

情報構造化環境プラットフォーム ROS-TMSとBig Sensor Boxの提案

倉 爪 亮^{*1} ピョ ユンソク^{*1} 辻 徳生^{*2} 河村 晃宏^{*1}

ROS-TMS and Big Sensor Box: Platforms for Informationally Structured Environment

Ryo Kurazume^{*1}, Pyo Yoonseok^{*1}, Tokuo Tsuji^{*2} and Akihiro Kawamura^{*1}

This paper proposes new software and hardware platforms for an informationally structured environment named ROS-TMS and Big Sensor Box. We started the development of a management system for an informationally structured environment named TMS (Town Management System) in Robot Town Project in 2005. Since then we are continuing our efforts for the improvement of the performance and the enhancement of the functions of the TMS. Recently, we launched a new version of TMS named ROS-TMS, which resolves some critical problems in TMS by adopting ROS (Robot Operating System) and utilizing the high scalability and a plenty of resources of ROS. In this paper, we firstly discuss about the requirement and the structure of a software platform for an informationally structured environment, and describe in detail our latest system, ROS-TMS version 4.0. Next, we show the requirement of a hardware platform for an informationally structured environment, and introduce a hardware platform named Big Sensor Box, in which a variety of sensors are embedded and service robots are operated according to the structured information under the management of the ROS-TMS. Robot service experiments including a fetch-and-give task and autonomous control of a wheelchair robot are also conducted in Big Sensor Box.

Key Words: Informationally Structured Environment, Service Robot, Cyber Physical System, Internet of Things, ROS

1. はじめに

ヒトと共生するサービスロボットを近い将来に実現する一手法として、ロボットを単体で知能化するだけでなく、ロボット周囲の環境を「知能化」する、いわゆる環境情報構造化（空間知能化）[†]が提唱されている。これはロボットが作業を行う空間に様々なセンサを配置し、空間内のモノやヒト、ロボットの位置や状態を逐次収集、蓄積、提供することで、ロボットをより導入しやすい日常生活環境を構築するものである。環境情報構造化は、東京大学のロボティックルーム [1] やインテリジェント・スペース [2], MIT MediaLab のスマートルーム [3], AILab のインテリジェントルーム [4], Georgia Tech. の Aware Home [5], 早稲田大学の Wabot House [6] など、多くの研究例があり、現在でも精力的に研究が進められている [7]~[11]。

筆者らも 2005 年より街レベルでの情報構造化環境の構築を目指すロボットタウンプロジェクトを開始し、そのソフトウェア基

盤として、TMS (Town Management System) を開発してきた。さらに近年では、TMS と汎用ロボットミドルウェアである ROS (Robot Operating System) [12] を融合した、ROS-TMS の開発を行っている。

環境に配置された多数のセンサにより、環境の情報を取得して状況を理解し、システムの動作を人の行動に適応させるアプローチは、アンビエントインテリジェンス [13] とも呼ばれる。アンビエントインテリジェンスにおけるロボットの活用の例として、EU の PF7 プロジェクト “CompanionAble project” (2008–2012) [14] では、アンビエントインテリジェンスとロボットの相乗 (synergy) 的な利用に関する研究が行われた。筆者らの研究も、アンビエントインテリジェンスやスマートホーム [15] [16], また近年ではあらゆるものをネットワーク化する IoT (Internet of Things) [17] や、仮想化空間と実空間を連携させてが実世界の諸問題を解決する CPS (Cyber Physical System) とも関連が深い。ロボットと結びつけた生活支援サービスの実現により重点が置かれている点の特徴である。さらに、個々のセンサシステムやサービスロボットではなく、それらを統合して管理する

原稿受付 2016 年 4 月 15 日

^{*1}九州大学

^{*2}金沢大学

^{*1}Kyushu University

^{*2}Kanazawa University

■ 本論文は有用性 (システム設計・構築分野) で評価されました。

[†]ここでは環境情報構造化、空間知能化とも環境の知能化が目的であると、区別していない。

マネジメントシステムの開発が最も重要なテーマであり、柔軟性、拡張性に優れたマネジメントシステムとして、ROS-TMSの開発を主目的としている点が他の研究と異なる。

本論文では、まず情報構造化環境を構築するためのハードウェア、およびソフトウェアプラットフォームの要求仕様について考察し、その要求仕様に従い、これまでに筆者らが開発してきたソフトウェアプラットフォーム「TMS」、「ROS-TMS」を紹介する。さらにソフトウェアプラットフォーム ROS-TMS を適用した情報構造化環境の実現例として、提案された要求仕様を満たすハードウェアプラットフォーム「Big Sensor Box」を紹介する。さらに、日常生活で頻繁に起こり得る軽作業である「物品取り寄せタスク」と、介護施設での利用を想定した「車いす型ロボットによる自動搬送」を想定し、開発した情報構造化環境のためのソフトウェアプラットフォーム ROS-TMS 4.0 とハードウェアプラットフォーム Big Sensor Box が、ロボットがサービスを提供するのに必要な機能・性能を有しているかを検証した結果を示す。

2. 情報構造化環境プラットフォーム

2.1 要求仕様

サービスロボットのための情報構造化環境を構築するためのハードウェア、およびソフトウェアプラットフォームの要求仕様を以下にまとめる。

ハードウェアプラットフォームの要求仕様

- (1) サービス提供に必要な環境情報を取得可能なセンサデバイスが設置されていること
- (2) サービス提供に必要なサービスロボットが存在し、利用可能であること
- (3) 収集された情報を集約し、分析、保管するための記憶装置（クラウドを含む）を有すること
- (4) センサデバイスやサービスロボット、記憶装置、ユーザを繋ぐ通信インフラを有すること

このうち、(1) はロボットの周囲環境に埋め込まれたセンサのみならず、ロボットに搭載されたセンサ、あるいはサービスを受けるユーザが保持しているセンサ（例えばウェアラブルカメラなど）も含まれる。また (3) は情報構造化環境において特徴的かつ必須の機能であり、データを集約して記録し、要求に応じて提供することで、情報の精度、確度を高めたり、すべてのセンサやロボット間を直接的に接続するのに比べて通信インフラを効率的に利用できる。また (4) は、センサと記録装置、アプリケーションソフトウェア、モノ、ロボット、ユーザをつなぐために必要な、情報構造化環境、あるいは IoT の実現には欠くことができない要件である。

ソフトウェアプラットフォームの要求仕様

- (1) すべてのセンサで必要な情報を取得するためのソフトウェア（デバイスドライバ等）を有すること
- (2) すべてのロボットの行動を制御するための制御ソフトウェア（モーションコントローラ等）を有すること
- (3) 収集されたセンサ情報から適切なロボットの行動を計画するアプリケーションソフトウェアを有すること
- (4) ユーザの要求を受け入れ、サービスを提供するためのユー

