TMS クラウド: ロボットタウンマネジメントのための スケーラブルな分散処理システム

TMS Cloud : Scalable Distributed Processing System for Robot Town Management

○辻 徳生 (九州大学) 表 允晳 (九州大学) 曾 昀 (北京航空航天大学) 永田 晃洋 (九州大学)
長谷川 勉 (九州大学) 倉爪 亮 (九州大学) 諸岡 健一 (九州大学) 村上 剛司 (九州産業大学)
○ Tokuo Tsuji, Yoonseok Pyo, Zeng Yun, Akihiro Nagata, Tsutomu Hasegawa,
Ryo Kurazume, Ken'ichi Morooka, Kouji Murakami

This paper presents a system for robot town management which aims a common platform where robots can work at dairy human life using distributed sensors. The developed system is scalable since data analysis of sensor signal is performed at cloud servers and analyzed data are stored at cloud servers.

1. はじめに

我々は，ロボットに搭載したセンサと環境側に分散配置したセンサ群との連携による「環境構造情報化」の研究を推進してきた (図 1)[1]．その中で，センサから得られる情報から，環境内の人，物及びロボットの状態を記録，整理，分析し，ロボットからの要求に応じて必要な情報を提供するタウンマネジメントシステム（Town Management System, 以下 TMS）を開発し，機能強化と機能検証を行ってきた．本報告では，センサ群の大規模化に対応するクラウド型のデータベースと分散処理システムの実装とその成果について述べる．

「ロボットタウンプロジェクト」では，その名称が象徴しているように，単一の部屋レベルの情報構造化ではなく，屋内外をシームレスに街のブロックレベルまで広げた情報構造化を最終的なねらいとしている．街全体に多数のセンサ群を分散配置しその情報を管理するためには，大量のデータを保持し必要に応じて提供する機構や，それらの膨大な情報を解析する機構が必要になる．ここで，仮にセンサに合わせて計算機等の端末を環境内に分散配置すると端末数が増大し維持管理が困難になる．また，センサの情報量の偏りなどにより，計算資源の使用率の低い計算機や逆にオーバフローを起こす計算機が発生するなど，資源の有効活用ができない．さらに，センサや新たな機能を追加する場合は，分散配置した各計算機で更新作業をする必要がでてくる．

そこで，タウンマネジメントシステムにおいて，サーバ群を集中管理し，必要に応じて資源を配分するクラウド型のシステムを実現する．センサの信号をネットワークを通してサーバ群に出力し，データ蓄積や解析をサーバ群で行う．サーバ群内で，データを分散して蓄積する．また，過剰な負荷がかからないようにデータ解析を分散する．本システムでは，計算資源を必要なだけ利用でき，センサの数が増え大規模化した場合でも既存のシステムを変更することなく，サーバ数を増やすだけで対応できる．新たな種類のセンサを追加する場合でも，ネットワークを通してサーバ群に出力する処理を実装すればよく，拡張も容易である．

具体的には以下の項目を開発検証した．

1. クラウド型データベース

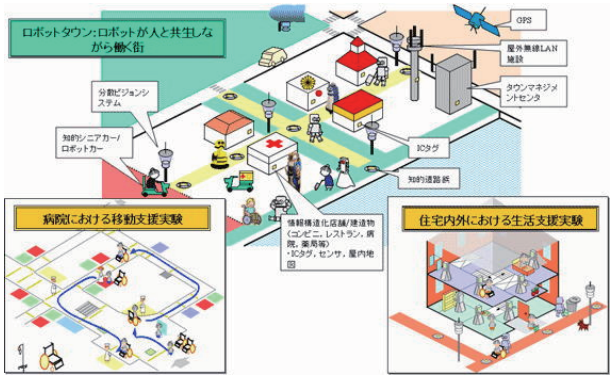

Fig. 1 ロボットタウン

2. データ解析の並列・パイプライン処理
3. 負荷自動分散システム
4. スマートフォン用のビューア

データ蓄積のクラウド化として，MySQL によるクラウド型データベースを利用する．データ解析のクラウド化として，データ解析の分散処理と負荷自動分散システムを開発する．分散処理は ROS をベースに実現する．各解析過程をパイプライン化し，ROS のメッセージによるデータ送受信を行う．これによりプロセスを分割して実行することが可能になる．さらに，負荷が集中しないように各サーバの負荷を計測し自動的に処理を割り当てる負荷分散システムを実現する．また，クラウドにより提供できるサービスの例として，スマートフォンなどの簡易端末で移動体の位置を表示するシステムを開発する．

