

## IoT システムを対象とした簡易な性能評価に関する一検討 A Study on Simple Performance Evaluation for IoT System

辻尾 良太<sup>†</sup> 鶴 薫<sup>†</sup>  
Tsujiyo Ryota Tsuru Kaoru

### 1. はじめに

システム開発においては、顧客の要件に合わせて、既に開発した機能の組み合わせを変更したり、一部機能をカスタマイズして構築する場合がある。このようなシステムの開発においても、性能評価が十分にできていない場合、検証工程で大きな手戻りが発生する。また、性能見積りを実施するには一定のノウハウが必要であり、本作業自体の工数が大きくなる場合があることに加えて、IoT システムでは、収集するデータや機能を後から追加されるケースもあり、事前の性能見積りが困難である。

一方、システムの拡張性は、クラウドやハイパーコンバージドインフラストラクチャといった技術により向上しており、従来と比較して容易に拡張できる傾向にある。このため、事前に計算機リソースの大まかなスケールを見積もった後に、誤差分の計算機リソースを追加または削減することを想定できる。

本報告では、この際に実施する性能見積りを容易に行う方式に関して報告する。

### 2. 先行技術の課題

システムの動作メカニズムをモデル化するホワイトボックスモデルとシステムの実測値を用いてモデル化するブラックボックスモデルがあり、これらを組み合わせた評価手法が提案されており、各モデルに対して機械学習を用いて改善する手法も用いられている[1]-[3]。

また、リソースの使用率と負荷の関係を多項式モデルで近似することで性能見積りを行う方式も提案されている[4]。

従来の手法では、いずれにおいても性能見積りを行うためのモデルの生成にノウハウが必要であり、性能見積りの作業が属人的になるため、これを解決する必要がある。

### 3. 提案手法

本報告では、性能見積りにおけるモデル化を容易に行うための手法を提案する。図 1 に提案手法の概要を示す。本手法では、ノウハウ保有者があらかじめアプリケーション

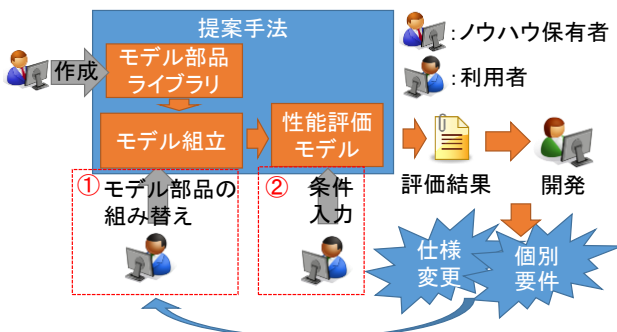


図 1 提案手法の概要

の機能ごとに性能特性をブラックボックス化したもの（以下、モデル部品）を作成し、配置しておく。利用者はモデル部品を組み立て、要求量などの条件を入力することで使用リソース量の見積りを容易に行うことを目的とする。本報告では、CPU 使用率の見積りに関して報告する。

### 3.1 実現方式

#### 3.1.1 モデル部品

モデル部品は既に開発したアプリの機能を動作させた際の測定値から作成する。アプリの機能のリソース使用率の測定値を基に式 (1) により、1 リクエスト当たりの消費リソースを導出する。この時、各機能を個別で実行している時の測定値を取得する。

$$1 \text{ リクエスト当たりの CPU 消費量} = \frac{1 \text{ コア当たりの CPU 使用率}}{\text{単位時間当たりのリクエスト数}} \dots (1)$$

#### 3.1.2 利用者が入力する情報

図 2 に利用者が入力する情報の例を示す。①はアプリケーション層の機能の流れであり、source から sink に流れるように記載する。②はシステム基盤の構成であり、サーバ名、コア数、並列数も併せて記載する。③は機能・システム基盤の紐付け情報であり、機能とシステム基盤の配置関係を表す。④は機能に対するリクエスト量であり、各機能に対する 1 分あたりのリソース量を記載する。

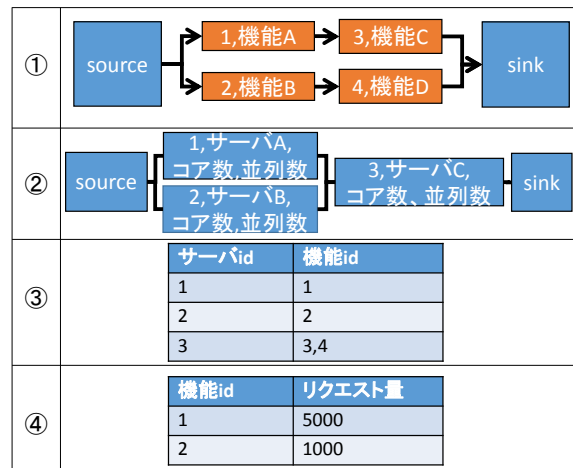


図 2 利用者が入力する情報

#### 3.1.3 性能見積りの方式

図 3 に性能見積り方式の概要を示す。3.1.1 で作成したモデル部品と 3.1.2 で入力した情報から使用率の見積りを行う。3.1.2 の内容から、各サーバで動作する機能とそれぞれの機能に対するリクエスト量を把握することができる。また、機能-システム基盤の関係を 3.1.1 のモデル部品と関連

