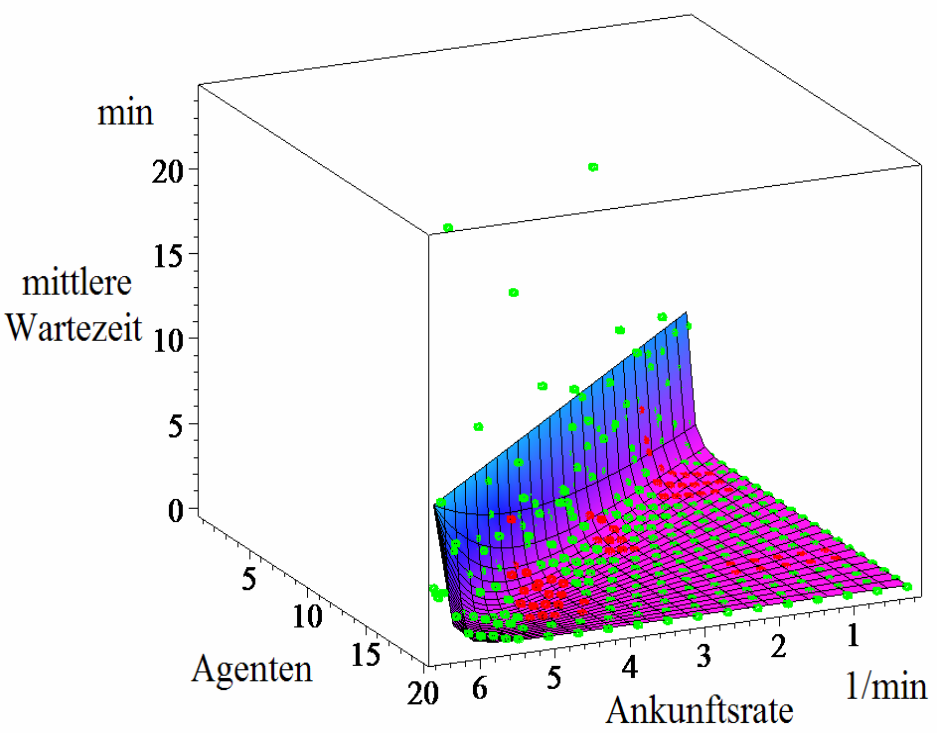
IWI Diskussionsbeiträge

 # 13 (10. Januar, 2005)[1]

ISSN 1612-3646

Optimierung von Warteschlangen-systemen in Call Centern auf Basis von Kennzahlenapproximation[2]

Frank Köller[3] und **Michael H. Breitner**[4]



[1] Kopien oder eine PDF-Datei sind auf Anfrage erhältlich: Institut für Wirtschaftsinformatik, Universität Hannover, Königsworther Platz 1, 30167 Hannover, http://www.iwi.uni-hannover.de.

[2] Dieser Aufsatz ist am 30.09.2004 für die GOR-Tagung „Entscheidungsunterstützende Systeme in Supply Chain Management und Logistik", 22. – 23.04.2005, in Paderborn eingereicht worden, vgl. http://www.dsor.de/gor-tagung.de, und am 10.01.2005 angenommen worden und erscheint imTagungsband Günther, H. O., Mattfeld, D. C., Suhl, L., (2005), „Entscheidungsunterstützende Systeme in Supply Chain Management und Logistik", Heidelberg, Physica.

[3] Diplom-Mathematiker (koeller@iwi.uni-hannover.de).

[4] Professor für Wirtschaftsinformatik und Betriebswirtschaftslehre (breitner@iwi.uni-hannover.de).

Inhaltsverzeichnis

Optimierung von Warteschlangensystemen in Call Centern auf Basis von Kennzahlen-approximation

Frank Köller, Michael H. Breitner

Institut für Wirtschaftsinformatik, Universität Hannover, Königsworther Platz 1, 30167 Hannover, {koeller;breitner}@iwi.uni-hannover.de

Abstract

In diesem Aufsatz wird der Fragestellung nachgegangen, ob neuronale Netze in der Lage sind Kennzahlen für Warteschlangensysteme zu approximieren. Da für die meisten in der Praxis vorkommenden Warteschlangenprobleme keine exakten, expliziten Lösungen für die Warteschlangenkennzahlen existieren, werden diese entweder mit aufwendigen, diskreten Simulationen gelöst, oder aber das Grundproblem wird soweit vereinfacht, dass es analytisch lösbar wird. Im Gegensatz dazu muss für das Training neuronaler Netze nicht die Struktur des Problems verändert werden. Weiterhin brauchen auch nur wenige Simulationspunkte gegenüber einer „flächendeckenden“ Auswertung mit einer Simulation generiert werden, da das unvermeidliche Rauschen in den Simulationsdaten durch die kontinuierliche, approximierte Lösung geglättet wird, d. h. die Kennzahlen genauer verfügbar sind. Aufgrund deutlich weniger Simulationen besteht ein erheblicher Zeitvorteil, denn der zusätzliche Schritt des Trainings der neuronalen Netze dauert i. d. R. nur wenige Sekunden.

Anhand von Simulationen für Inbound-Call-Center wird gezeigt, dass künstliche neuronale Netze Kennzahlen von Warteschlangenproblemen, bei denen analytische Lösungen existieren, sehr gut approximieren können. Dieser Aufsatz bildet also die Grundlage dafür, dass in einem weiteren Schritt künstliche neuronale Netze auch auf allgemeine Warteschlangenprobleme angewendet werden können, für die keine exakten, expliziten Lösungen für die Warteschlangenkennzahlen existieren[1].

[1] Meist können obere und untere Schranken bestimmt werden, die die Bandbreiten für Warteschlangenkennzahlen begrenzen. Somit ist überprüfbar, ob die approximierten Kennzahlen innerhalb dieser Bandbreiten liegen.

1 Einleitung

Kundenservice Center bilden die Schnittstelle zwischen Unternehmen und Kunden und haben somit eine Schlüsselposition inne: Von hier aus werden Geschäftsbeziehungen aufgebaut, gesteuert und ausgebaut, sowohl im B2B- als auch im B2C-Bereich. Insbesondere in den Unternehmen, wo heute bereits 90 % aller Kundenkontakte im Kundenservice Center abgewickelt werden, kommt dem Call Center eine nicht zu unterschätzende Bedeutung für den Gesamterfolg des Unternehmens zu. Es ist daher unabdingbar, die Abläufe im Call Center permanent im Blick zu haben und zu verbessern. Maßgebliche Erfolgsfaktoren sind hierbei Kosten und Performance. Der überwiegende Anteil, etwa dreiviertel des Gesamtbudgets, in einem Call Center sind personalbezogene Ausgaben (Call Center-Benchmark Kooperation 2004). In der Praxis erfolgt gegenwärtig die Personalbedarfsermittlung und -einsatzplanung in der Regel in den folgenden drei Schritten (vgl. Helber und Stolletz 2004):

1. Prognose des Anrufaufkommens je Periode (häufig 30- oder 60-Minutenintervalle).
2. Ermittlung der erforderlichen Zahl von Agenten je Periode für einen vorgegebenen Servicegrad hinsichtlich der Wartezeit (meist mit dem M/M/*c*-Modell).
3. Zeitliche Einplanung der Mitarbeiter über die Perioden (oder zeitliche Einplanung „anonymer“ Schichten mit anschließender Zuordnung der Mitarbeiter zu den Schichten).

An die Personaleinsatzplanung im Schritt 3 schließt sich noch eine Echtzeit-Steuerung an, in der in Abhängigkeit des aktuellen Systemzustandes z. B. die Pausen der Agenten, Besprechungen oder Trainingsmaßnahmen zeitlich festgelegt werden.

Im ersten Schritt, der Prognose, ist ein Anrufaufkommen vorherzusagen, das zwar innerhalb eines Tages oder einer Woche hochgradig variabel ist, dabei aber häufig wiederkehrende Muster aufweist (vgl. Abbildung 1). Die Datengrundlage für die Prognose wird dabei in der Regel von der automatischen Anrufverteilungsanlage (Automatic call distribution (ACD)-Anlage) geliefert. Relativ einfache Prognoseverfahren sind die exponentielle Glättung erster Ordnung auf Basis korrespondierender Zeitabschnitte oder eine Prognose durch gleitende Mittelwerte. Zieht man aufwendigere ARIMA-Methoden heran, vgl. Box et al. (1994), so erhält man bessere Ergebnisse. Wir werden uns in dieser Arbeit auf den Schritt 2 beschränken und Rückschlüsse auf eine mögliche Einsatzplanung an dem konkreten

Beispiel eines Inbound-Call-Centers machen[2]. Das folgende Kapitel gibt einen allgemeinen Überblick zu Call Centern. In der Praxis wird in Call Centern meist noch das M/M/*c*-(oder „Erlang-C“-)Warteschlangenmodell, welches in Kapitel 3 erläutert wird, bei der Personaleinsatzplanung eingesetzt. Da für das M/M/*c*-Modell analytische Lösungen für alle Kennzahlen existieren wird die mathematische Analyse von mit dem Neurosimulator FAUN[3] approximierten Kennzahlen möglich[4]. Dies geschieht nach einer kurzen Einführung in die künstliche Intelligenz in Kapitel 4 und der Erläuterung in Kapitel 5, wie die Simulationsdaten für das Training der neuronalen Netze generiert werden, in Kapitel 6.

