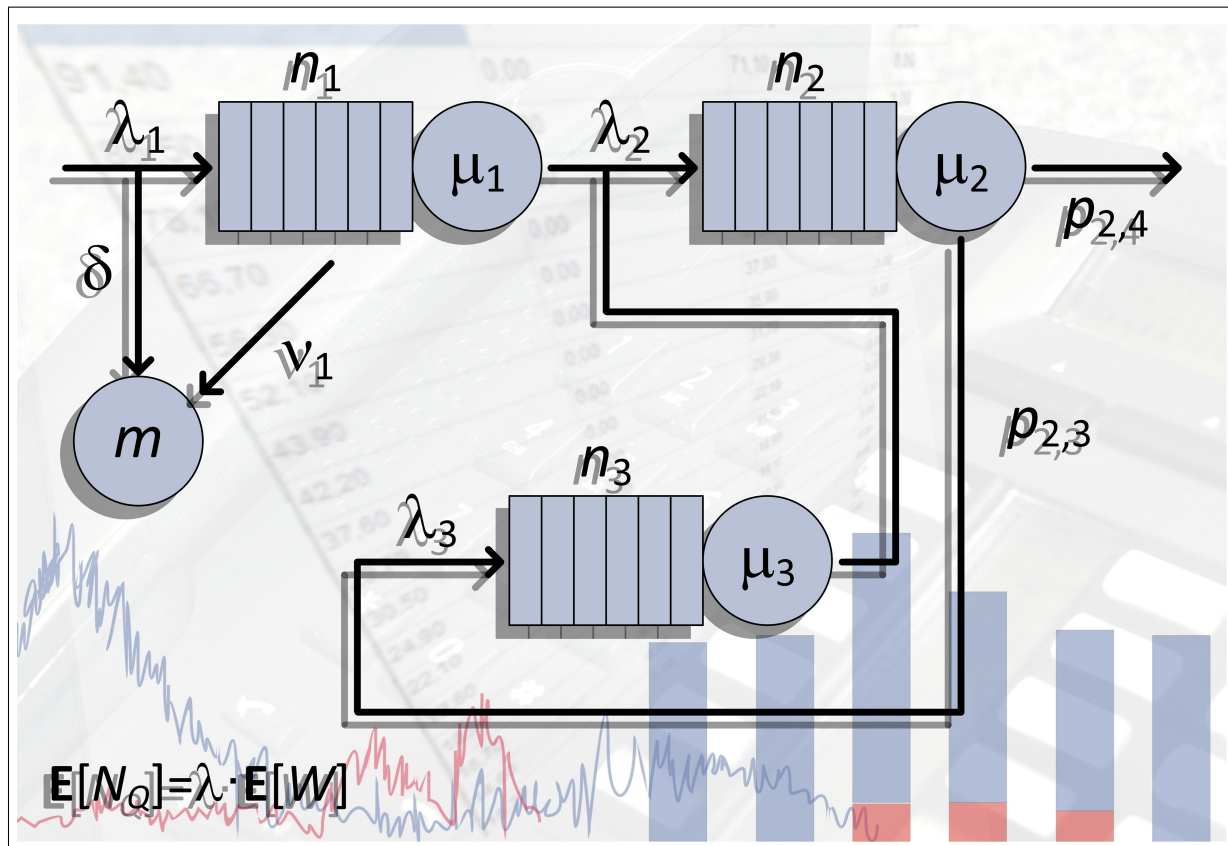


# Informationen für Dozenten

ALEXANDER HERZOG ([alexander.herzog@tu-clausthal.de](mailto:alexander.herzog@tu-clausthal.de))



Dieses Dokument bezieht sich auf die Version 5.6.0 des Warteschlangensimulators.  
Download-Adresse: <https://a-herzog.github.io/Warteschlangensimulator/>.

In diesem Dokument sind eine Reihe von Hinweisen für Dozenten, die den Warteschlangensimulator im Lehrbetrieb einsetzen möchten, zusammengefasst. Es wird beschrieben, welche Kompetenzen der Studierenden für die Verwendung des Simulators vorausgesetzt werden, welche Notation im Simulator verwendet wird und welche Fragestellungen sich als Einstiegsbeispiele besonders eignen.



# Kapitel 1

## Vorbereitungen

### 1.1 Voraussetzungen für die Verwendung des Warteschlangensimulators

Der Warteschlangensimulator ist auf die drei Anwendungsfelder Industrieinsatz, Forschung und Lehre ausgerichtet. Für den Einsatz in Unternehmen sind z. B. Datenbankanbindungen sowie die Einbindung in Reporting-Systeme wichtig. Zur Untersuchung von komplexen Forschungsfragen sind eine hohe Simulationsgeschwindigkeit bzw. die Möglichkeit, den Simulator ohne grafische Ausgabe auf einem entfernten (Linux-)Rechner ausführen zu können, von besonderer Bedeutung. Für den Einsatz in der Lehre verfügt der Simulator über eine möglichst intuitiv nutzbare Programmoberfläche sowie über zahlreiche Hilfe- und Erklärungstexte.

