A bounding histogram approach for network performance analysis

J.M. Fourneau

Laboratoire PRiSM, CNRS UMR 8144

Université de Versailles St-Quentin

Joint Work with F. Aït Salaht (PRiSM, Univ. Versailles), H. Castel Taleb (SAMOVAR, Telecom Sud Paris) and N. Pekergin (LACL, Univ. Creteil)

Motivation and Outline

- Analyze the performance of a network under general traffics derived from real traces
- Markov chains with huge state spaces
- Computation of the steady-state distribution is very difficult and often impossible
- Apply the stochastic bounding method for network performance analysis under histogram-based traffic

- Histogram-based approach because of measurements
- Supposed to be more precise than typical assumptions about the arrivals and services processes.
- Stochastic bound theory to reduce the size of the distribution
- Stochastic bound : a bound of the exact distribution
- It implies Bounds on performance measures which are non decreasing rewards
- Better than a previous method (HBSP defined by Hernandez-Orallo and his colleagues) which only provides approximation.
- Control of the size of the distribution
- Control of the complexity and Trade-off between accuracy and complexity

Traffic trace, Example

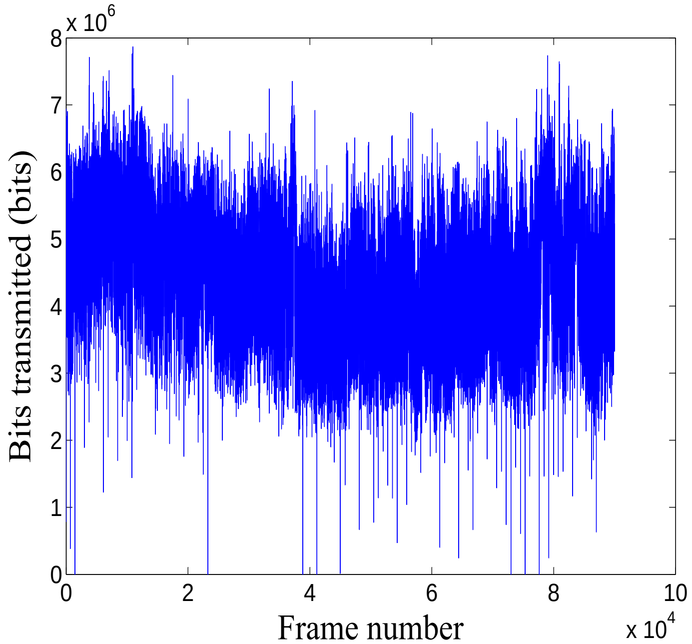


Figure 1: MAWI traffic trace corresponds to a 1-hour IP traffic, 9th of January 2007 between noon and 1PM

First Step

- Deriving a discrete distribution from the trace
- Main Assumption : Stationarity of the process
- Sampling Period (Here $T = 40\,ms$, to be consistent with previous works by Hernandez-Orallo)
- Future Work : Markov Modulated Arrivals

