Modellierung korrelierter Eingabedaten für Simulationen

Jan Kriege
Fakultät für Informatik,
TU Dortmund
jan.kriege@tu-dortmund.de

Original-Artikel erschienen in
Ausgezeichnete Informatikdissertationen 2012
GI-Edition Lecture Notes in Informatics (LNI)
ISBN 978-3-88579-417-2

Modellierung korrelierter Eingabedaten für Simulationen*

Jan Kriege
Fakultät für Informatik, TU Dortmund
jan.kriege@tu-dortmund.de

Zusammenfassung

Ein wichtiger Schritt bei der Erstellung von Simulationsmodellen ist die geeignete Repräsentation von korrelierten Eingabedaten. Für unabhängig und identisch verteilte Daten existieren zahlreiche Ansätze, während sich die Modellierung von Korrelation noch schwierig gestaltet. In diesem Beitrag wird ein Ansatz vorgestellt der Phasenverteilungen und ARMA-Prozesse kombiniert um korrelierte Daten abzubilden. Für diesen Ansatz wird ein Verfahren zur Parameteranpassung vorgestellt und anhand von echten Verkehrsdaten wird gezeigt, dass der Ansatz eine gute Darstellung von Verteilung und Korrelation ermöglicht.

1 Einleitung

Die Leistungsbewertung von Computer- und Kommunikationssystemen spielt eine wichtige Rolle beim Entwurf neuer Systeme oder bei Änderungen an vorhandenen Systemen. Aufgrund der zunehmenden Komplexität dieser Systeme ist es oftmals nicht mehr möglich, Experimente mit dem realen System durchzuführen und der Einsatz stochastischer Modelle ist zur Ermittlung von Leistungsmaßen erforderlich. Für eingeschränkte Modellklassen bieten sich numerische Techniken [23] zur Ermittlung dieser Maße an. Aufgrund der zuvor erwähnten Komplexität bleibt aber oft nur eine simulative Untersuchung [16], für die keine Einschränkungen des Modells erforderlich sind. Eine geeignete Darstellung der Eingabedaten (wie z.B. Ankunftszeiten oder Bedienzeiten) ist unerlässlich, um valide Simulationsmodelle und somit brauchbare Simulationsergebnisse zu erhalten. Oftmals liegen Beobachtungen aus einem realen System vor und es wird versucht, wesentliche Charakteristika dieser Beobachtungen durch Verteilungen oder stochastische Prozesse abzubilden. Für den Fall, dass diese Daten unabhängig und identisch verteilt sind, sich also durch eine Verteilung geeignet abbilden lassen, existiert eine umfangreiche Theorie [16]. Methoden zur Parameteranpassung von Verteilungen werden von Standardwerkzeugen [17] zur Verfügung gestellt und gängige Simulationssoftware unterstützt üblicherweise den Einsatz einer umfangreichen Auswahl von Verteilungen. Anders stellt sich die Lage dar, wenn die Beobachtungen aus dem realen System Abhängigkeiten und Korrelationen enthalten, wie es in vielen Anwendungsbereichen der Fall ist. Als Beispiel seien hier Rechner- und Kommunikationsnetzwerke [9, 21] genannt. Es ist bekannt, dass die Vernachlässigung dieser Korrelationen zu einer Unterschätzung des Ressourcenbedarfs führt und

*Englischer Titel der Dissertation: “Fitting Simulation Input Models for Correlated Traffic Data”

Verteilungen zur Modellierung der Eingabedaten hier nicht mehr ausreichen [19]. In der Vergangenheit wurden daher unterschiedliche Typen von stochastischen Prozessen vorgestellt, um sowohl die empirische Verteilung als auch die auftretenden Abhängigkeiten modellieren zu können, z.B. Autoregressive-Moving-Average (ARMA) Prozesse [4], ARTA Prozesse [7] und Markovsche Ankunftsprozesse (MAPs) [20].

Beim Einsatz dieser Prozesse in Simulationsmodellen ergeben sich aber noch zahlreiche Probleme: So gestaltet sich die Parameteranpassung für Prozesse deutlich schwieriger als für Verteilungen und ist meist nur prototypisch implementiert. Fehlende Unterstützung in den gängigen Simulationstools erschwert den Einsatz der Prozesse zusätzlich. Außerdem unterstützen viele Prozesstypen nur eine eingeschränkte Klasse von Verteilungen, die nicht ausreicht, um z.B. Zwischenankunftszeiten in Rechnernetzen adäquat zu modellieren.

