Fitting Markovian Arrival Processes by Incorporating Correlation into Phase Type Renewal Processes

Falko Bause
Computer Science IV
TU Dortmund

Gábor Horváth
Department of Telecommunications
Budapest University of Technology and Economics

Outline

1. Introduction
2. Structured Markovian Arrival Processes (SMAPs)
3. Fitting Lag-1 joint moments $E[X_0^i X_1^j]$
4. Fitting Higher Lag Joint Moments $E[X_0^i X_\ell^j], \ell \geq 1$
5. Applications
6. Conclusions

Introduction

Advantages of modeling inter-arrival/service times by
Phase-type (PH) distributions and Markovian Arrival Processes (MAPs)

- Markov property
- efficient numerical methods for QN models (MAP/PH/1, ...)
- easy to simulate

Fitting methods:

- EM based methods (use trace directly; applicable to "small" traces)
- matching based methods (fit statistical figures of a trace;
  applicable to larger traces)

Experience from distribution fitting:
Methods operating on special sub-classes of PH distributions are more effective (Acyclic PH, Hyper-Erlang,...)

Introduction (cont'd)

Recent matching methods for MAPs follow a "two-phase" approach:

1. fit distribution (giving a PH distribution)
2. fit correlation

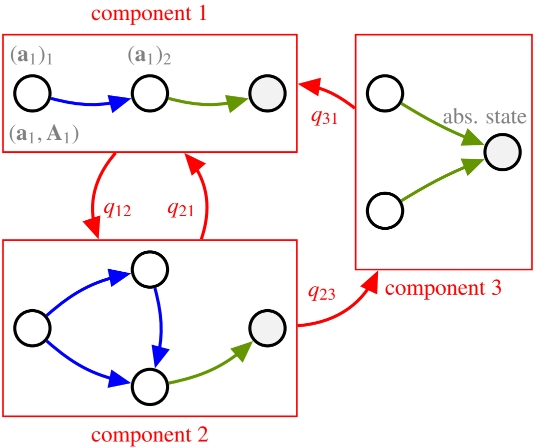
Which figures to fit?

Telek, Horváth (2007):
A (non-redundant) MAP with n states is characterized by n^2 moments
($2n-1$ marginal moments, $(n-1)^2$ lag-1 joint moments $E[X_0^i X_1^j]$)

This paper

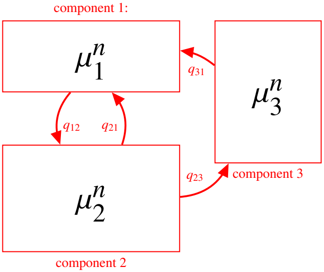
- Assumption: PH distribution representing inter-arrival times is given
- Fitting is based on joint moments $E[X_0^i X_\ell^j]$
- Sub-class of MAPs (Structured MAPs (SMAPs)) is considered

Structured Markovian Arrival Process (SMAP)



- set of component (PH) distributions $\{(\mathbf{a}_i, \mathbf{A}_i) \mid i = 1, \ldots, N\}$
- discrete time Markov chain, specifying which component generates next inter-arrival time; irreducible switching matrix $\mathbf{Q} = (q_{ij})$

Structured Markovian Arrival Process (cont'd)



$(\{(\mathbf{a}_i, \mathbf{A}_i) \mid i = 1, \ldots, N\}, \mathbf{Q})$:

$$E[X^n] = \sum_{i=1}^{N} \pi_i \mu_i^n \quad (1)$$

$$\eta^{(1)}_{n_1 n_2} = E[X_0^{n_1} X_1^{n_2}] = \sum_{i=1}^{N} \sum_{j=1}^{N} \pi_i \, q_{ij} \, \mu_i^{n_1} \mu_j^{n_2} \quad (2)$$

$\mu_i^n := n! \, \mathbf{a}_i (-\mathbf{A}_i)^{-n} \mathbf{1}^T$	n-th moment of distribution of component i
$\pi \mathbf{Q} = \pi, \pi \mathbf{1}^T = 1$	steady-state distribution of "switching process"

- Sizes of Eqs. (1) and (2) only depend on N

How to get Component Distributions $\{(\mathbf{a}_i, \mathbf{A}_i) \mid i = 1, \ldots, N\}$?