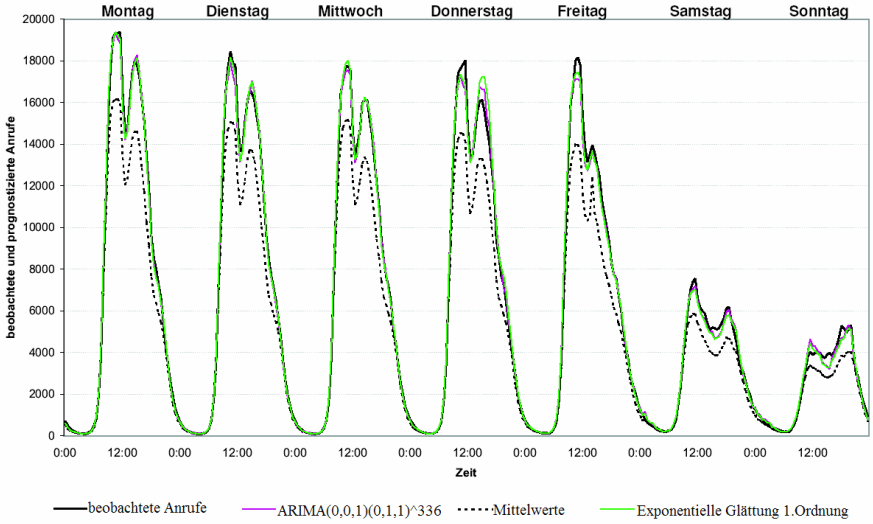


Abb. 1. Anrufaufkommen und Prognosen in Halbstundenintervallen in den Call Centern des Auskunftsdienstes der Deutschen Telegate AG vom 2. – 8.11.1998. Deutlich sind die Auswirkung der Mittagspausen und des Wochenendes zu erkennen (Helber und Stolletz 2004).

[2] In Helber und Stolletz (2004) wird eine gewinnmaximierende Agentenallokation vorgestellt, bei der gewissermaßen als „Nebenprodukt“ entsprechende Wartezeitmaße ermittelt werden. Hier wird dagegen nur das M/M/*c*-Modell betrachtet.

[3] „**F**ast **A**pproximation with **U**niversal **N**eural Networks“. Neurosimulator bezieht sich nicht auf die Simulation von Warteschlangen, sondern auf die komfortable, GUI-unterstützte Simulation gehirnanaloger Vorgänge, die als Training bzw. Lernen von künstlichen neuronalen Netzen bekannt sind (Breitner 2003).

[4] Für das M/M/1-Modell teilweise untersucht in Barthel (2003), einer Diplomarbeit betreut durch die Autoren.

2 Beispiel Call Center

In einem Call Center werden organisatorisch Telefonarbeitsplätze in Verbindung mit informations- und kommunikationstechnischer Unterstützung in Großraumbüros zusammengefasst. Die Mitarbeiter, welche in koordinierte Gruppen eingeteilt und auf die Durchführung von Telefongesprächen spezialisiert sind, werden auch als Agenten bezeichnet. Ziel eines jeden Call Centers ist ein verbesserter Kundenkontakt bzw. die Kundenbetreuung und -gewinnung bei gleichzeitiger Optimierung der Wirtschaftlichkeit. Dementsprechend wird ein Call Center als Dienstleistungsbetrieb bezeichnet, bei dem der Produzent des Dienstes und der Konsument zwar räumlich voneinander getrennt, aber zeitlich in der Regel aneinander gebunden sind. Stehen dem Agenten neben dem Telefon noch mehrere Kommunikationskanäle zur Verfügung, nennt man dies auch Contact Center oder Kundenservice Center.

Grundsätzlich werden hereinkommende Anrufe als Inbound-Gespräche und ausgehende Anrufe als Outbound-Gespräche bezeichnet. Entsprechend können Call Center in Inbound- und Outbound-Call-Center bzw. Mischformen unterteilt werden[5].

2.1 Call-Center-Marktentwicklung

Über die Call-Center-Marktentwicklung gibt es unterschiedliche Meinungen. Beispielsweise prognostizieren die Analysten von Datamonitor (Datamonitor 2004), dass der Call-Center-Markt weiter rasant wächst: Die Zahl der Call Center soll in Deutschland fortlaufend steigen, doch ebenso kontinuierlich die Zahl der Beschäftigten je Call Center sinken. Trotzdem sollen hier unter dem Strich viele Arbeitsplätze entstehen[6] (vgl. Abbildung 2). Allerdings warnen die Analysten auch davor, dass immer mehr Call Center nach Polen, Tschechien oder Ungarn abwandern, wo qualifiziertes und kundenfreundliches Personal zu niedrigeren Kosten bereit stehe. In der Call Center Benchmarkstudie 2003 wurde hingegen gezeigt, „…dass die Anforderungen des Marktes beinahe alle Betreiber vor die gleichen Probleme und Schwierigkeiten stellen. Inhouse-Center und Dienstleister haben gleichermaßen mit den Auswirkungen zu kämpfen, die wirtschaftlicher Stillstand, Kostendruck und dennoch hohe Service-Erwartungen mit

[5] In diesem Artikel beziehen wir uns nur auf die Inbound-Call-Center.

[6] Es ist darauf zu achten, dass zwischen der Zahl der Beschäftigten und der Zahl der Arbeitsplätze genau differenziert wird, da Call Center i. d. R. einen hohen Anteil an Teilzeitkräften einsetzen.

sich bringen. Die einstige „Boom“-Branche, in der „maximaler“ Service ohne Rücksicht auf die Kosten geboten wurde, expandiert nicht mehr, sondern konzentriert sich mit den vorhandenen Kapazitäten auf den Versuch, sich den veränderten Rahmenbedingungen anzupassen“ (Kestling 2004).

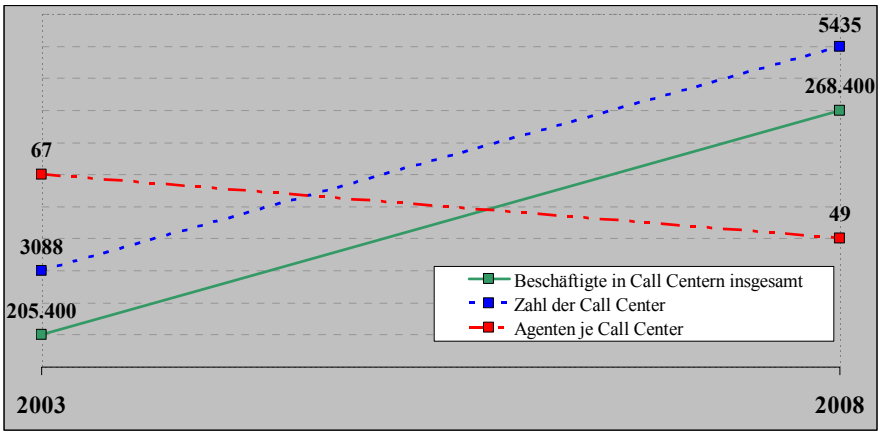


Abb. 2. Die Zahl der Call Center steigt in Deutschland fortlaufend, doch ebenso kontinuierlich sinkt die Zahl der Beschäftigten je Call Center (Datamonitor 2004).

2.2 Steigender Kostendruck in Call Centern

Das steigende Kommunikationsaufkommen in den beiden vergangenen Jahren und unreflektierter maximaler Service in Call Centern verursachten eine Kostenexplosion, die in keinem proportionalen Verhältnis zur Umsatzentwicklung steht (Call Center-Benchmark Kooperation 2004). Somit stehen Call Center nun vor der konkreten Aufgabe, Maßnahmen zur Kostensenkung aktiv umzusetzen. Dieses ist jedoch problematisch, da der überwiegende Anteil des Gesamtbudgets personalbezogene Ausgaben sind (Gehälter, Personalauswahl, Schulung und Training). Während dieser Kostenblock im Jahre 1998 noch mit rund 61% des Gesamtbudgets beziffert wird (Henn et al. 1998, S. 99), ist der Wert laut der Benchmarkstudie im Jahre 2003 schon auf rund 75% gestiegen. Die verbleibenden Positionen (Miete, lfd. Betriebskosten, Ausstattung, etc.) weisen jeweils nur eine nachgeordnete Größenordnung auf und können in der Praxis auch nicht weiter gesenkt werden. Somit sind nunmehr Ansätze gefordert, das angebotene Servicespektrum an die tatsächlichen Bedürfnisse anzupassen und gleichzeitig die Effizienz der Prozesse bzw. die Auslastung der Agenten zu steigern. Bei der Fokussierung auf Einsparpotenziale, wie die Freisetzung der tatsächlich entbehrlichen Kapazitäten, darf das Call-Center-Manage-

ment jedoch nicht die notwendige Kundenzufriedenheit gefährden (vgl. Abbildung 3).

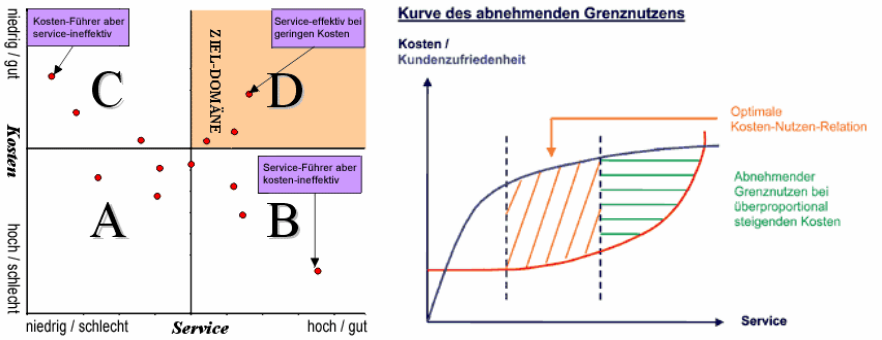


Abb. 3. Ziel eines jeden Call Centers ist es einen guten Service bei möglichst geringen Kosten anzubieten (links). Je höher aber der angebotene Service (und damit auch die Kundenzufriedenheit) ist, desto höher sind die hierfür aufzuwendenden Kosten (rechts) (Call Center-Benchmark Kooperation 2004).