Ziel dieser Arbeit ist es daher, die zuvor angesprochenen Probleme beim Einsatz stochastischer Prozesse in der Simulation zu mildern. In Abschnitt 2 werden wir zunächst einen kurzen Überblick über existierende Ansätze zur Modellierung von korrelierten Daten geben und einige Vor- und Nachteile herausarbeiten. In Abschnitt 3 wird schließlich eine neue Klasse von stochastischen Prozessen vorgestellt, die Phasenverteilungen zur Modellierung der empirischen Verteilung mit ARMA Prozessen zur Modellierung der Autokorrelation kombiniert. Die Umsetzung dieser theoretischen Verfahren in Werkzeuge zur Parameteranpassung und Simulation wird in Abschnitt 4 beschrieben. In Abschnitt 5 präsentieren wir einige experimentelle Ergebnisse, die zeigen dass die neue Klasse von stochastischen Prozessen geeignet ist, sowohl Verteilung als auch Korrelation von realen Netzwerkdaten geeignet abbilden zu können.

2 Grundlagen

Wir betrachten die Modellierung von stochastischen Prozessen Y_t anhand von Beobachtungen aus einem realen System, die üblicherweise in Form eines Traces $T = (t_1, t_2, \ldots, t_l)$ vorliegen. Sofern die Beobachtungen unabhängig und identisch verteilt sind, ist es ausreichend die Parameter einer geeigneten Verteilungsfunktion anzupassen, wozu eine umfangreiche Theorie existiert [16]. Die Qualität der Parameteranpassung wird häufig mit Hilfe der Likelihood-Funktion $L(T, X) = \prod_{i=1}^{l} f_X(t_i)$ für einen Trace T und eine angepasste Verteilung mit Dichtefunktion $f_X(x)$ bestimmt. Aus numerischen Gründen nutzt man oft die log-Likelihood $l(T) = \log L(T, X)$. Im Gegensatz zur Parameteranpassung von Verteilungen existieren für die Anpassung von stochastischen Prozessen, die sowohl die Verteilung als auch die Korrelation eines Traces abbilden deutlich weniger Ergebnisse. Wir werden im Folgenden einige Ansätze mit ihren Stärken und Schwächen vorstellen.

2.1 Phasenverteilungen und Markovsche Ankunftsprozesse

Phasenverteilungen (PH-Verteilungen) lassen sich mittels einer $n \times n$ Matrix D_0 mit Transitionsraten und einem Vektor π mit initialen Wahrscheinlichkeiten definieren und beschreiben unabhängig und identisch verteilte Zufallsvariablen als Zeit bis zur Absorption in einer Markov-Kette [20]. Die Momente und die Dichtefunktion einer PH-Verteilung ergeben sich wie folgt:

$$\mu_i = E(X^i) = i!\pi M^i e^T \text{ und } f_X(t) = \pi e^{D_0 t} e^T \tag{1}$$

mit $M = -(D_0)^{-1}$ und dem Vektor $e = (1, \cdots, 1)$ der Länge n. Phasenverteilungen sind eine sehr flexible Familie von Verteilungen, die jede Verteilung mit positiver Dichtefunktion in $(0, \infty)$ beliebig genau approximieren können [1]. Da die Darstellung von Phasenverteilungen redundant ist, gestaltet sich die Parameteranpassung im Allgemeinen schwierig. Die meisten Verfahren schränken daher die Klasse der unterstützten Verteilungen ein. Für azyklische PH-Verteilungen, bei denen sich D_0 als obere Dreiecksmatrix darstellen lässt, existieren Ansätze, die die Parameter anhand der empirischen Momente [6] oder durch Maximierung der Likelihood [24] anpassen.

Sofern auch Korrelation berücksichtigt werden soll, können PH-Verteilungen zu Markovschen Ankunftsprozessen (MAPs) erweitert werden. Insbesondere bei einer größeren Anzahl von Korrelationskoeffizienten ist die Parameteranpassung von MAPs allerdings schwierig, auch wenn neuere Ansätze gute Resultate erzielen [8].

2.2 ARMA Prozesse

Autoregressive Moving Average (ARMA) Prozesse [4] dienen zur Modellierung von Zeitreihen. Ein $ARMA(p, q)$ Prozess ist definiert als

$$Z_t = \alpha_1 Z_{t-1} + \alpha_2 Z_{t-2} + ... + \alpha_p Z_{t-p} + \beta_1 \epsilon_{t-1} + \beta_2 \epsilon_{t-2} + ... + \beta_q \epsilon_{t-q} + \epsilon_t \tag{2}$$

wobei die Rauschterme ϵ_t normalverteilt mit Mittelwert 0 und Varianz σ_ϵ^2 sind. Für $q = 0$ ergibt sich ein sogenannter $AR(p)$ Prozess der durch $Z_t = \alpha_1 Z_{t-1} + \alpha_2 Z_{t-2} + ... + \alpha_p Z_{t-p} + \epsilon_t$ beschrieben ist. ARMA Modelle sind sehr flexibel bei der Modellierung der Autokorrelation und Methoden zur Parameteranpassung werden durch Standardsoftware für Statistik zur Verfügung gestellt [4]. Die Verteilung der Modelle ist allerdings die gewichtete Summe von $N(0, \sigma_\epsilon^2)$ verteilten Zufallsvariablen, was bedeutet, dass nur Verteilungen modelliert werden können, die sich durch einfache Transformationen aus der Normalverteilung ergeben.