Assume PH distribution $(\boldsymbol{\pi}, \mathbf{H})$ for inter-arrival times is given, e.g. from moment matching or EM-based fitting methods (we used: G-FIT).

Define

$$\begin{aligned} N &:= \#\text{ non-zero elements in } \boldsymbol{\pi} \\ \mathbf{A}_i &:= \mathbf{H} \\ \mathbf{a}_i &:= \begin{cases} \mathbf{e}_i & \text{if } \pi_i > 0 \\ \mathbf{0} & \text{otherwise} \end{cases} \end{aligned}$$

<u>Constraint for switching matrix $\mathbf{Q}$:</u> $\boldsymbol{\pi}\mathbf{Q} = \boldsymbol{\pi}$ since

$$\sum_{i=1}^{N} \pi_i \mathbf{a}_i e^{\mathbf{A}_i t} \mathbf{1}^T = \sum_{i=1}^{N} \pi_i \mathbf{e}_i e^{\mathbf{H} t} \mathbf{1}^T = \boldsymbol{\pi} e^{\mathbf{H} t} \mathbf{1}^T \quad \text{for } t \geq 0$$

i.e. distribution $(\boldsymbol{\pi}, \mathbf{H})$ remains unchanged.

How to get Component Distributions? (cont'd)

Furthermore one can apply similarity transformations for more suitable representation of PH-dist $(\boldsymbol{\pi}, \mathbf{H})$

$$\boldsymbol{\pi}' = \boldsymbol{\pi}\mathbf{B}, \mathbf{H}' = \mathbf{B}^{-1}\mathbf{H}\mathbf{B}$$

Elementary transformation matrix

$$\mathbf{B}_{i,j}(b) = \begin{array}{c} \\ \\ \\ i \\ \\ \\ \end{array} \begin{bmatrix} 1 & 0 & 0 & 0 & \dots \\ 0 & 1 & 0 & 0 & \dots \\ \vdots & \vdots & \ddots & \ddots & \dots \\ 0 & b & 0 & 1-b & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$
$$\qquad\qquad\qquad j$$

Fitting Lag-1 joint moments

System of lin. Eqs. for $\mathbf{Q}$: (with constraints $\boldsymbol{\pi}\mathbf{Q} = \boldsymbol{\pi}$, $\mathbf{Q}\mathbf{1}^T = \mathbf{1}$, $\mathbf{Q} \geq \mathbf{0}$)

$$\eta^{(1)}_{n_1 n_2} = E[X_0^{n_1} X_1^{n_2}] = \sum_{i=1}^{N} \sum_{j=1}^{N} \boldsymbol{\pi}_i \, q_{ij} \, \mu_i^{n_1} \mu_j^{n_2}, \quad n_1, n_2 = 1, \ldots, K$$

should be close to given lag-1 joint moments $\zeta^{(1)}_{n_1 n_2}$ from trace

Fitting Lag-1 joint moments

System of lin. Eqs. for $\mathbf{Q}$: (with constraints $\boldsymbol{\pi}\mathbf{Q} = \boldsymbol{\pi}$, $\mathbf{Q}\mathbf{1}^T = \mathbf{1}$, $\mathbf{Q} \geq \mathbf{0}$)

$$\eta^{(1)}_{n_1 n_2} = E[X_0^{n_1} X_1^{n_2}] = \sum_{i=1}^{N} \sum_{j=1}^{N} \boldsymbol{\pi}_i \, q_{ij} \, \mu_i^{n_1} \mu_j^{n_2}, \quad n_1, n_2 = 1, \ldots, K$$

should be close to given lag-1 joint moments $\zeta^{(1)}_{n_1 n_2}$ from trace $\Longrightarrow$

NNLS (non-negative least-squares problem)

Given matrices $\mathbf{A}, \mathbf{B}$ and vectors $\mathbf{a}, \mathbf{b}$.
Determine solution $\mathbf{x}$ minimizing $||\mathbf{A}\mathbf{x} - \mathbf{a}||_2^2$ subject to $\mathbf{B}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0$

- Well-known problem (e.g. *Lawson/Hanson: Solving Least Squares Problems, Prentice-Hall, 1974*)
- Efficient algorithms are reported in the literature (can handle several hundreds/thousands of variables/constraints)
- Implementations are available, e.g. R (limSolve), MATLAB (lsqlin, which we used).

Fitting Lag-1 joint moments (cont'd)

Problem: Higher order joint moments dominate optimization.

