Performance Impacts of Autocorrelated Flows in Systems

Evgenia Smirni
Claiborne-Stephens Associate Professor

What is autocorrelation?
- Burstiness
- Self-similarity
- Dependence (short-range, long-range)
- Well-studied in networking

- Systems?
  - Some early studies in storage systems (HP traces from the early 90s)
  - Recently (USENIX’06) from Seagate

Dependent process (example)

• Independent process
Dependent process (example)

• Independent process

• Independent process

• Independent process

• Independent process
Dependent process (example)

• Independent process
  \[ x_0 \quad x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8 \]

• Dependent process
  \[ x_0 \quad x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad \]

Dependence Metrics

- Autocorrelation function (ACF) of a process \( \{X_0, X_1, X_2, X_3, \ldots\} \) with lag \( k \)

\[
\text{corr}[X_0, X_k] = \frac{E[(X_0 - E[X])(X_k - E[X])]}{\text{Var}[X]}
\]
**Dependence Metrics**

- Autocorrelation function (ACF) of a process \( \{X_0, X_1, X_2, X_3, \ldots \} \) with lag \( k \)
  \[
  \text{corr}[X_0, X_k] = \frac{E[(X_0 - E[X])(X_k - E[X])]}{\text{Var}[X]}
  \]

- Higher ACF, stronger dependence

**Dependence in Storage Systems**

- Graphs from Seagate Research
  - (a) Disk interarrival times
  - (b) Disk service times

**Examples of ACF**

- Independent SRD LRD
Examples of ACF

Impact of Correlated Arrivals

Dependent
Arrivals

Independent
Service
(Exponential)
Focus: Three-tiered systems

- Experiments
- Analytic models
- Policy Development

### Outline

- Closed System
  - Experimental evaluation using TPC-W
  - Autocorrelation propagation
  - Impact of autocorrelation
  - Two-queue closed system

- Policy Development
  - Load balancing under autocorrelated flows

- On-going work

Highly Correlated (Bursty) Arrivals

[related work]
Outline

- Closed System
  - Experimental evaluation using TPC-W
  - Autocorrelation propagation
  - Impact of autocorrelation
  - Two-queue closed system

- Policy Development
  - Load balancing under autocorrelated flows

- On-going work

TPC-W Specifications

- On-line book store Web site
- 14 Interactions
  - (browsing-based vs. ordering-based)
    - Browsing mix (95% vs. 5%)
    - Shopping mix (80% vs. 20%)
    - Ordering mix (50% vs. 50%)
- Databases (different number of items)

<table>
<thead>
<tr>
<th># Items</th>
<th>10K</th>
<th>100K</th>
<th>500K</th>
<th>1M</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB size</td>
<td>1.5GB</td>
<td>1.5GB</td>
<td>1.9GB</td>
<td>2.1GB</td>
</tr>
</tbody>
</table>

Multi-tiered E-commerce Site Set-up

- Clients (EB)
  - P4/2GHz
  - 256MB
- Front Server (Apache/Tomcat)
  - P3/1.3GHz
  - 2GB
- DB Server (MySQL 4.0)
  - Dual Xeon/1.5GHz
  - 768MB
### Multi-tiered E-commerce Site Set-up

![Diagram showing client departures, HTTP requests, HTTP reply, and MySQL reply]

- **Clients (EB)**
  - P4/2GHz
  - 256MB
- **Front Server (Apache/Tomcat)**
  - P3/1.3GHz
  - 2GB
- **DB Server (MySQL 4.0)**
  - Dual Xeon/1.5GHz
  - 768MB

### Dependence in Service Process

- Collected traces across tiers
  - Calculate ACF off-line
- Thinking time – exponential distribution
  - No ACF
- Service process in each server
  - Hard to obtain by measurements
  - Observing dependence in arrival and departure processes
  - Existence of dependence in service process
    - Increase of ACF for small lags
- DB server is the bottleneck

### ACF Propagation

- Browsing mix, 10K DB, 384 EBs

![Graph showing ACF propagation with lag (k)]
Observations

- Dependence in service processes
- Dependence in lower tiers affects the arrival process to the higher tiers but no dependence in the process of session generation!
- ACF propagation in all tiers
Observations

- Dependence in service processes
- Dependence in lower tiers affects the arrival process to the higher tiers
  - but no dependence in the process of session generation!
- ACF propagation in all tiers

Confirm the observations by an analytic model

TPCW Model

\[ Q_0 \quad 1 - p \quad Q_1 \quad p \quad Q_2 \]

- \( Q_0 \): Exponential distribution
- \( Q_1 \): Correlated MMPP process, high variance
- \( Q_2 \): Non-correlated Hyperexponential, high variance

ACF Propagation -- 384 MPL

\[ \text{ACF} \]

\[ \text{lag (k)} \]

\[ \text{Client Arr} \quad \text{Client Dep} \quad \text{Front Arr} \quad \text{Front Dep} \quad \text{DB Arr} \quad \text{DB Dep} \]
ACF Propagation -- 384 MPL