2.3 ARTA Prozesse

ARTA-Prozesse [3, 7] kombinieren einen $AR(p)$ Prozess $\{Z_t; t = 1, 2, ...\}$ mit einer Verteilung F_Y und versuchen so die genannten Schwächen des $AR(p)$ Prozesses bei der Modellierung der Verteilung zu beseitigen. Sie beschreiben eine Sequenz $Y_t = F_Y^{-1}[\Phi(Z_t)]$ $(t = 1, 2, ...)$, wobei Φ die Verteilungsfunktion der Standardnormalverteilung ist. Der $AR(p)$ dient zur Modellierung der Korrelation und wird so konstruiert, dass er standard-normalverteilt ist. Die Transformation $U_t = \Phi(Z_t)$ erzeugt dann gleichverteilte Zufallsvariablen im Intervall $(0, 1)$ [11] und $Y_t = F_Y^{-1}[U_t]$ ergibt eine Zeitreihe mit der gewünschten Verteilung F_Y. In [7, 3] wurden Verfahren zur Parameteranpassung von ARTA-Prozessen entwickelt. Da ARTA-Prozesse auf der Invertierung der Verteilungsfunktion basieren, eignen sie sich nur für Verteilungen für die F_Y^{-1} effizient berechnet werden kann, da die Invertierung sowohl für die Parameteranpassung als auch zur Erzeugung von Zufallszahlen durchgeführt werden muss. Für Phasenverteilungen kann die inverse Verteilungsfunktion im Allgemeinen nicht effizient berechnet werden. Allerdings ergibt sich für azyklische Phasenverteilungen, die sich als Kombination von endlichen Sequenzen von Exponentialverteilungen darstellen lassen, eine andere Möglichkeit Verteilung und $ARMA(p, q)$-Prozess zu kombinieren, wie wir im Folgenden zeigen werden.

3 Modellierung von korrelierten Daten mit Phasenverteilungen und ARMA-Prozessen

Um die Vorteile von azyklischen Phasenverteilungen (APH) gegenüber anderen Verteilungen, die für den ARTA-Ansatz geeignet sind, zu verdeutlichen, haben wir APHs (in diesem Fall Hyper-Erlang-Verteilungen mit r Zuständen ($HErD(r)$)) und einige andere Verteilungen für die F_Y^{-1} effizient berechnet werden kann an drei unterschiedliche Traces mit Netzwerktraffic angepasst. Aus Platzgründen zeigen wir hier nur die Ergebnisse für den Trace *BC-pAug89* [18]. Als Verteilungen wurden Exponential- und Lognormal-Verteilung gewählt, für die Maximum-Likelihood-Schätzer bekannt sind [16], sowie Verteilungen aus der Johnson-Familie [25] und die Weibull-Verteilung. Die Hyper-Erlang-Verteilungen wurden mit dem Expectation-Maximization-Ansatz aus [24] angepasst. Tabelle 1 enthält die Likelihood-Werte und die Momente der Verteilungen zusammen mit dem relativen Fehler $(|\mu_i - \hat{\mu}_i|/\hat{\mu}_i) \cdot 100$ in Prozent, wobei μ_i und $\hat{\mu}_i$ das i-te Moment der Verteilung bzw. des Traces ist. Tabelle 1 zeigt deutlich, dass mit Phasenverteilungen üblicherweise eine bessere Anpassung sowohl bezüglich der Likelihood als auch der Momente möglich ist, insbesondere, wenn sich die Zustandszahl der APH-Verteilung erhöht. Diese Ergebnisse verdeutlichen, dass es sinnvoll ist Phasenverteilungen zur Modellierung von Netzwerktraffic einzusetzen,

	Verteilung	Log-Likelihood	Moment 1	Moment 2	Moment 3
BC-pAug89	Exponential	−999999	1.0 (0.0%)	2.0 (52.7%)	6.0 (90.7%)
	Johnson SU	−959863	0.89 (11.0%)	1.8 (57.4%)	8.1 (87.5%)
	Weibull	−990007	0.95 (5.0%)	2.1 (50.3%)	7.3 (88.7%)
	Lognormal	−953799	1.05 (5.0%)	4.1 (2.9%)	56.9 (12.1%)
	HErD(2)	−911558	1.0 (0.0%)	4.5 (5.9%)	66.3 (2.4%)
	HErD(3)	−911135	1.0 (0.0%)	3.5 (16.9%)	34.4 (46.9%)
	HErD(4)	−874270	1.0 (0.0%)	4.1 (2.9%)	51.2 (20.9%)
	HErD(5)	−847551	1.0 (0.0%)	4.1 (2.9%)	50.8 (21.5%)