Mit Hilfe des Warteschlangensimulators können Fragestellungen aus der Warteschlangentheorie, für die ggf. keine exakten Formeln bekannt sind, untersucht werden. Es wird zwar versucht, die komplexen mathematischen Zusammenhänge und auch die informatischen Konzepte der ereignisorientierten, stochastischen Simulation so gut es geht im Hintergrund zu halten, dennoch ist aber ein gewisses Basiswissen für den erfolgreichen Einsatz des Simulators notwendig:

#### Stochastik

Wissen um die grundlegenden Zusammenhänge der Wahrscheinlichkeitsrechnung ist für die Erstellung von Modellen und für die Interpretation der Ergebnisse notwendig. Konkret sollten folgende Begriffe bekannt sein:

- Begriff der Wahrscheinlichkeit (z. B. für die Verzweigung von Kundenströmen in verschiedene Richtungen gemäß verschiedener Wahrscheinlichkeiten); damit verbunden der Begriff der Rate (als unnormierte Wahrscheinlichkeit)
- Wesentliche diskrete und kontinuierliche Wahrscheinlichkeitsverteilungen; in diesem Zusammenhang die Bedeutung von Dichte und Verteilungsfunktion
- Erwartungswerte, Standardabweichungen, Variationskoeffizient

Die Umrechnung zwischen Erwartungswert und Standardabweichung auf der einen Seite und den jeweiligen Verteilungsparametern auf der anderen Seite erfolgt im Simulator wenn immer möglich automatisch, d. h. die exakte Kenntnis der Formeln zur Berechnung der Verteilungsparameter aus den Kenngrößen (und umgekehrt) ist für die meisten üblicherweise eingesetzten Verteilungen nicht notwendig. Sehr wohl notwendig ist jedoch ein generelles Verständnis darüber, was z. B. Erwartungswert und Standardabweichung in Bezug auf eine Wahrscheinlichkeitsverteilung konkret bedeuten.

### Warteschlangentheorie

Aus der Warteschlangentheorie werden weniger konkrete Formeln benötigt, als viel mehr die wesentlichen Begriffe. Konkret sind folgende Begriffe für die Verwendung des Simulators wichtig:

- Genereller Aufbau eines Warteschlangenmodells (Ankunftsstrom, Warteraum, Bedienschalter; Objekte: Kunden, Bediener)
- Charakterisierung des Ankunftsstroms (Ankunftsrate und Zwischenankunftszeiten)
- Exponentialverteilung (inkl. ihrer Eigenschaft der Gedächtnislosigkeit und Begründung, warum dies für Ankunftsströme häufig passend ist)
- Charakterisierung des Bedienprozesses (Anzahl an Bedienern, Bedienrate und Bediendauer)
- Typische Wahrscheinlichkeitsverteilungen für die Bedienzeiten (z. B. Log-Normalverteilung)
- Übliche Kenngrößen (Arbeitslast, Auslastung, Wartezeit, Verweilzeit, Anzahl an Kunden in der Warteschlange oder im System)

Die MARKOV-Theorie und auch die KENDALL-Notation werden für die Verwendung des Simulators und auch für die Interpretation der Ergebnisse nicht benötigt. Natürlich kann es didaktisch interessant sein, z. B. die ERLANG-Formeln in der Hinführung zu den Simulationsmethoden zu erklären und dafür die MARKOV-Theorie einzuführen und die KENDALL-Notation zur Klassifikation einfacher Modelle vorzustellen.

### Über die Warteschlangentheorie hinaus

In der klassischen Warteschlangentheorie werden bevorzugt Modelle behandelt, die in der KENDALL-Notation beschrieben werden können. Für eine analytische Betrachtung stellt die vollständige Abdeckung der Möglichkeiten der KENDALL-Notation bereits eine sehr große Herausforderung dar<sup>1</sup>. Die Möglichkeiten von Simulationsmodellen gehen jedoch weit über diese Modelle hinaus.

Um interessante Fragestellung für die spätere Verwendung des Simulators zu identifizieren, ist es daher sinnvoll, vorab einige in realen Warteschlangensystemen relevante Eigenschaften zu erklären:

- Ungeduld von Kunden
- Betrachtung verschiedener Kundentypen mit z. B. verschiedenen Prioritäten oder einem Routing zu verschiedenen Teilmodellen (da z. B. nicht alle Callcenter-Agenten alle Kundentypen bedienen können)
- Rüstzeiten beim Wechsel von einem Kundentyp zu einem anderen an einer Maschine
- Schleifen entweder bedingt durch Nacharbeit oder aber durch Wahlwiederholer in Callcentern

### Ereignisorientierte Simulation

Der Warteschlangensimulator ist ein klassischer ereignisorientierter, stochastischer Simulator. Die Basis für die schnelle Abarbeitung der während der Simulation sukzessive entstehenden Ereignisse stellt eine effiziente Ereignisverwaltung dar. Für den Nutzer sind diese architektonischen Überlegungen nicht sichtbar und für die sinnvolle Nutzung des Simulators ist das Wissen um diese Konzepte auch nicht erforderlich.

Dennoch kann natürlich bei einer stärker informatischen Betrachtung auch ein Blick auf diese Zusammenhänge geworfen werden<sup>2</sup>:

- Welche Operationen treten wie häufig auf? (Einfügen von Ereignissen mit beliebigen Ausführungszeitpunkten und Entfernen des Ereignisses mit jeweils dem kleinsten Ausführungszeitpunkts jeweils

<sup>1</sup> siehe z. B. Shortle, John F., et al. *Fundamentals of queueing theory*. Vol. 399. John Wiley & Sons, 2018.

