Ad Hoc Distributed Simulation of Queueing Networks

Ya-Lin Huang
Christos Alexopoulos
Michael Hunter
Richard M. Fujimoto
Agenda

- Motivations
- Ad Hoc Distributed Simulations
- Queueing Network Simulations
- Experiments and Results
- Conclusions and Future Works
Motivations

- Sensor networks gain in importance
- On-line simulations emerge
Motivations (cont.)

- Operating on-line simulations within sensor networks show benefits
  - Reduced communication
  - Quick response to changes
  - Resilient to failures
- Ad hoc distributed simulation is an approach to embedded on-line simulations in sensor networks
  - Applied to transportation management systems
  - Generalizing the approach
  - Applying to queueing systems
Ad Hoc Distributed Simulations

- Ad hoc distributed simulation = \{autonomous logical processes (LPs)\} + space time memory (STM) + rollback mechanism
Ad Hoc Distributed Simulations (cont.)

- Each LP autonomously models a portion of the system under investigation
  - Partitioning of the system is arbitrary (compared to conventional distributed simulations)
  - The portion modeled by one LP may change over time

[Diagram showing conventional and ad hoc approaches]
Ad Hoc Distributed Simulations (cont.)

- LPs exchange information via the STM
  - STM holds time stamped updates of variables shared among LPs
  - LPs update values with a time interval within which the update is valid
  - LPs read values by specifying the desired variable and a time stamp
    - Sufficient number of predictions
    - Insufficient number of predictions
Ad Hoc Distributed Simulations (cont.)

- The rollback mechanism is to correct invalid input values used by LPs
  - The rolled back LP rewinds its simulation time back to when the invalid value was used
  - The LP restores its prior state
  - The LP retracts updates that should be canceled out
  - The LP restarts the simulation with new values

The rollback mechanism is to correct invalid input values used by LPs:
- The rolled back LP rewinds its simulation time back to when the invalid value was used.
- The LP restores its prior state.
- The LP retracts updates that should be canceled out.
- The LP restarts the simulation with new values.
Queueing Network Simulations

- Queueing Networks are used to model a variety of industrial systems
- The modeled open queuing network consists of 64 nodes arranged into an 8x8 rectangular configuration

The modeled open queuing network consists of 64 nodes arranged into an 8x8 rectangular configuration.
Queueing Network Simulations (cont.)

- Served jobs either leave the system or move to any of the neighboring nodes; routing probabilities vary on node types.

- The modeled network is large with complex routings, various path cycles, and traffic intensities of the nodes range from 0.3 to 0.8.
Applying Ad Hoc Approach - Partitioning

- A grid-like partitioning with significant overlapping among portions
  - Five portions, each with $n_1$, $n_2$, $n_3$, $n_4$, and $n_5$ LPS respectively
  - $\Sigma n_i = N$
  - Current design, $n_1 = n_2 = n_3 = n_4 = n_5 = 8$ ($N=40$)
Applying Ad Hoc Approach – Information Exchange

- Arrivals on input links are modeled as renewal processes where distribution parameters are estimated using the information from the STM
  - No data at first
  - Set to Poisson($\lambda=1/6$)

- Departure statistics of all links are written to the STM
  - Number of observed inter-departure times
  - First two moments of these inter-departure times
  - Denoted ($n_{t,\text{tr}}$, $m_{1,t,\text{tr}}$, $m_{2,t,\text{tr}}$)
Applying Ad Hoc Approach - Information Exchange (cont.)