ザインタフェースを備えること

- (5) 様々なハードウェア（センサやサービスロボット、記録装置、モノ、ユーザ）やアプリケーション間で、必要な情報を深刻な遅延なく送受信できる情報ネットワーク機能を有すること

このうち、(1), (2), (3), (4) はセンサやロボット、ユーザとシステム間のインタフェースに関するものであり、(5) はセンサやロボットなどをネットワーク接続するためのソフトウェアに関するものである。

特に (5) の通信ネットワークについては、情報構造化環境が多様なセンサ、ロボット、ユーザ、アプリケーションソフトウェアで構築されていることから、以下のような柔軟性、持続性、頑強性を有する分散システムであることが望ましい。

- (6) ハードウェアやアプリケーションの追加、削除、機能の変更が、システム全体を停止することなく容易にできること
- (7) システムの一部が予期せず終了した場合でも、全体の停止にはつながらず堅牢さを有すること

従来、特に単体のロボットシステムでは、多くの場合でセンサ、制御装置、ロボット制御ソフトウェアを一体として開発し、接続するハードウェアやアプリケーションを変更する場合には、すべてのプログラムを一時停止する必要があった。これに対し、情報構造化環境では多数のセンサやロボットが稼働しているため、センサ、ロボット、その他のハードウェア、ソフトウェアはそれぞれ独立、分散して動作し、それぞれがほかの動作に極力影響を与えない仕組みが極めて重要である。すなわち、この高い柔軟性、持続性、頑強性を有する分散システムをいかに構築するかが、情報構造化環境を実現するには重要な鍵となる。これに対し、筆者らは、これらの機能を備え、かつ多くのセンサやロボットに対して (1), (2) の機能を提供している ROS の利用が、この問題に対する有効な手段であると考え、ROS を基本アーキテクチャとして開発を行った。

さらに、センサ情報を単に記録するのみならず、様々な情報を統合して状況を正確に分析し、適切なロボット動作を計画するためには、(3) の機能も重要である。すなわち、(3) には、(8) 多数のセンサ情報を統合し、状況を正確に分析する機能 (9) 分析された状況を基に、サービスロボットの機能に応じて作業を適切に計画する機能などが求められる。一方、情報構造化環境を CPS (Cyber Physical System) を具現化する場として捉え、(10) 収集された環境情報を基に、実世界を模した仮想環境 (シミュレータ) を構築し、計画に利用できることも重要な要素である。

2.2 提案するハードウェア・ソフトウェアプラットフォームの概要

本研究では、サービスロボットによる日常生活支援サービスの提供を念頭に、上記の「ハードウェアプラットフォームの要求仕様」および「ソフトウェアプラットフォームの要求仕様」をできる限り満たすプラットフォームを、現在ロボット分野で広く用いられている技術を利用して、実際に動くシステムとして構築することを目標に開発を行った。

まず「ハードウェアプラットフォームの要求仕様」を満足する

センサ・通信インフラを備えた、情報構造化環境のためのハードウェアプラットフォームとして、ロボットサービス実証実験施設 Big Sensor Box を提案する。Big Sensor Box には、光学式トラッカーや LRF (Laser Range Finder)、分散 RGB-D カメラシステムなどの様々なセンサデバイスが整備され、またヒューマノイド型サービスロボットや移動マニピュレータ型サービスロボット、車いす型ロボットなどが利用可能である。また壁や柱には有線 LAN が埋め込まれ、同時に無線 LAN も提供されている。通信量の多いセンサは有線 LAN によりセンサ情報を伝送し、サービスロボットは無線 LAN を経由してデータベースにアクセスし、必要なセンサ情報を取得できる。

一方、Big Sensor Box 上で動作し、上記の「ソフトウェアプラットフォームの要求仕様」を満足するソフトウェアプラットフォームとして、本論文では、これまでに筆者らが開発してきた TMS を拡張した、新たなソフトウェアプラットフォーム ROS-TMS 4.0 を提案する。ROS-TMS 4.0 では、次章以降で詳しく説明するように、センサやロボット間の通信にオープンソースのミドルウェアである ROS を用いている。ROS では、①市販の多くのセンサやロボットに対応したデバイスドライバや制御ソフトウェアが提供されており、これらを用いれば「ソフトウェアプラットフォームの要求仕様」の (1) (2) を比較的容易に実現できる (4.4 項, 4.8 項)。また ROS では、②ノードおよびノード間のトピック通信、サービス通信など、様々なモジュール間で情報をやり取りするための分散通信ネットワークを標準で提供しており、センサやロボットの制御ソフトウェアや、記録・分析のための処理ソフトウェアをノードとして定義し、ROS の仕様に従ってそれらの間の通信を記述することで、「ソフトウェアプラットフォームの要求仕様」の (5) (6) (7) を実現できる。この①②が同時に実現され、「ソフトウェアプラットフォームの要求仕様」の重要な事項を満足していることが、新たに開発した TMS で ROS を採用した最大の理由である。

さらに、ROS-TMS 4.0 では、「ソフトウェアプラットフォームの要求仕様」の (3) (4) に対応する、音声入力、動作入力のためのユーザインターフェース (4.2 項) や、シナリオに沿った生活支援サービスを提供するためのロボット動作計画モジュール (4.3, 4.7 項) なども、ROS を用いて独自に実装されている。これまでに実現されている生活支援サービスは、第 6 章で示すように、物品取り寄せタスクや見回りタスク、車いす制御タスクなどである。さらに「ソフトウェアプラットフォームの要求仕様」の (10) における仮想環境も、Big Sensor Box では後述するように Choreonoid や標準可視化ツール RViz、および動力学シミュレータ Gazebo により実現されている。

一方、「ソフトウェアプラットフォームの要求仕様」の (8) (9) に対応する、センサ情報の統合、分析機能や高度なロボットの動作計画は、例えばカルマンフィルタを用いた光学式トラッカーとオドメトリ情報の統合 (5.2.3 項) や分散 RGB-D カメラシステムにおける人物統合 (5.1.2 項)、行動・動作推定 (4.5 項)、異常状態やトリガー状態の検出 (4.6 項)、シナリオに沿ったサービス提供と例外処理 (4.3 項) などの基本的な処理にとどまっており、今後開発の余地が大きい。