Illustration on MAWI

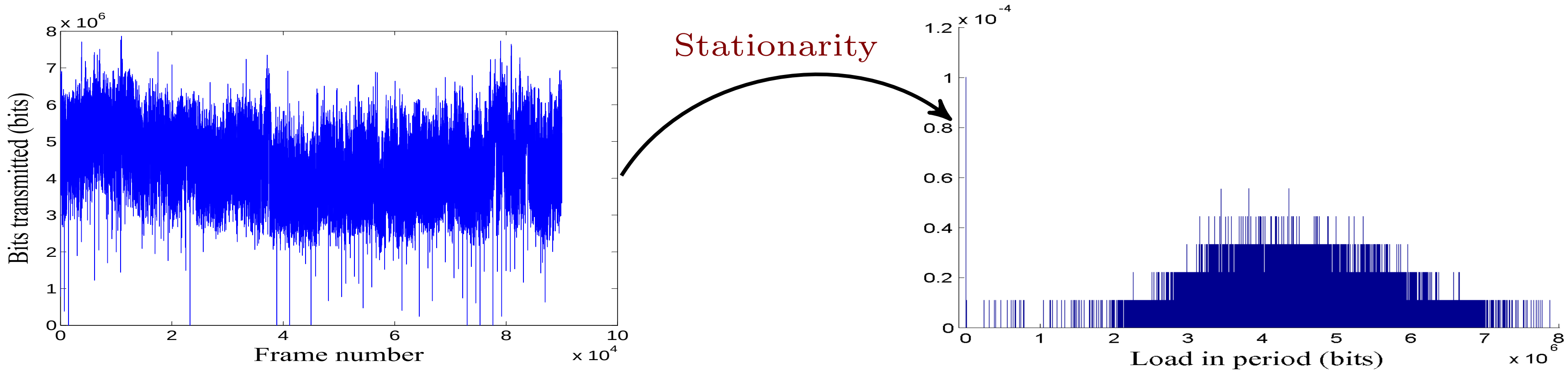


Figure 2: MAWI traffic trace (left), Histogram representation (right) The number of bins is 80511

Complexity Issues

- The size of the distribution of the arrival process (here, 80511) has a direct influence on the size of the Markov chain modeling the queue.
- Discrete Time Queues with iid batch arrivals (derived from the traces) and batch services with iid distribution
- Slot time : the sampling period. Thus we may have several services (i.e. the sampling period is not equal to a service time)
- In [EPEW2013], we have considered a model where the service capacity is constant.
- Here we generalized to batch services with iid distribution as a step to represent classes of packets with priority.
- Networks are analyzed by decomposition assuming independence of the queues (approximation)

Model of a Discrete Time Queue with finite buffer

- Arrival First
- Population at time n in the queue:

$$X_{n+1} = min(B, (X_n + A - S)^+) \quad (1)$$

- where A is the size of the batch of arrival,
- S is the size of the batch of services
- and B is the buffer size
- Independence implies Markov Chain.

