AMVA Techniques for High Service Time Variability

Derek L. Eager, Daniel J. Sorin, and Mary K. Vernon

Department of Computer Science
University of Saskatchewan

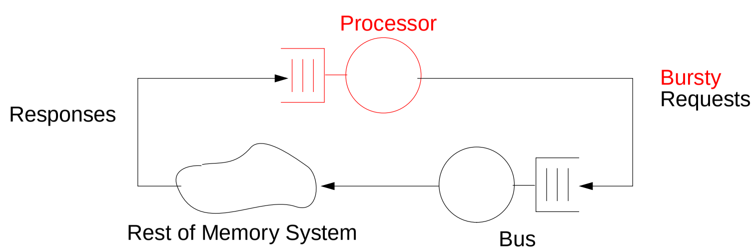
Computer Sciences Department
University of Wisconsin - Madison

Approximate Mean Value Analysis (AMVA)

- MVA is a technique for computing system performance
  - Compute mean values of residence times, queue lengths, etc.
  - \+ Easy to create and solve, if system is separable
  - \- Strict separability assumptions
- AMVA extends MVA
  - Replaces exact equations with simpler approximations
  - \+ Computationally cheaper
  - \+ Model non-separable system features with intuitive heuristics
    - e.g., non-exponential service times at FCFS queues
  - \- Validation required

Motivation and Problem

- Many systems of interest exhibit bursty behavior
  - e.g., memory requests from ILP processors



- Problem #1: Estimate R at the bursty processor
- Problem #2: Estimate R at the bus
- Current techniques are insufficient
  - Standard AMVA approximation is inaccurate
  - Decomposition is computationally expensive

Outline

✓ Motivation and Problem

- Existing Techniques for Modeling High Service Time Variability
  - Standard AMVA approximation
  - Decomposition approach
- Three New Techniques
  - A new interpolation for residual life
  - AMVA - Decomposition for high service time variability
  - Analysis of downstream queue with bursty arrivals
- Summary

Standard AMVA Approximation

- τ = service time
- CV_τ = coefficient of variation in service time
- Standard AMVA approximation for residual life at a service center with $CV_\tau \neq 1$

$$\text{residual life} = L = \frac{\tau}{2}(1 + CV_\tau^{\,2})$$

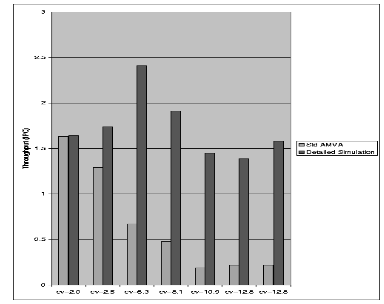
- Use L in AMVA equation for residence time

$$\text{mean residence time}, R = \tau\left[1 + \frac{N-1}{N}(Q-U)\right] + \frac{N-1}{N}U\boldsymbol{L}$$

+ Accurate for $CV_\tau \leq 1$
- Assumes that arrivals to bursty server are random
- Does not account for downstream burstiness

Standard AMVA Approximation, continued

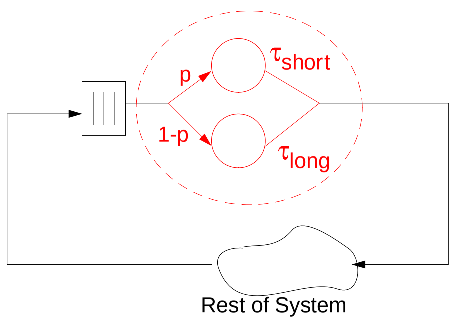
- Accurate for $CV \leq 1$
- Inaccurate for CV >> 1, for example:



- Inaccuracy due to overestimating residual life at bursty server

Decomposition: Model of Bursty Server

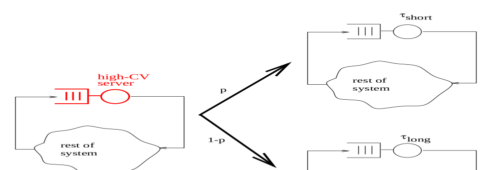
- Model bursty server as (2-stage) hyperexponential server



- Choose τ_{short}, τ_{long}, and p to match mean & CV of service time

Decomposition: System Analysis

- Decomposed queuing networks



+ Based on theory of near-complete decomposability
+ Typically highly accurate
+ Captures impact of burstiness at downstream queues
- Solution time exponential in number of bursty service centers
  - H high CV centers $\rightarrow 2^H$ networks to solve

Outline

- ✓ Motivation and Problem
- ✓ Existing Techniques
- Three New Techniques
  - A new interpolation for residual life
  - AMVA - Decomposition
  - A model for downstream burstiness
- Conclusions

A New Interpolation for Residual Life

- Replace standard approximation with the following:

$$\text{residual life} = \frac{T}{T + R_{other}}\tau + \frac{R_{other}}{T + R_{other}}L$$

where

$$T = \frac{\tau_{short}\tau_{long}}{\tau}$$

$$R_{other} = \text{mean residence time in rest of system}$$

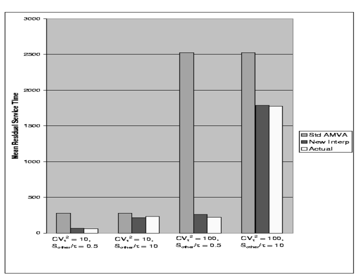
+Exact at endpoints:
- R_{other} >> T: residual life → L (arrivals back are random)
- R_{other} << T: residual life → τ (arrivals back immediately)