<sup>2</sup> siehe auch [github.com/A-Herzog/Warteschlangensimulator/wiki/Architecture](https://github.com/A-Herzog/Warteschlangensimulator/wiki/Architecture)

sehr häufig; Entfernung von (nicht abzuarbeitenden) Ereignissen mit beliebigen Ausführungszeitpunkten selten.)

- Wie lassen sich für die Operationen möglichst kleine  $O$ -Klassen erzielen? (Für die Abarbeitung der Ereignisse ist z. B. eine umgekehrte Reihenfolge im Speicher sinnvoll, so dass das jeweils nächste Ereignis am Ende der Liste steht und so das Entfernen dieses Ereignisses keine Umkopieroperationen nach sich zieht. Als Listentyp eignet sich ein Array, keine verkettete Liste, da nur in einem Array effizient per binärer Suche die Einfügeposition bestimmt werden kann.)
- Weitere Implementierungstricks zur Beschleunigung (z. B. interne Rechnung auf Millisekundenbasis mit Ganzzahlen, Verwendung mehrerer Teilereignislisten mit assoziativer Zuordnung der Ereignisse, Ringpuffer für Ereignisse, die als Folgeereignisse mit demselben Ausführungszeitpunkt wie der des aktuellen Ereignisses entstehen usw.)

Im Kontext der Ereignisverarbeitung kann auch diskutiert werden, welche Konzepte sich besonders gut mit ereignisorientierter Simulation abbilden lassen und welche nicht. Soll ein Kunde an einer Station z. B. für 60 Sekunden verzögert werden, so wäre die Definition einer (ständig) zu prüfenden Bedingung  $t \geq \text{Ankunft} + 60$  maximal kontraproduktiv. Es müssten hier extrem viele Ereignisse, an denen jeweils der Wahrheitswert der Bedingung geprüft wird, abgearbeitet werden. Legt man jedoch einfach ein Ereignis „Kunde freigeben“ mit dem Ausführungszeitpunkt  $\text{Ankunft} + 60$  an, so entfällt der gesamte Aufwand und statt z. B. 60.000 Ereignissen (bei Prüfung der Bedingung im Millisekundentakt) ist nur ein Ereignis notwendig.

## 1.2 Technische Voraussetzungen für den Betrieb des Warteschlangensimulators

Der Warteschlangensimulator ist ein Opensource-Programm, d.h. er kann ohne lizenzrechtliche Einschränkungen auf beliebigen Rechnern installiert werden. Auch ein späterer Übergang vom Lehrbetrieb zu konkreten Forschungsfragestellungen oder aber zur Modellierung von realen Fragestellungen im industriellen Kontext ist problemlos möglich.

Als einzige Voraussetzung für den Betrieb wird eine Java-Laufzeitumgebung benötigt (welche aber ebenfalls als Opensource für verschiedenste Betriebssysteme und CPU-Architekturen zur Verfügung steht). Je nach technischem Kenntnisstand der Nutzer kann es sinnvoll sein, Download-Links im Vorfeld einer Präsenzveranstaltung bereitzustellen bzw. sicherzustellen, dass tatsächlich eine Java-Laufzeitumgebung auf den entsprechenden Rechnern installiert ist. Für einen technisch versierten Nutzer sollte die Installation von Java und vom Warteschlangensimulator in jeweils weniger als einer Minute möglich sein.

Zu beachten ist, dass der Warteschlangensimulator sowohl mit als auch ohne Admin-Rechte installiert werden kann. (Auch eine portable Nutzung ist möglich. In diesem Fall muss nur eine `zip`-Datei entpackt werden.) Für die Installation einer Java-Laufzeitumgebung werden jedoch Admin-Rechte benötigt.

Für Windows-Rechner empfohlen wird die Verwendung der Java-Umgebung von Eclipse Adoptium<sup>3</sup>. Es existieren eine Reihe von Laufzeitumgebungen, die jedoch alle intern auf dem OpenJDK-Projekt aufbauen und untereinander kompatibel sind. Der Vorteil von Eclipse Adoptium besteht im Wesentlichen in der sehr einfachen Installation. Der Warteschlangensimulator kennt die Installationspfade aller üblichen Java-Umgebungen und muss daher nicht gesondert konfiguriert werden. Es wird lediglich eine beliebige Java-Laufzeitumgebung in der Version 8 oder höher vorausgesetzt. Insbesondere auf Rechnern mit hochauflösenden Bildschirmen (und daraus resultierend mit einer Skalierung der Darstellung mit einem Faktor größer als 100%) empfiehlt sich die Installation von Java 11 oder höher, da Java 8 noch nicht mit „High-DPI-Displays“ umgehen kann und das Programmfenster in diesem Fall unskaliert (und damit extrem klein) dargestellt wird.

---

<sup>3</sup> [adoptium.net](https://adoptium.net)

Auf Linux-Systemen lässt sich eine Java-Umgebung über den jeweiligen Paketmanager installieren. Der Name des Paketes beginnt üblicherweise mit „`openjdk`“, z. B. „`openjdk-17-jdk`“.