3 Warteschlangentheorie anhand eines Inbound-Call-Centers

Die Warteschlangentheorie beschäftigt sich mit den strukturellen Zusammenhängen innerhalb von Warteschlangensystemen. Sie sucht nach mathematischen Lösungen um die Kennzahlen von Warteschlangensystemen berechnen zu können. Das erste Mal wurde die Warteschlangentheorie im Jahre 1908 durch die optimale Dimensionierung von Telefonnetzen durch A. K. Erlang bekannt (Zimmermann 1997, S. 362). Mit der Entwicklung der Computer wurde es dann möglich, Warteschlangenprozesse zu simulieren, um für Systeme ohne analytische Lösung Kennzahlen zu ermitteln, ohne dabei einen direkten mathematischen Systemzusammenhang herzustellen.

3.1 Warteschlangensysteme

Eine Warteschlange entsteht beispielsweise, wenn Personen in einem Call Center anrufen und dort alle Agenten besetzt sind. Meist werden sie dann in einer Warteschleife abgefangen und warten solange bis der nächste Agent frei ist. Im Kontext von Warteschlangen werden alle Personen oder Jobs als Kunden und die Warteschleife als Warteschlange bezeichnet. Die

Elemente, die ein Warteschlangensystem bilden, sind in Abbildung 4 anhand eines Schalters, wie er z. B. bei einer Post vorkommt, dargestellt.

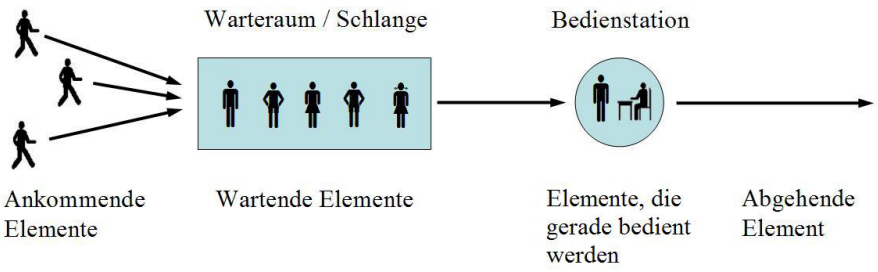


Abb. 4. Das Warteschlangensystem

Dabei sind die wichtigsten Elemente im Einzelnen:

- **Der Ankunftsprozess**. Wenn Kunden zu den Zeiten t_1 , t_2 , … , t_n eintreffen, so werden die Zeiten $\tau_j = t_j - t_{j-1}$ als Zwischenankunftszeiten bezeichnet. Es wird allgemein angenommen, dass die τ_j eine Folge von unabhängigen und identisch verteilten (iid) Zufallsvariablen sind.
- **Die Verteilung der Bedienzeit**. Die Zeit, die jede Person am Schalter bzw. im Gespräch mit dem Call Center Agenten verbringt wird ihre Bedienzeit genannt. Die Bedienzeiten werden ebenfalls als unabhängige, identisch verteilte Zufallsvariablen angenommen.
- **Anzahl an Bedieneinheiten**. Oft arbeiten in einem Call Center mehrere Agenten, die alle dieselben Dienste anbieten. In diesem Fall spricht man von mehreren Bedieneinheiten. Bieten die Agenten jedoch verschiedene Dienste an, so werden sie in Gruppen mit gleichem Angebot gegliedert, die dann jeweils eine Warteschlange bilden[7].

Die Kapazität der Warteschleife ist in Call Centern begrenzt, d. h. wenn die Warteschleife voll ist, werden weitere Anrufer abgewiesen bzw. erhalten ein Besetztzeichen. Dennoch wird zur Vereinfachung der Berechnung der Personaleinsatzplanung in Call Centern eine unbegrenzte Warteschlange angenommen. Ebenso ist bei dieser Berechnung die Anzahl aller potentiellen Kunden (Population) unendlich und die Kunden werden in der Reihenfolge bedient, in der sie ankommen (First Come, First Served (FCFS)).

[7] Ausführliche Darstellungen zur Warteschlangentheorie findet man z.B. in Schassberger (1973), Bolch (1989), Meyer und Hansen (1996, S. 210 ff.) oder Hillier und Lieberman (1997, S. 502 ff.)

Sind zusätzlich noch der Ankunftsprozess poissonverteilt, d. h. die Zwischenankunftszeiten sind iid und exponentialverteilt und die Bedienzeit exponentialverteilt, so wird dies als M/M/*c*-Modell bezeichnet. Dabei stehen die beiden „M“ für „Markovian“ und entsprechen den Exponentialverteilungen der Zwischenankunftszeiten und der Bedienzeit. *c* ist hierbei die Anzahl der Bedieneinheiten.

3.2 Das M/M/*c*-System und ein Inbound-Call-Center

Die in der Praxis eingesetzte Personaleinsatzplanungssoftware zieht regelmäßig das so genannte M/M/*c*- (oder “Erlang-C”-) Warteschlangenmodell heran, mit dem a priori unter bestimmten Annahmen zum einen

- die Wahrscheinlichkeit $P(W \leq t)$, dass die zufällige Wartezeit W nicht länger als t Zeiteinheiten ist, und zum anderen
- die mittlere Wartezeit $\mathrm{E}(W)$ der Anrufer

berechnet werden kann. In diesem Modell wird unterstellt, dass in dem Call Center Anrufe mit der durchschnittlichen Rate λ eingehen und jeder der c identischen Agenten Anrufe mit einer durchschnittlichen Rate μ bearbeitet. Die Zwischenankunftszeiten seien ebenso wie die Bearbeitungszeiten unabhängig exponentialverteilt, der Warteraum unendlich groß und alle Anrufer geduldig. Unter diesen Bedingungen ist das System stabil in dem Sinn, dass die Anzahl der Anrufer im System nicht über alle Grenzen steigt, wenn die Anrufrate λ strikt kleiner ist als die kombinierte Bearbeitungsrate (oder -geschwindigkeit) $c\mu$ aller c Agenten:

$$\lambda < c\mu\,,\ a := \frac{\lambda}{\mu} \text{ und } \rho := \frac{1}{c}\frac{\lambda}{\mu} = \frac{a}{c}\,. \tag{1}$$

Dabei stellt a das Arbeitsvolumen in der dimensionslosen Einheit „Erlangs“ dar. Die stationären Lösungen im Allgemeinen Fall für die Kennzahlen des M/M/*c*-Systems existieren genau dann, wenn der Servicegrad $\rho < 1$ ist[8], welches hier durch die Annahme $\lambda < c\mu$ schon gegeben ist. Der stationäre Zustand von Warteschlangenprozessen ist eine wichtige Eigenschaft in der Warteschlangentheorie. Er dient, zusammen mit der Markov-Eigenschaft, als Voraussetzung dafür, dass die Kennzahlen von Warteschlangensystemen und deren Verteilungen überhaupt analytisch bestimmt werden können[9]. Eine wichtige Kenngröße für eine M/M/*c*-Warteschlange

[8] Siehe hierzu auch Kapitel 5.2.

[9] Ein stochastischer Prozess ist stationär, wenn sich der Erwartungswert und die Varianz in der Zeit nicht ändern

ist die Wahrscheinlichkeit, dass ein ankommender Kunde warten muss. Der Ausdruck dafür ist bekannt unter dem Namen Erlangsche C-Formel oder Erlangsche Warteformel. Sie ist gegeben durch

$$P(N \geq c) = \frac{\frac{a^c}{c!}}{\left(1-\frac{a}{c}\right)\sum_{n=0}^{c-1}\frac{a^n}{n!}+\frac{a^c}{c!}} =: C(c,a) \qquad (2)$$

,

wobei N die Zahl der Kunden im System ist, d.h. die Zahl der Wartenden N_q plus der Zahl der Kunden, die gerade bedient werden N_s.

Die zu erwartende Zahl der Kunden in der Schlange ist

$$E(N_q) = \frac{\rho}{1-\rho} C(c,a). \qquad (3)$$

Die zu erwartende Wartezeit in der Schlange ist also

$$E(W_q) = \frac{1}{\lambda} E(N_q). \qquad (4)$$

4 Neuronale Netze

Im Idealfall lernen neuronale Netze ähnlich wie ein Gehirn an Beispielen. In künstlichen neuronalen Netzen werden einige Strukturen eines Nervensystems in karikativer Weise imitiert, um so ein Programm zu erhalten, mit dem Daten in einer bestimmten Weise verarbeitet werden können. Ein künstliches neuronales Netz besteht aus einer Menge von Knoten und deren Verbindungen untereinander, wobei jeder Knoten eine einzelne Nervenzelle modelliert. Vereinfacht ist ein neuronales Netz ein gerichteter und gewichteter Graph. Jeder Knoten j wird durch eine Variable $a_j(t)$ zum Zeitpunkt t beschrieben, die seinen Aktivierungszustand anzeigt. Für jede Verbindung zwischen zwei Knoten wird eine weitere Variable w_{ij} eingeführt, die die Stärke der Verbindung zwischen den Nervenzellen modelliert und als das Gewicht von Neuron i nach Neuron j bezeichnet wird (vgl. Abbildung 5).