従来のロボットに関係するクラウド化の取り組みとしては，Rsi-cloud[2] がある．クラウド内で様々なサービスを実行し，ロボット上での処理コストやトータルの実装コストを低減するための全体の枠組みを提案している．しかし，クラウド内でのセンサデータの並列分散処理やデータベースの形態までは提案されていない．そこで，我々は TMS を発展させクラウド内のセンサ群による計測データの並列解析処理と解析データの蓄積システムを設計開発する．

SY0012/12/0000 - 0633 © 2012 SICE

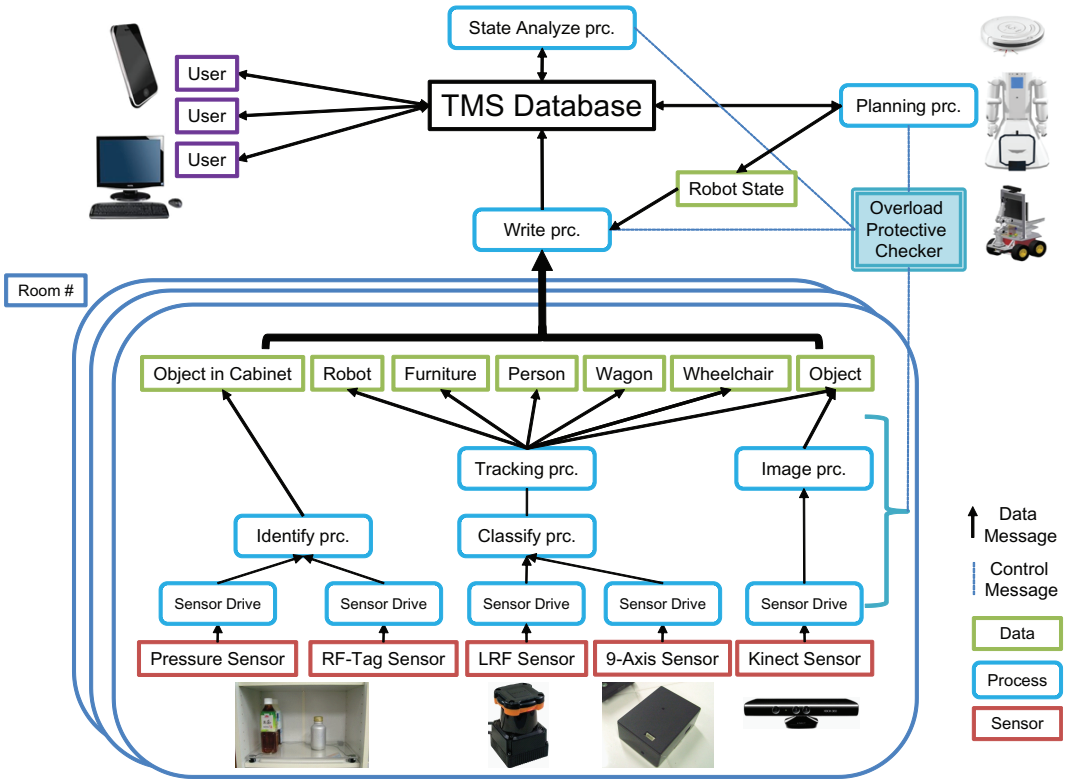

Fig. 2 システム概要

2. タウンマネジメントシステム構成

2·1 概要

TMS は，センサから得られた情報を整理・解析する処理システムと，その情報を格納しロボットやユーザからの問い合わせに応じて情報を出力するデータベースから構成される．扱う情報は，建築構造物や什器などで固定された「場」，食品や携帯電話など人が日常生活で操作あるいは消費する対象である「物」，さらに活動主体である人やロボットなどの「体」から構成される．場の情報は，地図として保管されロボットに提供される．また，物体に取り付けた電子タグにより，その物体の位置の追跡を行い，ロボットに作業情報として提供する．人やロボットの位置はレーザレンジファインダなどで計測され，実時間で解析されデータベースに格納される．それぞれ固有のデータテーブルにより定義されている．