Tabelle 1: Likelihood und Momente für die angepassten Verteilungen

Eine weitere Einschränkung der ARTA-Prozesse ergibt sich aus der Nutzung des $AR(p)$ Prozesses. Ein $AR(p)$ eignet sich, um p Autokorrelationskoeffizienten exakt zu modellieren, weitere Koeffizienten werden allerdings nicht mehr modelliert. Da Computernetzwerke üblicherweise eine signifikante Autokorrelation über eine große Anzahl von Koeffizienten aufweisen, führt dies zu sehr großen Modellbeschreibungen. Mit $ARMA(p, q)$ ist es dagegen möglich eine genaue Approximation von deutlich mehr als $p + q$ Koeffizienten zu erzielen. Diese Beobachtungen motivieren die Nutzung von Phasenverteilungen und $ARMA(p, q)$ Prozessen zur Modellierung von Netzwerktraffic, die wir im Folgenden vorstellen.

Dabei dient der $ARMA(p, q)$ der Modellierung der Autokorrelation und die Phasenverteilung mit n Zuständen der Modellierung der empirischen Verteilung. Dieser neuartige Prozesstyp aus [14] wird als $CAPP(n, p, q)$ (*Correlated Acyclic Phase-type Process*) bezeichnet. Zunächst wird kurz vorgestellt, wie APH-Verteilung und ARMA-Prozess in einen stochastischen Prozess kombiniert werden können.

Für die Kombination ist es erforderlich, die APH-Verteilung anders darzustellen als durch die übliche Matrix-Notation. APH-Verteilungen lassen sich als eine Menge von Pfaden (elementary series) darstellen [10], wobei jeder Pfad eine Sequenz

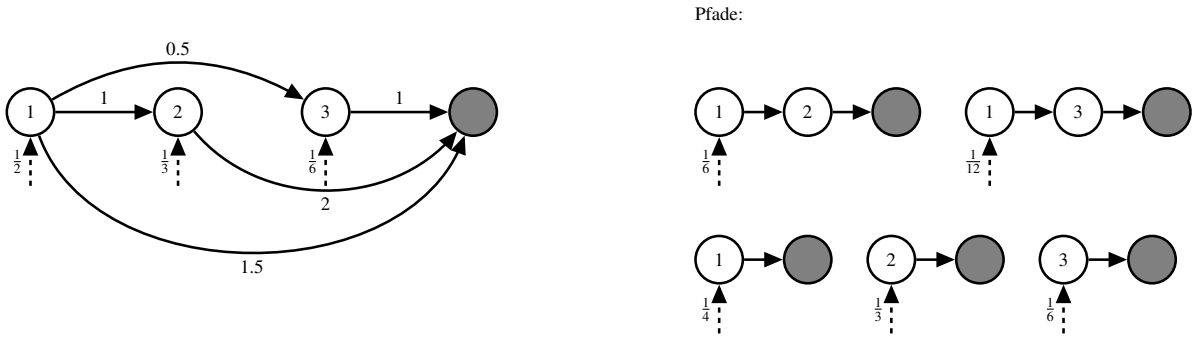


Abbildung 1: APH-Verteilung mit Pfaden

von Zuständen von einem Initialzustand zum absorbierenden Zustand beschreibt. Jeder dieser Pfade beschreibt eine Hypo-Exponential-Verteilung, also eine Folge von Exponentialverteilungen mit unterschiedlichen Raten. Außerdem wird jedem Pfad eine Wahrscheinlichkeit zugeordnet, die sich aus der initialen Wahrscheinlichkeit des ersten Zustands des Pfades und den Transitionsraten auf dem Pfad ergibt. Ein Beispiel ist in Abbildung 1 dargestellt.

Sei τ_i die Wahrscheinlichkeit des i-ten Pfades ($i = 1, ..., m$). Wir definieren

$$\begin{aligned} \underline{b}_1 &= 0 \\ \bar{b}_i &= \underline{b}_i + \tau_i \qquad i = 1, ..., m \\ \underline{b}_i &= \bar{b}_{i-1} \qquad i = 2, ..., m \end{aligned} \qquad \text{und} \qquad \delta(U, i) = \begin{cases} 1, & U \in [\underline{b}_i, \bar{b}_i) \\ 0, & \text{sonst} \end{cases} \tag{3}$$