4.1 Neurosimulator FAUN

Die Entwicklung des Neurosimulators FAUN begann 1997 an der TU Clausthal und wird mit der FAUN-Projektgruppe an der Universität Han-

nover weitergeführt[10]. Heute ist es mit FAUN Release 1.0 komfortabel möglich, Probleme des überwachten Lernens mit künstlichen neuronalen Netzen zu lösen. Als Netze sind so genannte 3- und 4-lagige Perzeptrons und Radial-Basis-Netze mit und ohne Direktverbindungen verfügbar (vgl. Abbildung 5). Direktverbindungen zwischen der Eingabeschicht und der Ausgabeschicht erhöhen die Flexibilität eines künstlichen neuronalen Netzes. Es können „schwach nichtlineare" Abhängigkeiten in den Ein- und Ausgabezusammenhängen leichter und besser approximiert werden. Im Vergleich zu anderen Neurosimulatoren trainiert FAUN Netze extrem schnell und konvergiert, dank globaler Optimierung, sehr zuverlässig (Breitner (2003)). Für FAUN 1.0 ist eine sehr komfortable, graphische Benutzeroberfläche unter Microsoft Windows und LINUX verfügbar. Mit der Benutzeroberfläche kann das Training der künstlichen neuronalen Netze einfach gesteuert und überwacht werden. Ferner können die besten trainierten Netze einfach ausgewählt und durch Bereitstellung des C- und FORTRAN-Quellcodes evaluiert werden.

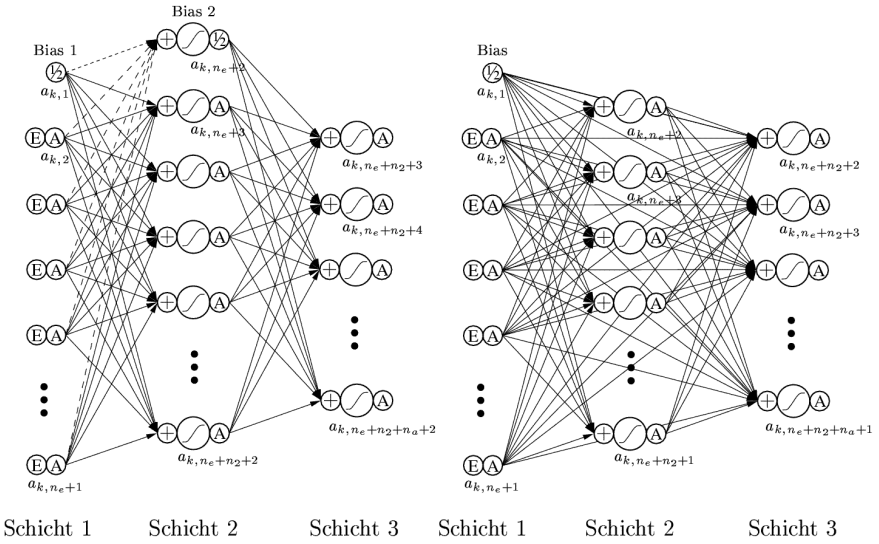


Abb. 5. Vollständig verbundenes dreilagiges Perzeptron ohne (links) bzw. mit Direktverbindungen (rechts) mit n_2 inneren Neuronen für eine n_e-dimensionale Eingabe $\boldsymbol{x}_k$ und eine n_a-dimensionale Ausgabe $f_{app}(\boldsymbol{x}_k; \boldsymbol{p})$.

[10] Siehe auch www.iwi.uni-hannover.de/faun.html.

4.2 Überwachtes Lernen

Überwachtes Lernen bedeutet, dass Ein-/Ausgabezusammenhänge $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ - so genannte Muster - mit Input $\boldsymbol{x}_i \in \mathbb{R}^{n_e}$ und Soll-Output $\boldsymbol{y}_i \in \mathbb{R}^{n_a}$, $i = 1, 2, \ldots, n_m$, aus einem Musterdatensatz $\mathtt{D}_m$ gegeben sind, für die eine „möglichst gute“ C^∞-Approximationsfunktion $f_{app}(\boldsymbol{x}; \boldsymbol{p}^*)$ berechnet werden soll, wobei $f_{app}(\boldsymbol{x}; \boldsymbol{p}^*) : \mathbb{R}^{n_e} \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_a}$ ist. $f_{app}(\boldsymbol{x}; \boldsymbol{p})$ hängt unendlich oft differenzierbar von $\boldsymbol{x}$ und dem wählbaren Parametervektor $\boldsymbol{p}$ ab. Dies ist u. a. wichtig für die Verwendbarkeit in der Praxis bzw. das Lösen schwieriger, multivariater Approximationsprobleme, wie z. B. Prognosen für Aktien, Indizes oder Zinsen sowie Kapitalmarktanalysen und -bewertungen (auch für Derivate). Dabei müssen die Muster in $\mathtt{D}_m$ problemgerecht auf den Trainingsdatensatz $\mathtt{D}_t := \{(\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots, (\boldsymbol{x}_{n_t}, \boldsymbol{y}_{n_t})\}$ und den Validierungsdatensatz $\mathtt{D}_t := \{(\boldsymbol{x}_{n_t+1}, \boldsymbol{y}_{n_t+1}), \ldots, (\boldsymbol{x}_{n_m}, \boldsymbol{y}_{n_m})\}$ aufgeteilt werden. Wichtig ist eine Equilibrierung und Skalierung $\boldsymbol{x}_i \in [-1,1]^{n_e}$ und $\boldsymbol{y}_i \in [-c,c]^{n_a}$ mit $c \in]0,1[^+$ für alle Muster. Für das Training der neuronalen Netze wird in der Regel der Trainings- und Validierungsfehler

$$\begin{aligned} \varepsilon_t(\boldsymbol{p}) &:= \sum_{i=1}^{n_t} \sum_{k=1}^{n_a} \left(f_{app_k}(\boldsymbol{x}_i; \boldsymbol{p}) - y_{i,k} \right)^{2q}, \\ \varepsilon_v(\boldsymbol{p}) &:= \sum_{i=n_t+1}^{n_m} \sum_{k=1}^{n_a} \left(f_{app_k}(\boldsymbol{x}_i; \boldsymbol{p}) - y_{i,k} \right)^{2q} \end{aligned} \tag{5}$$

benutzt, wobei $q \in \mathbb{N}$ gelten muss und oft $q = 1$ verwendet wird.

Eine gute Approximationsfunktion $f_{app}(\boldsymbol{x}; \boldsymbol{p}^*)$ weist einen kleinen Fehler $\varepsilon_t(\boldsymbol{p}^*)$ pro Muster auf, d. h. $f_{app}(\boldsymbol{x}; \boldsymbol{p}^*)$ synthetisiert die Ein-/Ausgabezusammenhänge aus $\mathtt{D}_t$ ausreichend genau. Darüber hinaus ist ein gutes globales Approximations- bzw. Extrapolationsverhalten von $f_{app}(\boldsymbol{x}; \boldsymbol{p}^*)$ erforderlich. Dafür ist notwendig, dass auch der Fehler $\varepsilon_v(\boldsymbol{p}^*)$ pro Muster klein ist. In der Praxis muss $f_{app}(\boldsymbol{x}; \boldsymbol{p}^*)$ noch weiteren Anforderungen genügen, wie z. B. eine kleine Maximal- oder Gesamtkrümmung aufweisen (Breitner 2003).

5 Simulation von Warteschlangenmodellen

Simulation ist „der experimentelle Zweig des Operations Research“ (Hillier und Lieberman 1997). Komplexe Zusammenhänge werden auf dem

Rechner nachgespielt, weil Ausprobieren in der Realität oft zu teuer ist oder das Objekt dabei zerstört wird. Beispielsweise werden im Flugsimulator kritische Turbulenzen untersucht.

Simulation kann auch dann verwendet werden, wenn es keine (exakten) mathematischen Lösungsverfahren gibt, oder wenn es zwar prinzipiell mathematische Lösungsmöglichkeiten gibt, diese jedoch zu kompliziert sind. Oft erfordern mathematisch exakte Lösungen zudem einschränkende Annahmen. Etwa bei der Untersuchung von stochastischen Zufallseinflüssen, wie z. B. dem Wartesystem M/M/*c*.

5.1 Simulation des Inbound-Call-Centers

Für ein Inbound-Call-Center sind einschränkende Annahmen bei dem M/M/*c*-Wartesystem, dass die Zwischenankunftszeiten ebenso wie die Bearbeitungszeiten unabhängig exponentialverteilt seien, alle Anrufer geduldig sind und der Warteraum unendlich groß sei. Auf viele Inbound-Call-Center treffen diese Annahmen des Erlang-C-Modells eher nicht zu. Meist steht nur eine begrenzte Zahl an Wartepositionen zur Verfügung, das heißt, wenn dieser Warteraum voll ist, erhält der Anrufer ein Besetztzeichen. Meist weisen Call Center mehrere Klassen von Anrufern oder Agenten auf oder die Anrufer sind ungeduldig und legen vorzeitig auf. Sind die Zwischenankunftszeiten und die Bearbeitungszeiten nicht exponentialverteilt, so ist es nur schwer bzw. gar nicht möglich, eine analytische Lösung zu finden. Dennoch können grundlegende Zusammenhänge auf der Basis dieses einfachsten Modells in konzeptionell klarer Weise erläutert werden und so wird es regelmäßig bei der Personaleinsatzplanung in der Praxis eingesetzt.

