On the Throughput Optimization in Large-Scale Batch-Processing Systems

Sounak Kar[1], Robin Rehrmann[2], Arpan Mukhopadhyay[3], Bastian Alt[1], Florin Ciucu[3], Heinz Koeppl[1], Carsten Binnig[1], Amr Rizk[4]

Partly funded by Collaborative Research Center 1053 MAKI of German Research Foundation

What is this talk about?

- We analyze a data-processing system with n clients (job producer) and m parallel servers serving jobs in batches.
- Seek to maximize system throughput Θ which critically depends on batch size k.
- Numerical search for optimal batch size k^* (corresponding to optimal throughput Θ^*) prohibitively expensive *and* standard/naive CTMC analysis takes $\omega\left(n^4\right)$ time.
- We provide a mean-field model for calculating k^* in $O(1)$ time.
  - Findings *validated* in a prototype of large commercial database.

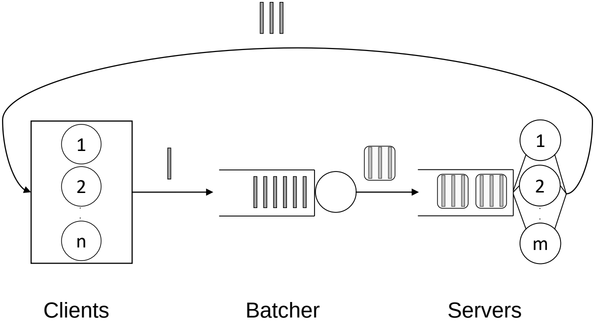
Outline

1 Background

System Description

- **Closed system**: Client becomes *active* only after receiving response to previously submitted query, i.e., total no. of jobs $= n$.
- **Service speedup**: Average batch service time $g(k)$ is a sub-additive function of batch size k.
- **Utilization**: Beyond a batch size, servers start idling yielding a non-trivial optimization problem

Flow Diagram

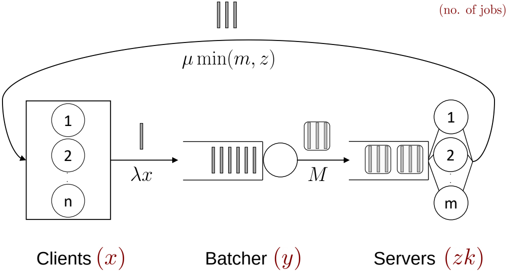


2 Optimal Batch Size Maximizing Throughput: Approaches

Exhaustive search

- Probe throughput for all possible batch size $k \in \{1, 2, \ldots, n\}$ to find k^*.
  - *Infeasible* for real systems with large number of clients.

Model Assumptions



- **Number of jobs**: x, y and zk are number of jobs at client, batching and service station, respectively. Thus, $x + y + zk = n$.
- **Exponential sleeping time**: Clients produce jobs at rate λx when x of them are *active*.

- **Exponential batching time**: The batcher produces batches of size k at rate $M\lfloor y/k \rfloor$ when there are y available jobs.
- **Exponential batch service time**: The service station consists of a single queue and m parallel servers, each having a service rate $\mu(k) = \frac{1}{g(k)}$. Usually, $M >> \mu$.
  - Overall *batch* service rate is $\mu \min(m, z)$ when z batches are available.
- **Speedup forms**: Speedup has either of the following sub-additive forms: linear, logarithmic, power.

Speedup Assumptions: Explanations

- Speedup influences throughput Θ but estimating average service time $g(k), \forall k$ is expensive.
- Assuming convenient forms lets us estimate parameters of $g(k)$ efficiently.
  - Choose batch sizes to probe given a fixed budget (e.g., 5%).
  - Derive OLS estimates for parameters of speedup form.
  - Speedup form with least error picked as estimate.

Approaches under Model Assumptions

Standard CTMC Analysis:

- Derive steady-state distribution of the CTMC.
- Calculate corresponding throughput $\forall k$.
- Find optimal batch size k^*.

Mean-field Analysis:

- Take no. of jobs $n \to \infty$ and no. of servers $m \to \infty$ s.t. $m/n \to \alpha$.
- Calculate steady state throughput as function of k.
- Find optimal batch size k^*.

3 Standard CTMC Analysis: Details

Steps

- Estimate model parameters to populate intensity matrix $\mathbf{Q}(k)$ of the CTMC.
- Obtain steady state distribution $\boldsymbol{\pi}$ by solving $\boldsymbol{\pi} \cdot \mathbf{Q} = 0$.
- Expected steady state throughput obtained as

$$\mathbf{E}[\Theta(k)] = \sum_{(x,y,zk)} \boldsymbol{\pi}(x, y, zk)\, k\, \mu(k) \min(m, z).$$

state prob. batch size state throughput

- Find optimal batch size $k^* = \mathrm{argmax}_k\, \mathbf{E}[\Theta(k)]$.