Principe

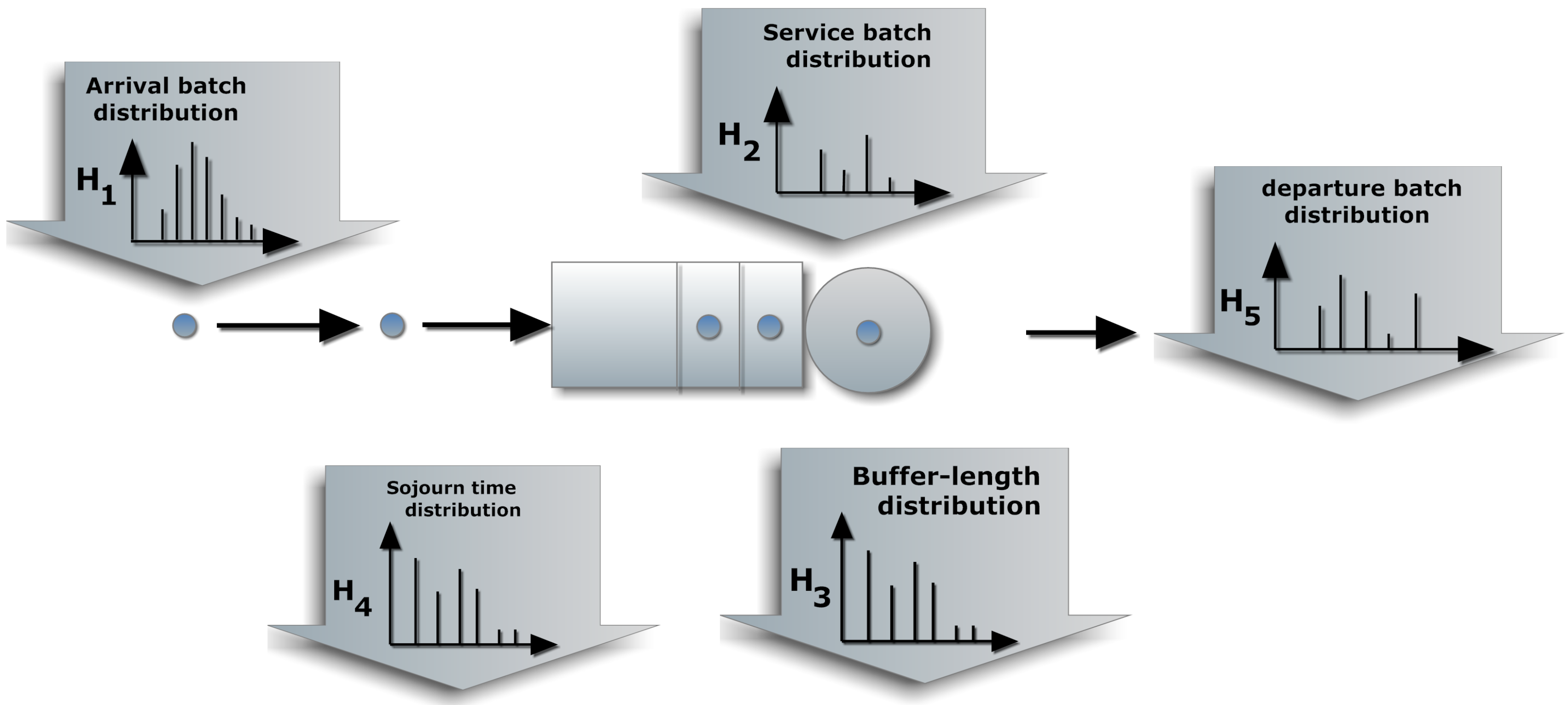


Figure 3: Analysis of a Queue, H_1 and H_2 are known, H_3, H_4 and H_5 are numerically computed.

Stochastic Bound and Complexity Issues

- Stochastically monotone. Intuition: If we consider the same distribution for the services and we stochastically increase the arrivals, we stochastically increase the distributions H_3, H_4 and H_5.
- Based on the stochastic ordering $\leq_{st}$ of distributions.
- Key Idea: replace the distribution of arrivals with N bins by another one with less bins (say $K << N$) and with is stochastically larger or lower.
- Two Methods to find such a : a linear algorithm proposed by Tancrez and Semal and the algorithm we have presented in [WODES12] which provide the most accurate distribution according to a non negative reward (with a larger complexity, based on dynamic programming).
- HBSP: builds an approximation of the distribution rather than a bound.

A Brief Introduction to Stochastic Ordering

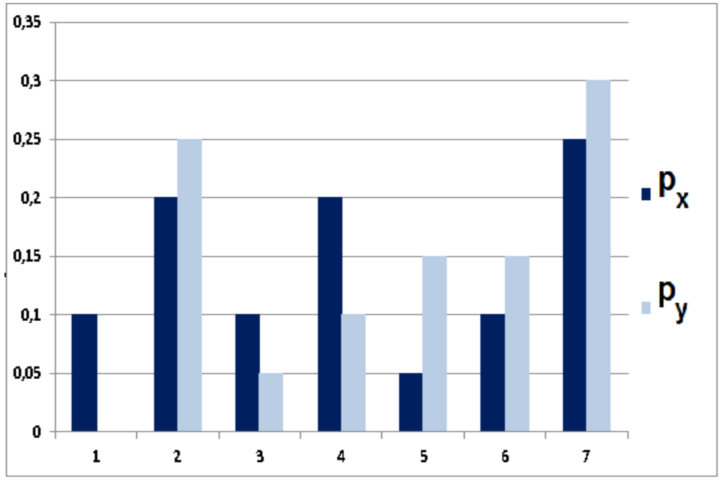
- $\mathcal{G} = \{1, 2, \ldots, n\}$ a finite state space, X, Y: discrete distributions over $\mathcal{G}$, $p_X(i) = prob(X = i)$ and $p_Y(i) = prob(Y = i)$ for $i \in \mathcal{G}$.