Die etablierten verschiedenen Simulationsprachen, wie z. B. GPSS (ab 1962 entwickelt), SIMSCRIPT, SIMULA oder DYNAMO besitzen integrierte Prozeduren, die es ermöglichen einige Warteschlangenprobleme in kurzer Zeit zu modellieren. Die Prozeduren der Programme sind aber nur allgemein anwendbar und nicht direkt problemspezifisch angepasst[11]. Deshalb und weil verschiedenste komplexere Warteschlangenprobleme ohne einschränkende Annahmen simuliert werden sollen, wurde ein eigenes Simulations-Tool erst in Maple, dann in C++ entworfen. Während die Simulationen auf einem Intel Pentium 4 mit 1,8 GHz und 512 MB RAM in Maple durchaus eine Stunde betragen können, sind es bei dem C++ Programm

[11] Vertiefende Beispiele und Erläuterungen zu den Simulationsprogrammiersprachen sind in Zimmermann (1997, S. 338), Domschke (2002, S. 220) und Siegert (1991) zu finden.

nur wenige Sekunden[12]. Für das Maple-Tool spricht jedoch, dass jede Simulation sofort mit Maple sowohl mathematisch, als auch graphisch analysiert werden kann. Weiterhin besitzt der Neurosimulator FAUN eine Maple Schnittstelle, die es einfach ermöglicht, die neuronalen Netze nicht nur mit den Simulationspunkten zu vergleichen, sondern auch mit den exakten analytischen Lösungen der Kennzahlen. Mit den Simulationsprogrammen können alle Kennzahlen simuliert werden, wir gehen hier aber nur speziell auf die mittlere Wartezeit in der Schlange und auf die Auslastung des Systems ein (vgl. Abbildung 6 und 11), da dies die relevanten Entscheidungsvariablen für einen Call-Center-Manager zur Personaleinsatzplanung sind.

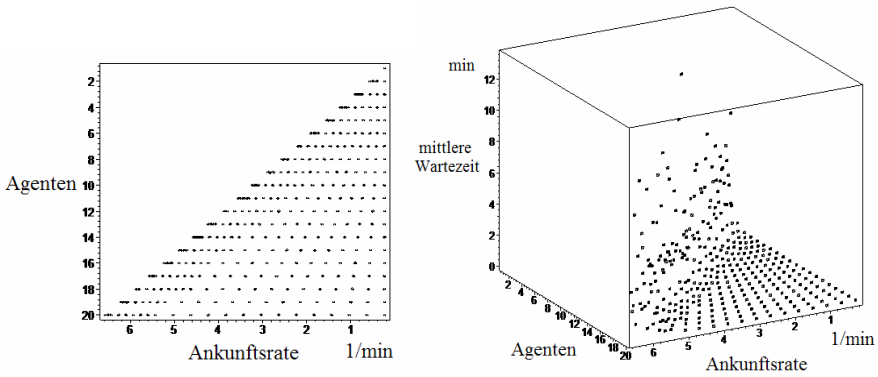


Abb. 6. Links: 349 Simulationspunkte, wobei für die Ankunftsrate $\lambda < c\mu$ gilt und $c = 1, 2, \ldots, 20$ die Anzahl der Agenten und $\mu = 1/3$ die Bedienrate ist. Rechts: Die 349 Punkte und die dazugehörige mittlere Wartezeit, wobei für jeden Punkt 5.000 ankommende Anrufer simuliert wurden.

Rechts in Abbildung 6 sind die Simulationspunkte für die mittlere Wartezeit (in Minuten) für die Kunden in der Warteschleife in Abhängigkeit von der Ankunftsrate und der Anzahl an Agenten zu sehen. Die Verteilung der Simulationspunkte in der Ebene, die aufgespannt wird durch die Anzahl der Agenten und der Ankunftsrate, ist links zu sehen. Es ist eindeutig zu erkennen, dass die Bedingung aus Gleichung (1), welche besagt, dass die Anrufrate λ strikt kleiner ist als die kombinierte Bearbeitungsrate $c\mu$ aller c Agenten, eingehalten wird. Hierbei wird angenommen, dass ein Beratungsgespräch durchschnittlich bei allen Agenten drei Minuten dauert, also

[12] Die Zeit für die Simulationen hängt direkt proportional ab von der Anzahl der simulierten Punkte und der Anzahl an Kunden, die pro Punkt simuliert werden.

die Bedienrate $\mu = 1/3$ ist und maximal 20 Agenten eingesetzt werden. Geht die Ankunftsrate gegen die Bedienrate, so ist in der Simulation zu erkennen, dass die mittleren Wartezeiten schlagartig ansteigen (vgl. Abbildung 6 rechts), während sie vorher nahezu Null sind. In dem Bereich, wo die mittlere Wartezeit nahezu Null ist, werden die Simulationspunkte durch eine variable Schrittweite bezüglich der Ankunftsrate „ausgedünnt", um nicht zu viele redundante Informationen für das Training der künstlichen neuronalen Netze zur Verfügung zu stellen und um die Zeit für die Simulationen zu senken[13]. Die Anzahl der Simulationspunkte in diesem Bereich sollte aber ungefähr genauso groß sein wie die Anzahl der übrigen Punkte, da sonst die zu approximierende Funktion hier einen zu hohen Fehler aufweist und nicht wie die analytische Lösung, bzw. auch die Simulation, eine mittlere Wartezeit von nahe Null hat (vgl. Kapitel 6).

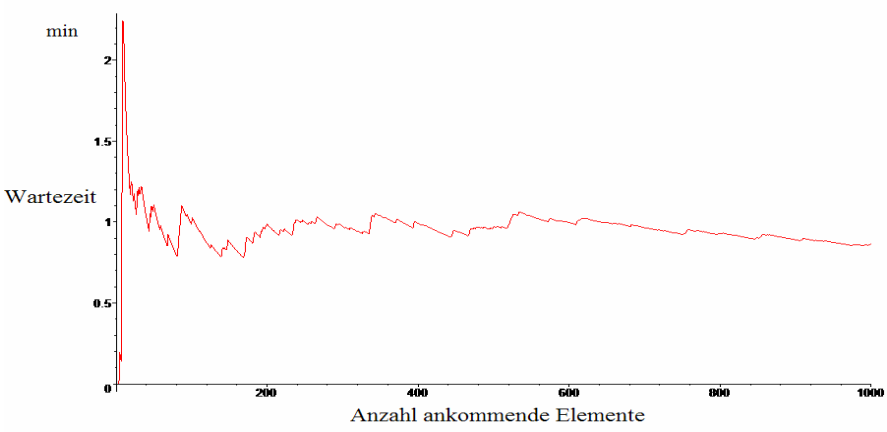


Abb. 7. Stationäres Verhalten der durchschnittlichen Wartezeit bei einer Simulation von n = 1.000 Ankünften und einer Bedienstation (ein Call Center Agent): bis n = 200 Ankünfte unterliegt das System noch starken Schwankungen, stabilisiert sich dann aber und konvergiert gegen seinen Erwartungswert.

5.2 Genauigkeit stochastischer Simulationen

Der stationäre Zustand des simulierten Systems schwankt im Zeitablauf, hat aber einen Mittelwert, um den die einzelnen Zustände schwanken bzw. zu dem sie konvergieren (vgl. Abbildung 7). Simulationen, die mit Verteilungen arbeiten, generieren Zufallsvariablen. Bei unendlich vielen Versu-

[13] Der Zeitfaktor ist hauptsächlich von Bedeutung, wenn das Maple-Tool benutzt wird bzw. auch bei dem C++ Tool, wenn wesentlich mehr als 100.000 Ankünfte pro Simulationspunkt simuliert werden sollen.

chen würde der Zustand des Systems durch die generierten Zufallsvariablen genau seinen Erwartungswert $E(x)$ treffen. Die Unendlichkeit in diesem Zusammenhang zu simulieren ist aber unmöglich. Die Genauigkeit des Erwartungswertes kann jedoch nach einer Gesetzmäßigkeit verbessert werden. Die Gesetzmäßigkeit besagt, wenn die Versuche um das n-fache steigen, verbessert sich der Fehler um das $\frac{1}{\sqrt{n}}$ -fache. Wenn der Fehler also auf nur noch 1/10 verbessert werden soll, müssen die Versuche verhundertfacht werden (Siegert 1991, S. 167). Dieses Verhältnis zeigt auf, wie zeitaufwendig eine solche Simulation sein kann, ohne dass eine wesentliche Verbesserung der Genauigkeit erreicht wird.

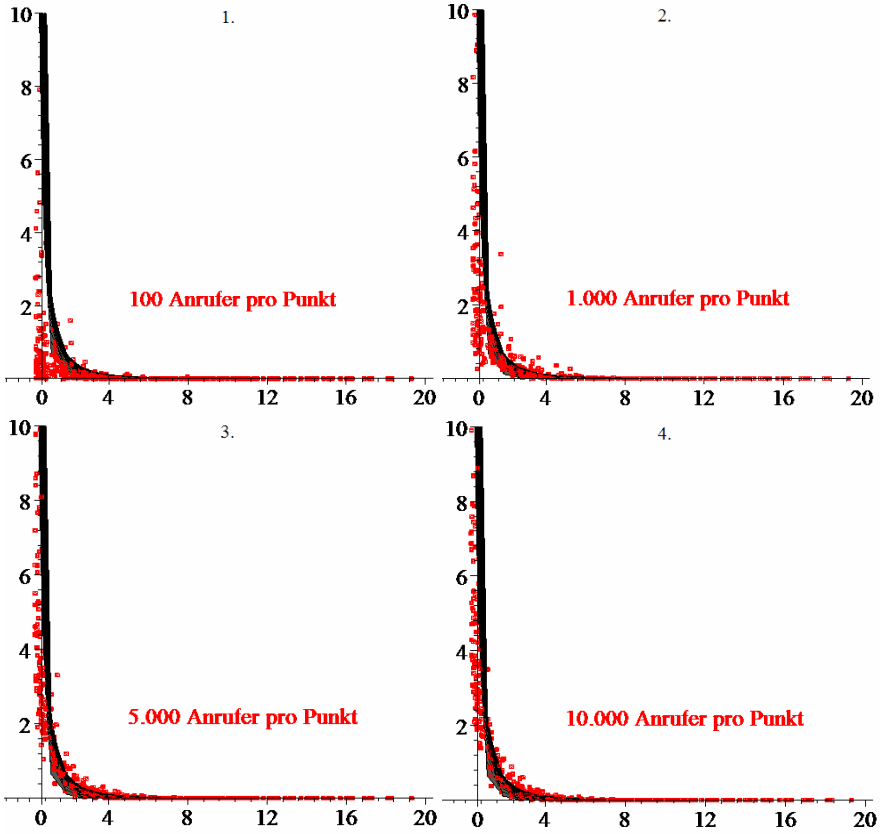


Abb. 8. Genauigkeit der Simulationen für die mittlere Wartezeit: jeweils die Seitenansicht der Abbildung 6 (rechts) mit 1.) 100 Anrufer pro Punkt, 2.) 1.000 Anrufer pro Punkt, 3.) 5.000 Anrufer pro Punkt und 4.) 10.000 Anrufer pro Punkt; die Punkte „ziehen von unten“ immer näher an die tatsächliche analytische Lösung, da das Einschwingen an Bedeutung verliert.