Issues

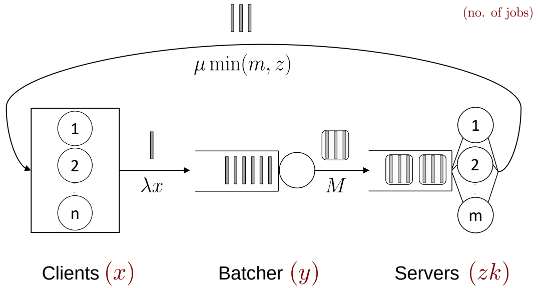
- Intensity matrix $\mathbf{Q}$ has non-linear rates implying $\boldsymbol{\pi} \cdot \mathbf{Q} = 0$ cannot be solved analytically.
- Numerical solution takes $\omega\left(n^4\right)$ time, n being number of clients.
- **Estimate** of k^* **matches closely with** the findings in the commercial database **system**, as we will see later.

4 Mean-field Analysis

Additional Assumption

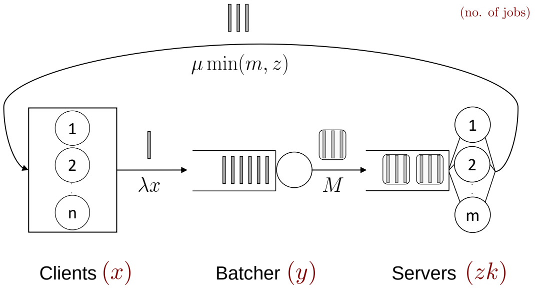
- Batching step is *instantaneous*.
  - Batching station has $(k-1)$ jobs $\implies$ Upon arrival of a new job, a batch is forwarded to the service station immediately.
  - *Realistic* as the batching step is ~ 50 times faster than the service step in the considered system.

Implication



- $z = \lfloor \frac{n-x}{k} \rfloor$ and $y = n - x - z$.
- $x \leftrightarrow (x, y, zk)$, i.e., the state is adequately represented by number of active clients x.

Steady State Dynamics: Client Station



- Expected job input rate $= k\mu(k)\mathbf{E}\left[\min\left(m, \left\lfloor \frac{n-X}{k} \right\rfloor\right)\right]$.
- Expected job output rate $= \lambda \mathbf{E}[X]$.

Steady State Dynamics: Client Station

Under stationarity,

$$\lambda \mathbf{E}[X] = k\mu(k)\mathbf{E}\left[\min\left(m, \lfloor \frac{n-X}{k} \rfloor\right)\right], \quad (= \mathbf{E}[\Theta])$$

$$\implies \lambda \mathbf{E}[X] \leq k\mu(k) \min\left(m, \lfloor \frac{n-\mathbf{E}[X]}{k} \rfloor\right), \quad (Jensen's\ inequality)$$

$$\implies \frac{\lambda \mathbf{E}[X]}{n} \leq \min\left(\frac{m}{n}k\mu(k), \frac{\lambda\mu(k)}{\lambda+\mu(k)}\right).$$

Now, LHS = Expected *relative* steady state throughput $\mathbf{E}[\Theta^{(n)}/n]$ and the bound is *asymptotically tight* when $m/n \to \alpha \in \mathbb{R}_+$ **as** $n \to \infty$.

Back to Optimal Throughput (*or* Batch Size)

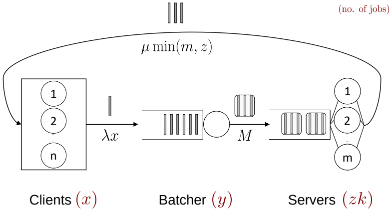
Optimal relative throughput

$$\mathbf{E}\left[\frac{\Theta^*}{n}\right] \to \max_k \min\left(\alpha k\mu(k), \frac{\lambda\mu(k)}{\lambda + \mu(k)}\right).$$

i.e., optimal batch size

$$k^* = \operatorname*{argmax}_k \min\left(\alpha k\mu(k), \frac{\lambda\mu(k)}{\lambda + \mu(k)}\right).$$

