Analytical Approaches for Performance Evaluation of Networks-on-Chip

Part 4: Queueing Theory

Abbas Eslami Kiasari

KTH Royal Institute of Technology
Stockholm, Sweden
Objectives

- To provide basic understanding of queueing systems
- To show how queueing theory can be applied to average-case timing analysis for NoCs
Outline

- Basic concepts
- Event and node models
- Single queue systems
- Network of queues
- Conclusion
General Queueing System

Arriving customers → Queue → Server(s) → Departed customers
General Queueing System

Interarrival time distribution: $a(t)$

Service time distribution: $b(t)$

Waiting time distribution: $w(t)$

Queueing System Diagram:
- Interarrival time distribution $a(t)$
- Service time distribution $b(t)$
- Waiting time distribution $w(t)$

© Abbas Eslami Kiasari, KTH, 2012

CASES Tutorial 1, ESWeek'12, Tampere, Finland
General Queueing System

Idle server

\[ w_{n+1} = 0 \]

Busy server

\[ w_{n+1} = w_n + x_n - t_{n+1} \]
Waiting Time Distribution

\[ w_{n+1} = \max (0, w_n + x_n - t_{n+1}) = (w_n + x_n - t_{n+1})^+ \]  
Lindley’s recursion

\[ W(t) = \int_0^\infty C(t-x)\,dW(x) \quad t \geq 0 \]  
Lindley’s integral equation

\[ C(t) = \int_0^\infty B(t+x)\,dA(x) \]

\[ W^*(s) = \exp \left\{ -\frac{1}{j2\pi} \int_c \frac{s}{z(s-z)} \log[1 - A^*(-z)B^*(z)]\,dz \right\} \]  
Pollaczeck formula
Performance Measures

- Average delay
- Delay variation (jitter)
- Network throughput
- Resource utilization
- Probability of buffer overflow
Stability and Steady-State

- A queueing system is stable if arrival rate ($\lambda$) < service rate ($\mu$).
- The ratio $\rho = \lambda / \mu$ is called the utilization factor and describes the loading of a queue.
- In an unstable system ($\rho \geq 1$) packets accumulate in various queues and/or get dropped.
- For unstable systems with large buffers some packet delays become very large.
  - Flow/admission control may be used to limit the packet arrival rate.
  - Prioritization of flows keeps delays bounded for the important traffic.
- Stable systems with time-stationary arrival traffic approach a steady-state.
Little’s Law

- $N = \lambda T$
  - $N$: average number of packets in the system (queue + server)
  - $\lambda$: packet arrival rate
  - $T$: average packet delay

- $N_q = \lambda W$
  - $N_q$: average number of packets in the queue
  - $W$: average waiting time in the queue
Outline

- Basic concepts
- Event and node models
- Single queue systems
- Network of queues
- Conclusion
Event Model

- Distribution of the interarrival times
  - CDF: \( A(t) = P\{\text{interarrival time} \leq t \} \)
  - pdf: \( a(t) = \frac{d}{dt} A(t) \)
- \( P\{t_1 \leq \text{interarrival time} \leq t_2 \} = \int_{t_1}^{t_2} a(t) \, dt = A(t_2) - A(t_1) \)
Poisson Traffic

- Number of packets follows a Poisson process.
  \[ P\{X = k\} = \lambda^k e^{-\lambda} / k! \]
  where \( \lambda \) is packet arrival rate.

- Packets interarrival times are independent and exponentially distributed with the average interarrival time of \( 1/\lambda \) secs/packet.

\[ A(t) = \begin{cases} 
1 - e^{-\lambda t} & t \geq 0 \\
0 & t < 0 
\end{cases} \]

\[ a(t) = \lambda e^{-\lambda t} \]
Bursty Traffic

How to model the burstiness of NoC workload probabilistically?

