

計測値に基づいた性能モデルの誤り検出

木村 大地[†] 河津 正人[†] 矢野尾 一男[†]NEC 情報・ナレッジ研究所[†]

1. はじめに

ソフトウェアや情報処理システムが期待通りの性能を出せるかどうかを評価することは、設計プロセスの効率化・高品質化を実現するうえで重要である。

ソフトウェアなどの性能を予測するには、待ち行列やペトリネットを用いた性能モデルの作成が有効な手段の一つである[1]。このような性能モデルでは、各スレッドの CPU 使用率などの複数のパラメータが含まれており、それらは、理論値や低負荷の場合の計測情報から導出される場合が多い。しかしながら、そのようなパラメータを用いた場合の性能モデルの予測する値が、実際に計測した値からずれることはしばしば起こりうる。例えば、CPU 使用率が負荷に依存して変化する場合（待ち行列理論で言えば、平均サービス時間が到着率に依存する場合）には、当然、低負荷の場合の計測情報から得たパラメータは、高負荷の場合の予測には適用できないだろう。このような場合には、計測値をもとにパラメータを調整することで、より実態に即した性能予測を行う必要がある。調整すべきパラメータが明らかであれば、カルマンフィルタなどによって調整する方法が提案されている[2]。しかし、複数のパラメータの中で、本当にそのパラメータを調整するのが妥当か、という点については別に判断基準が必要である。妥当ではないパラメータを調整しても予測値を計測値に合わせることはできるが、これは性能モデルの再利用に有害である。妥当ではない性能モデルを再利用すると、また誤った結果を生む。しかし、調整すべきパラメータ、すなわち、誤りを含むパラメータがわかれば、他のパラメータについては再利用しても問題ないということが程度保証されるだろう。

本稿では、どのパラメータを調整するのが妥当か、すなわち、どのパラメータに誤りが含ま

れているかを、計測値に基づいて検出する方法を提案する。

2. 提案手法

提案手法を以下の図 1 を参照しながら説明する。

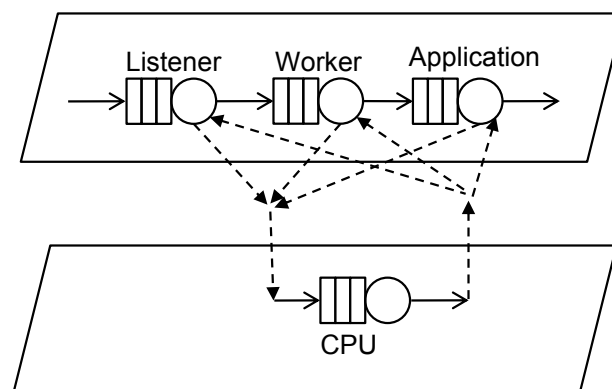


図 1 階層型待ち行列を用いた性能モデル

図 1 は、標準的な Web アプリケーションサーバについての、階層型待ち行列を用いた性能モデルを示す。この Web アプリケーションサーバは 3 つのスレッドから構成されており、それぞれ”Listener”, ”Worker”, ”Application”と呼ぶ。図 1 の上の層の待ち行列は各スレッドに対応し、下の層の待ち行列は CPU に対応する。

この性能モデルが含むパラメータは、各スレッドの平均サービス時間であり、そのうちのいずれに誤りが含まれているかを次のように推定する。まず、TAT、および、CPU 使用率の平均値について、性能モデルが予測する値と、計測値が合うように、どれか 1 つのパラメータだけを調整する。調整後、TAT の度数分布を、性能モデルが予測する値と、計測値を比較し、距離を算出する。調整した結果、度数分布の距離が最も小さくなるようなパラメータに誤りが含まれていたとみなす。

上記のように推定する理由は、待ち行列網の形が異なれば、TAT の平均値が等しくても、度数分布の形が一般には異なるということに基づいている。すなわち、平均値を合わせるように妥当なパラメータを調整すれば、調整の対象ではない度数分布もおのずと合うことが期待され