- Simulation starts in empty and idle state; information exchange begins after $\nu$ seconds in simulation time
- Every $d_{\text{read}}$ seconds, each LP “reads” the statistics of the input links
  - Reuse last statistics (i.e., the one corresponding to $d_{\text{read}}$ seconds earlier)
  - Inform the STM of the usage
  - Be rolled back if the STM finds the statistics invalid
- Every $d_{\text{write}}$ seconds, each LP writes statistics of all modeled links
- The observation period is over the last $d_{\text{observe}}$ seconds with the rolling window mechanism, preventing the statistics from being sensitive to random fluctuations
- Current design,
  - $\nu=300$ (seconds)
  - $d_{\text{read}}=d_{\text{write}}=30$ (seconds)
  - $d_{\text{observe}}=300$ (seconds)
Applying Ad Hoc Approach – Arrival Process Approximation

- When the service times follow exponential distribution, the arrival processes on the input links are approximated as Poisson processes.
- When the service times are non-exponential, the arrival processes are approximated by renewal processes with gamma inter-arrival times\(^1,2\).
- The parameters (for either case) are estimated using the method of moments:
  - Let \( m_1 \) and \( m_2 \) be the first two moments.
  - In former case, the rate parameter \( \lambda \) follows
    \[
    \hat{\lambda} = \frac{1}{m_1}
    \]
  - In later case, the shape parameter \( \sigma \) and the scale parameter \( \beta \) follows
    \[
    \hat{\alpha} = \frac{m_1^2}{m_2 - m_1^2}, \quad \hat{\beta} = \frac{m_2 - m_1^2}{m_1}
    \]

Applying Ad Hoc Approach - Rollback Mechanism & Data Aggregation

- Few predictions may bias the aggregated value; the rollback mechanism is applied if there are at least $k$ predictions (i.e., $k$ triples of $(n_{t,l}, m_{1,t,l}, m_{2,t,l})$ with the same $t$ but different $l$)
- The rollback detection function is invoked when
  - An LP informs the statistics it is using
  - Statistics are written to the STM
  - Statistics are deleted (retracted) from the STM

- First, the rollback detection function requests an aggregated value calculated from the pooled-averaging approach

$$
\hat{m}_{1,t} = \frac{\sum_{l} n_{t,l} \times m_{1,t,l}}{\sum_{l} n_{t,l}} \quad \hat{m}_{2,t} = \frac{\sum_{l} n_{t,l} \times m_{2,t,l}}{\sum_{l} n_{t,l}}
$$
Applying Ad Hoc Approach – Rollback Mechanism & Data Aggregation

Second, the difference between the first moments are considered; a rollback is triggered if the relative difference is greater than $\varepsilon_{\text{relative}}$

Third, if a rollback is necessary, another aggregated value is requested as new input to the LP being rolled back

The random sampling and pooled-variance approach is adopted

$$\hat{m}_{1,t} = \text{RandUni}(\{m_{1,t,l}\}) + h\varepsilon$$

$$\hat{m}_{2,t} = \frac{\sum_{l} n_{t,l} - 1}{\sum_{l} n_{t,l}} \times \frac{\sum_{l} (n_{t,l} - 1) s_{t,l}^2}{\sum_{l} (n_{t,l} - 1)} + \hat{m}_{1,t}, \text{ with } s_{t,l}^2 = \frac{n_{t,l} (m_{2,t,l} - m_{1,t,l}^2)}{n_{t,l} - 1}$$
Applying Ad Hoc Approach - Rollback Mechanism & Data Aggregation

- When an LP is rolled back to simulation time $t$,
  - Retracts its read and write operations associated to time greater than or equal to $t$
  - Rewinds further back to $t - d_{\text{rollback}}$ so that the mismatches do not result in abrupt changes in output statistics
    - No writes to the STM during the $d_{\text{rollback}}$-second coast-forward phase
    - Interpolating the input values during the coast-forward phase
      - Linear interpolation on first moments
      - Linear interpolation on coefficients of variation to calculate second moments

- Current design,
  - $k = 8$
  - $\varepsilon_{\text{relative}} = 10\%$
  - $d_{\text{rollback}} = 300$ (seconds)
Experiments