Possible Solutions:

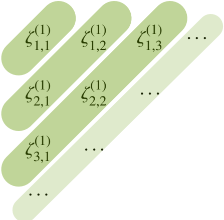
- weighting of joint moments, but no general rule which weight functions are appropriate

Here:

- relative errors; NNLS: $\sum_{n_1,n_2}\left(\frac{\sum_{i,j}\pi_i\mu_i^{n_1}\mu_j^{n_2}q_{ij}-\zeta^{(1)}_{n_1,n_2}}{\zeta^{(1)}_{n_1,n_2}}\right)^2$
- step-by-step

  solve NNLS for subset of
  $S := \{\zeta^{(1)}_{n_1 n_2} \mid n_1, n_2 = 1, \ldots, K\}$
  encode resultant solutions $\eta^{(1)}_{n_1 n_2}$ as linear constraints for next NNLS problem

  (*"one-step, if subset=S"*)



Fitting Higher Lag Joint Moments

Idea: incorporate memory into switching process, i.e. determine
$q^{(k)}_{(i_k|i_0\ldots i_{k-1})} := P[\text{component } i_k \mid \text{components } i_0 \ldots i_{k-1}]$ which

- approximates lag-k joint moments and
- keeps fitting results for lag-ℓ joint moments, $\ell < k$

Fitting Higher Lag Joint Moments

Idea: incorporate memory into switching process, i.e. determine

$q^{(k)}_{(i_k|i_0\ldots i_{k-1})} := P[\text{component } i_k \mid \text{components } i_0 \ldots i_{k-1}]$ which

- approximates lag-k joint moments and
- keeps fitting results for lag-ℓ joint moments, $\ell < k$

$$r^{(k)}_{i_0\ldots i_{k-1}} := P[\text{comps } i_0 \ldots i_{k-1}] = \pi_{i_0} q^{(1)}_{(i_1|i_0)} q^{(2)}_{(i_2|i_0 i_1)} \cdots q^{(k-1)}_{(i_{k-1}|i_0\ldots i_{k-2})}$$

$$E[X_0^{n_1} X_k^{n_2}] = \sum_{i_0,\ldots,i_k=1}^{N} \mu_{i_0}^{n_1} \mu_{i_k}^{n_2} r^{(k)}_{i_0\ldots i_{k-1}} q^{(k)}_{(i_k|i_0\ldots i_{k-1})}$$

Fitting Higher Lag Joint Moments

Idea: incorporate memory into switching process, i.e. determine $q^{(k)}_{(i_k|i_0\ldots i_{k-1})} := P[\text{component } i_k \mid \text{components } i_0 \ldots i_{k-1}]$ which

- approximates lag-k joint moments and
- keeps fitting results for lag-ℓ joint moments, $\ell < k$

$$r^{(k)}_{i_0\ldots i_{k-1}} := P[\text{comps } i_0 \ldots i_{k-1}] = \pi_{i_0} q^{(1)}_{(i_1|i_0)} q^{(2)}_{(i_2|i_0 i_1)} \cdots q^{(k-1)}_{(i_{k-1}|i_0\ldots i_{k-2})}$$

$$E[X_0^{n_1} X_k^{n_2}] = \sum_{i_0,\ldots,i_k=1}^{N} \mu_{i_0}^{n_1} \mu_{i_k}^{n_2} r^{(k)}_{i_0\ldots i_{k-1}} q^{(k)}_{(i_k|i_0\ldots i_{k-1})}$$

Constraints: $r^{(k)}_{i_1\ldots i_k} = \sum_{i_0=1}^{N} r^{(k)}_{i_0\ldots i_{k-1}} q^{(k)}_{(i_k|i_0\ldots i_{k-1})}$ $\qquad \sum_{i_k=1}^{N} q^{(k)}_{(i_k|i_0\ldots i_{k-1})} = 1$

relative error; NNLS: $\sum_{n1,n2} \left(\frac{\sum_{i_0,\ldots,i_k=1}^{N} \mu_{i_0}^{n_1} \mu_{i_k}^{n_2} r^{(k)}_{i_0\ldots i_{k-1}} q^{(k)}_{(i_k|i_0\ldots i_{k-1})} - \zeta^{(k)}_{n_1,n_2}}{\zeta^{(k)}_{n_1,n_2}} \right)^2$