Sei $\{X_t^{(\Lambda_i)}\}$ eine Sequenz von unabhängig und identisch verteilten Zufallsvariablen mit Hypo-Exponential-Verteilung, beschrieben durch einen Vektor Λ_i mit Länge S_i der die Transitionsraten des i-ten Pfades enthält. Sei $\{Z_t\}$ ein $ARMA(p, q)$ Prozess wie in Gleichung 2 definiert. Wir nehmen an, dass σ_ϵ^2 so gewählt wurde, dass $\{Z_t\}$ Standard-Normalverteilung hat und setzen $U_t = \Phi(Z_t)$, wobei Φ wieder die Standard-Normalverteilung ist. $\{U_t\}$ beschreibt eine Sequenz von gleichverteilten Zufallsvariablen im Intervall $(0, 1)$ [11], die korreliert sind, da die $\{Z_t\}$ ebenfalls Korrelation aufweisen. Die Reihe

$$Y_t = \sum_{i=1}^{m} \delta(U_t, i) X_t^{(\Lambda_i)} \tag{4}$$

beschreibt dann eine Sequenz von korrelierten Zufallsvariablen mit der gewünschten Phasenverteilung. Der ARMA-Prozess dient hierbei der Auswahl eines Pfades gemäß der Wahrscheinlichkeit τ_i. Da die Werte aus dem ARMA-Prozess korreliert sind, weist auch die Sequenz Y_t Korrelation auf.

Um die Prozessbeschreibung nutzen zu können, ist es allerdings notwendig, die Korrelationen $Corr[Y_t, Y_{t+h}]$ und $Corr[Z_t, Z_{t+h}]$ in Beziehung zu setzen. Gesucht ist eine Prozessbeschreibung $\{Z_t\}$ mit Autokorrelation $Corr[Z_t, Z_{t+h}]$, so dass $\{Y_t\}$ die gewünschte Korrelation $Corr[Y_t, Y_{t+h}]$ aufweist, die beispielsweise aus einem Trace ermittelt wurde.

Die Autokorrelation von $\{Y_t\}$ kann ausgedrückt werden durch

$$Corr[Y_t, Y_{t+h}] = \frac{E[Y_t Y_{t+h}] - E[Y]^2}{Var[Y]}. \tag{5}$$

$E[Y]$ und $Var[Y]$ sind bekannt und können mit Hilfe von Gleichung 1 berechnet werden. Für den fehlenden Term $E[Y_t Y_{t+h}]$ ergibt sich nach einigen Umformungen (für die detaillierte Herleitung sei auf [14] verwiesen):

$$E[Y_t Y_{t+h}] = \sum_{i,j} \left(\left(\sum_{s=1}^{S_i} \frac{1}{\Lambda_i(s)} \right) \left(\sum_{s=1}^{S_j} \frac{1}{\Lambda_j(s)} \right) \int_{\Phi^{-1}(\underline{b}_j)}^{\Phi^{-1}(\bar{b}_j)} \int_{\Phi^{-1}(\underline{b}_i)}^{\Phi^{-1}(\bar{b}_i)} \varphi_{\rho_h}(z_t, z_{t+h}) dz_t dz_{t+h} \right) \tag{6}$$

wobei $\varphi_{\rho_h}(z_t, z_{t+h})$ die Dichtefunktion der bivariaten Standard-Normalverteilung mit Korrelation $\rho_h = Corr[Z_t, Z_{t+h}]$ ist. Der Term $E[Y_t Y_{t+h}]$ ergibt sich also als Summe über die Kombinationen der möglichen Pfade. Jeder Summand ist dabei das Produkt aus dem Mittelwert des ersten Pfads, dem Mittelwert des zweiten Pfads und dem Integral über die Dichtefunktion der bivariaten Standard-Normalverteilung. Für die Berechnung des Integrals existieren schnelle numerische Verfahren [12].

Die Parameteranpassung von CAPPs kann in zwei Schritten erfolgen, d.h. Phasenverteilung und ARMA-Prozess können getrennt voneinander angepasst werden, wodurch sich das Optimierungsproblem in zwei leichter zu lösende Probleme aufteilt. Für die Anpassung der APH-Verteilung kann auf ein beliebiges existierendes Verfahren zurückgegriffen werden (siehe z.B. Abschnitt 2.1). Ein wesentlicher Schritt bei der Konstruktion von CAPPs ist die Erzeugung des $ARMA(p, q)$ Prozesses, der zwei Anforderungen erfüllen muss. Zunächst muss die Autokorrelation so bestimmt sein, dass der CAPP die gewünschte Autokorrelation, die beispielsweise aus einem Trace ermittelt wurde, aufweist. Außerdem wurde verlangt, dass der Prozess standard-normalverteilt ist ($Z_t \sim N(0, 1)$). Die Parameteranpassung des ARMA-Prozesses gliedert sich also in drei Phasen: Bestimmung der Autokorrelation anhand der gewünschten Autokorrelation des CAPPs, Bestimmung der Parameter und Anpassung des Prozesses, so dass er standard-normalverteilt ist.