- **Definition of $\leq_{st}$ order**:
$X \leq_{st} Y \;\; iff \quad \sum_{k=i}^{n} p_X(k) \leq \sum_{k=i}^{n} p_Y(k), \qquad \forall i.$

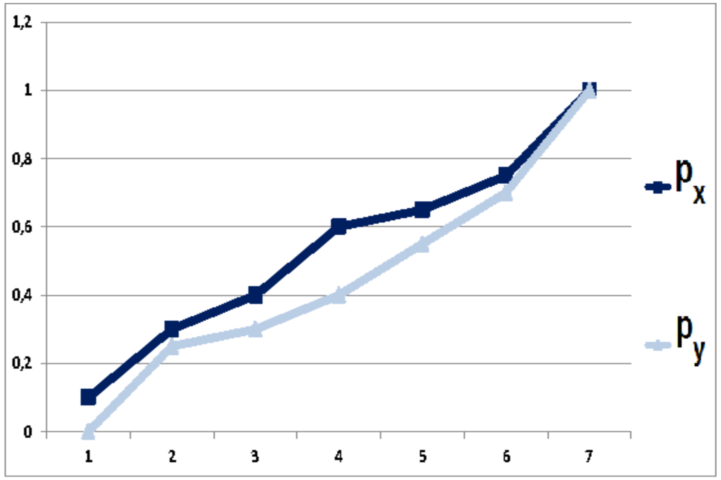
- Comparison of non decreasing rewards:

$$X \leq_{st} Y \iff E[f(X)] \leq E[f(Y)]$$

for all non decreasing functions f, whenever expectations exist.



The pmf of a discrete distributions X and Y



Cumulative distribution functions

Figure 4: $\mathcal{G} = \{1, 2, \ldots, 7\}$, $p_X = [0.1,\ 0.2,\ 0.1,\ 0.2,\ 0.05,\ 0.1,\ 0.25]$ and $p_Y = [0,\ 0.25,\ 0.05,\ 0.1,\ 0.15,\ 0.15,\ 0.3]$.

Theoretical Result

- Theorem: The finite queue with batch arrivals and batch services is stochastically monotone under the Tail Drop per unit assumption.
- Thus, if we consider two distributions H_1^l and H_1^u on K bins such that $H_1^l \leq_{st} H_1 \leq_{st} H_1^u$, then we obtain:
  - $H_3^l \leq_{st} H_3 \leq_{st} H_3^u$
  - $H_4^l \leq_{st} H_4 \leq_{st} H_4^u$
  - $H_5^l \leq_{st} H_5 \leq_{st} H_5^u$
  - We also obtain upper and lower stochastic bounds for the distribution of the losses.
- Use $K << N$. Typically $K = 100$ or 500 and $N = 80511$.

Computing Population Distribution, H_3

- We have to solve the steady-state distribution of the chain.
- Easy when the size is small.
- A new algorithm based on the convolution of distributions (Hernandez)
- with some improvements to take into account that the system is stochastically monotone
- Provides a proof of convergence.