+Exact when R_{other} is exponentially distributed

A New Interpolation for Residual Life: Accuracy

- Example accuracy for other cases:

Mean Residual Service Time Estimates
Networks with 2 FCFS Centers
N=5, p=0.99, τ=50

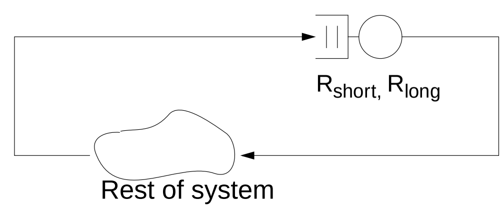


- Still inaccurate for arrival queue length
- Still ignores downstream burstiness

AMVA Decomposition

- Key idea: Decompose *only at level of individual bursty center*

$$R_{bursty} = pR_{short} + (1-p)R_{long}$$

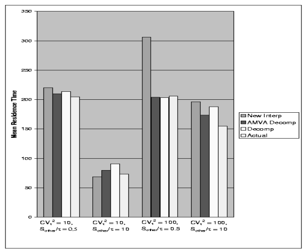


- Use standard AMVA for all queues

AMVA Decomposition: Accuracy

+ Solution time is much faster than Decomposition
- Example accuracy

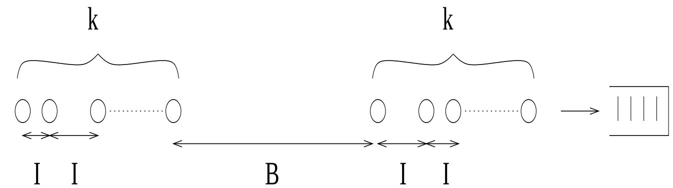
Mean Residence Time Estimates
FCFS Queue with High Service Time CV



+Similar accuracy to Decomposition Approach
- Still ignores downstream burstiness

A Model for Downstream Burstiness

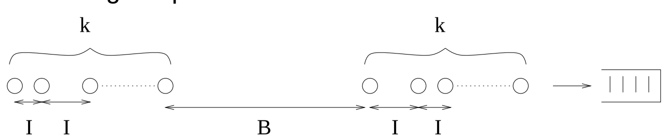
- So far, we've ignored arrival burstiness at downstream queues
- Model downstream burstiness with 3 parameters:



  - k = mean number of customers per burst (geometric dist.)
  - I = mean inter-arrival time during a burst (exponential)
  - B = mean time between bursts (exponential)

A Model for Downstream Burstiness, cont'd

- Determining the parameter values



- Can estimate CV of arrival process by Sevcik et al. method
- From throughput and $CV_{arrivals}$, can estimate model parameters
- Solve underconstrained problem (3 parameters, 2 constraints)
  - e.g., set I equal to τ_a

A Model for Downstream Burstiness, cont'd

- Using the model to compute mean residence time downstream
  - S_{down} = service time at the downstream queuing center
  - Q_{nb} = mean queue length during the time between bursts

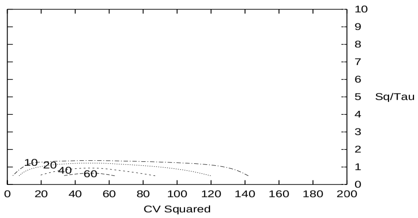
$$R_{down} = S_{down}\left(1 + \frac{N-1}{N}Q_{nb} + (k-1)\right) - (k-1)I$$

  - Skipping some steps ...

$$Q_{nb} = Q - X\left[\frac{I(k-1)(S_{down} - I)}{S_{down}}\right]$$

Accuracy of AMVA-Decomp with Downstream Burstiness

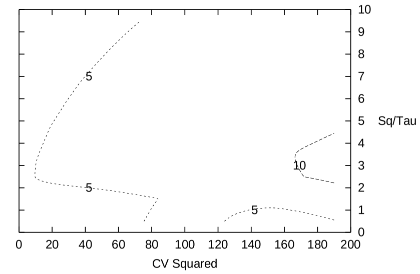
- Mean residence time at center with bursty arrivals



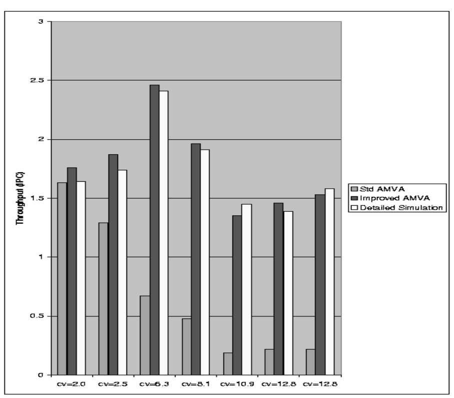
+ Highly accurate except over small region of design space
- Inaccurate region only for cases where center is negligible

Accuracy of AMVA Decomp with Downstream Burstiness

- Mean residence time in system



Applying the Techniques to ILP Multiprocessor Model



Summary and Future Work

- Modeling bursty behavior is important
- Standard AMVA equation is inaccurate for CV > 1
- Traditional decomposition is accurate but expensive
- AMVA-Decomp is accurate and less expensive
- Modeling downstream burstiness improves overall accuracy
- Future work: multiple customer classes