

初期化処理を含む待ち行列を用いた計算機リソースサイジング Computing Resource Sizing Method Using Queue Model with Initialization Process

佐藤 尚也[†] 小杉 優[†] 魚住 光成[†] 鶴 薫[†]
Naoya Sato Yu Kosugi Mitsunari Uozumi Kaoru Tsuru

1. はじめに

システム構築において、ハードウェアの構成、能力といった計算機リソースを決定するためのサイジングはオンプレミスもしくはクラウドのいずれの環境においても、重要な項目である。計算機リソースのサイジングには、単体の性能を元に要件として求められる性能を整数倍にして求め、これに従って計算機リソースの充分性を確認する方法が考えられるが、実際にはリソース競合の影響により、単純な整数倍とならず、調達したリソースが不足、または過剰となることがある。

サイジングの誤りはコストの増大に繋がるため、適切なサイジングを行うための性能予測は重要である[1]。しかし、性能予測に用いられる既存の待ち行列モデル[2]は、処理が連続して行われない時に発生する初期化処理を考慮しておらず、予測結果と実測値で大きく異なる場合がある。

本稿では、初期化処理を含んだ待ち行列モデルを提案してモデル式化を行い、計算機リソースのサイジング精度向上を目的とする。

2. 先行技術

大規模なシステムは、サーバへのリクエストの到着はランダム性があると考えられることから、待ち行列モデル(図1)を利用してサーバに求められる処理能力を推定することが適当である。

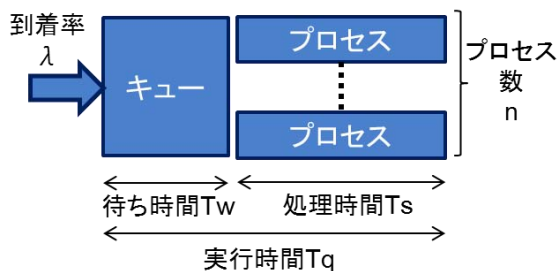


図1 待ち行列モデル

リクエストの到着から出力までの実行時間 Tq は、キューの待ち時間を Tw 、プロセスの処理時間を Ts とすると、式(1)で表せる。

$$Tq = Tw + Ts \quad \dots \text{式(1)}$$

待ち時間 Tw は、同時に処理できる最大プロセス数(=窓口数)を n 、リクエストの単位時間当たりの到着数(=到着率)を λ 、全窓口が塞がる確率を $P(n, \lambda, Ts)$ とすると、式(2)で表せる。

$$Tw(n, \lambda, Ts) = \frac{P(n, \lambda, Ts) * Ts}{n(1 - \frac{\lambda * Ts}{n})} \quad \dots \text{式(2)}$$

式(1), (2)より、実行時間 Tq は窓口数 n 、到着率 λ 、処理時間 Ts を用いれば算出が可能であり、性能予測に使用することができる。

3. 対象システム

プログラムによっては処理時間 Ts だけでなく、初期化の為に時間 Ti を要し、連続した処理の場合に初期化処理を省略する実装が行われる。例として、最初にリクエストが到着した際にリソースのオープン処理を行い、連続して処理が実行されない場合にリソースのクローズ処理を行う実装が挙げられる。

本稿にてサイジングを行う対象のシステムは、リクエスト到着時に窓口空きがある場合に初期化処理を実行し、いずれの窓口にも空きがない場合に初期化処理を省略する動作を行うものとする(図2)。

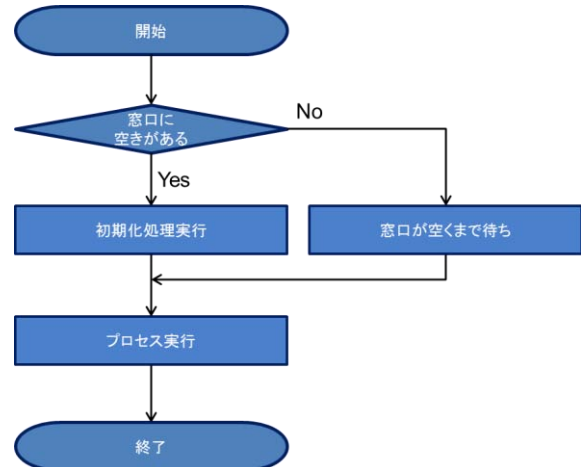


図2 初期化処理を含む処理(フローチャート)

4. 課題

図2のように初期化処理を実行するシステムの場合、サーバへのリクエストが低トラフィックの時と高トラフィックの時で動作が大きく異なる(図3)。

[†] 三菱電機株式会社, Mitsubishi Electric Corporation

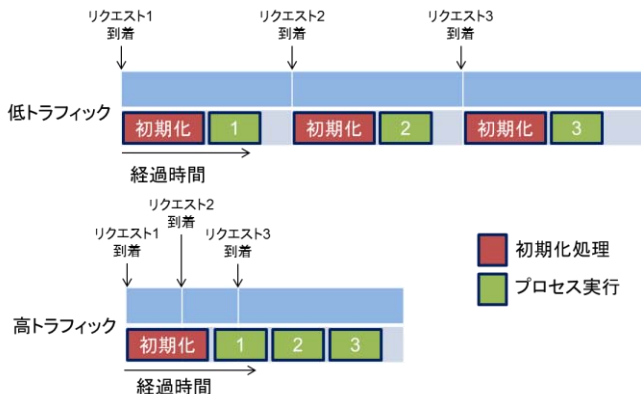


図3 初期化処理を含む処理(タイムチャート)