Werden nur wenige Anrufer simuliert, wie z. B. 100 Anrufer (Abbildung 8 1.), so ist schnell ersichtlich, dass bei 20 Call Center Agenten für die ersten 20 Anrufer keine Wartezeiten und für die folgenden kaum Wartezeiten entstehen, da die Agenten den Anstrom an Kunden sehr leicht bewältigen können. Die Simulation für die mittlere Wartezeit befindet sich noch in der Einschwingphase (Anlaufphase) in den stationären Zustand. Wird die Anzahl der simulierten Anrufe schrittweise von 100 auf 1.000, 5.000 und 10.000 erhöht, so hat die Anlaufphase immer weniger Auswirkung auf die Simulationsdaten (vgl. Abbildung 8 2., 3. und 4.) und das System stabilisiert sich. Der stationäre Zustand der einzelnen Simulationspunkte ist abhängig von der Ankunfts- und der Bedienrate. Während bei einem Agenten nur ca. 1.000 Anrufer simuliert werden müssen, sind es bei 20 schon über 5.000 Anrufer um nahezu den stationären Zustand zu erreichen[14]. Im Folgenden werden daher nur noch die Simulationsdaten mit 5.000 bzw. 10.000 simulierten Anrufern pro Punkt für das Training der neuronalen Netze benutzt. Es sei hier schon darauf hingewiesen, dass sich das System nur annähernd im stationären Zustand befinden muss, da die neuronalen Netze ein Rauschen in den Simulationsdaten sehr gut ausgleichen können. Dennoch sollten sich die Simulationsdaten sehr nah an der analytischen Lösung befinden. Besonders wenn der Bereich betrachtet wird, wo die Ankunftsrate gegen die Bedienrate geht und somit die mittlere Wartezeit sprunghaft ansteigt und die Simulationsdaten nur noch unterhalb der analytischen Lösung sind (vgl. Abbildung 8 und Abbildung 6 rechts). Aber in einem Call Center sind aus Servicegründen nur geringe Wartezeiten der Kunden erwünscht, so dass eigentlich nur der Bereich analysiert werden muss, wo die Wartezeiten nahezu Null sind bzw. nur leicht ansteigen[15]. In diesem Bereich liegt die analytische Lösung bei 5.000 bzw. 10.000 simulierten Anrufern direkt in den Simulationsdaten (vgl. Abbildung 8 3. und 4. jeweils der untere Bereich bis zu 2 Minuten Wartezeit).

[14] Es gibt keine exakten statistischen Verfahren zur Bestimmung der Anlaufphase, nur heuristische Ansätze wie die Regeln nach Conway, bzw. nach Tocher oder nach Morse, wobei letztere eine Abschätzung liefert: Anlaufzeit > 3 · Ankunftsrate / (Ankunftsrate – Bedienrate)^2, vgl. Page (1991) und Ripley (1987).

[15] In der Praxis beträgt die maximale Zeit, die ein Kunde warten darf, meist nur wenige Sekunden. Wir betrachten hier dennoch den Bereich weit über zwei Minuten Wartezeit, dementsprechend müssen 5.000 bis 10.000 Anrufer simuliert werden, obwohl für den Praxisfall Call Center weniger gereicht hätten.

6 Approximation von Kennzahlen für Warteschlangensysteme

Wesentliche Vorteile der Approximation gegenüber der einfachen diskreten Simulation von Kennzahlen sind,

- dass eine kontinuierliche Funktion zur Kostenminimierung generiert wird, und
- dass die approximierte Funktion eine bessere Annäherung an die analytische Lösung aufweist als die Simulationsdaten.

Letzteres ist dadurch begründet, dass die Simulationsdaten immer ein Rauschen aufweisen und die approximierte Funktion in diesen Daten liegt. Da auch stärkere Schwankungen der verwendeten Musterdatensätze durch das neuronale Netz wieder ausgeglichen werden, ist die Simulation, die der Approximation durch den Neurosimulator FAUN vorangestellt ist, zeitlich wesentlich weniger aufwendig, als wenn die gewünschte Kennzahl nur alleine durch Simulation bestimmt werden soll. Wichtig ist jedoch, dass die zugrunde liegende Simulation annähernd den stationären Zustand erreicht und somit hinreichend nahe der analytischen Lösung ist (vgl. Kapitel 5). Da der weitere Arbeitsschritt durch die Approximation mit FAUN nur wenige Sekunden beträgt, entsteht hierdurch kein wesentlicher Nachteil.

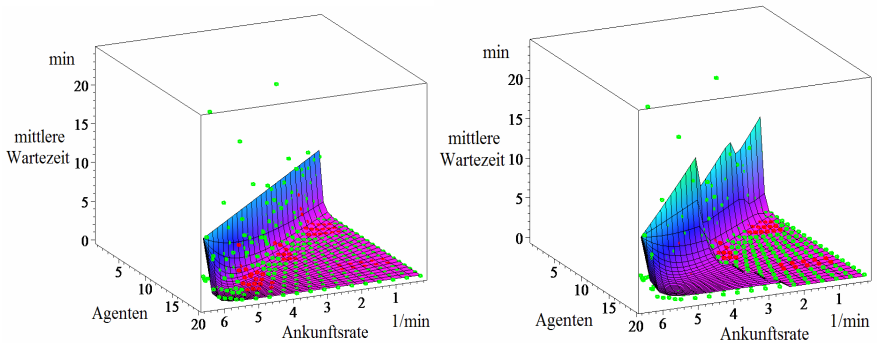


Abb. 9. Das Neuronale Netz mit einem verdeckten Neuron (links) weist einen höheren Trainingsfehler auf als das neuronale Netz mit drei inneren Neuronen (rechts), dennoch ist das rechte Netz unbrauchbar für die Kennzahlenbestimmung, da es in den Daten oszilliert.

6.1 Approximation mit FAUN 1.0

Nach der Aufteilung der 349 Simulationsdaten[16] in $n_t = 285$ Trainings- und $n_v = 64$ Validierungsdaten[17] und der Equilibrierung und Skalierung aller Muster stellt sich beim Training der neuronalen Netze schnell heraus, dass dreilagige Perceptrons ohne Shortcuts mit nur einem inneren Neuron in der verdeckten Schicht die besten Resultate für die Approximation der mittleren Wartezeit liefern[18] (vgl. Abbildung 9). Dabei wurden Topologien untersucht, bei denen die innere Neuronenanzahl n_2 von 1 bis 10 variierte. Gemäß (7) wurde zu den einzelnen Topologien der Trainingsfehler ε_t und Validierungsfehler ε_v bestimmt.

Neuronale Netze mit einer höheren Anzahl an inneren Neuronen weisen zwar einen geringeren Trainings- und Validierungsfehler auf (vgl. Tabelle 1), sind aber zur Kennzahlenbestimmung unbrauchbar, da sie nicht mehr eine „glatte Fläche“ aufweisen, sondern „wellig“ sind (vgl. Abbildung 9 und 10). Dies ist auch schon bei zwei inneren Neuronen der Fall.

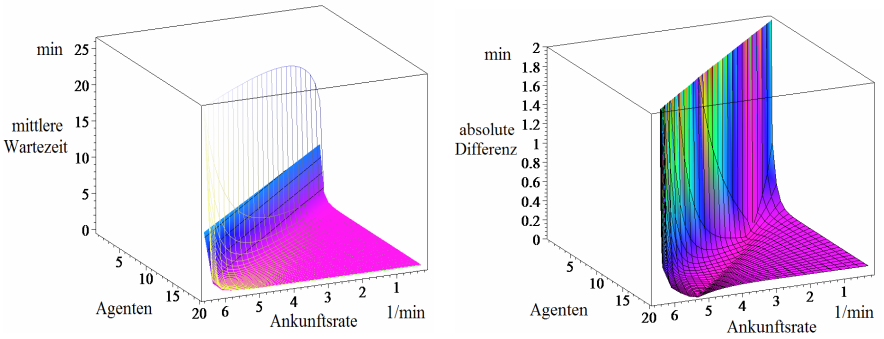


Abb. 10. Links: Vergleich der analytischen Lösung für die mittlere Wartezeit (Gitternetz) mit dem neuronalen Netz (Fläche) bei 10.000 simulierten Anrufern pro Punkt. Rechts: Absolute Differenz der beiden Lösungen in Minuten.