## Kapitel 2

### Mathematische Notation

Während die Namen der Bezeichner in der Stochastik noch weitgehend einheitlich sind, werden in der Warteschlangentheorie häufig sehr verschiedene Bezeichnungen verwendet. Teilweise steht dabei ein einzelner Buchstabe entweder für einen Begriff wie „Wartezeit“ oder aber gleich für eine Kenngröße wie „mittlere Wartezeit“. Im Warteschlangensimulator erfolgt hier eine klare Trennung: Buchstaben bezeichnen stets Begriffe; Kenngrößen werden stets durch funktionsähnliche Bezeichnungen angedeutet:

- $P(X = x)$ : Wahrscheinlichkeit, dass Bezeichner  $X$  den Wert  $x$  annimmt.
- $\mathbf{E}[X]$ : Erwartungswert von  $X$ .
- $\mathbf{Std}[X]$ : Standardabweichung von  $X$ .
- $\mathbf{Var}[X]$ : Varianz von  $X$ .
- $\mathbf{CV}[X]$ : Variationskoeffizient von  $X$ .
- $\mathbf{Sk}[X]$ : Schiefe von  $X$ .
- $\mathbf{Kurt}[X]$ : Exzess von  $X$ . Der Exzess ist ein von der Wölbung abgeleitetes Maß.

#### Mittelwerte und Erwartungswerte

Der Bezeichner  $\mathbf{E}[X]$  für den Erwartungswert wird dabei sowohl für die theoretischen Verteilungen auf der Eingabeseite verwendet, als auch für die Mittelwerte auf der Ausgabeseite. In der Stochastik wird häufig sehr genau darauf hingewiesen, dass sich ein Erwartungswert immer auf eine theoretische Verteilung bezieht und ein Mittelwert sich auf eine Messreihe bezieht. (Und durch das Gesetz der großen Zahlen wird dann gezeigt, dass der Mittelwert gegen den Erwartungswert konvergiert – und folglich beides genau zu unterscheidende Begriffe sind.) Die Tatsache, dass der Warteschlangensimulator hier eine etwas ungenaue Bezeichnungsweise auf der Ausgabeseite benutzt, hat einen ganz praktischen Grund:  $\bar{x}$  lässt sich in einfachen Ausgabetexten schlecht darstellen;  $\mathbf{E}[X]$  ist hingegen ohne weitere Formatierungen darstellbar.

#### Bedingte Erwartungswerte usw.

Bei allen obigen Kenngrößen kann noch eine Einschränkung bzw. nähere Information, worauf sich diese beziehen, hinzugefügt werden. Dazu wird die Notation der bedingten Erwartung verwendet:

$\mathbf{E}[X|\text{Kundentyp A}]$

bezeichnet den Erwartungswert von  $X$  bezogen auf Kunden des Typs A.

#### Warteschlangentheorie

Für die üblichen Begriffe in der Warteschlangentheorie werden folgende Abkürzungen verwendet:



- $N$  = Anzahl an Kunden an einer Station oder im System (*number*).
- $NQ$  = Anzahl an Kunden in der Warteschlange an einer Station oder im System (*number in queue*).
- $NS$  = Anzahl an Kunden im Bedienprozess an einer Station oder im System (*number in service process*).
- $W$  = Wartezeiten der Kunden insgesamt oder an der Station.
- $T$  = Transportzeiten der Kunden insgesamt oder an der Station.
- $S$  = Bedienzeiten der Kunden insgesamt oder an der Station.
- $V$  = Verweilzeiten der Kunden insgesamt oder an der Station.

Alle Kenngrößen werden jeweils global, bezogen auf das gesamte System, pro Kundentyp und auch pro Station aufgezeichnet. Bewegen sich in einem Modell alle Kunden ausschließlich durch eine einzelne Station, so sind die Kenngrößen für System, Kundentyp und Station jeweils identisch. Bewegen sich jedoch Kunden verschiedener Typen durch verschiedene Teilmengen der Bedienstationen, so werden die Kenngrößen zwischen den Erfassungsarten variieren.

Um Darstellungen, die sich aus der Ergebnisansicht nicht als einfacher Text kopieren lassen, zu vermeiden, werden die Bezeichner  $NQ$  und  $NS$  für die Anzahl an Kunden in der Warteschlange bzw. im Bedienprozess verwendet. In einem mathematischen Kontext werden für diese Begriffe häufig die Bezeichner  $N_Q$  und  $N_S$  verwendet.

### Beispiele

- $\mathbf{E}[W|\text{Typ A}]$  ist die mittlere Wartezeit der Kunden vom Typ A.
- $\mathbf{CV}[V]$  ist der Variationskoeffizient der Verweilzeiten betrachtet über alle Kundentypen und alle Stationen.
- $\mathbf{E}[NQ|\text{Station X}]$  ist die mittlere Warteschlangenlänge an Station X.

In der Notation des Warteschlangensimulators würde die häufig als  $L = \lambda \cdot W$  beschriebene Formel von LITTLE wie folgt aussehen:

$$\mathbf{E}[NQ] = \lambda \cdot \mathbf{E}[W].$$

## Kapitel 3

# Nutzung des Simulators

### 3.1 Erstellen eines einfachen Beispiels

Mit dem „Erlang-C-Vergleichsmodell“ (aufzurufen über den Menüpunkt „Beispiel laden“ im „Datei“-Menü) steht ein bereits komplett vorbereitetes, einfaches Modell, in dem ein sinnvoll parametrisiertes M/M/c-System abgebildet wird, zur Verfügung. Aus didaktischen Gründen ist es jedoch empfehlenswert, ein einfaches Modell zunächst manuell aufzubauen.