MMPP: Markov Modulated Poisson Process

Poisson

MMPP
Node Model

- CDF of the service time
  - $B(x) = P\{\text{service time} \leq x\}$
- Number of servers
  - Single server
  - Multiple servers
- Capacity of queues
  - Finite or infinite
- Service discipline
  - FCFS, Random, Round-robin, Priority
Classify Queueing Models

- Kendall’s Notation (A/B/C/K/N-S)
  - A: Distribution of the interarrival time (M, D, G)
  - B: Distribution of the service times (M, D, G)
  - C: Number of servers
  - K: Maximum capacity of queue (default: infinite)
  - N: The size of the population (default: infinite)
  - S: Service discipline (default: FCFS)

- M/G/2/10-Random queueing system
  - The customer interarrival times are exponentially distributed.
  - The service time distribution is arbitrary.
  - There are two servers in the system.
  - The queue can store at most 10 customers.
  - The customers in the queue are served in random order.
Outline

- Basic concepts
- Event and node models
- Single queue systems
- Network of queues
- Conclusion
# Single Queue System

<table>
<thead>
<tr>
<th>Queueing model</th>
<th>Avg. delay in queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>M/M/1</td>
<td>$\frac{\rho}{\mu - \lambda}$</td>
</tr>
<tr>
<td>M/G/1</td>
<td>$\frac{\rho(1 + C_B^2)}{2\mu(1 - \rho)}$</td>
</tr>
<tr>
<td>G/G/1</td>
<td>$\approx \frac{\rho(C_A^2 + C_B^2)}{2\mu(1 - \rho)}$</td>
</tr>
</tbody>
</table>
Example: Single Node

- 2D packet-switched mesh network
- The packet arrivals to the input channels follow Poisson processes.
- There is enough space in routers to store backlogged packets.
- Messages are broken into some packets of fixed length of 8 flits.
- We consider an ejection channel as a server in which the packets have preferential treatment based on priorities associated with them.

\[ \lambda_1 = 0.025 \text{ North} \]
\[ \lambda_2 = 0.015 \text{ East} \]
\[ \lambda_3 = 0.05 \text{ South} \]
\[ \lambda_4 = 0.01 \text{ West} \]

\[ \mu = 0.125 \]

Ejection

© Abbas Eslami Kiasari, KTH, 2012

CASES Tutorial 1, ESWEEK'12, Tampere, Finland
Example: Single Node (cont.)

- M/D/1 priority queue

\[ \bar{W}_k = \frac{1}{2\mu} \sum_{i=1}^{4} \rho_i \]

\[ \bar{W}_1 = \frac{3.2}{1-0.2} = 4.0 \text{ cycles} \]
\[ \bar{W}_2 = \frac{3.2}{(1-0.2)(1-0.32)} = 5.9 \text{ cycles} \]
\[ \bar{W}_3 = \frac{3.2}{(1-0.32)(1-0.72)} = 16.8 \text{ cycles} \]
\[ \bar{W}_4 = \frac{3.2}{(1-0.72)(1-0.8)} = 57.1 \text{ cycles} \]

\[ \bar{N}_1 = \lambda_1 \bar{W}_1 = 0.10 \text{ packets} \]
\[ \bar{N}_2 = \lambda_2 \bar{W}_2 = 0.09 \text{ packets} \]
\[ \bar{N}_3 = \lambda_3 \bar{W}_3 = 0.84 \text{ packets} \]
\[ \bar{N}_4 = \lambda_4 \bar{W}_4 = 0.57 \text{ packets} \]
Network of Queues

- Burke’s Theorem

- Jackson Network
  \[ p(n_1, n_2, \ldots, n_k) = p(n_1)p(n_2) \ldots p(n_k) \]

- Not accurate for M/G/1 and G/G/1 case
Approximations

- **Kleinrock independence approximation**
  - Perform a delay calculation in each queue independently of other queues
  - Add the results (including propagation delay)

- Tends to be more accurate in networks with low and medium workload.
**Example: Network of Nodes**