部屋の情報を解析するシステム [3] を例として，TMS の概要を図 2 に示す．センサ群とそのデータを解析するプロセス群，データベースサーバ，人・ロボットへのインタフェースから構成される．部屋の数が多くなるとモジュールの数が大規模化することがわかる．

分散配置されたセンサの情報を処理する各種モジュールと TMS 間で通信を行うために，ROS(Robot Operating System)[4] を利用する．ROS では，各種プロセスはノードと呼ばれ，複数のプロセス間で通信を行うメッセージ機能を提供している．このメッセージ機能を利用し並列処理を実現する．

2·2 クラウド型データベース

データベースは，データを保存する機能とネットワークを介して簡便にアクセスするためのインタフェース用 API（Application Program Interface）ライブラリから構成されている．具体的には，データベースには MySQL を，インタフェースの通信方式には SOAP（Simple Object Access Protocol）をベースとする Web サービスを利用して実装されている．その SQL による入出力をラッピングし，ロボットからの問い合わせに応じてデータベースから情報を返すリクエスト／リプライ形式と，問い合わせによらずデータベースからロボットへ情報を送信する割り込み形式の両方を実装している．割り込み形式はイベントの発生をロボットに通知する場合などに用いられる．それぞれの実装形態は API の中に隠蔽されるため，ロボット側でこれらを考慮する必要はない．

API は，C++のライブラリ，OpenRTM のポート，ROS のメッセージとしてユーザに提供されている．OpenRTM のモジュールに対しては，サービスポートによりデータの入出力ができ，イベントはデータポートにより出力され，接続されたモジュールに通知される．さらに ROS のメッセージによる入出力に対応している．ROS に関する詳細な仕様は次節にて説明する．

センサや追跡する対象物の数が多くなれば，それにともないデータ量が増加する．人や物体の追跡履歴も，時間が経つにつれてデータ量が増加する．これらのデータを圧縮する取り組みも行っているが，限界がある．

そこで，クラウド型サーバ群によるデータベースを利用する．MySQL は標準でクラウド機能をサポートしており，サーバ群によるデータ管理を行える．また，既存の商用のクラウドデータベースサービスも利用できる．

2·3 ROS によるデータ解析の分散処理

ROS ノードにより構成されるシステムの概要について説明する．分散配置されたセンサから取得された生データは，Sensor

drive によって ROS のメッセージとして配信され，購読を希望するデータ解析用ノードにそれぞれ送信される．そこで解析された情報は，ROS を通じデータ保存用ノード (Write prc.) に送信される．データ保存用ノードは送り元と送信されるデータの形式から内容を分析し，データ到着時刻等必要な情報を付加後，MySQL の API を利用してデータベース上の適切なテーブルに保存処理する．

従来の我々のシステムでは，センサから出力された情報の解析を，そのセンサが接続されている計算機上で 1 つのみのプロセスで実行していた．そのため，解析処理が重くなると，リアルタイム性や計測データが失われる場合があった．これを変更し，センサの出力を ROS のメッセージとして配信することとした．その結果，複数のプロセスがデータを受信することができ，処理を並列化することが可能になる．また，逆に複数のセンサの情報を 1 つのプロセスが受け取る事も容易になる．さらに，プロセスをステップ毎に分割し，パイプライン化することも可能である．このように，センサからの情報の取得と処理の組み合わせを自由に変更でき，またプロセスを分割することが可能になる．加えて，新しいセンサや解析処理も，ROS のノードとして追加すればよいため，実装が容易になる．