3. TMS から ROS-TMS へ

TMS は、文部科学省科学技術振興調整費により実施された「ロボットタウンの実証的研究」(研究代表者・長谷川勉九州大学名誉教授・2005 年 11 月～2008 年 3 月) [18] において、環境情報構造化空間におけるソフトウェアプラットフォームとして開発が開始され、2006 年 5 月に TMS API ver.1 が公開された。このプロジェクトでは、カメラ、レーザ、IC タグリーダなどから収集された環境内の物品、ヒト、ロボットの情報を、環境データベースである TMS に蓄積し、ロボットからの要求に応じて提供する枠組みを提案した [19]。

具体的には、このプロジェクトでは、取り扱う情報を、分散センサシステムやロボットに搭載したセンサから取得された移動体の情報 (人間やロボットの種類、位置、速度等、「体」と定義)、被操作対象の情報 (種別や位置等、「物」と定義)、および環境構造の情報 (環境地図、IC タグ配置図等、「場」と定義) と整理した。TMS は、この「体」「物」「場」(あるいは「ヒト、モノ、ソコ」) の情報を整理、保持し、ロボットからの問い合わせに応じて情報を受け渡すデータベース、およびこのデータベースにネットワークを介して簡便にアクセスするための API (Application Program Interface) から構成されていた。API は、ロボット・TMS 間のネットワーク通信処理とデータベース操作の二つのプログラムを、関数ライブラリとして共通化したものである。本プラットフォームの利用者は、用意された種々の API 関数を用いてネットワーク経由で TMS へアクセスし、データベースに登録された「体」「物」「場」の情報を取得できる。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た [20]。

その後、2011 年には、データベースに蓄積できるデータの種類を増やすとともに、セキュリティ機能の追加など、API 群の機能を強化した TMS API v2 を公開した。また、NEDO 次世代ロボット知能化技術開発プロジェクトでは、TMS の RT-Middleware 化を行い、TMS-RTM として Web で公表している [21]。さらに、TMS のスケラビリティを高めるため、TMS をクラウドシステムで動作するように拡張した TMS-Cloud [22] を発表した。

API を用いた初期の TMS は、制御ソフトウェアのコンパイル時に API 関数をリンクすることでデータベースにアクセスする仕組みであった。このため、TMS に新たな機能の追加を行うたびに、すべてのロボットやセンサでプログラムをコンパイルする必要があり、拡張性、柔軟性に問題があった。また TMS 内部の処理プログラムはそれぞれ独立に開発され、同様な処理を行うプログラムが複数存在し、プログラムの再利用性が低かった。さらに、この TMS 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単一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。

そこで、2012 年からはオープンソースのミドルウェアである ROS を用いることで、様々なセンサや多様なロボットをより柔軟に統合できる ROS-TMS [23] の開発を開始し、2015 年 9

月には、最新バージョン ROS-TMS 4.0 を公開した [24]. この ROS-TMS には、レーザや IC カードリーダーなどのセンサ処理モジュールや、ロボットの動作プランニングモジュール、タスクスケジューラ、GUI モジュールなど、150 以上の環境情報構造化モジュールが含まれている。

ROS を使用して多種多様なセンサや処理プロセスをモジュール化することにより、処理モジュールの再利用性が高まり、効率的なシステム開発が実現できるようになった。具体的には、例えばデータベースとのやり取りを行う TMS_DB モジュール (4.1 項) の各ノード (DB_Writer, DB_Reader) は、それぞれ一つのノードがほかのほぼすべてのモジュールから繰り返し接続され、利用される。すなわち、ほかのノードの開発時には、データベースへの入出力の具体的な実装を考慮する必要がなく、データベース本体が ROS-TMS ver. 3.4.2 での MySQL から ver. 4.0 の mongodb に変更された場合でも、TMS_DB モジュール以外のノードにはまったく影響がない。また処理の分散化により、センサドライバなどの低レベル処理には Raspberry Pi などのシングルボードコンピュータや NUC (Next Unit of Computing) を、認識や計画にはデスクトップパソコンを使うなど、適切な計算機リソース配置が実現できた。

また、一連の処理をステップごとに細かく分割し、モジュール化することにより、センサ情報の取得から加工、利用、保存などの処理の組み合わせを柔軟に変更できるようになった。新しいセンサや処理プロセスも、ROS のノードとして追加すればよいので、一部の処理の追加、変更によるシステム全体への影響も少なくてすむ。加えて、世界中で公開されている多くの処理パッケージをそのまま利用することができ、必要な機能に絞った効率的なアプリケーション開発が可能となった。例えば、TMS_SD モジュール (4.4) のうち、LRF や RGB-D カメラ、Vicon モーショントラッカのパッケージは、ROS で公開されているパッケージ (urg_node, openni_camera, vicon_bridge) を利用し、RFID タグやロードセルなどの ROS ノードのみを独自に開発した。また没入感ディスプレイ (4.2 項) の RViz 用プラグイン (oculus_rviz_plugins) や、タスク実行マシン SMACH (4.3 項) も、ROS_Wiki で公開されているモジュールである。このように ROS に準拠してソフトウェアプラットフォームを開発することで、必要な機能を有するパッケージがすでに公開されているならば、それらを容易に入手、導入できる。

Fig. 1 は、室内におけるセンサ群とノード群の構成例である。圧力センサとタグリーダは、冷蔵庫や収納庫などに設置され、内部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。LRF は、床面上に設置され、その出力を識別ノード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に与えることにより、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・地磁気センサは、ワゴンや車椅子などに移動する家具に取り付けて、LRF の情報と統合し、それらの位置を計測する。RGB-D カメラ (Kinect) 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。

また、**Fig. 2** に実証実験環境、および仮想化された実証実験

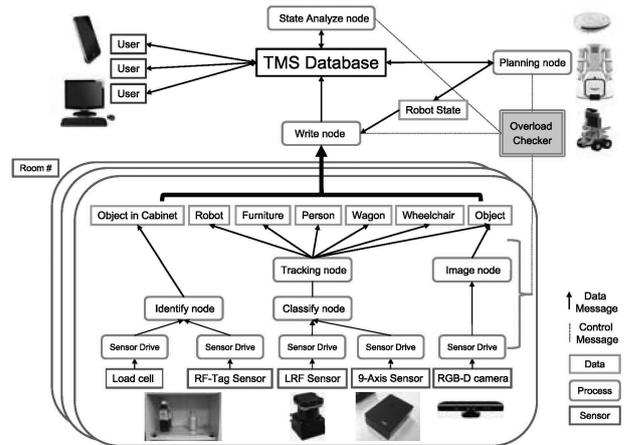


Fig. 1 Processing nodes of TMS

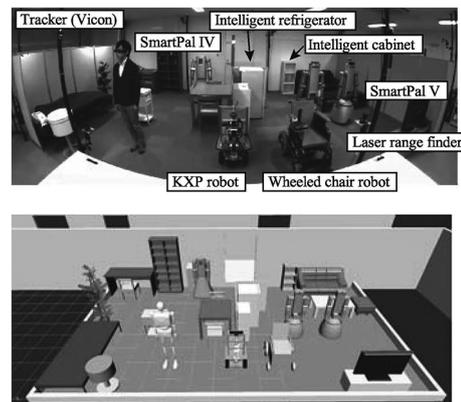


Fig. 2 Informationally structured environment and virtual environment

環境の例を示す。これまでに我々は、この ROS-TMS を用いて、ロボットによるサービスを提供するための物品管理や経路計画、人物の行動記録等のサービスを実装し、実証実験を行っている。