Die Autokorrelationsstruktur kann mit Hilfe von Gleichung 6 numerisch mit beliebiger Genauigkeit bestimmt werden. Sei $\hat{\rho_h}$ die lag-h Autokorrelation die der CAPP letztendlich aufweisen soll. Gesucht ist dann die ARMA-Autokorrelation ρ_h, so dass der CAPP gemäß Gleichung 6 Autokorrelation $\hat{\rho_h}$ hat. Die Bestimmung kann numerisch mit einem einfachen Suchverfahren [22] erfolgen. Nachdem die Autokorrelationskoeffizienten $\rho = (\rho_1, \rho_2, ..., \rho_K)$ für den ARMA-Prozess bestimmt worden sind, müssen Parameter gefunden werden, so dass der Prozess tatsächlich die gewünschte Autokorrelation aufweist. Dazu ist das Optimierungsproblem

$$min \sum_{k=1}^{K} \left(\frac{\rho_k^*}{\rho_k} - 1 \right)^2 \tag{7}$$

zu lösen, wobei ρ_k die gewünschte Korrelation und ρ_k^* die Korrelation des $ARMA(p, q)$-Modells ist, das während der Minimierung erzeugt wird.

Im letzten Schritt muss die Varianz σ_ϵ^2 der Rauschterme angepasst werden, so dass $Z_t \sim N(0, 1)$ Da für einen ARMA-Prozess Kovarianz und Varianz beide von σ_ϵ^2 abhängen [5], kann σ_ϵ^2 wie gewünscht gesetzt werden, ohne dabei die Autokorrelation zu ändern.

Für Gleichung 6 lassen sich einige interessante Eigenschaften nachweisen, die für die Durchführung des beschriebenen Verfahrens notwendig sind. Diese Eigenschaften sollen hier nur kurz erwähnt werden, für Details sei auf [14] verwiesen. Zum Einen ist die Autokorrelation des CAPPs eine monoton steigende Funktion bezüglich der ARMA-Korrelation ρ, was für das oben beschriebene Suchverfahren nötig ist. Zum Anderen ist die Funktion stetig, so dass tatsächlich alle möglichen Korrelationswerte erreicht werden können. Für die Extremfälle $\rho \pm 1$ ist es außerdem möglich Gleichung 6 ohne numerische

Berechnung des Integrals zu lösen. Dies ermöglicht eine einfache Berechnung der minimal und maximal möglichen Korrelation, die ein CAPP mit bestimmter Phasenverteilung aufweisen kann. Da die Darstellung von Phasenverteilungen nicht eindeutig ist hilft dies außerdem bei der Bestimmung von Transformationen in eine andere Darstellung derselben Verteilung, die eine höhere maximale Korrelation ermöglicht.

Neben CAPPs wurden in [14] noch CHEPs (Correlated Hyper-Erlang Processes) definiert. CHEPs sind ein Spezialfall von CAPPs, für die als PH-Verteilung eine Hyper-Erlang-Verteilung genutzt wird. Durch diesen Verteilungstyp vereinfachen sich einige der Notationen, das Vorgehen und die Ideen sind aber im Prinzip identisch.

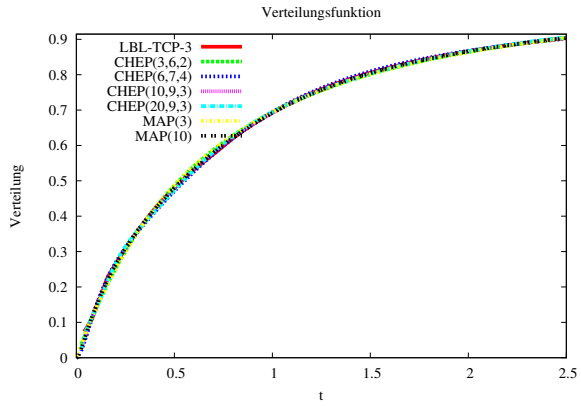
4 Tool-Unterstützung

Um die eingangs erwähnte mangelhafte Unterstützung von stochastischen Prozessen in Simulationswerkzeugen zu mildern, wurden mehrere Tools entwickelt, die den Einsatz von stochastischen Prozessen erleichtern sollen. Die Implementierung des oben beschriebenen Vorgehens zur Parameteranpassung von CAPPs wurde in das Toolset ProFiDo [2] integriert. ProFiDo ist ein Framework zur Parameteranpassung von Verteilungen und Prozessen, das bereits Verfahren für Phasenverteilungen enthält und somit eine einfache Einbindung der Anpassung von CAPPs ermöglicht. Für das Simulationstool OMNeT++ [13], das hauptsächlich zur Simulation von Rechnernetzen eingesetzt wird, für die seit langem bekannt ist, dass z.B. Zwischenankunftszeiten von Paketen Korrelationen und Abhängigkeiten aufweisen, wurde ein Modul zur Erzeugung von Zufallszahlen aus stochastischen Prozessen entwickelt [15], das neben CAPPs auch MAPs, ARMA- und ARTA-Prozesse unterstützt.

