B8").value
```

```
dc_y = wsRP.Range("B9").value
```

```
replication = 0
```

```
disav = 0
```

```
totalTime = GetTickCount
```

```
replication = replication_nr + replication
```

```
lTime = GetTickCount
```

```
fulldistance = 0
```

```
InitializeRNSeed (replication)
```

```
already_used_error = 0
```

```
p = 3
```

```
Do
```

```
    found_valid = False
```

```
Do
```

```
    x = DiscreteUniform(x_min, x_max)
```

```
    y = DiscreteUniform(y_min, y_max)
```

```
    coordinate_string = CStr(x) & "/" & CStr(y)
```

```
If Not already_used_coordinates.exists(coordinate_string) Then

    found_valid = True

    already_used_coordinates.Add coordinate_string, coordinate_string

    distance = 1.234 * Sqr((x - dc_x) ^ 2 + (y - dc_y) ^ 2)

    fulldistance = fulldistance + distance

    wsRP.Cells(p, 16).value = x

    wsRP.Cells(p, 17).value = y

    p = p + 1

Else

    already_used_error = already_used_error + 1

End If

Loop Until found_valid = True

Loop Until customer = already_used_coordinates.Count

f1_error = already_used_error + f1_error

f2_error = outofbounce_error + f2_error

f3_error = norm_error + f3_error

wsRP.Range("J3").value = replication

wsRP.Range("K3").value = fulldistance

lTime = GetTickCount - lTime

wsRP.Range("L3").value = (lTime / 1000)
```

```
wsRP.Range("M3").value = already_used_error
```

```
already_used_coordinates.RemoveAll
```

```
With Application
```

```
    .EnableEvents = True
```

```
    .Calculation = xlCalculationAutomatic
```

```
    .ScreenUpdating = True
```

```
End With
```

```
End Sub
```

```
Public Sub generate_normal_repeat()
```

```
    Dim lTime As Long
```

```
    Dim evalTime As Long
```

```
    Dim totalTime As Long
```

```
    Dim avg_time As Long
```

```
With Application
```

```
    .EnableEvents = False
```

```
    .Calculation = xlCalculationManual
```

```
    .ScreenUpdating = False
```

```
End With
```

```
Dim wb As Workbook
```

```
Set wb = ThisWorkbook
```

```
Dim wsRP As Worksheet  
Set wsRP = wb.Sheets("Repeat_Replication")  
wsRP.Range("J3:Q100000").ClearContents
```

```
Dim x_min As Double
```

```
Dim x_max As Double
```

```
Dim y_min As Double
```

```
Dim y_max As Double
```

```
Dim f1_error, f2_error, f3_error As Long
```

```
f1_error = 0
```

```
f2_error = 0
```

```
f3_error = 0
```

```
Dim replication As Integer
```

```
Dim replication_nr As Integer
```

```
Dim customer As Integer
```

```
Dim x As Integer
```

```
Dim y As Integer
```

```
Dim dc_x, dc_y As Integer
```

```
Dim mu_x As Double
```

```
Dim sig_sq_x As Double
```

```
Dim mu_y As Double
```

```
Dim sig_sq_y As Double
```

```
Dim norm_error As Long
```

```
Dim outofbounce_error As Long
```

```
Dim distance As Double
```

```
Dim fulldistance As Double
```

```
Dim disav As Double
```

```
Dim i As Integer
```

```
Dim norm_coordx As Double
```

```
Dim norm_coordy As Double
```

```
Dim already_used_coordinates As Object
```

```
Set already_used_coordinates = CreateObject("Scripting.Dictionary")
```

```
Dim coordinate_string As String
```

```
Dim found_valid As Boolean
```

```
Dim already_used_error As Long
```

```
x_min = wsRP.Range("H8").value
```

```
x_max = wsRP.Range("H9").value
```

```
y_min = wsRP.Range("H10").value
```

```
y_max = wsRP.Range("H11").value
```

```
mu_x = wsRP.Range("H14").value
```

```
sig_sq_x = wsRP.Range("H15").value  
mu_y = wsRP.Range("H17").value  
sig_sq_y = wsRP.Range("H18").value  
  
replication = 0  
replication_nr = wsRP.Range("B4").value  
customer = wsRP.Range("B5").value  
dc_x = wsRP.Range("B8").value  
dc_y = wsRP.Range("B9").value
```

```
lTime = GetTickCount  
replication = replication_nr + replication  
fulldistance = 0  
InitializeRNSeed (replication)  
already_used_error = 0  
norm_error = 0  
outofbounce_error = 0  
i = 3
```