なお、Johnson ら [25] は、MOVAID [26] などの汎用サービスロボットよりも、市販のロボット家電やタスクに特化したロボットをモジュール化して組み合わせることで、より低価格で導入や実現の容易なロボットシステム (GIVING-A-HAND System) が開発できるとし、我々の提案する情報構造化環境下におけるサービスロボットシステムに近いコンセプトを提案している。しかし実現しているタスクは、電子レンジと低機能ロボットを組み合わせた fetch-and-carry タスクなど少数にとどまっており、また汎用性、柔軟性に欠ける。

4. ソフトウェアプラットフォーム ROS-TMS

Fig. 1 のノード群を、機能ごとに分類、整理したものを **Fig. 3** に示す。それぞれのノード群を以降ではモジュールと呼ぶ。User Request (TMS_UR), Sensor Driver (TMS_SD), Robot Controller (TMS_RC) の各モジュールは人、センサ、ロボットと TMS のインターフェースであり、これらを通してそれぞれ情報をやり取りする。タスクスケジューラ (TMS_TS) モジュールは、TMS_UR から得られたユーザからのタスク要請に基づ

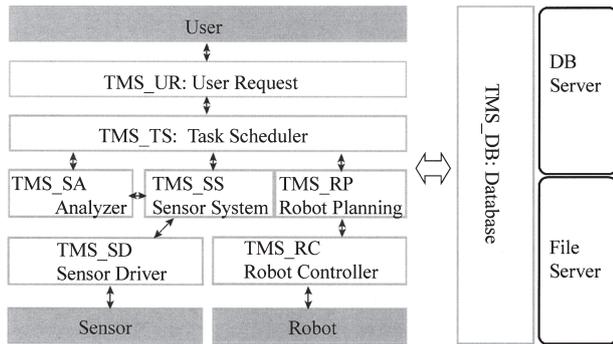


Fig. 3 ROS-TMS architecture

き、データベース (TMS_DB) モジュールからタスクを実現するためのサブタスク列の情報を読み取り、サブタスクの実行スケジュールを計画する。さらにロボットプランナ (TMS_RP) モジュールは、計画されたサブタスク列の実行を管理し、適切なタイミングでサブタスクの実行命令をロボットコントローラ (TMS_RC) モジュールに送る。

各モジュールの概要を以下に記す。

- TMS_DB: データベース本体と読み書きを行うノード群
- TMS_UR: ユーザの要請を受け取り、ROS_TS に送信するノード群
- TMS_TS: ロボットのタスクを計画するノード群
- TMS_SD: LRF, カメラ, RFID タグリーダなどのセンサデータを取得するノード群
- TMS_SS: センサデータを統合し、より高次の情報を出力するノード群
- TMS_SA: 環境の状態を推定するノード群
- TMS_RP: ロボットの作業を計画するノード群
- TMS_RC: 作業計画に従いロボット動作を制御するノード群

このような階層的なモジュール構成は、ほかの多くのロボット制御アーキテクチャやスマートハウス制御アーキテクチャ [27] ~ [35] でも見られる。例えば、Fong ら [32] は、人間と共同作業するロボットのためのミドルウェアとして、“Human-Robot Interaction Operating System” (HRI/OS) を提案しており、その中にもそれぞれの処理を分散したモジュール構成を見ることができる。必要な機能をモジュール化し、処理に応じて組み合わせを変えることで、開発の柔軟性、効率性を高める手法は、大規模システム開発ではすでに一般的であるが、提案する ROS-TMS は、このモジュール構成をすべて ROS ノードで構築している点が特徴である。すなわち、第 2 章で述べたように、ROS を基本としたシステムは、柔軟性、持続性、頑強性が高く、またすでに多くのリソースが公開され、利用可能であることから、的を絞った効率的な開発が可能であることが、ROS-TMS で採用したモジュール構成の特徴である。

これら ROS-TMS を構成する各モジュールについて、以下でより詳しく説明する。

4.1 Data Base (TMS_DB)

TMS_DB は、データベース本体と、情報の書き込みを行う DB_Writer、情報の読み出しを行う DB_Reader から構成される。データベース本体の実装には ROS-TMS ver. 3.4.2 では

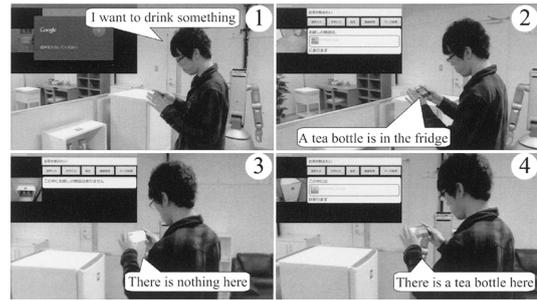


Fig. 4 Object recommendation using smart phone

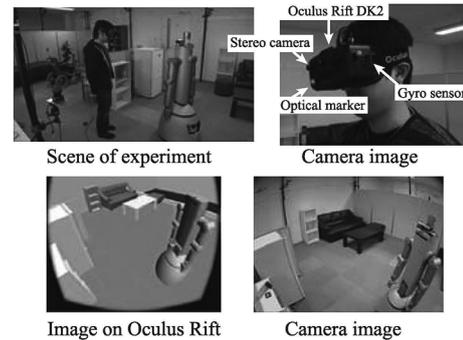


Fig. 5 Immersive display device consisting of Oculus Rift DK2, Ovrvision, and Vicon tracker