<table>
<thead>
<tr>
<th>Flow</th>
<th>Priority</th>
<th>Packet Length ($m_i$)</th>
<th>Packet Generation Rate ($\lambda_i$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1$</td>
<td>high</td>
<td>8</td>
<td>0.02</td>
</tr>
<tr>
<td>$f_2$</td>
<td>low</td>
<td>16</td>
<td>0.01</td>
</tr>
<tr>
<td>$f_3$</td>
<td>medium</td>
<td>12</td>
<td>0.02</td>
</tr>
<tr>
<td>$f_4$</td>
<td>low</td>
<td>8</td>
<td>0.03</td>
</tr>
</tbody>
</table>
Network of Nodes (cont.)

wire delay = 1 cycle
router delay = 2 cycles
wormhole switching
one flit buffer per input channel
deterministic routing

\[ d_1 = 4t_{\text{router}} + 5t_{\text{wire}} + (m_1 - 1) t_{\text{wire}} = 8 + 5 + 7 = 20 \text{ cycles} \]

\[ \overline{D}_1 = d_1 + \overline{W}_{\text{Inj}_1 \rightarrow L_1} + \overline{W}_{L_1 \rightarrow L_4} + \overline{W}_{L_4 \rightarrow L_5} + \overline{W}_{L_5 \rightarrow \text{Ej}_6} \]
Network of Nodes (cont.)

An Analytical Latency Model for Networks-on-Chip
Abbas Eslami Kiasari, Zhonghai Lu and Axel Jantsch
IEEE Transactions on Very Large Scale Integration
(VLSI) Systems, January 2012

\[
\bar{W}_{L_5 \rightarrow E_{j_6}} = \bar{W}_{L_4 \rightarrow L_5} = 0
\]

\[
\bar{W}_{L_1 \rightarrow L_4} = \frac{1}{2} \left( C_A^2 + C_{sL_4}^2 \right) \frac{\lambda_{L_4}}{\mu_{L_4}^2} = 3.1
\]

\[
\bar{W}_{\text{ln}j_1 \rightarrow L_1} = \frac{1}{2} \left( C_A^2 + C_{sL_4}^2 \right) \frac{\lambda_{L_1}}{\mu_{L_1} \left( \mu_{L_1} - \lambda_{L_1} \right)} = 2.9
\]

\[
\bar{D}_1 = d_1 + \bar{W}_{\text{ln}j_1 \rightarrow L_1} + \bar{W}_{L_1 \rightarrow L_4} = 26
\]

G/G/1-Priority

\[
\rho_j^N \left( C_A^2 + C_{s_j}^2 \right)
\]

\[
\frac{2 \left( \mu_j^N - \lambda_{i \rightarrow j}^N \right)}{2 \left( \mu_j^N - \sum_{k=1}^{i-1} \lambda_{k \rightarrow j}^N \right)^2}
\]
## Application in NoCs

Mathematical Formalisms for Performance Evaluation of Networks-on-Chip

Abbas Eslami Kiasari, Axel Jantsch and Zhonghai Lu

Accepted for publication in the ACM Computing Surveys

<table>
<thead>
<tr>
<th>Queueing model</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>M/M/1 [1]</td>
<td>Link capacity allocation</td>
</tr>
<tr>
<td>M/M/1/K [2]</td>
<td>Buffer sizing</td>
</tr>
<tr>
<td>M/G/1 [3]</td>
<td>Average packet latency, Network throughput, Buffer utilization</td>
</tr>
<tr>
<td>MMPP/G/1 [4]</td>
<td>Bursty traffic</td>
</tr>
<tr>
<td>G/G/1-Priority [5, 6]</td>
<td>Average packet latency, Throughput, Module placement</td>
</tr>
<tr>
<td>Statistical physics [7]</td>
<td>Non-stationary effects of the system</td>
</tr>
</tbody>
</table>
Conclusion

- Queueing theory is concerned with the average-case analysis of systems whose demands occurrences and lengths can be specified probabilistically.

- Queueing theory can be used in the early design phase of NoCs.


Thanks for your attention!