Fitting Higher Lag Joint Moments (cont'd)

From $q^{(k)}_{(i_k|i_0 \dots i_{k-1})}$ an SMAP $((\{(\mathbf{a}_i, \mathbf{A}_i) \mid i = 1, \dots, N^k\}, \mathbf{Q}))$

can be obtained by encoding the memory of the switching process into the components

$$q_{a,b} = \begin{cases} q^{(k)}_{(i_k|i_0 \dots i_{k-1})} & \text{if } a = (i_0, \dots, i_{k-1}) \text{ and } b = (i_1, \dots, i_k) \\ 0 & \text{otherwise} \end{cases}$$

PH distribution of component $(i_1, \dots, i_k)$ is given by $(\mathbf{a}_{i_k}, \mathbf{A}_{i_k})$.

Fitting of SMAPs up to lag *k* *("basic idea")*

taking similarity transformations into account
(experience: improves fitting results)

Given PH distribution $(\boldsymbol{\pi}, \mathbf{H})$
for several similarity transformations **do**
- generate component distributions
- fit up to lag k *(either step-by-step or in one-step)*

end for
return best result $(\boldsymbol{\pi}_{best}, \mathbf{H}_{best}, \mathbf{Q}_{best})$

Application Examples

1. Fitting of a given MAP:

$$\mathbf{D}_0=\begin{bmatrix} -3.721 & 0.5 & 0.02 \\ 0.1 & -1.206 & 0.005 \\ 0.001 & 0.002 & -0.031 \end{bmatrix},\ \mathbf{D}_1=\begin{bmatrix} 0.2 & 3.0 & 0.001 \\ 1.0 & 0.1 & 0.001 \\ 0.005 & 0.003 & 0.02 \end{bmatrix},\ \boldsymbol{\pi} = \boldsymbol{\pi}\left(-\mathbf{D}_0\right)^{-1}\mathbf{D}_1,\ \boldsymbol{\pi}\mathbf{1}^T = 1$$

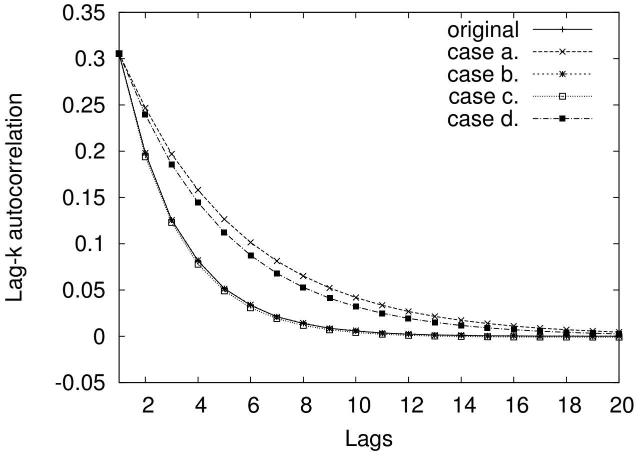
	Method	Subject of fitting	# states of MAP
case a.	step-by-step	$\eta_{i,j}^{(k)},\quad i,j=\{1,2\}, k=1$	9
case b.	step-by-step	$\eta_{i,j}^{(k)},\quad i,j=\{1,2\}, k=\{1,2\}$	27
case c.	step-by-step	$\eta_{i,j}^{(k)},\quad i,j=1, k=\{1,2,3\}$	81
case d.	one-step	$\eta_{i,j}^{(k)},\quad i,j=\{1,2\}, k=1$	9

a and b consider higher order joint moments

c considers only order 1 joint moments, but higher lags

d =a, but one-step

Results of Fitting a given MAP



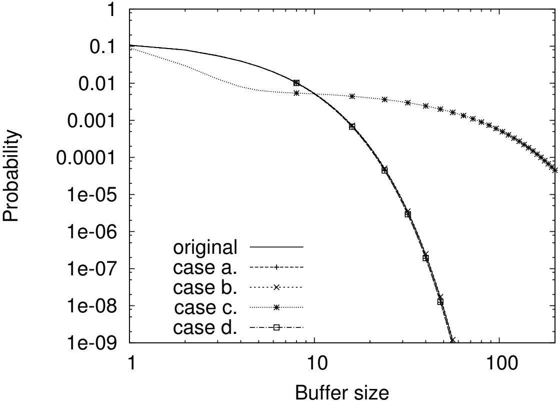
lag-k autocorrelation:

$$\rho_k = \frac{(\eta_{11}^{(k)} - E[X]^2)}{(E[X^2] - E[X]^2)}$$

- cases b,c give better fitting than a,d (since they consider higher lags)
- no difference between step-by-step and one-step