```
Do
```

```
    found_valid = False
```

```
Do
```

```
Modul3.normal mu_x, mu_y, sig_sq_x, sig_sq_y, norm_error,  
norm_coordx, norm_coordy
```

```
x = Round(norm_coordx, 0)
```

```
y = Round(norm_coordy, 0)
```

```
If x > x_min And x < x_max And y > y_min And y < y_max Then
```

```
coordinate_string = CStr(x) & "/" & CStr(y)
```

```
If Not already_used_coordinates.exists(coordinate_string) Then
```

```
found_valid = True
```

```
already_used_coordinates.Add coordinate_string, coordinate_string
```

```
distance = 1.234 * Sqr((x - dc_x) ^ 2 + (y - dc_y) ^ 2)
```

```
fulldistance = fulldistance + distance
```

```
wsRP.Cells(i, 16).value = x
```

```
wsRP.Cells(i, 17).value = y
```

```
i = i + 1
```

```
Else
```

```
already_used_error = already_used_error + 1
```

```
End If
```

```
Else
```

```
outofbounce_error = outofbounce_error + 1
```

```
        End If

        Loop Until found_valid = True

        Loop Until customer = already_used_coordinates.Count

f1_error = already_used_error + f1_error

f2_error = outofbounce_error + f2_error

f3_error = norm_error + f3_error

wsRP.Range("J3").value = replication

wsRP.Range("K3").value = fulldistance

lTime = GetTickCount - lTime

wsRP.Range("L3").value = (lTime / 1000)

wsRP.Range("M3").value = already_used_error

wsRP.Range("N3").value = outofbounce_error

wsRP.Range("O3").value = norm_error

already_used_coordinates.RemoveAll

With Application

    .EnableEvents = True

    .Calculation = xlCalculationAutomatic

    .ScreenUpdating = True

End With

End Sub
```

```
Public Sub generate_triangular_repeat()

    Dim lTime As Long

    Dim evalTime As Long

    Dim totalTime As Long

    Dim avg_time As Long

    With Application

        .EnableEvents = False

        .Calculation = xlCalculationManual

        .ScreenUpdating = False

    End With

    Dim wb As Workbook

    Set wb = ThisWorkbook

    Dim wsRP As Worksheet

    Set wsRP = wb.Sheets("Repeat_Replication")

    wsRP.Range("J3:Q100000").ClearContents

    Dim x_min As Double

    Dim x_max As Double

    Dim y_min As Double

    Dim y_max As Double

    Dim x_prob As Double
```

```
Dim y_prob As Double
```

```
Dim f1_error, f2_error, f3_error As Long
```

```
f1_error = 0
```

```
f2_error = 0
```

```
f3_error = 0
```

```
Dim replication As Integer
```

```
Dim replication_nr As Integer
```

```
Dim customer As Integer
```

```
Dim x As Integer
```

```
Dim y As Integer
```

```
Dim dc_x, dc_y As Integer
```

```
Dim lauf_x As Integer
```

```
Dim laufp_x As Integer
```

```
Dim lauf_y As Integer
```

```
Dim laufp_y As Integer
```

```
Dim distance As Double
```

```
Dim fulldistance As Double
```

```
Dim disav As Double
```

```
Dim i As Integer
```

```
Dim tri_coordx As Double
```

```
Dim tri_coordy As Double
```

```
Dim m_stroke_x As Double
```

```
Dim m_stroke_y As Double
```

```
Dim tri_coord As Double
```

```
Dim norm_error As Long
```

```
Dim outofbounce_error As Long
```

```
Dim already_used_coordinates As Object
```

```
Set already_used_coordinates = CreateObject("Scripting.Dictionary")
```

```
Dim coordinate_string As String
```

```
Dim found_valid As Boolean
```

```
Dim already_used_error As Long
```

```
x_min = wsRP.Range("E22").value
```

```
x_max = wsRP.Range("E23").value
```

```
y_min = wsRP.Range("E24").value
```

```
y_max = wsRP.Range("E25").value
```

```
x_prob = wsRP.Range("E28").value
```

```
y_prob = wsRP.Range("E31").value
```

```
replication = 0
```

```
replication_nr = wsRP.Range("B4").value
```

```
customer = wsRP.Range("B5").value
```

```
dc_x = wsRP.Range("B8").value
dc_y = wsRP.Range("B9").value
disav = 0
totalTime = GetTickCount

replication = replication_nr + replication

lTime = GetTickCount

fulldistance = 0

InitializeRNSeed (replication)

already_used_error = 0

i = 3

m_stroke_x = (x_prob - x_min) / (x_max - x_min)
m_stroke_y = (y_prob - y_min) / (y_max - y_min)