MySQL を、ver. 4.0 以降は mongodb を用いており、以下の種類の情報が格納される。

- ロボットの形状や外観などの情報
 - 環境埋込センサに関する情報
 - 地図などの静的な環境情報
 - 計測された移動体位置などの動的な環境情報
 - ロボットのタスクや、それを分割したサブタスクの情報
- データベースへの挿入や更新には DB_Writer を、読み出しは DB_Reader を用いる。センサから出力される情報は 10 [Hz] 以上で更新されるため、すべての情報を保存すると情報が重複して無駄が生じる。そのため、ロボットの移動量が少ない場合や収納庫内の物品が変化していない場合は、DB_Writer を用いてデータベースへの挿入や更新を制限する機能を持たせている。

4.2 User Request (TMS_UR)

TMS_UR は、スマートフォン、タブレットなどの端末からユーザからのタスク要請を受け取り、TMS_TS に送信する。また、データベースの登録内容の提示や、ユーザへの仮想空間の提示なども行う。Fig. 4 にスマートフォンを用いたユーザインタフェースの例を示す。このシステムは、音声入力により指定された物品の情報をユーザに提示するだけでなく、スマートグラスやスマートフォンのカメラにより家具を識別し、データベースに登録された家具内の物品を提示することができる。また Fig. 5 は没入感ディスプレイを用いた仮想現実感インタフェースであり、仮想化空間と実空間をシームレスに体感できる。

4.3 Task Scheduler (TMS_TS)

タスクスケジューラモジュール (TMS_TS) は、ユーザリクエスト (TMS_UR) から送信された要求 (タスク) を受け付け、

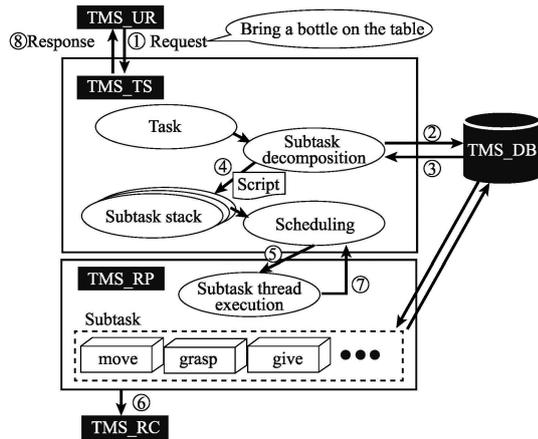


Fig. 6 Architecture of TMS-TS

必要な情報をデータベース (TMS_DB) と連携して補充しながら要求を解析、ロボットの動作コマンドを発行して作業を実行する機能を持つ [36]。タスク実行の流れを Fig. 6 に示す。

データベースには、作業情報構造化 [37] [38] の概念に基づき、様々なタスクを分割した様々なサブタスクと、特定のタスクを実行するためのサブタスクの組み合わせ情報が登録されている。TMS_TS は、要請されたタスクに基づき、データベースから必要なサブタスク列を取得する。まず、ユーザからタスクが要請されたとき (1)、そのタスクがどのようなサブタスクの集合からなるかをデータベースに問い合わせる (2)、(3)。次に、得られたサブタスクの集合を解釈して、タスク実行マシン (SMACH [39]) への入力となるスクリプトを発行する (4)。そして、スクリプトに記述された順序・並列度にしたがって、各サブタスクをスレッドで実行させる (5)。タスクスケジューラから実行命令が出されたサブタスクは、ロボットプランニングモジュール (TMS_RP) において、与えられた抽象的な情報をデータベースと連携しながら具体的な数値データに変換し、指定されたロボットを動かすためにロボットコントローラモジュール (TMS_RC) にコマンドを送信する (6)。

各サブタスクはスレッドで起動されるので、タスクスケジューラでは、現在走っているスレッド数と終了したスレッド数の整合性をとる必要がある。そこで、各サブタスクは、サブタスクが正常に終了したかやロボットプランニング、ロボットコントローラレベルでエラーが発生し、タスクが異常終了したかといった通知を実行タイミング制御ノードに送信する (7)。すべてのサブタスクが正常終了すると、要求されたタスクの完了を通知する (8)。

このように、タスクをサブタスクの組に分解し、異なるタスクで同じサブタスクを用いることで、プログラムの再利用性を高めることができる。さらに、すべてのロボットに共通する部分 (TMS_TS) と、ロボットに固有の部分 (TMS_RP, TMS_RC) を分離することで、ロボットが変わっても同じ要求に対して同様のタスクが実行できる。

サブタスクの実行管理には、タスク実行マシン SMACH [39] を用いる。SMACH は、Python ライブラリとして提供されており、タスクレベルのロボット作業を記述・実行するための階

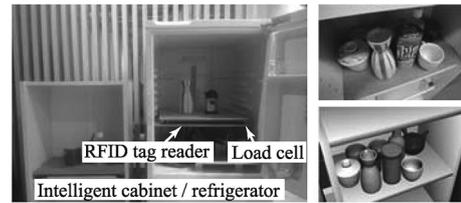


Fig. 7 Intelligent cabinet and intelligent refrigerator

層型オートマトンを生成できる。本システムでは、一つの状態を一つのサブタスクに割り当て、サブタスクを直列または並列に接続した状態遷移図でタスクを表現する。

4.4 Sensor Driver (TMS_SD)

LRF, RGB-D カメラ, RFID タグ, ロードセル, Vicon モーショントラッカ, 9 軸加速度, 角加速度, 地磁気センサ, 要介護者に装着した心拍センサや脳波センサなどのセンサを起動し、センサデータを取得する。取得されたセンサデータは、上位の TMS_SS に送られ、解析される。

4.5 Sensor System (TMS_SS)

TMS_SD から取得したセンサデータを、より高次の情報へ変換する。例えば、LRF を用いた床上物体を検出や人の行動推定、複数の RGB-D カメラを用いた人物の動作推定、RFID タグリーダとロードセルを用いた知的収納庫の物品検出などである。Fig. 7 に、RFID タグやロードセルを利用した知的収納庫、知的冷蔵庫、および知的収納庫による物品検出の例を示す。

4.6 State Analyzer (TMS_SA)

室内の特定状況の検出と、生じた状態の推定を行うためのモジュールである。TMS_SS から得られた室内のデータは、例えば物の位置や人の位置などの情報であり、それらを人の行動などの意味情報に変換し、ロボットサービスの実行依頼として TMS_TS に送信する。例えば、人が手で持って移動、放置した物品の探索では、あらかじめ人が物品を収納庫から取り出したことを検出し、人が部屋から退出するまでの歩行軌跡を取得する。その滞在時間から、物品が放置された可能性が高い固定家具を推定し、固定家具の ID と物品探索サービスの実行依頼を TMS_TS に送信する。そのほかにも、転倒した人の検出や状態確認のためのロボットの派遣なども実装されている。