付けることで、各サーバでの大まかなリソース消費量を求める。

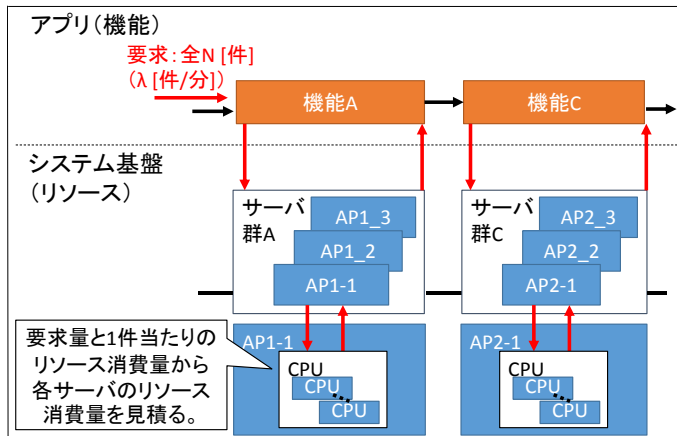


図3 性能見積り方式の概要

### 3.2 評価

センサデータ収集を行うシステムに適用し、リソース消費量の見積りの妥当性を評価する。

#### 3.2.1 評価シナリオ

評価シナリオを図4に示す。4コアのAP1サーバが10台、AP2サーバが2台、DBサーバが1台配置されており、AP1サーバでは、機能1、機能2が、AP2サーバでは機能3が、DBサーバでは機能4、機能5がそれぞれ実行される。機能1の処理結果は機能4に渡され、機能2、機能3の処理結果は機能5に渡されるものとする。

また、機能1には1分あたり20万件、機能2には1分あたり100万件、機能3には1分あたり6万件のリクエストが到着するものとする。

モデル部品は、上記のシナリオを実行した際の値を用いて作成し、測定は上記のシナリオの他、機能1、機能2のリクエスト量を1/10、機能3のリクエスト量を1/20としたシナリオで比較する。

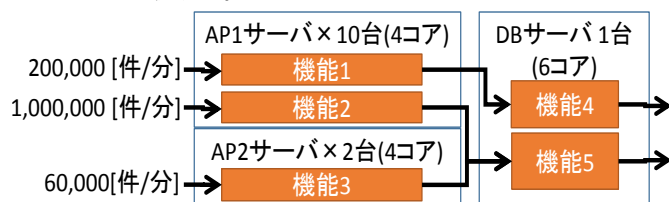


図4 評価シナリオ

#### 3.2.2 評価結果

図5に本手法により見積り値と測定値を示す。縦軸は使用率であり、横軸は利用者が入力した各機能へのリクエスト量を1とした際の比を表している。ここで、横軸が2の場合は、各機能のリクエスト量がそれぞれ2倍になったことを示している。

測定値には、測定ごとにばらつきがあるため、サーバの上限、下限を示している。

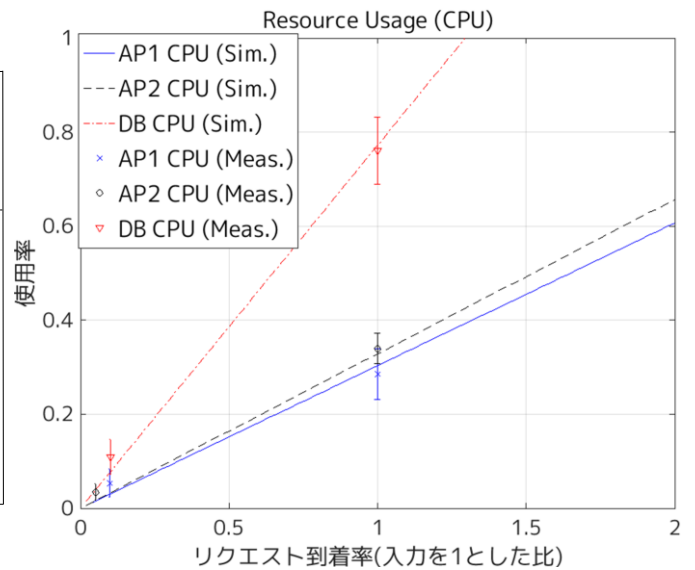


図5 見積り値と測定値

本手法によるリソース消費量の見積りは、測定値のばらつきの範囲に収まっていることが確認できる。

### 4. おわりに

本報告では、既開発した機能の組み合わせやカスタマイズしてシステムを構築する場合を想定した性能見積りの容易化を目的とした見積り手法を検討した。

ノウハウが特に必要となるシステムのモデル化作業に対して、あらかじめ作成しておいた機能部品ごとのモデルを組み合わせることでシステム全体のモデル化を行うことで、利用者は容易に消費リソース量の見積りを実施できる手法を提案した。

今後、見積り精度を向上させるため、ホワイトボックスモデルの適用を検討する。また、本方式の更なる容易性向上のためにモデル部品生成や選択を補助する方式を検討する。

#### 参考文献

- [1] 木村 大地, 榊 啓, 矢野尾 一男, “ホワイトボックスモデルとブラックボックスモデルの組み合わせによる情報処理システムの性能評価”, FIT2011 (第10回情報科学技術フォーラム), pp.589-590 (2011).
- [2] Eno Thereska, Gregory R. Ganger, “IRONModel Robust Performance Models in the Wild,” ACM SIGMETRICS Performance Evaluation Review Vol.36, No.1, pp.253-264 (2008).
- [3] Diego Didona, Francesco Quaglia, Paolo Romano, Ennio Torre, “Enhancing performance prediction robustness by combining analytical modeling and machine learning.” Proceedings of the 6th ACM/SPEC international conference on performance engineering, pp.145-156. (2015).
- [4] 市原 利浩, 原田 篤史, 樋口 毅, “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), pp.81-86, (2014).

† 三菱電機株式会社 情報技術総合研究所