- We differ service time distribution to construct three scenarios
  - Scenario 1: Exponential(λ=1)
    - Rate parameter: λ
    - Mean: 1
    - Variance: 1
  - Scenario 2: Gamma(α=2, β=0.5)
    - Shape parameter: α
    - Scale parameter: β
    - Mean: 1
    - Variance: 0.5
  - Scenario 3: Gamma(0.25, 4)
    - Mean: 1
    - Variance: 4
- We are interested in the steady-state mean utilization and queue length of each node
  - The data are collected after one hour in simulation time and the collection lasts for another one hour
Results - Scenario 1

Relative Errors of Utilizations Based on 100 IID Runs under Scenario 1

- Sequential Simulations
- Ad Hoc Distributed Simulations

Relative error: compared against analytical solutions
Results - Scenario 1 (cont.)

Point Estimates and 90% Confidence Intervals for Steady-state Mean Queue Length Based on 100 IID Runs under Scenario 1

Queue Length

Server 3
Server 2
Server 1
Server 0

Ad Hoc Distributed Simulations
Sequential Simulations
Point Estimates and 90% Confidence Intervals for Steady-state Mean Queue Length Based on 100 IID Runs under Scenario 1
Results - Utilization

Relative difference:
compared against sequential simulations

Relative Differences of Utilizations Based on 100 IID Runs

- Scenario 1
- Scenario 2
- Scenario 3
Results - Queue Length

Relative Differences of Queue Lengths Based on 100 IID Runs

- Scenario 1
- Scenario 2
- Scenario 3
Conclusions and Future Works

- We generalized the ad hoc distributed simulation approach
- We applied it to a queueing network simulation to show its capabilities and some weakness
  - Work comparably well compared to sequential simulations especially when the variation of service times is small
  - Reveal some issues when the variation of service time is large, an area of future work
- Future works include
  - Relaxing restrictions (e.g., fixed partitioning)
  - Examining response to unexpected changes in sensor measurements
  - Evaluating resilience to failures and errors in the underlying sensor network
Questions?
Appendix
Results - Scenario 1

- Relative errors of utilizations based on 100 IID runs using ad hoc approach

<table>
<thead>
<tr>
<th>Server 0</th>
<th>Server 1</th>
<th>Server 2</th>
<th>Server 3</th>
<th>Server 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25 %</td>
<td>0.05 %</td>
<td>0.13 %</td>
<td>0.35 %</td>
<td>-0.10 %</td>
</tr>
<tr>
<td>Server 10</td>
<td>Server 11</td>
<td>Server 18</td>
<td>Server 19</td>
<td>Server 27</td>
</tr>
<tr>
<td>0.22 %</td>
<td>0.51 %</td>
<td>0.32 %</td>
<td>0.33 %</td>
<td>0.31 %</td>
</tr>
</tbody>
</table>

- Relative errors of queue lengths based on 100 IID runs using ad hoc approach

<table>
<thead>
<tr>
<th>Server 0</th>
<th>Server 1</th>
<th>Server 2</th>
<th>Server 3</th>
<th>Server 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.45 %</td>
<td>-0.18 %</td>
<td>0.25 %</td>
<td>0.58 %</td>
<td>-0.40 %</td>
</tr>
<tr>
<td>Server 10</td>
<td>Server 11</td>
<td>Server 18</td>
<td>Server 19</td>
<td>Server 27</td>
</tr>
<tr>
<td>0.98 %</td>
<td>1.53 %</td>
<td>1.37 %</td>
<td>1.69 %</td>
<td>1.15 %</td>
</tr>
</tbody>
</table>
**Results - Scenario 1**

- Relative differences of utilizations based on 100 IID runs using ad hoc approach