4.7 Robot Planning (TMS_RP)

TMS_TS から指令されたサブタスクに対し、ロボットの種類を考慮してサブタスクを遂行するための動作計画を行うモジュールである。例えばロボットの移動であれば、障害物を避けながら目的地まで移動する安全な経路を計画し、TMS_RC に経由点を送信する。

4.8 Robot Controller (TMS_RC)

TMS_RP で計画されたサブタスクを、実際にロボットで実行するモジュールである。ロボットの移動であれば、TMS_RP から送られた目標点に到達するように、ロボットの前進・回転などの動作を計画し、実行する。ロボットアームによる把持であれば、指定された目標物体に指定方向からアプローチし、把持を行う。



Fig. 8 Big Sensor Box

5. ハードウェアプラットフォーム Big Sensor Box

これまでに説明した ROS-TMS の実証実験施設として、新たに情報構造化環境のハードウェアプラットフォーム「Big Sensor Box」を構築した。Big Sensor Box は、Fig. 8 に示すように、寝室や居間を備えた 2LDK のマンションタイプの居住空間である。

本プラットフォームには、以下のような様々なセンサ、ロボット、家電や家具が配置されている。

センサ

- 光学式モーショントラッキングシステム (Vicon Bonita)
- 分散 RGB-D カメラシステム
- 知能化家電

サービスロボット

- ヒューマノイド型サービスロボット (SmartPal V)
- 移動ロボット (KXP, Kobuki)
- 車いす型見守りロボット (Mimamoru-kun)

以下では、主なセンサ、知能化家電、およびサービスロボットを紹介する。

5.1 センサ

5.1.1 光学式モーショントラッキングシステム

本プラットフォームには、光学式トラッキングシステム (Bonita, Vicon Motion Systems) が、合計 18 台備え付けられている。これは複数の赤外線 LED ライトと赤外線カメラを用いて、反射マーカの三次元位置を数ミリ精度で計測するものである。反射マーカを移動家具やロボットに取り付けることにより、本プラットフォーム内におけるそれぞれの三次元位置、姿勢情報を極めて高精度 (1 [mm] 以下) で取得できる。

5.1.2 分散 RGB-D カメラシステム

プラットフォーム内の 9 箇所に、RGB-D カメラ (Kinect v2) を設置し、居住者の動作の推定を行った。複数の RGB-D カメラの位置姿勢は、デプス画像と RGB 画像を用いた格子パターンのキャリブレーションによりあらかじめ推定しておく。複数の RGB-D カメラから得られた距離画像に対し、Kinect for Windows SDK 2.0 により、それぞれの画像内で人物の位置と関節位置を推定する。次に、人物ごとにカメラ間で対応関係を推定し、カメラまでの位置と相対姿勢に基づき定義された信頼度に基づき、同一の人物情報のうち信頼度の低い情報を棄却する。残された関節位置の推定値を、人物の位置とともに TMS_DB に送り、データベースに格納する。Fig. 9 に空間内に設置した複数の RGB-D カメラ (Kinect v2)、および居住者の動作推定の様子を示す。

5.1.3 知能化家電

プラットフォーム内に設置された収納庫、冷蔵庫には、ロー

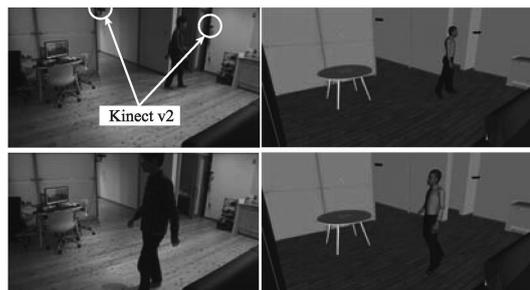


Fig. 9 Motion estimation using RGB-D cameras (Kinect v2)

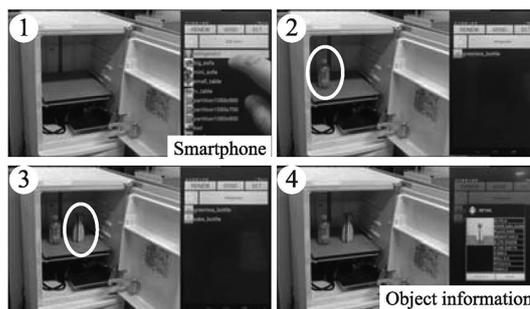


Fig. 10 Registration of objects in intelligent refrigerator

ドセルと RFID リーダが取り付けられ、格納された物品の種別、位置を計測できる (Fig. 7) [40]. プラットフォーム内の物品にはすべて RFID タグが貼り付けられており、ロードセルに加わる物品の荷重の重心位置から物品の位置を推定し、また同時に RFID タグリーダでその物品の識別を行う。計測された物品情報は、ROS-TMS のデータベースに格納される。

またこれらの家具には AR マーカが貼られており、4.2 節で紹介したように、スマートフォンやスマートグラスのカメラで撮影することで、ドアを開けることなく内容物が確認できる。Fig. 10 は、スマートフォンを用いて ROS-TMS にアクセスし、知的冷蔵庫内の登録物品を確認している様子である。

5.2 サービスロボット

5.2.1 ヒューマノイド型サービスロボット

ROS-TMS により制御されるヒューマノイド型サービスロボット (SmartPal V, 安川電機) により、物品取り寄せなどのサービスを提供する。Fig. 11 に、全方向レーザスキャナを用いた障害物のリアルタイム回避行動の様子を示す [41]. ヒューマノイド型サービスロボットの上部には、全方向レーザスキャナ (Velodyne HDL-32E) と、RGB-D カメラ (Xion, ASUS) が搭載されている。計測された障害物上の点群からアームに対し仮想的な斥力を発生させ、アームを障害物から退避させる。物体形状を復元せずに直接点群を用いることにより、実時間での回避動作を生成をしている。

5.2.2 移動ロボット

移動マニピュレータ型サービスロボット (KXP, Fig. 12 左) は、Katana (Neuronics), Xtion (ASUS), Pioneer3-AT (MobileRobots) を組み合わせて開発したロボットである。Pioneer3-AT は本来、4 輪スキッドステアの移動台車型ロボットであるが、後輪をキャストに変更し、室内でその場回転ができるなど、

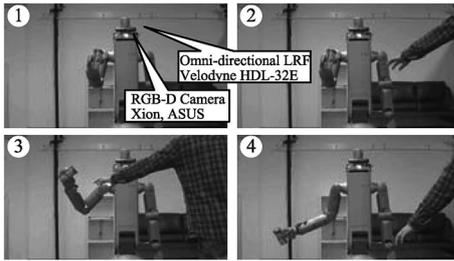


Fig. 11 Humanoid-type service robot (SmartPal V, Yaskawa Electric Corporation) and collision avoidance using Velodyne HDL-32e