Departure Process H_5

- H_3 is the steady-state distribution just before the arrival instants. It is the distribution of the state seen by a batch of arrivals. The arrivals modify this distribution, adding a new group of data units represented by distribution (H_1). after arrivals, we observe a buffer length distributed with H_q:

$$H_q = H_3 \otimes H_1 \tag{2}$$

- The departure histogram H_5 is defined on $\mathcal{S}$ such that $\mathcal{S} = \{k \,|\, \forall\, i \in E^{H_q} \text{ and } \forall\, j \in E^{H_2},\ k = \min(i, j)\}$ and computed from H_q as follows

$$H_5(w) = \sum_{i \in E^{H_q}} \sum_{j \in E^{H_2}} H_q(i)\, H_2(j)\, \mathbf{1}_{\{\min(i,\,j)=w\}}, \quad \forall\, w \in \mathcal{S} \tag{3}$$

- An easy numerical computation

Computing Response Time Distribution, H_4

- for FIFO queues
- We compute upper and lower bounds for H_4 because the data units arriving in the same time slot will not necessarily experienced the same delay.
- Algorithms are presented in the proceedings.
- Other techniques to compute bounds (not presented here) for queues with a work conserving discipline.

Losses

- H_L distribution of the number of data units lost at the entrance of a finite queue with a Tail Drop policy and an Arrival First assumption
- We first compute $H_n = H_3 \otimes H_1 \otimes (-H_2)$
- The distribution of losses under the Tail Drop policy is:

$$\left\{ \begin{array}{lcll} H_L(k-\mathbf{B}) & = & H_n(k) & k > \mathbf{B} \\ H_L(0) & = & \sum_{k \leq \mathbf{B}} H_n(k) & \end{array} \right.$$

Examples on MAWI trace and a single queue

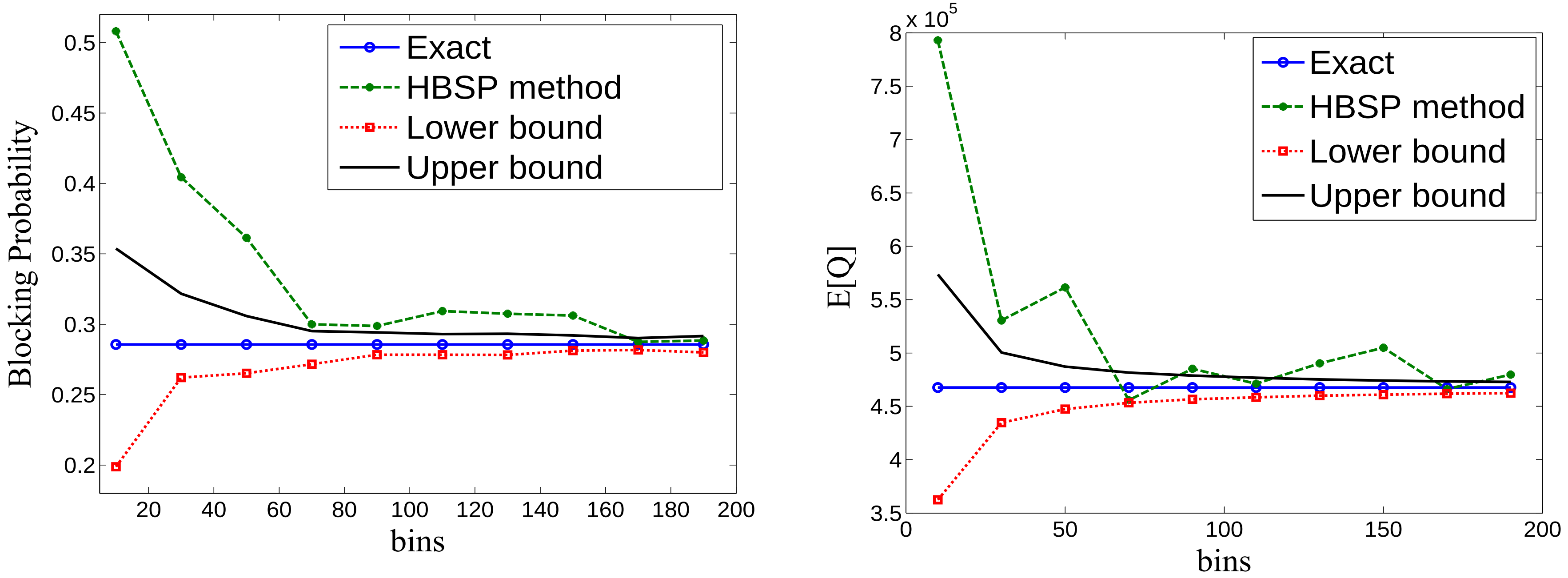


Figure 5: Blocking Probability (left), Mean buffer occupancy (right)