図 2 は，室内におけるセンサ群とノード群の構成例である．圧力センサ (Pressure sensor) とタグリーダ (RF-Tag Sensor) は，収納庫などに設置され，収納庫内の物品の位置と種類の情報を取得し，その情報をデータ保存用ノードを通してデータベースに保存する．レーザレンジファインダ (LRF Sensor) は，床面上に設置され，その出力を識別ノード (Classify proc.) が受け取って識別結果を出力する．その識別結果を追跡ノード (Tracking proc.) に与える事により，ロボット，人や物品の移動を推定する．9 軸加速度・角加速度・地磁気センサ (9-axis sensor) は，ワゴンや車椅子などに移動する家具に取り付けて，レーザレンジファインダの情報と統合し，それらの位置を計測する．Kinect sensor は，移動ロボットに取り付けられており，物体の識別と位置計測に利用する．ロボットは，プランニングノードにより計画された動作を実行し，ユーザにサービスを提供する．

例えば，床面上に設置されたレーザレンジファインダの情報の解析はパイプライン化され，識別処理，追跡処理ノードが順次実行される．このような並列化・パイプライン化によりリアルタイムで多数の処理を実行できる．

2·4　負荷分散システム

ROS を用いてプロセスを分割することにより，並列化パイプライン化が可能になる．しかし，分割されたそれぞれのプロセスをどのサーバで実行するかを自動で管理する必要がある．なぜなら，それらのプロセスは数が多くなると手動で管理していくのは困難であり，さらに，各プロセスの処理時間は関連するセンサの測定状況で動的に変化するからである．そこで，システムを安定化するためには，自動負荷分散が必要になる．

CPU の使用量を監視し，必要に応じて処理をクラウド内のサーバ間で自動的に分散させる (図 3)．ROS のノードとして実装され，マスタノードとスレーブノードが存在する．スレーブノードはクラウド内の各サーバで実行している．その役割は，各サーバの CPU 使用率を監視しマスタノードに報告することと，マスタノードからの指令を受けプロセスを起動したり停止したりすることである．マスタノードはシステム全体で一つのインスタンスだけで実行する．役割はスレーブノードからもらったクラウド内の各計算機の状態を分析し，スレーブノードに指令