Neuronale Netze mit mehreren inneren Neuronen neigen dazu zwischen den Daten zu oszillieren um diese auswendig zu lernen. Diese Oszillation ist aber in vielen Praxisanwendungen, so auch hier, nicht erwünscht und so liefern neuronale Netze mit nur wenigen inneren Neuronen trotz eines höheren Trainingsfehlers bessere Ergebnisse. Daher ist eine graphische Analyse empfehlenswert bzw. in einer mathematischen Analyse muss der

[16] Vgl. Abbildungen 5 und 9.

[17] Dabei sollten Trainings- und Validierungsdaten so gewählt werden, dass $1 \le n_t/n_v \le 9$ gilt (vgl. Breitner (2003)).

[18] Dies wurde auch schon für das M/M/1 gezeigt (Barthel 2003).

Krümmungstensor möglichst klein sein (vgl. Breitner 2003). Dies zeigt auch der graphische Vergleich mit der analytischen Lösung für die zu erwartende Wartezeit in der Schlange aus (4) in Abbildung 10 (links). Die absolute Differenz der beiden Lösungen (Abbildung 10 rechts) ist fast über den ganzen Bereich nahezu immer Null. Nur in dem Bereich, wo die Ankunftsrate gegen die kombinierte Bedienrate geht, steigt die absolute Differenz sprungartig an, da hier die analytische Lösung gegen unendlich divergiert. Anhand dieser graphischen Betrachtung ist schon zu erkennen, wie gut das neuronale Netz diese Kennzahl approximiert. Die entsprechende mathematische Analyse bezüglich der Abweichungen der beiden Verfahren wird in Tabelle 2 dargestellt.

Tabelle 1. Trainings- und Validierungsfehler der besten Approximationsfunktionen

Topologie	1 inneres Neuron	2 innere Neuronen	3 innere Neuronen	5 innere Neuronen	10 innere Neuronen
ε_t*	2,64	2,56	2,49	1,95	1,91
prozentualer Fehler	7,3 %	7,2 %	7,1 %	6,3 %	6,2 %
ε_v*	0,64	0,74	0,55	0,47	0,53
Rechenzeit in sec.	6,4	10,7	14,9	23,0	43,5

6.2 Qualität der Approximation

Um die Approximation der mittleren Wartezeit in der Warteschleife mit der analytischen Lösung und den tatsächlichen Simulationspunkten vergleichen zu können, ist zu beachten, dass die Inputwerte für das neuronale Netz skaliert eingehen, während die Ausgabe zurückskaliert werden muss, so dass die Wartezeit wieder in Minuten abzulesen ist.

In der Tabelle 2 werden die drei Verfahren Approximation, Simulation und analytische Lösung der mittleren Wartezeit in der Warteschleife für das entsprechende Beispiel des Inbound-Call-Centers mit maximal 20 Agenten verglichen. Dazu werden für drei verschiedene Bereiche des Lösungsraumes, aufgespannt durch die Ankunftsrate, die kombinierte Bedienrate und die zugehörige mittlere Wartezeit, die minimale, maximale

und durchschnittliche Abweichung der drei Verfahren untereinander in Minuten bestimmt.

Die approximierte Lösung auf Basis von neuronalen Netzen hat bis auf einen Wert immer eine geringere durchschnittliche Abweichung zu der analytischen Lösung als die beiden Simulationen einmal mit 5.000 Anrufern pro Punkt und einmal mit 10.000 Anrufern pro Punkt (vgl. Tabelle 2)[19].

Wird der gesamte Bereich (jeweils die ersten drei Zeilen der Tabelle 2) betrachtet, so treten hier die größten maximalen Abweichungen auf. Dies ist dadurch begründet, dass, wenn die Ankunftsrate λ sich der kombinierten Bearbeitungsrate $c\mu$ annähert, die analytische Lösung sehr schnell gegen unendlich geht und auch die Simulationspunkte in diesem Bereich größere Schwankungen aufweisen. Dennoch beträgt die durchschnittliche Abweichung des neuronalen Netzes zur exakten Lösung jeweils nur etwas mehr als eine Minute, da auch der Bereich betrachtet wird, wo alle drei Verfahren eine mittlere Wartezeit von nahezu Null haben.

Tabelle 2. Vergleich des besten künstlichen neuronalen Netzes (NN) mit der analytischen Lösung (Ana.) und den Simulationsdaten (Simu.) (alle Werte in Minuten angegeben)

Anzahl Anrufer		**Ana. vs. NN**		**NN vs. Simu.**		**Ana. vs. Simu.**	
		5.000	**10.000**	**5.000**	**10.000**	**5.000**	**10.000**
$E(W_a)$ gesamt 349 Pkt	min Abw.	0,0003	0,0007	0,0002	0,0001	$2 \cdot 10^{-21}$	$2 \cdot 10^{-21}$
	max Abw.	19,04	17,25	9,37	16,76	22,87	22,15
	Ø Abw.	**1,45**	**1,32**	0,51	0,67	1,65	1,45
$E(W_q) \leq 5$ 280 Pkt	min Abw.	0,0003	0,0007	0,0002	0,0001	$2 \cdot 10^{-21}$	$2 \cdot 10^{-21}$
	max Abw.	1,32	1,33	3,09	2,94	3,64	2,89
	Ø Abw.	**0,12**	**0,16**	0,19	0,23	0,20	0,16
$0{,}1 \leq E(W_q) \leq 5$ 122 Pkt	min Abw.	0,0003	0,0007	0,003	0,002	0,0005	0,01
	max Abw.	1,32	1,33	3,09	2,94	3,64	2,89
	Ø Abw.	**0.24**	**0.26**	0,39	0,42	0,47	0,37

[19] Ausnahme bildet der zweite Bereich bei 10.000 simulierten Anrufern, da sind beide Werte gleich 0,16.

Dementsprechend werden noch zwei zusätzliche Bereiche analysiert. Zum einen wird der Lösungsraum betrachtet für den die mittlere Wartezeit nicht größer als fünf Minuten ist ($E(W_q) \leq 5$), da hier angenommen wird, dass dies für die Kunden des Inbound-Call-Centers eine zumutbarer Wartezeit ist[20]. Zum anderen wird der Lösungsraum analysiert für den zusätzlich der Bereich nicht betrachtet wird, wo alle drei Verfahren fast Null sind ($0{,}1 \leq E(W_q) \leq 5$). Es ist ersichtlich, dass das neuronale Netz, welches durch nur 5.000 Anrufer pro Simulationspunkt generiert wurde, geringere Werte aufweist als das beste neuronale Netz mit 10.000 Anrufern pro Punkt. Die durchschnittliche Abweichung beträgt dann nur noch ungefähr sechs (für $E(W_q) \leq 5$) bzw. 12 Sekunden (für $0{,}1 \leq E(W_q) \leq 5$) und die maximale Abweichung etwas mehr als eine Minute, vgl. Tabelle 2. Es ist daher anzunehmen, dass für diese eingeschränkten praxisrelevanten Bereiche jedoch 5.000 simulierte Anrufer zur Generierung der neuronalen Netze ausreichen, obwohl die approximierte Lösung, generiert durch 10.000 Anrufer pro Punkt, insgesamt eine bessere durchschnittliche Abweichung aufweist. Das beste neuronale Netz weicht auf einem Intervall von null bis fünf Minuten für die mittlere Wartezeit nur im Durchschnitt sechs Sekunden von der analytischen Lösung ab.

6.3 Auswertung des Inbound-Call-Centers

Neben der durchschnittlichen Wartezeit der Kunden in der Warteschleife ist für einen Call-Center-Manager noch der Auslastungsgrad bzw. Servicegrad ρ seiner Agenten entscheidend. Wenngleich auch der Auslastungsgrad hier durch (1) sehr einfach bestimmt werden kann, wurde er mit simuliert und dann mit FAUN approximiert, da ρ bei weit aus schwierigeren Problemen nicht mehr so leicht zu bestimmen ist (z. B. sind nicht alle Agenten identisch und haben alle die gleiche Bedienrate μ).

Während die Ankunftsrate tageszeitabhängig und exogen ist (vgl. Abbildung 1), d. h. nicht beeinflussbar vom Call-Center-Manager[21], ist die Anzahl an Agenten dagegen endogen, also steuerbar. In Abbildung 11 ist zu erkennen, dass, wenn die Ankunftsrate gegen die kombinierte Bedienrate geht, die durchschnittliche Wartezeit (links) lange Zeit Null bleibt, dagegen aber der Auslastungsgrad (rechts) ständig steigt. Ein Call-Center-Manager ist also bei vorgegebener Ankunftsrate bestrebt, einen möglichst

[20] Oft liegt diese obere Grenze in der Praxis doch weit niedriger.

[21] Die Warteschleifen in Call Centern haben eine vom System bzw. auch vom Call-Center-Manager vorgegebene bzw. einstellbare Kapazität, so dass Kunden bei voller Warteschleife abgewiesen werden, und somit kann indirekt Einfluss genommen werden.

hohen zumutbaren Auslastungsgrad seiner Agenten gegenüber möglichst geringen Wartezeiten der Kunden zu erreichen. Die approximierten Lösungen dieser beiden Kennzahlen sind also nur dann sinnvoll einsetzbar, wenn ein hinreichend genaues Prognoseverfahren für die Ankunftsrate der nächsten Stunden bzw. Tage zur Verfügung steht.

