Pull versus Push Mechanism in Large Distributed Networks: Closed Form Results

Wouter Minnebo, Benny Van Houdt

*Dept. Mathematics and Computer Science
University of Antwerp - iMinds
Antwerp, Belgium

Universiteit Antwerpen

Outline

- Problem description
- Pull / Push / Hybrid strategies
- Finite system model
  - Continuous time Markov chain
- Infinite system model
  - Ordinary differential equations
  - Closed form results
- Finite system simulations
- Conclusion

Universiteit Antwerpen

Problem Description

Set of M / M /1 queues

- N queues (single server with infinite waiting room)
- Each has its own arrival stream of jobs: Poisson (λ)
- Processing time is exponential ($\mu = 1$)

Distribute work to reduce response time

- Some servers may be empty while others have jobs waiting $\rightarrow$ **inefficient**
- Distribute waiting jobs (load sharing)
  - **Pull** strategies (work stealing):
    Lightly-loaded servers attempt to attract work
  - **Push** strategies:
    Heavily-loaded servers attempt to forward work
  - **Hybrid** strategies:
    Combine Pull and Push strategy

Load sharing strategies

Communication via probe messages

- Random queues are probed according to an Interrupted Poisson process (r)
- If the target accepts, a task is transfered to it
- Transfer and probe time is considered zero, i.e., instantaneous

Strategies

- **Pull**: Idle servers generate probes (r), busy servers accept
- **Push**: Servers with pending jobs generate probes (r), idle servers accept
- **Hybrid**: Idle and servers with pending jobs generate probes ($r = r_1 + r_2$), servers accept accordingly

Performance

- What is the **required probe rate** of a strategy to achieve a specified **mean delay**?

Load sharing strategies

Relation to existing work

- Other works frequently use a maximum of Lp probes (in batch or one-by-one) at task arrival / completion instead of rate-based probing
- Parameter r allows to match any predefined R
- It is fair to compare strategies if they have a similar (preferably equal) overall probe rate R

Remarks

- Rate-based probing performs better, given the same overall probe rate R, if the Lp probes are sent in batch
- Both methods are equivalent if overall probe rate R is matched, given that the Lp probes are sent one-by-one until success or the maximum is reached

Finite system model - Continuous time Markov chain

System state

- N queues, independent arrivals and completions
- Define $X^{(N)}(t) = (X_1^N(t), X_2^N(t), \ldots)_{t \geq 0}$
- $X_i^{(N)}(t) \in \{0, \ldots, N\}$:
  Number of nodes with at least i jobs in queue at time t
- Note that for any state $x = (x_1, x_2, \ldots)$, $x_i \geq x_{i+1}$ for all $i \geq 1$

State transitions

- $q^{(N)}(x, y)$ is the transition rate between state $x = (x_1, x_2, \ldots)$ and $y = (y_1, y_2, \ldots)$
- Events incurring a state change:
  - Arrival
  - Completion
  - Succesful job transfer

Finite system model - Continuous time Markov chain

State transitions

- Arrival : $y = x + e_i$
  $q^{(N)}(x, y) = \lambda(x_{i-1} - x_i)$
- Completion : $y = x - e_i$
  $q^{(N)}(x, y) = (x_i - x_{i+1})$
- Succesful task transfer : $y = x + e_1 - e_i$
  $q^{(N)}(x, y) = \frac{r(N - x_1)(x_i - x_{i+1})}{N}$

Load sharing strategies

- These transitions describe both pull, push and hybrid strategies, although semantics differ
- Probe rate $\times$ target is idle $\times$ initiator has exactly i jobs
  $\rightarrow$ **Push**
- Probe rate $\times$ initiator is idle$\times$ target has exactly i jobs
  $\rightarrow$ **Pull**

From Finite to Infinite

Transition rates

- Define $\beta_\ell(x/N) = q^{(N)}(x, x+\ell)/N$, such that

$$\begin{aligned} \beta_{e_i}(x/N) &= \lambda(x_{i-1}/N - x_i/N), && \text{for } i \geq 1 \\ \beta_{-e_i}(x/N) &= (x_i/N - x_{i+1}/N), && \text{for } i \geq 1 \\ \beta_{e_1-e_i}(x/N) &= r(1 - x_1/N)(x_i/N - x_{i+1}/N) && \text{for } i \geq 2 \end{aligned}$$

- Define

$$F(x) = \textstyle\sum_{i\geq 1}(e_i\beta_{e_i}(x) - e_i\beta_{-e_i}(x)) + \sum_{i\geq 2}(e_1 - e_i)\beta_{e_1-e_i}(x)$$

- Then the ODEs $\frac{d}{dt}x(t) = F(x(t))$ describe the evolution of the infinite system model
- Density dependent Markov chain [Kurtz] of infinite dimensionality

Infinite system model - Ordinary differential equations

Description

- Let $x_i(t)$ be the fraction of nodes with at least i jobs at time t
- The set $\frac{d}{dt}x(t) = F(x(t))$ can be written as

$$\frac{d}{dt}x_1(t) = (\overbrace{\lambda}^{\rightarrow \text{ Arrivals } \leftarrow} + \underbrace{r x_2(t))\overbrace{(1 - x_1(t))}}_{\text{Incoming job transfers}} - \underbrace{(x_1(t) - x_2(t))}_{\text{Completions}}$$

$$\frac{d}{dt}x_i(t) = \underbrace{\lambda(x_{i-1}(t) - x_i(t))}_{\text{Arrivals}} - (\overbrace{1}^{\rightarrow \text{ Completions } \leftarrow} + \underbrace{r(1 - x_1(t)))\overbrace{(x_i(t) - x_{i+1}(t))}}_{\text{Outgoing job transfers}}$$