Eine vollständige Anleitung dazu kann in dem über das „Hilfe“-Menü aufrufbaren Tutorial nachgelesen werden.

In Kurzform lässt sich die Modellerstellung wie folgt zusammenfassen:

- Über die Schaltfläche „Element“ in der vertikalen Symbolleiste am linken Fensterrand kann die Vorlagenleiste ein- und ausgeblendet werden.
- Aus der Vorlagenleiste müssen für ein erstes Modell die drei Stationen „Quelle“, „Bedienstation“ und „Ausgang“ auf die Zeichenfläche gezogen werden.
- Über die Schaltfläche „Kante“ in der vertikalen Symbolleiste am linken Fensterrand wird die Funktion zum Hinzufügen von Verbindungskanten ein- und ausgeschaltet. Ist die Funktion aktiv, so müssen nacheinander das Start- und das Zielelement für eine neue Verbindungskante angeklickt werden. (Im vorliegenden Fall also zunächst die Quelle und dann die Bedienstation und danach die Bedienstation und dann der Ausgang).
- Um einem Erstanwender möglichst wenig Schwierigkeiten zu bereiten, sind an der Quelle und an der Bedienstation bereits Zwischenankunfts- bzw. Bedienzeiten voreingestellt, mit denen sich schnell ein lauffähiges Modell ohne Überlast erstellen lässt: An der Quelle gilt vorgabemäßig  $\mathbf{E}[I] = 60$  Sekunden und an der Bedienstation  $\mathbf{E}[S] = 50$  Sekunden. In beiden Fällen ist die Exponentialverteilung ausgewählt, so dass sich also ein M/M-Modell ergibt. Es wird nur  $c = 1$  Bediener benötigt; als Arbeitslast ergibt sich dann  $a := \frac{1/60}{1/50} \approx 0,83$ .
- Im Gegensatz zur klassischen Gedankenwelt der analytischen Warteschlangentheorie sind die Bediener in einem Simulationsmodell nicht integraler Bestandteil einer Bedienstation, sondern können zwischen verschiedenen Stationen geteilt werden. Daher existieren sie als eigenständige Objekt ohne feste Zuordnung zu einer Bedienstation. Bedienergruppen können über den Modelleigenschaftendialog angelegt und dann an einer Bedienstation als notwendige Ressource definiert werden. Allerdings existieren Funktionen, um diesen Prozess abzukürzen: Auf der Dialogseite „Bediener“ des Konfigurationsdialogs der Bedienstation (aufrufbar per Doppelklick auf das Bedienstation-Symbol auf der Zeichenfläche) kann eine Bedienergruppe angelegt und sofort der aktuellen Station zugeordnet werden. Hierbei sind zwei Zahlenwerte zu unterscheiden: Aus wie vielen Bedienern soll die Gruppe bestehen und wie viele Be-

diener sollen notwendig sein, um einen Kunden bedienen zu können. Während der erste Wert dem  $c$  aus der Warteschlangentheorie entspricht, ermöglicht der zweite Wert später komplexe Modelle und sollte für unser Modell auf 1 gesetzt werden (was auch der Sichtweise in der analytischen Warteschlangentheorie entspricht). Da eine Arbeitslast von  $a = \frac{5}{6} < 1$  vorliegt, genügt für unser Modell eine Gruppengröße von  $c = 1$ .

### Autokorrektur

Vor dem Start einer Animation oder einer Simulation wird das Modell geprüft und ein Start verweigert, wenn Fehler, die eine erfolgreiche Abarbeitung des Modells verhindern würden, vorliegen. Bei einfachen Fehlern bzw. Fehlern, bei denen eine offensichtliche und in vielen Fällen sinnvolle Lösung existiert, schlägt der Simulator direkt eine Korrektur vor. In Bezug auf das obige Modell würde der Simulator automatisch folgende Fehler korrigieren können:

- Fehlende Verbindungskante von der Quelle zur Bedienstation.
- Fehlende Verbindungskante von der Bedienstation zum Ausgang.
- Fehlende Bedienergruppe an der Bedienstation.

Theoretisch würde es daher ausreichen, die drei Stationen, aus denen das Modell bestehen soll, aus der Vorlagenleiste auf die Zeichenfläche zu ziehen. Alle weiteren Verknüpfungen und Einstellungen kann die Autokorrektur notwendigenfalls selbst durchführen. Da die Autokorrektur natürlich nur in einem sehr begrenzten Rahmen erkennen kann, wie sich ein konkreter Fehler sinnvoll beheben lässt (z.B. ist bei größeren Modellen nicht mehr offensichtlich, welche Station sinnvoll mit welcher verknüpft werden soll), wäre es didaktisch von Nachteil, sich auf diese Korrekturfunktion zu sehr zu verlassen.