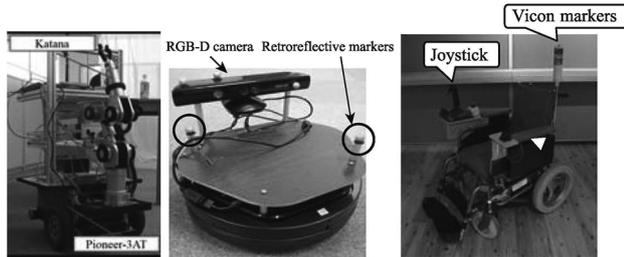


Fig. 12 Manipulator-type service robot (KXP), Kobuki, and Wheel-chair robot

小回りが利くように改良してある。Katana は 5 自由度を有する可搬重量 0.5 [kg] の小型マニピュレータである。手先に 1 自由度のハンドが装着されており、床上の物品を把持することができる。

一方、Kobuki (Fig. 12 中) は、Yujin Robot 社が開発した移動ベースで、直径 30 センチの円柱型である。ROS で基本的なパッケージが提供されており、移動用のプラットフォームとしても幅広く活用されている。Kobuki には、RGB-D カメラ (Kinect, Microsoft) と Vicon によるトラッキングのために反射マーカを取り付けている。

5.2.3 車いす型見守りロボット

居住者の移動支援と生体情報モニタリングを目的に、Fig. 12 右に示す車いす型見守りロボットを試作した [42]。搭乗者は脳波センサ (MindWave Mobile, NeuroSky), 心拍センサ (T31C トランスミッター, POLAR) などの生体センサを装着し、得られた生体情報 (脳波, 心拍) は無線で ROS-TMS に送信され、蓄積される。介護者は ROS-TMS に接続し、手元の端末を用いて遠隔地から生体データを確認できる。このシステムは、従来の TMS がロボットやモノの管理を行っていたのに加え、その範囲をヒトの健康把握にも拡張したものと考えられる。

また車いす型見守りロボットの位置は、前述の光学式トラッキングシステムにより逐次計測され、カルマンフィルタにより車輪回転角度の情報と統合されて ROS-TMS に保存される。車いす型見守りロボットの移動は、ROS-TMS による自律移動と、ジョイスティックによる手動制御が可能である。

5.3 シミュレーション環境

シミュレーション環境として、産業技術総合研究所で開発された Choreonoid, ROS の標準可視化ツール RViz, および動力学シミュレータ Gazebo が利用可能である。Fig. 13 に Chore-

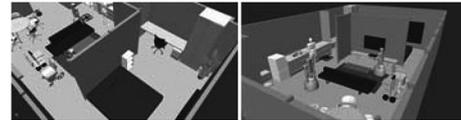


Fig. 13 3D model of the environment (Choreonoid)

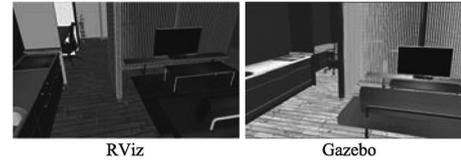


Fig. 14 3D model of the environment (RViz and Gazebo)

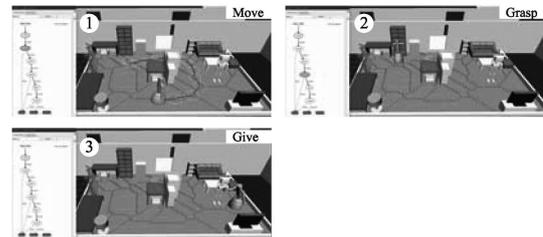


Fig. 15 Fetch and carry task performed by service robot (Simulation)

onoid によるシミュレーション画像の例を、Fig. 14 に可視化ツール RViz, 動力学シミュレータ Gazebo で作成した Big Sensor Box 環境の例を示す。また、Fig. 15 に取り寄せタスクを実行したときの Choreonoid の画面を示す。

6. ROS-TMS と Big Sensor Box を用いたサービス実験

本章では、

“開発した情報構造化環境のためのソフトウェアプラットフォーム ROS-TMS 4.0 とハードウェアプラットフォーム Big Sensor Box が、ロボットが対象とする既定されたサービスを提供するのに必要な機能・性能を有しているか”

について、実験により検証を行った。ただし、ここでは対象とするサービスとして、日常生活で頻繁に起こり得る軽作業である「物品取り寄せタスク」と、介護施設への導入を念頭に「車いす型ロボットによる自動搬送」に絞って、開発システムで実現できるかを検証した。

具体的には、

- (1) 異なる種類のロボットにより、データベースを参照して物品の位置を把握し、任意の位置のユーザへ手渡す「物品取り寄せ」が実現できるか。
- (2) 車いす型ロボットにより、タブレットをタッチするだけで搭乗者が目的地へ移動できる自動操縦サービスが提供できるか

を検証することを目的とした。なお、物品取り寄せでは、ユーザは物品の位置を指示するのではなく、物品の名前のみを ROS-TMS に知らせ、ROS-TMS はデータベースを参照して物品の位置を特定し、現在位置、物品の位置、およびユーザの位置を結

Table 1 Task descriptions in TMS_DB

type	id	name	etcddata		
task	8001	get_object	9001\$roid 9002\$roid + 9003\$uid +		
task	8002	patrol	9001\$rid 9006\$roid 9007\$roid +		
type	id	name	type	id	name
subtask	9001	move	subtask	9002	grasp
subtask	9003	give	subtask	9006	sensing
subtask	9007	random_walk			

ぶ安全な経路を計画する。その後、タスク実行マシンにより、移動、把持、移動の順に、ロボットが指令された動作を実行する。

6.1 複数種類のロボットによる物品取り寄せサービス実験

6.1.1 タスク実行手順の詳細

開発した ROS-TMS を用いたサービス実験として、人から要求された物品を、別の場所まで取りに行き、把持して手渡し「物品取り寄せタスク」を行った。想定するサービスシナリオは、室内の任意の場所にいるロボットに、室内の別の場所にある物品の取り寄せを指示することにより、ROS-TMS が必要な行動を計画し、実行するものである。