Number of classes vs Accuracy: QoS parameters

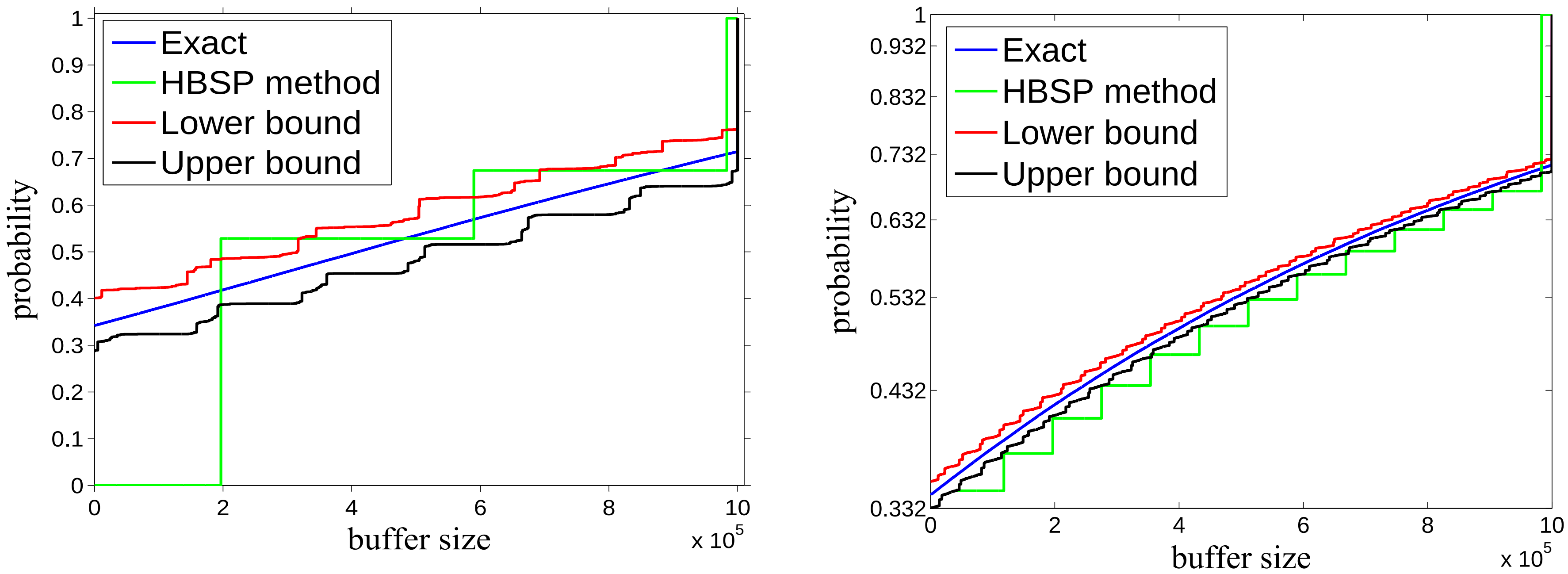


Figure 6: Cumulative probability (cdf) of buffer occupancy under MAWI traffic trace for 20 bins or 100 bins

Conclusion

- Some Active Queue Management models will be added in the method (proof of monotony)
- Add End to End Bounds for the delay
- Consider more complex arrival processes (for instance modulated by a Markov chain)
- Consider several classes of customers with priority for the resource or fair queueing.
- Not limited to networks, may be used with any large measurements data.