Do

    found_valid = False

Do

    Modul6.tri_x m_stroke_x, tri_coord

    tri_coordx = x_min + (x_max - x_min) * tri_coord

    x = Round(tri_coordx, 0)

    Modul6.tri_y m_stroke_y, tri_coord

    tri_coordy = y_min + (y_max - y_min) * tri_coord
```

```
y = Round(tri_coordy, 0)
```

```
coordinate_string = CStr(x) & "/" & CStr(y)
```

```
If Not already_used_coordinates.exists(coordinate_string) Then
```

```
    found_valid = True
```

```
    already_used_coordinates.Add coordinate_string, coordinate_string
```

```
    distance = 1.234 * Sqr((x - dc_x) ^ 2 + (y - dc_y) ^ 2)
```

```
    fulldistance = fulldistance + distance
```

```
    wsRP.Cells(i, 16).value = x
```

```
    wsRP.Cells(i, 17).value = y
```

```
    i = i + 1
```

```
Else
```

```
    already_used_error = already_used_error + 1
```

```
End If
```

```
Loop Until found_valid = True
```

```
Loop Until customer = already_used_coordinates.Count
```

```
f1_error = already_used_error + f1_error
```

```
f2_error = outofbounce_error + f2_error
```

```
f3_error = norm_error + f3_error
```

```
wsRP.Range("J3").value = replication  
wsRP.Range("K3").value = fulldistance  
lTime = GetTickCount - lTime  
wsRP.Range("L3").value = (lTime / 1000)  
wsRP.Range("M3").value = already_used_error  
wsRP.Range("N3").value = outofbounce_error  
wsRP.Range("O3").value = norm_error  
already_used_coordinates.RemoveAll
```

With Application

```
.EnableEvents = True  
.Calculation = xlCalculationAutomatic  
.ScreenUpdating = True
```

End With

End Sub

Modul 5

Option Explicit

```
Private Declare PtrSafe Function GetTickCount Lib "kernel32.dll" () As Long
```

```
Public Sub generate_triangular()
```

```
Dim lTime As Long
```

```
Dim evalTime As Long
```

```
Dim totalTime As Long
```

```
Dim avg_time As Long
```

```
With Application
```

```
    .EnableEvents = False
```

```
    .Calculation = xlCalculationManual
```

```
    .ScreenUpdating = False
```

```
End With
```

```
Dim wb As Workbook
```

```
Set wb = ThisWorkbook
```

```
Dim wsG As Worksheet
```

```
Set wsG = wb.Sheets("One_Generator")
```

```
Dim wsP As Worksheet
```

```
Set wsP = wb.Sheets("Parameter")
```

```
wsG.Range("A3:F100000").ClearContents
```

```
wsG.Range("H3:N5").ClearContents
```

```
wsG.Range("I7:I12").ClearContents
```

```
wsG.Range("I15:I18").ClearContents
```

```
wsG.Range("I22:M24").ClearContents
```

```
wsG.Range("O3:P100000").ClearContents
```

```
Dim x_min As Double
```

```
Dim x_max As Double
```

```
Dim y_min As Double
```

```
Dim y_max As Double
```

```
Dim x_prob As Double
```

```
Dim y_prob As Double
```

```
Dim f1_error, f2_error, f3_error As Long
```

```
f1_error = 0
```

```
f2_error = 0
```

```
f3_error = 0
```

```
Dim replication As Integer
```

```
Dim replication_nr As Integer
```

```
Dim customer As Integer
```

```
Dim x As Integer
```

```
Dim y As Integer
```

```
Dim dc_x, dc_y As Integer
```

```
Dim lauf_x As Integer
```

```
Dim laufp_x As Integer
```

```
Dim lauf_y As Integer
```

```
Dim laufp_y As Integer
```

```
Dim distance As Double
```

```
Dim fulldistance As Double
```

```
Dim disav As Double
```

```
Dim i As Integer
```

```
Dim tri_coordx As Double
```

```
Dim tri_coordy As Double
```

```
Dim m_stroke_x As Double
```

```
Dim m_stroke_y As Double
```

```
Dim tri_coord As Double
```

```
Dim norm_error As Long
```

```
Dim outofbounce_error As Long
```

```
Dim already_used_coordinates As Object
```

```
Set already_used_coordinates = CreateObject("Scripting.Dictionary")
```

```
Dim coordinate_string As String
```

```
Dim found_valid As Boolean
```

```
Dim already_used_error As Long
```

```
x_min = wsP.Range("K11").value
```

```
x_max = wsP.Range("K12").value
```

```
y_min = wsP.Range("k13").value
```

```
y_max = wsP.Range("k14").value
```

```
x_prob = wsP.Range("K17").value
```

```
y_prob = wsP.Range("K20").value
```

```
replication = 1

replication_nr = wsP.Range("B7").value

customer = wsP.Range("B8").value

dc_x = wsP.Range("B11").value

dc_y = wsP.Range("B12").value

disav = 0

totalTime = GetTickCount

Do

    lTime = GetTickCount

    fulldistance = 0

    InitializeRNSeed (replication)

    already_used_error = 0

    i = 1

    m_stroke_x = (x_prob - x_min) / (x_max - x_min)

    m_stroke_y = (y_prob - y_min) / (y_max - y_min)