## 3.2 Simulation und Animation

Das ursprüngliche primäre Anwendungsszenario für den Warteschlangensimulator bestand in der möglichst schnellen Simulation komplexer Warteschlangenmodelle gefolgt von der Ausgabe der Kenngrößen des Modells. Da der quantitative Erkenntnisgewinn aus Animationen begrenzt ist, stand die Animation von Modellen nicht im Vordergrund. Während in vielen Simulationsprogrammen immer eine Animation durchgeführt wird und eine Simulation lediglich eine Animation mit deaktivierter Grafikausgabe ist, sind die beiden Modi im Simulator strikter getrennt: In der horizontalen Symbolleiste am oberen Fensterrand befinden sich die beiden Schaltflächen „Animation“ und „Simulation“. Für beide kommt zwar derselbe Simulationskern zum Einsatz, allerdings kann dieser sein volles Leistungspotential (insbesondere die Mehrkernunterstützung) nur im Simulationsmodus voll ausschöpfen.

Während daher für spätere Untersuchungen der Simulationsmodus relevant werden wird, ist insbesondere für Einsteiger für die Vermittlung von unmittelbaren Erfolgserlebnissen der Animationsmodus wichtig.

Da auch bei einer Animation im Hintergrund der vollständige Simulationskern arbeitet, stehen nach dem Abschluss einer per Animation ausgeführten Simulation ebenfalls die vollständigen Statistikdaten zur Verfügung. Die Simulation im Animationsmodus wird lediglich im Vergleich zu einer eigentlichen Simulation deutlich länger dauern.

Die Dauer einer Simulation (egal ob im Animations- oder im Simulationsmodus) hängt unmittelbar von der Anzahl an simulierten Kundenankünften ab. (Außerdem hängt die statistische Validität natürlich auch von der Anzahl an simulierten Ankünften ab.) Die Anzahl an zu simulierenden Ankünften kann auf der Dialogseite „Simulation“ des Modelleigenschaftendialogs (aufrufbar über die Schaltfläche „Modell“ in der vertikalen Symbolleiste am linken Fensterrand) eingestellt werden. Für neue Modelle ist standardmäßig eine Anzahl von 10.000.000 Ankünften eingestellt. Bezogen auf das oben beschriebene einfache M/M/1-Modell benötigt der Warteschlangensimulator (ohne Animationsanzeige) rechnerabhängig ca. 0,5 bis 5

Sekunden für die Simulation dieser Anzahl an Ankünften. Für eine Animation mit maximaler Geschwindigkeit sind für diese Anzahl an Ankünften Werte zwischen 30 Sekunden und 3 Minute realistisch. Soll der Animationsmodus intensiv genutzt werden und sollen dennoch am Ende die Kenngrößen betrachtet werden, so kann es daher sinnvoll sein, die Anzahl an zu simulierenden Ankünften zu reduzieren.

### 3.3 Statistische Stabilität der Simulationsergebnisse

In den Simulationsprozess fließen (im obigen Beispiel für die Zwischenankunfts- und die Bedienzeiten) Zufallszahlen ein. Daher liefert jede Ausführung der Simulation (leicht) unterschiedliche Ergebnisse. Um hierfür ein Gefühl zu erhalten, lohnt es sich, das oben aufgestellte Modell mehrfach unverändert zu simulieren und jeweils die relevanten Kenngrößen zu notieren.

Als Ergebnis lässt sich feststellen: Für  $n \rightarrow \infty$  gilt zwar  $\bar{x}_n \rightarrow \mathbf{E}[X]$ , aber selbst 10.000.000 simulierte Ankünfte sind noch weit genug von „ $\infty$ “ entfernt, so dass z. B. die mittlere Wartezeit der Kunden zwischen verschiedenen Wiederholungen der Simulation durchaus noch um einige Prozentpunkte schwanken werden.

Der Warteschlangensimulator kann Konfidenzintervalle auf Basis der Batch-Means-Methode (die notwendig ist, da die Wartezeiten der aufeinander folgenden Kunden nicht unabhängig voneinander sind) ausgeben. Dafür muss jedoch zunächst eine entsprechende Batch-Größe, bei der die Rest-Autokorrelation unterhalb eines akzeptablen Schwellenwertes bleibt, ermittelt werden. Der Simulator verfügt über Programmfunktionen, die bei der Ermittlung dieser Werte unterstützen. Dennoch ist jedoch einiges Hintergrundwissen zu den mathematischen Konzepten notwendig, um diese Funktionen sinnvoll nutzen zu können. Daher ist es empfehlenswert, diese Überlegungen zumindest für ein erstes Modell zurückzustellen. (Siehe ansonsten die Programmfunktion „Statistik-Batch-Größe bestimmen“ im „Simulation“-Menü sowie die Dialogseite „Ausgabeanalyse“ im Modelleigenschaftendialog.)

Soll das Thema der Schwankungen der Ergebnisse zwischen verschiedenen Simulationen desselben Modells später noch weiter vertieft werden, so steht dafür im „Simulation“-Menü mit der Varianzanalyse eine entsprechende Funktion zur Verfügung.

### 3.4 Vergleich von Simulationsergebnissen und Erlang-C-Kenngrößen

Am Ende einer Simulation werden u. a. die Kenngrößen  $\mathbf{E}[N]$ ,  $\mathbf{E}[NQ]$ ,  $\mathbf{E}[W]$  und  $\mathbf{E}[V]$  ausgegeben. Diese lassen sich auch gut mit der Erlang-C-Formeln berechnen. Auf diese Weise können die Simulationsergebnisse analytisch nachvollzogen werden.