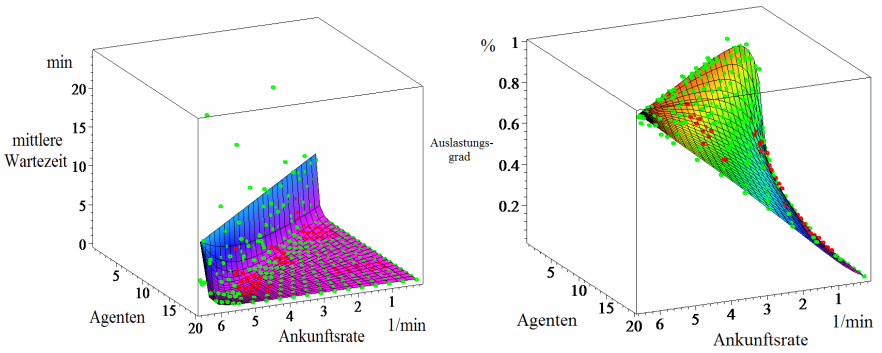


Abb. 11. Approximierte durchschnittliche Wartezeit der Kunden in der Warteschleife (links) und der dazugehörige approximierte Auslastungsgrad (rechts) mit den jeweiligen Simulationspunkten.

Soll zum Beispiel die mittlere Wartezeit eines Kunden nur eine Minute betragen und die prognostizierte Ankunftsrate für die nächsten Zeitintervalle fünf Kunden pro Minute beträgt, so liefert die approximierte Lösung, dass 16,32 Agenten, also 17, eingesetzt werden müssen. Dabei beträgt der approximierte Auslastungsgrad der 17 Agenten 87,85%. Die analytische Lösung für die mittlere Wartezeit in der Schleife liefert einen Wert von 16,49 Agenten, also auch 17, und der Auslastungsgrad der 17 Agenten beträgt bei der analytischen Lösung 88,23%.

7 Fazit und Ausblick

Der Neurosimulator FAUN bietet eine Möglichkeit, für alle Warteschlangensysteme eine approximierte, explizite Lösung für deren Kennzahlen zu generieren. Dieser Aufsatz zeigt anhand des Standardmodells M/M/*c* wie dies möglich wird und welche Güte die approximierte Lösung im Gegensatz zur – hier ermittelbaren – analytischen Lösung besitzt. Die so gewonnenen Erkenntnisse können auf Modelle ohne analytische Lösung übertragen werden, die bisher nur mit Simulationen gelöst werden können.

Vorteile der Approximation von Warteschlangenkennzahlen bei schwierigen Warteschlangenproblemen gegenüber der Analyse durch diskrete Simulationen sind,

- dass eine analytische Funktion zur Personaleinsatzplanung und Kostenminimierung generiert wird, die extrem schnell auswertbar ist, und
- dass das unvermeidliche Rauschen in den Simulationsdaten geglättet wird, d. h. die Kennzahlen genauer verfügbar sind bzw. deutlich weniger Simulationen nötig sind.

Es brauchen nicht besonders viele Punkte zum Training der neuronalen Netze simuliert werden gegenüber einer „flächendeckenden" Auswertung mit einer Simulation. Dies ist ein erheblicher Zeitvorteil, da der zusätzliche Schritt des FAUN-Trainings i. d. R. nur wenige Sekunden dauert.

Ein weiterer Vorteil ist, dass bei der Simulation der Muster für das Training mit FAUN unterschiedlichste Verteilungen für die Ankunfts- und Bedienrate, so wie sie in der Praxis tatsächlich vorkommen, eingesetzt werden können. Beispielsweise kann so anhand des realen Anruferaufkommens in einem Call Center die tatsächliche Verteilung über einen längeren Zeitraum bestimmt und für die Simulation verwendet werden. Analog kann mit dem Bedienprozess verfahren werden. Aus realen Daten können dann Simulationspunkte für das Training der künstlichen neuronalen Netze generiert werden, um so die Abläufe in einem Call Center durch realistischere Bestimmung der Warteschlangenkennzahlen wesentlich praxisnäher abzubilden. Es muss also nicht das M/M/*c*-Modell mit all seinen Einschränkungen als Grundlage für die Mustergenerierung dienen. Dieses aus der Praxis gewonnene Datenmaterial kann durchaus verrauscht sein, da neuronale Netze mit wenigen inneren Neuronen sich „in die Daten legen" und so oft ein gleichmäßiges, oft weißes Rauschen ausgleichen.

Bevor jedoch schwierigere Warteschlangenprobleme ohne analytische Lösung anhand neuronaler Netze analysiert werden, muss weiter untersucht werden, ab wann die Simulationspunkte in Abhängigkeit von der Ankunfts- und Bedienrate nahezu ihren stationären Zustand erreichen. Wichtig ist, dass der jeweilige Einschwingvorgang der Simulationen nur wenig mitgelernt wird. Ein Ansatz zur besseren Generierung der Simulationsdaten wäre zum Beispiel, dass die Auswertung der Wartezeiten bzw. die Bestimmung der Kennzahlen erst nach einer gewissen Anzahl von ankommenden Kunden anfängt, um so den Vorgang des Einschwingens abzuschneiden.

Literatur

Barthel A (2003) Effiziente Approximation von Kenngrößen für Warteschlangen mit dem Neurosimulator FAUN 1.0. Diplomarbeit am Institut für Wirtschaftswissenschaft der Universität Hannover, Königsworther Platz 1, D-30167 Hannover

Bolch G (1989) Leistungsbewertung von Rechensystemen mittels analytischer Warteschlangenmodelle. Teubner, Stuttgart

Box GEP, Jenkins G M, Reinsel G C (1994) Time Series Analysis: Forecasting and Control, 3. Aufl. Prentice Hall, New Jersey

Breitner MH (2003) Nichtlineare, multivariate Approximation mit Perzeptrons und anderen Funktionen auf verschiedenen Hochleistungsrechnern. Akademische Verlagsgesellschaft Aka GmbH, Berlin

Call Center-Benchmark Kooperation (2004) Kooperationsprojekt: Purdue University, Universität Hamburg, Initiator der profiTel MANAGEMENT CONSULTING. http://www.callcenter-benchmark.de/index3.html. Letzter Abruf: 10.10.2004

Datamonitor (2004) Datamonitor. http://www.datamonitor.com. Letzter Abruf 21.05.2004

Domschke W, Drexl A (2002) Einführung in Operations Research, 5. Aufl. Springer, Berlin

Helber S, Stolletz R (2004) Call Center Management in der Praxis: Strukturen und Prozesse betriebswirtschaftlich optimieren. Springe, Berlin

Henn H, Kruse JP, Strawe OV (1998) Handbuch Call Center Management (2. Aufl.). Telepublic Verlag, Hannover

Hillier FS, Lieberman GJ (1997) Operations Research, 5. Aufl. Oldenbourg, München Wien

Kestling V (2004) Das Call Center-Jahr 2003 – Rückblick und Ausblick. Markt & Trends CallCenter, CRM, IT/TK, Telesales und –services, Presse Mitteilung

Meyer M, Hansen K (1996) Planungsverfahren des Operations Research (4. Aufl.). Vahlen, München

Page B (1991) Diskrete Simulation. Springer, Berlin

Ripley BD (1987) Stochastic Simulation. John Wiley & Sons, New York

Schassberger R (1973) Warteschlangen. Springer, Berlin

Siegert HJ (1991) Simulation zeitdiskreter Systeme. Oldenbourg, München Wien

Zimmermann W (1997) Operations Research: quantitative Methoden zur Entscheidungsvorbereitung (8. Aufl.). Oldenbourg, München Wien

IWI Discussion Paper Series

ISSN 1612-3646

Michael H. Breitner, *Rufus Philip Isaacs and the Early Years of Differential Games*, 36 p., # 1, January 22, 2003.

Gabriela Hoppe and Michael H. Breitner, *Classification and Sustainability Analysis of E-Learning Applications*, 26 p., # 2, February 13, 2003.

Tobias Brüggemann and Michael H. Breitner, *Preisvergleichsdienste: Alternative Konzepte und Geschäftsmodelle*, 22 S., # 3, February 14, 2003.

Patrick Bartels and Michael H. Breitner, *Automatic Extraction of Derivative Prices from Webpages using a Software Agent*, 32 p., # 4, May 20, 2003.

Michael H. Breitner and Oliver Kubertin, *WARRANT-PRO-2: A GUI-Software for Easy Evaluation, Design and Visualization of European Double-Barrier Options*, 35 p., # 5, September 12, 2003.

Dorothée Bott, Gabriela Hoppe and Michael H. Breitner, *Nutzenanalyse im Rahmen der Evaluation von E-Learning Szenarien*, 14 p., # 6, October 21, 2003.

Gabriela Hoppe and Michael H. Breitner, *Sustainable Business Models for E-Learning*, 20 p., # 7, January 5, 2004.

Heiko Genath, Tobias Brüggemann and Michael H. Breitner, *Preisvergleichsdienste im internationalen Vergleich*, 40 p., # 8, June 21, 2004.

Dennis Bode and Michael H. Breitner, *Neues digitales BOS-Netz für Deutschland: Analyse der Probleme und mögliche Betriebskonzepte*, 21 p. # 9, July 5, 2004.

Caroline Neufert and Michael H. Breitner, *Mit Zertifizierungen in eine sicherere Informationsgesellschaft*, 20 p., # 10, July 5, 2004.

Marcel Heese, Günter Wohlers and Michael H. Breitner, *Privacy Protection against RFID Spying: Challenges and Countermeasures*, 22 p., # 11, July 5, 2004.

Liina Stotz, Gabriela Hoppe and Michael H. Breitner, *Interaktives Mobile(M)-Learning auf kleinen Endgeräten wie PDAs und Smartphones*, 31 p., # 12, August 18, 2004.

Frank Köller and Michael H. Breitner, *Optimierung von Warteschlangensystemen in Call Centern auf Basis von Kennzahlenapproximation*, 24 p., # 13, Januar 10, 2005.