Performance Comparison

(a) Average queue length
(b) Average utilization

This simple model captures TPC-W behavior qualitatively.
Outline

- Closed System
  - Experimental Evaluation using TPC-W
  - Autocorrelation propagation
  - Impact of autocorrelation
  - Two-queue closed system
- Policy Development
  - Load balancing under autocorrelated flows
- On-going work

Comparison

- Comparison with independent services
- same moments
  - mean, cv and higher moments

<table>
<thead>
<tr>
<th>Q1</th>
<th>Q2 (bottleneck)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACF</td>
<td>Dependent</td>
</tr>
<tr>
<td>NOACF</td>
<td>Independent</td>
</tr>
</tbody>
</table>

Performance Comparison

(a) Average round-trip time

Left column: NOACF  Right column: ACF

Dependent flows degrade overall system performance.
Performance Comparison
(b) Average queue length

Dependent flows “balance” the load among queues.

Performance Comparison
(c) Average utilization

Utilization is low under correlated flows.
Observations

- Dependence has significant effect on system performance
- ACF propagates into all tiers
- Overload (VERY long response times) can happen under medium load if dependence exists
  - Dependence should be considered in capacity planning
  - Tails do not necessarily come from the bottleneck device

Outline

- Closed System
  - Experimental Evaluation using TPC-W
  - Autocorrelation propagation
  - Impact of autocorrelation
  - Two-queue system

- Policy Development
  - Load balancing under autocorrelated flows

- On-going work

Clustered Servers

- Front-end Dispatcher
- Back-end Nodes

Load Balancing
Clustered Servers

- Round Robin (RR)
- Random
- Join Shortest Queue (JSQ)
- Join Shortest Weighted Queue (JSWQ)

Front-end Dispatcher

Back-end Nodes

Load Balancing

- Round Robin (RR)
- Random
- Join Shortest Queue (JSQ)
- Join Shortest Weighted Queue (JSWQ)
- AdaptLoad

Arrivals:
- Self-similarity
- Autocorrelation

Heavy tailed service time
Clustered Servers

Front-end Dispatcher

Back-end Nodes

Load Balancing

Heavy tailed service time

Arrivals:
Self-similarity Autocorrelation

Effect of ACF on Load Balancing

Response Time

Size-based Policies do NOT win!

Size-based Policies do NOT win!

WHY?
Review: AdaptLoad

Step 1: Build histogram on-line
Step 2: At the end of monitoring window, find the boundaries to partition the total work (area) equally.
Review: AdaptLoad

Step 1: Build histogram on-line
Step 2: At the end of monitoring window, find the boundaries to partition the total work (area) equally

S_EQAL

- Server $i$ increase $p_i$ of its work
- Corrective factor $p_i$ : $\sum p_i = 0$
- negative (reducing work) vs. positive (increasing work)
- $p_1 = -R$ (pre-determined corrective constant)
- $p_i$ using semi-geometric method to decide

Performance of S_EQAL

- Service time: WorldCup 1998 Trace
- Inter-arrival time: MMPP(2)
  - Same moments
  - With short range dependence (SRD)
- 4 servers in the cluster
- Average utilization per server: 62%

Average Slowdown by $R$
Inside Each Server

Dynamic Policy: D_EQAL

- R is initialized as 0
- Adjust R for a small value Adj at the end of each monitoring window
- The adjustment should improve both slowdown and response time
- If not, wrong direction

How to obtain a good R?
**Performance of D_EQAL**

- Slowdown vs. Response time
- Monitoring windows

**Effectiveness of D_EQAL**

- Monitoring windows

**Summary**

- Workload characterization in multi-tiered closed systems
  - ACF propagates into all the tiers
  - Exists in storage systems
  - But also other parts (e.g., cache behavior, memory pressure)
  - Overload can happen under medium load if dependence exists
  - Tier with ACF affects performance a lot (although not bottleneck)
  - Classic analytic modeling techniques do not apply
    - e.g., MVA or approximation methods
  - Yet, simple models that capture ACF in service process capture trends

- Policy development
  - ACF-aware load balancing policy for cluster with dependent flows

**On-going work**

- Theory
  - Closed systems (i.e., multi-tiered example)
    - Perils of simulation, need for analytic models
    - Approximation methods
  - Open systems: Departure process
    - Two approximations, ETAQA-based and Lumping

- Policy development
  - QoS policies
  - Storage systems to schedule foreground/background jobs
  - General scheduling policies with minimum information
Acknowledgements

- **Students**
  - Qi Zhang (just graduated, soon to join Microsoft)
  - Ningfang Mi

- **Collaborators**
  - Alma Riska and Erik Riedel (Seagate Research)
  - Lucy Cherkasova (HP Labs)

- **NSF, Seagate, HP**

- More information (several papers)
  http://www.cs.wm.edu/~esmirni