このような実装の場合、従来の待ち行列モデルでは初期化処理が考慮されていないため、初期化処理が頻繁に発生する低トラフィック時の実行時間 Tq が実測値と比較して大きく乖離し、サイジングの精度が低下する課題が挙げられる。

本稿では、新たに初期化処理を含む待ち行列モデルを考案し、初期化処理を含む待ち行列モデルに基づくモデル式によって課題の解決を図る。

5. 解決策

いずれかの窓口が塞がっていない時に初期化処理を実行するモデルについて考える(図4)。

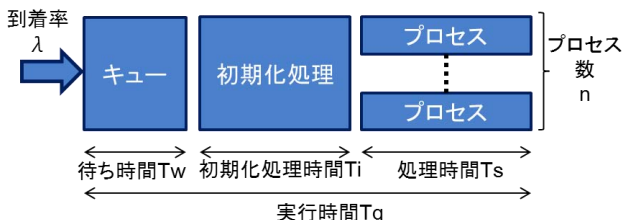


図4 初期化処理を含む待ち行列モデル

リクエストの到着から出力までの実行時間 Tq は、キューの待ち時間を Tw 、初期化処理時間を Ti 、プロセスの処理時間を Ts とすると、式(3)で表せる。

$$Tq = Tw + Ti + Ts \quad \dots \text{式(3)}$$

初期化処理時間 Ti は、窓口数を n 、リクエストの到着率を λ 、全窓口が塞がる確率を $P(n, \lambda, Ts)$ 、平均初期化処理時間を Ini とすると、式(4)で表せる。

$$Ti = (1 - P(n, \lambda, Ts)) * Ini \quad \dots \text{式(4)}$$

式(3), (4)より、初期化処理を含む待ち行列モデルに基づいたモデル式では、窓口数 n 、処理時間 Ts 、平均初期化処理時間 Ini を仮定することで、到着率 λ に対する実行時間 Tq を算出することができる。

6. 評価

前章で提案した初期化処理を含む待ち行列モデルに基づいたモデル式の妥当性を検証するため、初期化処理を含む処理を模擬したサンプルアプリケーションの実行時間と、モデルによる理論値との比較を行った。

例として窓口数1、平均処理時間0.1秒、平均初期化処理時間1.0秒とした場合の各モデル式との比較を行った(図5)。それぞれの到着率 λ と実行時間 Tq をプロットした。図5中の点がそれぞれにあたる。

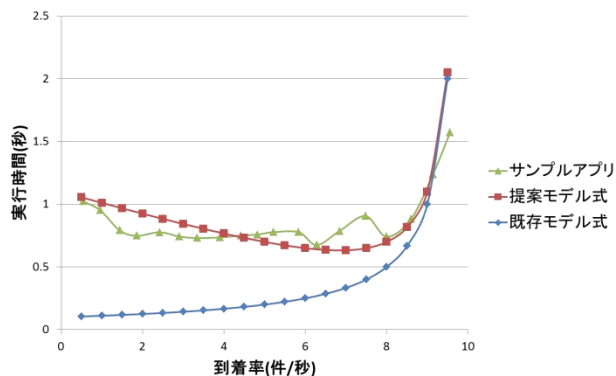


図5 評価結果

図5より、提案モデル式の実行時間は、既存モデル式のものとは比べてトラフィック全域でサンプルアプリケーションの実行時間に近似していることが確認できる。

このモデル式を利用することで、小規模な構成における計測から適当な窓口数 n 、処理時間 Ts 、平均初期化処理時間 Ini を決定して実行時間 Tq を試算することができ、より適切な計算機リソースのサイジングに利用することができる。

7. おわりに

大規模システムにおけるサイジングは、オンプレミス環境、クラウド環境いずれに対するシステム構築であっても大きな課題である。

本稿では、初期化処理を含むプログラムにおいて、新たな待ち行列モデルに基づいて実行時間を数式化して試算する方法を提示した。

サンプルアプリケーションを使ってモデル式を評価した結果、実行時間が概ね近似していることが確認された。本手法は、既存の待ち行列モデルに基づいたモデル式と比較して性能予測の精度向上に有用であるといえる。

今後は、初期化処理の異なる発生パターンを考慮したモデル式についても評価を進め、サイジングへの適用を検討していく。

参考文献

[1] 市原利浩ら、予備系システムのダウンサイジング手法及び評価、FIT2014 (第13回情報科学技術フォーラム), 2014
 [2] Kendall,D.G.: Some Problems in the Theory of Queues, J.Roy.Statist.Soc.,Ser.B, vol.13,no.2,pp.151.158,1951