    Do

        found_valid = False

    Do

        Modul6.tri_x m_stroke_x, tri_coord

        tri_coordx = x_min + (x_max - x_min) * tri_coord

        x = Round(tri_coordx, 0)
```

```
Modul6.tri_y m_stroke_y, tri_coord
```

```
tri_coordy = y_min + (y_max - y_min) * tri_coord
```

```
y = Round(tri_coordy, 0)
```

```
coordinate_string = CStr(x) & "/" & CStr(y)
```

```
If Not already_used_coordinates.exists(coordinate_string) Then
```

```
    found_valid = True
```

```
    already_used_coordinates.Add coordinate_string, coordinate_string
```

```
    distance = 1.234 * Sqr((x - dc_x) ^ 2 + (y - dc_y) ^ 2)
```

```
    fulldistance = fulldistance + distance
```

```
    ' wsG.Cells(i, 20).value = x
```

```
    ' wsG.Cells(i, 21).value = y
```

```
    ' i = i + 1
```

```
Else
```

```
    already_used_error = already_used_error + 1
```

```
End If
```

```
Loop Until found_valid = True
```

```
Loop Until customer = already_used_coordinates.Count
```

```
f1_error = already_used_error + f1_error

f2_error = outofbounce_error + f2_error

f3_error = norm_error + f3_error

replication = replication + 1

disav = disav + fulldistance

wsG.Cells(replication + 1, 1).value = replication - 1

wsG.Cells(replication + 1, 2).value = fulldistance

lTime = GetTickCount - lTime

avg_time = avg_time + lTime

wsG.Cells(replication + 1, 3).value = (lTime / 1000)

wsG.Cells(replication + 1, 4).value = already_used_error

already_used_coordinates.RemoveAll

lTime = 0

Loop Until replication = replication_nr + 1

' MsgBox ("Generated 100 coordinates, with " & CStr(already_used_error) & " new
generations")

' MsgBox (CStr(already_used_coordinates.Count))

' MsgBox (CStr(already_used_coordinates.Count))

evalTime = GetTickCount

Call create_boxplot(replication_nr, disav, avg_time, f1_error, f2_error, f3_error)
```

```
evalTime = GetTickCount - evalTime  
totalTime = GetTickCount - totalTime  
wsG.Range("I8").value = (evalTime / 1000)  
wsG.Range("I7").value = (totalTime / 1000)  
wsG.Range("G:G").ClearContents  
wsG.Range("T:U").ClearContents
```

With Application

```
.EnableEvents = True  
.Calculation = xlCalculationAutomatic  
.ScreenUpdating = True
```

End With

End Sub

Modul 6

Option Explicit

```
Public Sub tri_x(ByRef m_stroke_x As Double, ByRef result As Double)
```

```
Dim g1 As Double
```

```
g1 = MRG32k3a()
```

```
If g1 >= 0 And g1 <= m_stroke_x Then
```

```
result = Sqr(m_stroke_x * g1)
```

```
Else
```

```
result = 1 - Sqr((1 - m_stroke_x) * (1 - g1))
```

```
End If
```

End Sub

Public Sub tri_y(ByRef m_stroke_y As Double, ByRef result As Double)

Dim g2 As Double

g2 = MRG32k3a()

If g2 >= 0 And g2 <= m_stroke_y Then

result = Sqr(m_stroke_y * g2)

Else

result = 1 - Sqr((1 - m_stroke_y) * (1 - g2))

End If

End Sub

Eidesstattliche Versicherung

Name, Vorname

Matr.-Nr.

Ich versichere hiermit an Eides statt, dass ich die vorliegende ~~Bachelorarbeit~~/Masterarbeit* mit dem Titel:

„Vergleich von Verfahren für die Generierung von Kundenstandorten zur Simulation von urbanen Versorgungskonzepten“

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort, Datum

Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50.000,00 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden. (§ 63 Abs. 5 Hochschulgesetz - HG -)

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird ggfls. elektronische Vergleichswerkzeuge (wie z.B. die Software „turnitin“) zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.
Die oben stehende Belehrung habe ich zur Kenntnis genommen:

Ort, Datum

Unterschrift