Die Berechnung der zugehörigen Erlang-C-Kenngrößen kann manuell erfolgen (sofern die entsprechenden Formeln zuvor eingeführt wurden) oder aber auch mit Hilfe der Warteschlangenrechner-Funktion (siehe „Extras“-Menü) im Simulator. Während eine möglichst manuelle Berechnung für den Anfang, um die Zusammenhänge zu verstehen, sinnvoll sein kann, ist für spätere Betrachtungen die Statistikseite „Erlang-C-Vergleich“ hilfreich: Auf dieser Seite innerhalb der Statistikergebnisdarstellung werden die Simulationsergebnisse direkt den analytischen Ergebnissen gegenüber gestellt. Diese Statistikseite steht nur zur Verfügung, wenn das Modell einfach genug gestaltet ist, so dass der Simulator die vorgenommenen Einstellungen in eindeutiger Weise auf ein analytisches Modell abbilden kann. Ist die Ergebnisseite vorhanden, so wird sie direkt unterhalb der „Ergebnisübersicht (Text)“-Seite angezeigt.

Bei dem Vergleich von Simulations- und analytischen Ergebnissen wird auch jeweils die relative, prozentuale Abweichung ausgegeben. Wurde ein M/M-Modell simuliert, so resultieren diese Abweichungen einzig darin, dass die Simulation nicht lang genug gelaufen ist, um wirklich exakte Ergebnisse zu liefern. Allerdings steht der Erlang-C-Vergleich auch dann im Simulator zur Verfügung, wenn z. B. abweichende Verteilungen für die Zwischenankunfts- oder die Bedienzeiten verwendet werden. Hier kann als Übungs-

aufgabe untersucht werden, wie die Simulationsergebnisse und die Erlang-C-Ergebnisse zunehmend divergieren, wenn die in der Simulation betrachteten Wahrscheinlichkeitsverteilungen sich zunehmend von einem Variationskoeffizienten von 1 entfernen.

## Kapitel 4

# Untersuchbare Fragestellungen

Menschen neigen dazu, in linearen Zusammenhängen zu denken, was in vielen Bereichen Probleme nach sich zieht. Die Zusammenhänge zwischen Eingabe- und Kenngrößen in der Warteschlangentheorie sind in vielen Fällen nicht linear (z. B. wächst  $\mathbf{E}[W]$  mit steigendem  $\rho$  überexponential an). Daher ist es allgemein sehr interessant, die Auswirkungen von Änderungen bestimmter Eingabegrößen auf die Kenngrößen eines Modells systematisch zu untersuchen.

Theoretisch kann dies erfolgen, in dem eine bestimmte Eingabegröße wiederholt verändert wird, eine Simulation durchgeführt wird und die Ergebnisse notiert werden. Am Ende lässt sich dann ein Diagramm, in dem der Einfluss der Eingabe- auf die Kenngrößen dargestellt wird, aufstellen. Da diese repetitive Arbeit auf Dauer keinen Vorteil darstellt, sondern eher die Untersuchung weiterer Zusammenhänge behindert, besitzt der Warteschlangensimulator Funktionen zur Automatisierung entsprechender Untersuchungen.

### 4.1 Parameterreihenfunktion

Die Parameterreihenfunktion kann über die Schaltfläche „Parameterreihe“ in der horizontalen Symbolleiste aufgerufen werden. Auf der Zeichenfläche muss dafür ein gültiges Modell vorhanden sein, welches die Basis für die Parameterreihe darstellen soll. In der Parameterreihenfunktion können dann manuell Eingabe- und Ausgabeparameter definiert werden und eingestellt werden, in welchem Bereich der oder die Eingabeparameter variiert werden sollen.

Für viele (einfache) Fragestellungen lässt sich dieser Prozess abkürzen: Im Kontextmenü z. B. für die Quelle und die Bedienstation existieren bereits Einträge, die direkt eine passend konfigurierte Parameterreihe anlegen. Es muss dann nur noch ein sinnvoller Bereich, in dem z. B. die Zwischenankunfts- oder die Bedienzeiten variiert werden sollen, angegeben werden. Auf diese Weise automatisch angelegte Parameterreihenkonfigurationen unterscheiden sich nicht von manuell angelegten Konfigurationen, d. h. es können nach wie vor nach Bedarf weitere Ausgabegrößen hinzugefügt oder entfernt werden usw.

### 4.2 Direkter Vergleich

Soll jedoch nur untersucht werden, welche Auswirkung die Veränderung eines Parameters von einem Wert zu einem anderen hat, so wird dafür die Parameterreihenfunktion nicht benötigt. Der Simulator behält das Ergebnis der jeweils letzten Simulation bei der Durchführung einer neuen Simulation im Hintergrund im Speicher. Bei jeder Statistikseite kann über die Schaltfläche „Vorherige“ rechts neben der Ansicht der aktuellen Ergebnisse die Auflistung der Ergebnisse des jeweils unmittelbar vorherigen Simulationslaufes eingeblendet werden. So können, ohne Ergebnisse extern zu notieren, schnell Vergleiche vorgenommen werden.