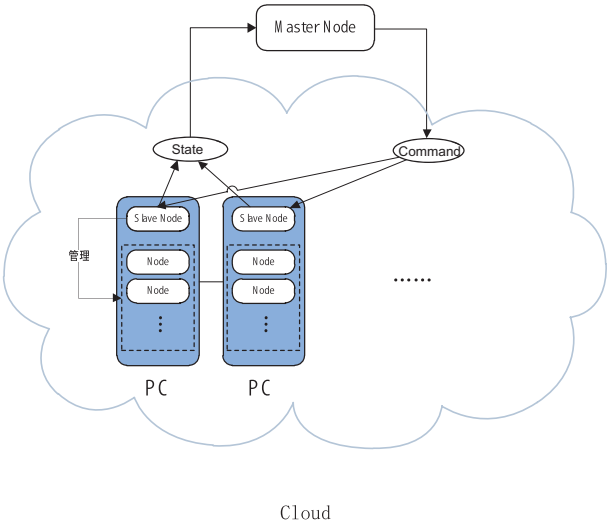

Fig. 3 負荷分散システム

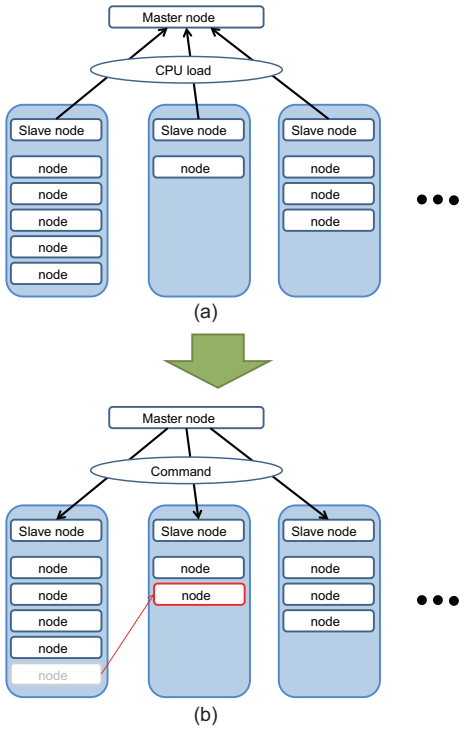

Fig. 4 負荷分散システムの動作

し，CPU の制限率が制限を超えた計算機のタスクを低負荷の計算機に渡すことである．

例えば，図 4(a) のように一台の計算機で多くのノードが実行されており負荷が指定値を超えた場合に，(b) のようにノードの実行計算機を切り替える．

TMS では，センサの数が増大すると多くの解析処理が必要になる．このように負荷を自動的に分散することにより，計算資源を有効に活用できる．

2·5　スマートフォンビューア

計算機のクラウド化がすすめば，ユーザ側で解析するための計算機を保持する必要はない．しかし，施設内の患者の見守り

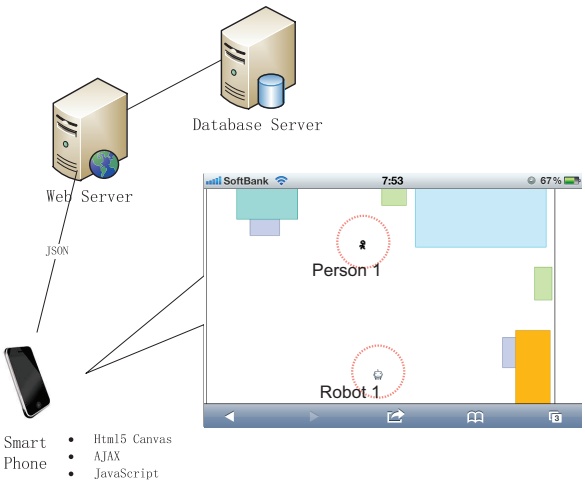

Fig. 5 端末画面

等ではデータベースの内容を閲覧したいという需要がある．そこで，web ベースのビューアを作成し、スマートフォンなどの簡易な端末で部屋内の情報を表示するシステムを構築した。

部屋内の情報を直観的に理解するためには、人やロボットの位置を可視化することは有効である．図 5 に示す通り，部屋内の固定されている家具等の配置は，TMS 内の情報から予め背景画像に変換し表示する．人とロボットの位置座標をデータベースから取得し，アイコンによって表示する．

ブラウザとサーバー間の通信は，データ量も小さくネットワークの負荷も小さい JSON (JavaScript Object Notation) を使用する．ブラウザは，Ajax(Asynchronous JavaScript + XML) と html5 の canvas 機能を使い，図を動的に更新する．現在普及しているスマートフォンに内蔵しているブラウザーは，殆ど JavaScrpit と html5 をサポートしているため，一般的なスマートフォンで動作する．さらに，web ページの形で表示するためアプリのインストールも不要であるという特徴がある．

3. まとめ

センサデータの蓄積と解析処理をサーバ群で実行し，センサ群の大規模化に対応できるクラウド型タウンマネジメントシステムを開発した．現在 USB などの端子を持つセンサの出力先の計算機は各部屋に必要であるが，イーサネット経由で通信可能なレーザレンジファインダやカメラも発売されており，今後センサ群もクラウド型のネットワークに組み込まれると想定している．

文 献

[1] 長谷川勉: “環境プラットフォーム「ロボットタウン」”, 日本ロボット学会誌, Vol. 26, No. 5, pp. 411-414, 2008.

[2] Izui, T. ; Tsuchiya, Y. ; Narita, M. ; Ueki, M. ; Murakawa, Y. ; Okabayashi, K.; “RSi-cloud for integrating Robot Services with internet services”, 37th Annual Conference on IEEE Industrial Electronics Society, pp.2158-2163, 2011.

[3] 長谷川勉, 野原康伸, 村上剛司: “生活環境における日用品情報構造化のための床面センシングシステム”, 日本ロボット学会誌, Vol.28, No.9, pp.1144-1147, 2010.

[4] “ROS の概要”, http://www.ros.org/wiki/ja/ROS/Introduction