Main Result: Asymptotic Tightness of the Bound



The fraction of active clients $w^{(n)}(t) = X^{(n)}(t)/n, t \geq 0$ has jump rates

$$q^{(n)}(w \to w - 1/n) = nw\lambda,$$
$$q^{(n)}(w \to w + k/n) = n\mu(k) \min\left(\alpha, \frac{1}{n}\lfloor\frac{n - nw}{k}\rfloor\right), w = \frac{x}{n}. \qquad (4.1)$$

Theorem 1. *(i) If* $w^{(n)}(0) \to w_0 \in [0,1]$ *as* $n \to \infty$ *in probability, then we have*

$$\sup_{0 \le t \le T} \left\| w^{(n)}(t) - w(t) \right\| \to 0$$

in probability as $n \to \infty$*, where* $w(t)$ *is the unique solution of the following ODE:*

$$\dot{w}(t) = f(w(t)), \quad w(0) = w_0, \quad \textit{with}$$

$$f(w) = k\mu(k) \min\left(\alpha, \frac{1-w}{k}\right) - \lambda w. \quad \textit{(total drift from 4.1)}$$

(ii) *For any* $w_0 \in [0,1]$, *we have* $w(t) \to w^*$ *as* $t \to \infty$, *where*

$$w^* = \min\left(\frac{\mu(k)}{\lambda + \mu(k)}, \frac{\alpha k \mu(k)}{\lambda}\right). \quad \textit{(unique solution of } f(w) = 0\textit{)}$$

(iii) *The sequence of stationary measures* $\pi_w^{(n)}$ *of the process* $(w^{(n)}(t), t \geq 0)$ *converges weakly to* δ_{w^*} *(Dirac delta) as* $n \to \infty$.

Proof Idea

- (i) The limiting drift of w is given by f which is Lipschitz continuous implying convergence in probability by Kurtz's theorem [1].
- (ii) One can bound $(w(t) - w^*)$ and show that it is non-increasing in t. Thus w^* is *globally attractive*.
- (iii) Observe that $\pi_w^{(n)}$ is tight as it is defined on the compact interval $[0, 1]$.

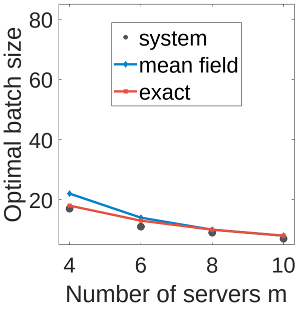
5 Experiments

Validation through a Prototype in a Commercial Database*

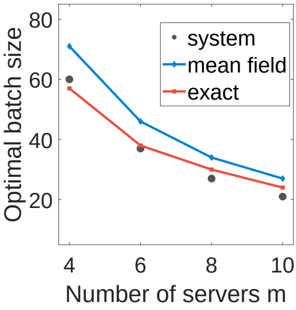
- Query rate estimated from observations.
- For a fixed probing budget, batch sizes are chosen s.t. variance of the estimate is minimized. (D-optimal design)
  - E.g., when $n = 100$ and one can probe 10 batch sizes, $\{1, 2, \ldots, 5, 96, 97, \ldots, 100\}$ should be chosen.
  - The speedup form yielding minimum error is chosen.

*SAP HANA

Results



(a) $n = 100$



(b) $n = 300$

system $\equiv$ prototype, exact $\equiv$ naive CTMC approach, n = number of clients

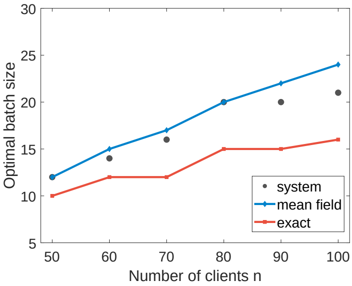
6 Multiple Job Types: Preemptive Priority

Further Results

- We prove similar results for **two** job types (e.g., *read* and *write* in databases).
  - One type is assumed to have preemptive priority over the other.
  - **Batch size** for different types can possibly be **different**.
- For **equal batch sizes**, the result was proved for **any** number of types.

Experiments

A similar experiment was done in a large commercial database where *write* jobs had non-preemptive* priority over *read* jobs. (4 servers.)



*due to system constraints, we have seen equivalence of both priorities in simulation

Summary

- We analyze a closed batch-processing system with the objective of maximizing throughput.
- Despite convenient assumptions, naive CTMC approach determines optimal batch size k^* with considerable precision. (takes $\omega(n^4)$ time)
- Mean-field approach provides a close match for k^* in $O(1)$ time.
- We also establish similar results for multiple job types under certain constraints.

Bibliography

[1] T. G. Kurtz. 1970. Solutions of Ordinary Differential Equations as Limits of Pure Jump Markov Processes. Journal of Applied Probability7, 1 (1970), 49–58.