### 4.3 Vorschläge für mögliche Untersuchungen

- Welche Auswirkungen hat die Veränderung der Auslastung auf die mittlere Wartezeit und die mittlere Warteschlangenlänge? (Die Auslastung kann dabei – bezogen auf das oben betrachtete einfache M/M/c-Modell – sowohl über die Zwischenankunftszeiten, als auch über die Bedienzeiten als auch über die Anzahl an Bedienern variiert werden.)
- Welche Auswirkungen hat es, wenn  $CV[I]$  oder  $CV[S]$  bei ansonsten gleichbleibender Auslastung variiert werden? (Hierfür müssen für die Zwischenankunfts- bzw. die Bedienzeiten von der Exponentialverteilung abweichende Wahrscheinlichkeitsverteilungen eingestellt werden.)
- Ohne Verwendung der Parameterreihenfunktion, durch direkte Modellvariation: Welche Auswirkungen hat der Wechsel von FIFO zu LIFO in der Bedienreihenfolge? ( $E[W]$  ändert sich nicht, aber  $CV[W]$  steigt signifikant an. Hinweis: Die Bedienreihenfolge kann auf der Dialogseite „Prioritäten und Batch-Größen“ im Einstellungendialog der Bedienstation konfiguriert werden. „w“ steht hier für „Priorität wächst linear mit wachsender Wartezeit“, d. h. FIFO, und „-w“ für „Priorität nimmt linear mit wachsender Wartezeit ab“, d. h. LIFO. Per „random()“ als Priorität lässt sich eine Bedingung in zufälliger Reihenfolge einstellen.)
- Das FIFO-LIFO-Experiment kann noch erweitert werden: Besitzt das Modell zwei Kundenquellen mit unterschiedlichen Namen (woraus sich dann auch die Kundentypnamen ableiten), so kann eine Kampagnenfertigung untersucht werden: Über den gleichnamigen Schalter auf der Dialogseite „Prioritäten und Batch-Größen“ lässt sich einstellen, dass die Prioritätsformeln zunächst nur innerhalb des gerade aktiven Kundentyps gelten. Nur wenn kein Kunde von dem Typ, der als letztes bedient wurde, wartet, wird zu den Kunden des anderen Typs gewechselt. Eine Kampagnenfertigung erhöht ebenfalls den Variationskoeffizienten der Wartezeiten, ist also zunächst von Nachteil. Allerdings kehrt sich dieser Nachteil um, wenn Rüstzeiten (ebenfalls im Einstellungendialog der Bedienstation definierbar) beim Wechsel von einem zu einem anderen Kundentyp auftreten.
- Ein weiteres klassisches Untersuchungsgebiet stellt die Frage dar, ob z. B. zwei getrennte Bediensysteme mit getrennten Warteschlangen oder eine Bedienstation mit zwei Bedienern effizienter ist. Im Simulator existieren hierzu (aufrufbar über „Beispiel laden“ im „Datei“-Menü) mehrere Beispielmodelle.

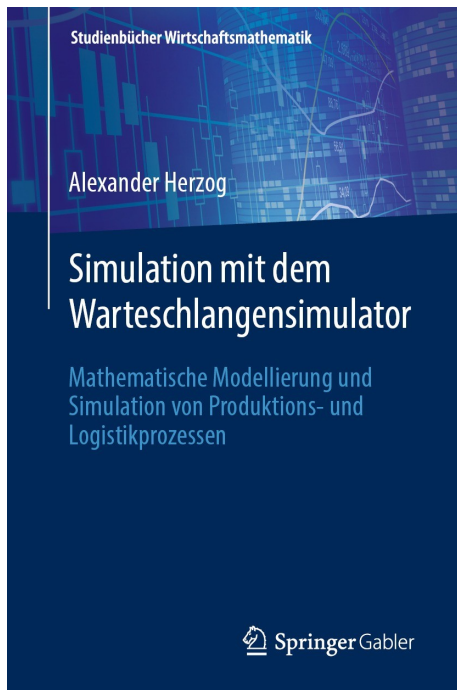
## Kapitel 5

### Weitere Ressourcen

Die Sammlung der direkt in den Warteschlangensimulator eingebauten Beispielm Modelle (aufrufbar über „Beispiel laden“ im „Datei“-Menü) stellt eine gute Quelle für Inspirationen für weitere mit dem Simulator untersuchbare Fragestellungen dar.

Im „Hilfe“-Menü sind eine Reihe von weiteren Dokumenten verlinkt: Ein Tutorial, eine Übersicht über alle verfügbaren Stationen usw. Ist der Bearbeitendialog für eine Station geöffnet, so kann über  direkt eine Online-Hilfe aufgerufen werden.

Des Weiteren sei auf das Lehrbuch zum Warteschlangensimulator verwiesen. Hier finden sich neben Einführungen in die Warteschlangentheorie und in die ereignisorientierte Simulation viele Beispiele zur Modellierung und Analyse von konkreten warteschlangentheoretischen Fragestellungen:



Alexander Herzog:  
*Simulation mit dem Warteschlangensimulator*,  
Springer, 2021.

ISBN: 978-3-658-34667-6

[link.springer.com/book/10.1007/978-3-658-34668-3](https://link.springer.com/book/10.1007/978-3-658-34668-3)