Universiteit Antwerpen

Infinite system model - Closed form results

Unique fixed point

- Describes cumulative queue length distribution at $t = \infty$
- $\pi = (\pi_1, \pi_2, \ldots)$ with $\sum_{i \geq 1} \pi_i < \infty$, explicitly:

$$\boxed{\pi_i = \lambda \left(\frac{\lambda}{1 + (1 - \lambda) r} \right)^{i-1}}$$

Performance

- The fixed point is a global attractor
- Proof by the Krasovskii-Lasalle principle
- The mean response time is then given by:

$$\boxed{D = 1 + \frac{\lambda}{(1 - \lambda)(1 + r)}}$$

Universiteit Antwerpen

Infinite system model - Closed form results

Load sharing strategies

$$D = 1 + \frac{\lambda}{(1-\lambda)(1+r)}$$

- Valid for Push / Pull / Hybrid
- Difference lies in the generated overall probe rate R:

$$R = (1-\lambda)r_1 + \frac{r_2\lambda^2}{1+(1-\lambda)r}$$

- Using $r = r_1 + r_2$ for the Hybrid strategy
- Setting $(r_1, r_2) = (r, 0)$ and $(0, r)$ results in R_{pull} and R_{push} respectively

Universiteit Antwerpen

Infinite system model - Closed form results

Mean response time

- Hybrid strategy is always inferior
- Using the relationship R, rewrite:

$$\boxed{\begin{aligned} D_{push} &= \frac{\lambda}{(1-\lambda)(\lambda+R)}, && \text{for } R < \lambda^2/(1-\lambda) \\ D_{push} &= 1, && \text{for } R \geq \lambda^2/(1-\lambda) \\ D_{pull} &= \frac{1+R}{1-\lambda+R} \end{aligned}}$$

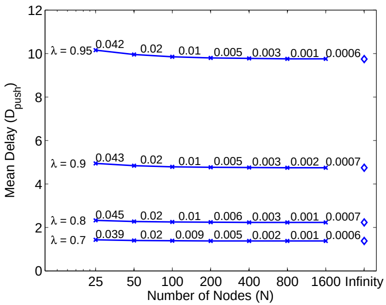
- Resulting in $D_{push} < D_{pull}$ if $\lambda < \frac{\sqrt{(1+R)^2+4(1+R)}-(1+R)}{2}$
- If $R \geq 0$, $\quad D_{push} < D_{pull}$ if $\lambda < \phi - 1$ with $\phi = (1+\sqrt{5})/2$

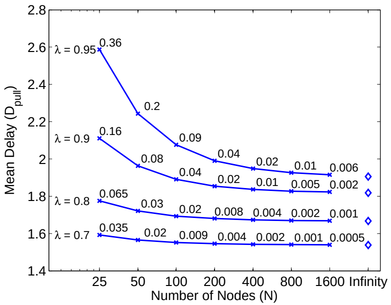
Universiteit Antwerpen

Finite system simulations

Finite vs. Infinite

- Errors are proportional to both load (λ) and probe rate (r)
- Using a push strategy r decreases with λ,
  $\rightarrow$ nearly load insensitive error
- Using a pull strategy r increases with λ,
  $\rightarrow$ larger error under high load
- Infinite model is accurate for large N :

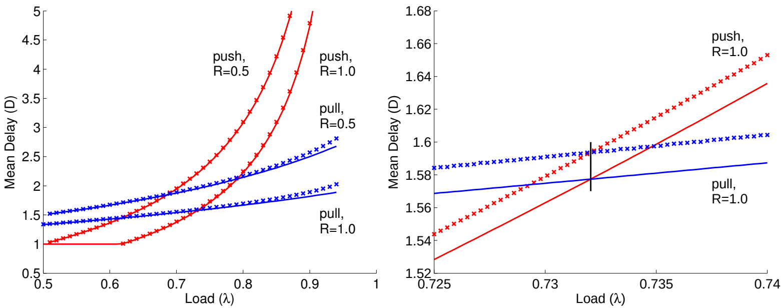




Finite system simulations

Finite vs. Infinite

- Simulation results using $N = 100$ (crosses) vs. infinite model:



- Infinite model accurately predicts λ where both strategies perform equally well, even for systems of moderate size

Universiteit Antwerpen

Conclusion

Push / Pull / Hybrid strategies

- Push outperforms pull (for $N = \infty$) if and only if:

$$\lambda < \frac{\sqrt{(1+R)^2+4(1+R)}-(1+R)}{2}$$

- Hybrid strategy is always inferior to pure push or pull strategy (proof in paper)

Finite vs. Infinite

- Infinite model predicts finite model accurately
- Technical issues to formally prove the convergence of the steady state measures of the finite system model to the infinite system model were identified
(see paper for details)

Universiteit Antwerpen