Results of Fitting a given MAP (cont'd)

MAP/M/1 queue length distribution (low load $\rho = 0.38$)



- cases a,b,d (considering higher order lag-1 joint moments) give better results than case c
- similar results for higher loads

Conforms to theory: "lag-1 joint moments determine MAP"

Application Examples (cont'd)

2. Fitting of LBL-TCP-3 trace

Distribution fitted with G-FIT giving a Hyper-Erlang distribution
(4 branches, each with two-phase Erlang distribution)

	Method	Subject of fitting	Time (sec.)	# states of MAP
case a.	step-by-step	$\eta_{i,j}^{(k)}, \quad i,j = \{1,2,3\}, k = 1$	≈ 2	16
case b.	step-by-step	$\eta_{i,j}^{(k)}, \quad i,j = \{1,2\}, k = \{1,2\}$	≈ 5	44
case c.	step-by-step	$\eta_{i,j}^{(k)}, \quad i,j = 1, k = \{1,2,3\}$	≈ 20	176
case d.	one-step	$\eta_{i,j}^{(k)}, \quad i,j = \{1,2,3\}, k = 1$	≈ 2	11
case e.	one-step	$\eta_{i,j}^{(k)}, \quad i,j = \{1,2\}, k = \{1,2\}$	≈ 5	32

(Intel Quad Core, 2.8GHz)

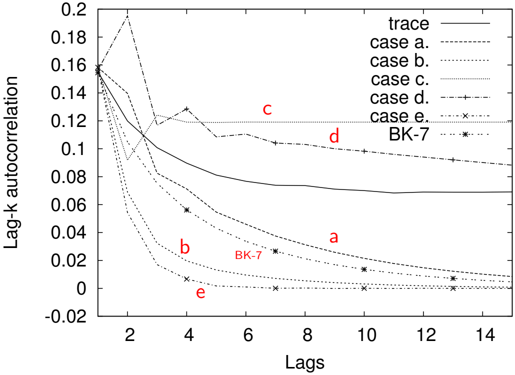
a,b,c decreasing order of joint moments, increasing lags
d=a, e=b but one-step

For comparison best MAP of

P. Buchholz, J. Kriege. *A Heuristic Approach for Fitting MAPs to Moments and Joint Moments.* QEST 2009.

was used: "BK-7" with 7 states.

Results of Fitting LBL Trace

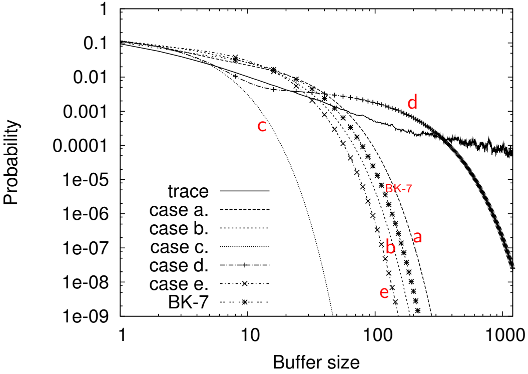


a,b,c
(decr. order of joint moments, incr. lags)
d=a, e=b (but one-step)

- fitting based on lag-1 joint moments gave better results

Results of Fitting LBL Trace (cont'd)

MAP/M/1 queue length distribution (high load $\rho = 0.8$)



a,b,c
(decr. order of joint moments, incr. lags)
d=a, e=b (but one-step)

- higher order lag-1 joint moments seem to be "more important" than higher lags

Conclusions

- Presented method allows incorporation of correlation into PH-type distributions
  - utilizing similarity transformations
- Special structure of SMAPs makes it possible to formulate joint moment fitting as an NNLS problem
  - also considering higher lags
  - using relative errors
  - with fitting step-by-step or in one-step
    (LBL: different results concerning rel. errors of joint moments, but showed no relevant improvement concerning acf and QN results)
  - complexity depends on the number of components
- resultant MAPs might get large; no problem in our experiments (at least usable for simulation)

Future work:

Systematically increase number of components for better fitting