<table>
<thead>
<tr>
<th>Server 0</th>
<th>Server 1</th>
<th>Server 2</th>
<th>Server 3</th>
<th>Server 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.23 %</td>
<td>0.14 %</td>
<td>0.14 %</td>
<td>0.29 %</td>
<td>-0.05 %</td>
</tr>
<tr>
<td>Server 10</td>
<td>Server 11</td>
<td>Server 18</td>
<td>Server 19</td>
<td>Server 27</td>
</tr>
<tr>
<td>0.24 %</td>
<td>0.59 %</td>
<td>0.18 %</td>
<td>0.21 %</td>
<td>0.18 %</td>
</tr>
</tbody>
</table>

- Relative differences of queue lengths based on 100 IID runs using ad hoc approach

<table>
<thead>
<tr>
<th>Server 0</th>
<th>Server 1</th>
<th>Server 2</th>
<th>Server 3</th>
<th>Server 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.30 %</td>
<td>-0.09 %</td>
<td>0.39 %</td>
<td>0.33 %</td>
<td>-0.45 %</td>
</tr>
<tr>
<td>Server 10</td>
<td>Server 11</td>
<td>Server 18</td>
<td>Server 19</td>
<td>Server 27</td>
</tr>
<tr>
<td>0.93 %</td>
<td>1.72 %</td>
<td>0.81 %</td>
<td>0.83 %</td>
<td>0.53 %</td>
</tr>
</tbody>
</table>
Results - Scenario 2

- Relative differences of utilizations based on 100 IID runs using ad hoc approach

<table>
<thead>
<tr>
<th></th>
<th>Server 0</th>
<th>Server 1</th>
<th>Server 2</th>
<th>Server 3</th>
<th>Server 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server 0</td>
<td>-0.11 %</td>
<td>-0.03 %</td>
<td>-0.29 %</td>
<td>0.05 %</td>
<td>0.24 %</td>
</tr>
<tr>
<td>Server 10</td>
<td>0.20 %</td>
<td>0.23 %</td>
<td>0.31 %</td>
<td>0.37 %</td>
<td>0.30 %</td>
</tr>
</tbody>
</table>

- Relative differences of queue lengths based on 100 IID runs using ad hoc approach

<table>
<thead>
<tr>
<th></th>
<th>Server 0</th>
<th>Server 1</th>
<th>Server 2</th>
<th>Server 3</th>
<th>Server 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server 0</td>
<td>-0.17 %</td>
<td>-0.16 %</td>
<td>-0.29 %</td>
<td>0.16 %</td>
<td>0.13 %</td>
</tr>
<tr>
<td>Server 10</td>
<td>0.94 %</td>
<td>0.63 %</td>
<td>1.53 %</td>
<td>1.17 %</td>
<td>1.10 %</td>
</tr>
</tbody>
</table>
Results - Scenario 3

- Relative differences of utilizations based on 100 IID runs using ad hoc approach

<table>
<thead>
<tr>
<th></th>
<th>Server 0</th>
<th>Server 1</th>
<th>Server 2</th>
<th>Server 3</th>
<th>Server 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server 0</td>
<td>-0.12 %</td>
<td>0.35 %</td>
<td>0.23 %</td>
<td>0.18 %</td>
<td>0.48 %</td>
</tr>
<tr>
<td>Server 10</td>
<td>-0.09 %</td>
<td>0.72 %</td>
<td>0.03 %</td>
<td>0.37 %</td>
<td>0.42 %</td>
</tr>
</tbody>
</table>

- Relative differences of queue lengths based on 100 IID runs using ad hoc approach

<table>
<thead>
<tr>
<th></th>
<th>Server 0</th>
<th>Server 1</th>
<th>Server 2</th>
<th>Server 3</th>
<th>Server 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server 0</td>
<td>-0.65 %</td>
<td>1.41 %</td>
<td>0.32 %</td>
<td>1.21 %</td>
<td>0.96 %</td>
</tr>
<tr>
<td>Server 10</td>
<td>0.97 %</td>
<td>5.74 %</td>
<td>2.95 %</td>
<td>4.07 %</td>
<td>5.01 %</td>
</tr>
</tbody>
</table>