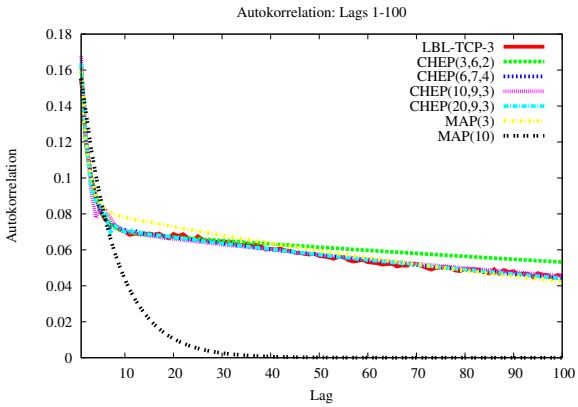
5 Experimente

Um die Qualität unseres Verfahrens zur Parameteranpassung einschätzen zu können wurden Experimente mit künstlich erzeugten und realen Traces durchgeführt. Aus Platzgründen sollen hier nur einige repräsentative Ergebnisse gezeigt werden. Eine umfangreiche experimentelle Bewertung findet sich in [14]. Für die Bewertung wurden die Parameter von CHEPs/CAPPs an Tracedaten angepasst. Zum Vergleich wurden außerdem z.B. MAPs angepasst, da diese ebenfalls PH-Verteilungen als Grundlage nutzen, die Korrelation aber anders modelliert wird. Zur Einschätzung der Qualität zeigen wir Plots der Autokorrelationskoeffizienten und der Verteilungsfunktion. Zusätzlich wurden die Prozesse zur Erzeugung von Ankünften im einem simplen Warte/Bediensystem mit Queue-Länge 10 genutzt. Hier dient die Verteilung der Warteschlangenlänge als Vergleichsmaß.

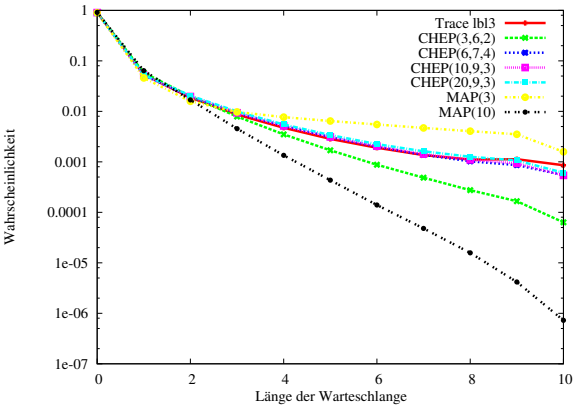
Abbildung 2 zeigt die Ergebnisse für den Trace *LBL-TCP-3* [21]. Aus der Abbildung lässt sich erkennen, dass die Anpassung der Verteilung für alle Modelle adäquat möglich war. Bei der Anpassung der Autokorrelation zeigt sich allerdings, das sich die Parameteranpassung für MAPs insbesondere bei einer größeren Anzahl von Zuständen schwierig gestaltet. Für CHEPs/CAPPs war hingegen in allen Fällen eine gute Anpassung möglich. Diese Erkenntnisse spiegeln sich auch im Queueing-Verhalten wider. Insbesondere für größere CHEPs/CAPPs ist die Verteilung der Queue-Länge nah an dem Verhalten für den realen Trace, während mit MAPs nur eine schlechtere Approximation möglich war.



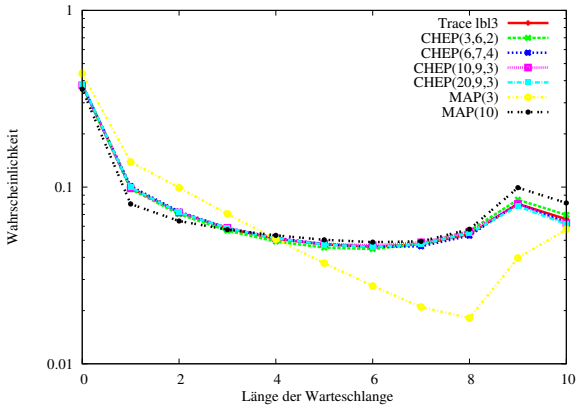
a) Verteilungsfunktion



b) Autokorrelation



c) Queue-Länge (Auslastung 0.3)



d) Queue-Länge (Auslastung 0.9)

Abbildung 2: Ergebnisse für den Trace *LBL-TCP-3*

6 Fazit

In diesem Artikel wurde ein Ansatz zur Modellierung von korrelierten Verkehrsdaten vorgestellt, der ARMA-Prozesse zur Modellierung der Korrelation und Phasenverteilungen zur Modellierung der empirischen Verteilung in einem neuen Prozesstyp kombiniert. Zur Parameteranpassung wurde ein Algorithmus skizziert, der in das bestehende Toolset ProFiDo integriert wurde. Anhand von Experimenten mit realen Daten wurde gezeigt, dass die Prozesse in der Lage sind sowohl Verteilung als auch Korrelation geeignet abzubilden.