Performance modeling error detection based on measurements

[†]Daichi Kimura, Masato Kawatsu, Kazuo Yanoo

[†]Knowledge Discovery Research Laboratories,

NEC Corporation

る。他方で、妥当ではないパラメータを平均値が合うように調整しても、度数分布は合わないことが期待される。

3. 検証

提案手法を検証するために、図 1 に示す性能モデルを用いて数値シミュレーションを行った。具体的には、正解として定義した各スレッドの平均サービス時間をもとにシミュレーションを行った結果（CPU 使用率の平均値、TAT の平均値・度数分布）を正解データとして用意した。各スレッドの平均サービス時間の初期値として、表 1 に示すように「誤り」を含んだ値を与えた。

表 1 各スレッドの平均サービス時間

スレッド	初期値	正解
Listener	0.00005	0.00005
Worker	0.0005	0.0005
Application	0.0025	0.0035

表 1 に示すように、初期値は Application スレッドに誤りを含むように設定した。本稿では、パラメータの調整の方法として、CPU 使用率、および、TAT の平均値について、予測値と計測値の誤差を目的関数とした滑降シンプレックス法を用いた。また、リクエストの到着は平均値が 200 のポアソン過程に、各待ち行列のサービス時間は指数分布に従うとした。

パラメータを初期値から調整した結果、および、正解データについて、CPU 使用率、および、TAT の平均値を表 2 に示す。

表 2 CPU 使用率、および、TAT の平均値

スレッド	CPU 使用率	TAT
Listener	81.4088	0.0206343
Worker	82.0301	0.0206007
Application	81.0847	0.0205757
正解	81.1583	0.0206049

例えば、Listener スレッドの行は、Listener スレッドの平均サービス時間だけを調整した結果を示している。正解データとそれぞれの調整の結果を比べて、誤差は最大でも約 2%程度になっている。図 2 に、パラメータ調整後、および、正解データの TAT の度数分布を示す。図が示すように、平均値がほぼ同等であっても、分布が異なることがわかる。各調整結果と正解データについての度数分布の距離は、Listner, Worker, Application の順に、0.03416, 0.02213, 0.009812 であり、Application の場合が最小であった。こ

のように、誤りを含んだパラメータを調整した場合には、度数分布の距離が最小になることがわかった。

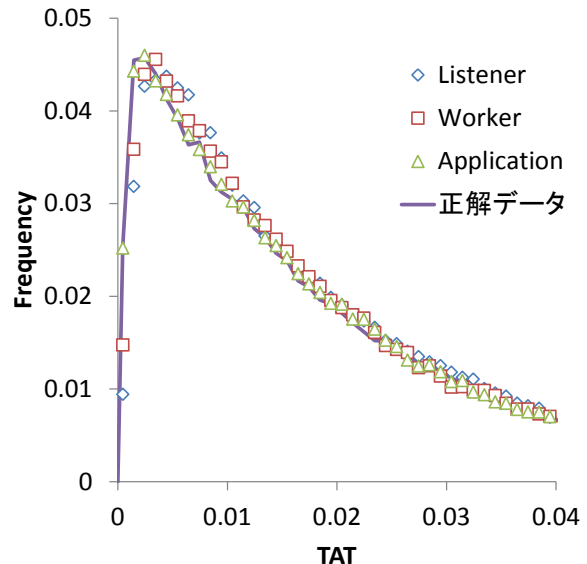


図 2 TAT の度数分布

4. まとめ

本稿では、性能モデルのどのパラメータを調整するのが妥当か、すなわち、どのパラメータに誤りが含まれているかを、計測値に基づいて検出する方法を提案し、数値シミュレーションを用いて手法の妥当性を検証した。数値シミュレーションの結果、誤りを含んだパラメータを調整した場合とそうでない場合では、CPU 使用率や TAT の平均値がほぼ同等でも、TAT の度数分布の距離が 2 倍以上異なっていた。このように度数分布の距離が明確に異なっており、本手法が性能モデルの誤り検出に有効である可能性が示唆された。今後は、図 1 に示すような各スレッドが直列につながっている場合以外の構造（確率による分岐、同期・非同期など）を含む場合にも、本手法が有効かどうかを検証していく予定である。

参考文献

[1] Daniel A. Menasce, Virgilio A.F. Almeida, Lawrence W. Dowdy, Performance by Design, Prentice Hall, U.S.A., 2004.
 [2] Murray Woodside, Tao Zheng, Marin Litoiu, "Service System Resource Management Based on a Tracked Layered Performance Model," Proc. IEEE International Conference on Autonomic Computing, pp. 175-184, June 2006.