Literatur

[1] S. Asmussen and C. A. O'Cinneide. Matrix-Exponential Distributions - Distributions with a Rational Laplace Transform. In *Encyclopedia of Statistical Sciences*, pages 435–440, New York, 1997. John Wiley & Sons.

[2] F. Bause, P. Buchholz, and J. Kriege. ProFiDo - The Processes Fitting Toolkit Dortmund. In *Proceedings of QEST 2010*, pages 87–96. IEEE Computer Society, 2010.

[3] B. Biller and B.L. Nelson. Fitting Time-Series Input Processes for Simulation. *Operations Research*, 53(3):549–559, 2005.

[4] G. E. P. Box and G. M. Jenkins. *Time Series Analysis - Forecasting and Control*. Holden-Day, 1970.

[5] P.J. Brockwell and R.A. Davis. *Time Series: Theory and Methods*. Springer, 1998.

[6] P. Buchholz and J. Kriege. A Heuristic Approach for Fitting MAPs to Moments and Joint Moments. In *Proceedings of QEST 2009*, pages 53–62, 2009.

[7] M.C. Cario and B.L. Nelson. Autoregressive To Anything: Time-Series Input Processes for Simulation. *Operations Research Letters*, 19(2):51–58, 1996.

[8] G. Casale, E.Z. Zhang, and E. Smirni. Trace Data Characterization and Fitting for Markov Modeling. *Performance Evaluation*, 67(2):61–79, 2010.

[9] M.E. Crovella and A. Bestavros. Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes. In *Proceedings of SIGMETRICS '96*. ACM, 1996.

[10] A. Cumani. On the Canonical Representation of Homogeneous Markov Processes Modeling Failure-Time Distributions. *Micorelectronics and Reliability*, 22(3), 1982.

[11] L. Devroye. *Non-Uniform Random Variate Generation*. Springer, New York, 1986.

[12] Z. Drezner and G. O. Wesolowsky. On the Computation of the Bivariate Normal Integral. *Journal of Statistical Computation and Simulation*, 35:101 – 107, 1990.

[13] R. Hornig and A. Varga. An Overview of the OMNeT++ Simulation Environment. In *Proceedings of SIMUTools*, 2008.

[14] J. Kriege. *Fitting Simulation Input Models for Correlated Traffic Data*. PhD thesis, Technische Universität Dortmund, Fakultät für Informatik, 2012.

[15] J. Kriege and P. Buchholz. Simulating Stochastic Processes with OMNeT++. In *Proceedings of the 4th International OMNeT++ Workshop*, 2011.

[16] A. M. Law and W. D. Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, Boston, 3rd edition, 2000. ISBN 0-07-059292-6.

[17] A.M. Law and M.G. McComas. How The Expertfit Distribution-Fitting Software Can Make Your Simulation Models More Valid. In *Proceedings of the Winter Simulation Conference 2003*, pages 169–174, 2003.

[18] W. Leland, M. Taqqu, W. Willinger, and D. Wilson. On the Self-Similar Nature of Ethernet Traffic. *IEEE/ACM Transactions on Networking*, 2(1):1–15, 1994.

[19] M. Livny, B. Melamed, and A. K. Tsiolis. The Impact of Autocorrelation on Queueing Systems. *Management Science*, 39:322–339, 1993.

[20] M. F. Neuts. *Matrix-Geometric Solutions in Stochastic Models*. Johns Hopkins University Press, 1981.

[21] V. Paxson and S. Floyd. Wide-Area Traffic: The Failure of Poisson Modeling. *IEEE/ACM Transactions on Networking*, 3:226–244, 1995.

[22] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C - The Art of Scientific Computing*. Cambridge University Press, 2nd edition, 1993.

[23] W.J. Stewart. *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, 1994.

[24] A. Thümmler, P. Buchholz, and M. Telek. A Novel Approach for Phase-Type Fitting with the EM Algorithm. *IEEE Transactions on Dependable and Secure Computing*, 3(3):245–258, 2006.

[25] R.E. Wheeler. Quantile Estimators of Johnson Curve Parameters. *Biometrika*, 67(3), 1980.

Jan Kriege studierte von 2001-2006 Informatik an der TU Dortmund und ist seit 2006 wissenschaftlicher Mitarbeiter in der Arbeitsgruppe „Modellierung und Simulation“. Nach seiner Diplomarbeit über logistische Netze beschäftigte er sich während der Promotion, die er 2012 abschloss, mir der Modellierung von Computer- und Kommunikationsnetzen. Seit 2012 ist Jan Kriege Lehrbeauftragter an der Universität Duisburg-Essen.