TMS_UR を通して受け取ったタスク指令に対し、TMS_TS はタスクを移動 (Move, 上)、把持 (Grasp, 中)、手渡し (Give, 下) のサブタスクに分解する。各サブタスクは、Fig. 15 の左の SMACH の画面のように、順々に TMS_RP に送られる。TMS_RP では、受けとったサブタスクの実行命令から、TMS_DB に記録されている物品の位置 (棚の中)、棚の位置、ロボットの現在位置、ヒトの位置などを用いて、移動目標軌道などの具体的な動作指令を生成し、TMS_RC に送信する。TMS_RC は、指示された動作指令をロボットごとに異なる具体的な制御命令に変えて実行する。この仕組みにより、異なるロボットに対しても、タスク、サブタスクの計画には共通のモジュールを使うことができる。また、サブタスクに分解することで、異なるタスクに対して同じサブタスクを利用でき、プログラムの再利用性が高まる。

より具体的に説明する。データベースには、Table 1 のように、タスクを構成するサブタスクと、その引数、および実行順が文字列として格納されている。例えば、「物品取り寄せタスク」である get_object タスク (ID=8001) は、サブタスク move (ID=9001)、grasp (ID=9002)、および give (ID=9003) から構成されている。さらに、それぞれのサブタスクは、物体固有の ID (\$roid) やユーザ固有の ID (\$uid) ロボット固有の ID (\$rid) を引数として呼び出される。サブタスクの記述は逆ポーランド記法を採用し、トークンの区切り文字を「(スペース)」、サブタスクを直列に接続するか、並列に接続するかをそれぞれ演算子「+」、「|」で表す。物体固有の ID (\$roid) やユーザ固有の ID (\$uid) は、実行時にデータベースを参照して、具体的な位置座標へと変換される。なお、get_object タスクなどのタスクを呼び出す際は、ロボット固有の ID (\$rid) に実行を依頼するロボットの番号 (SmartPal は 2002、KXP は 2003 など) をあらかじめ設定しておく。

\$rid = 2002 (SmartPalV), \$roid = 7001 (チップスター), \$uid = 1001 (ユーザ) とした場合に、TMS_TS から発行された get_object タスクに対する SMACH への入力スクリプトの

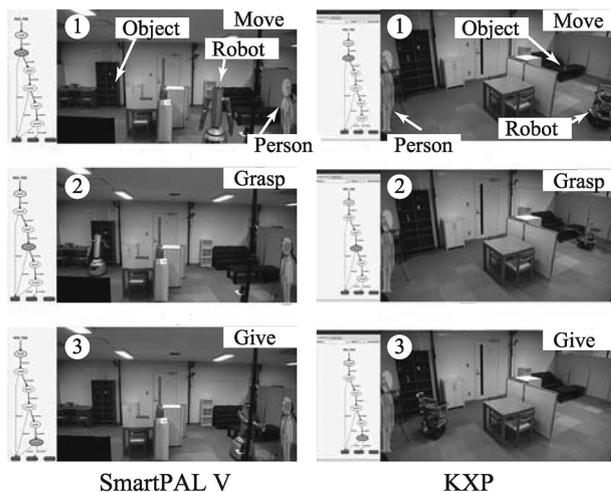


Fig. 16 Fetch and carry task performed by service robot (SmartPAL V and KXP)

一部を以下に示す。なお、SMACH では計画された動作が時間内に完了しない場合には、aborted に遷移してロボットの動作を直ちに停止させることで、システム全体は停止しない仕組み (第 2 章のソフトウェアプラットフォームの要求仕様 (7)) を実現している。

```
smach.StateMachine.add('move',
    ServiceState('rp_cmd', rp_cmd,
        request = rp_cmdRequest(9001,2002,[7001]),
        transitions={'succeeded':'grasp'})
smach.StateMachine.add('grasp',
    ServiceState('rp_cmd', rp_cmd,
        request = rp_cmdRequest(9002,2002,[7001]),
        transitions={'succeeded':'give'})
smach.StateMachine.add('give',
    ServiceState('rp_cmd', rp_cmd,
        request = rp_cmdRequest(9003,2002,[1001]),
        transitions={'succeeded':'succeeded',
            aborted':'aborted'})
```

6.1.2 タスク実行結果

実機による実験に先立ち、まず Choreonoid を用いて、生成された行動が安全に実行できるかを確認した。Fig. 15 に、前項で示したスクリプトを用いて物品取り寄せタスクに対して計画された行動を示す。Fig. 15 中の各図の右側は計画された動作であり、床面の点線は、TMS_RP により計画された移動目標軌道を示している。これは環境や障害物の位置から最も離れた線 (Voronoi 境界) であり、ロボットはこの Voronoi 境界上を安全に移動する。また Fig. 15 の各図の左側は、タスク実行マシン (SMACH) により管理されているサブタスクの実行状態である。

シミュレータにより安全が確認された後、実際に実機を用いた動作実験を行った。Fig. 16 は、計画されたサービス動作をサービスロボット SmartPAL V および KXP により行っている様子である。このように、ユーザはロボットや物品の ID を変えて同じタスクを呼び出すだけで、それぞれのロボットはデータベースから必要な情報を読み出し、適切にタスクを実行できる。

さらに同様の物品取り寄せタスクを Big Sensor Box で行った

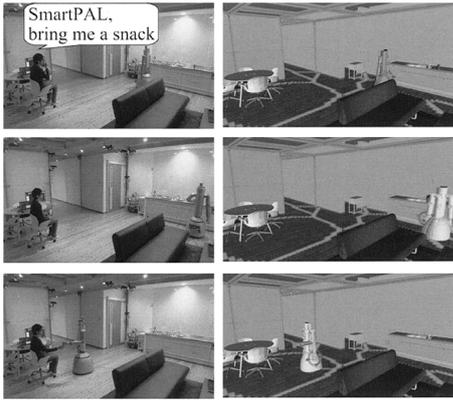


Fig. 17 Fetch and carry task performed by service robot in Big Sensor Box (Simulation and experiment)

結果を Fig. 17 に示す。ただしここでは、ユーザインタフェースには 4.2 項で示した音声入力を用いており、「スマートパル、お菓子持ってきて」という指示に基づいて検索された物品を、サービスロボット SmartPAL V が把持し、ユーザの位置へお菓子を運ぶシナリオを想定して実験を行った。Fig. 17 右は RViz を用いた安全確認のためのシミュレーションの様子であり、Fig. 17 左が安全確認後に実行された実機によるサービス動作の様子である。なお、サービスロボット、ユーザ（眼鏡）の位置は、それぞれに取り付けたマーカを光学式モーショントラッキングシステムで追跡することで計測され、物品や什器の位置はあらかじめデータベースに登録されている。光学式モーショントラッキングシステムによるマーカの三次元位置計測精度は 1 [mm] 以下であり、サービスロボットは環境に接触することなく、把持動作、移動動作が実行できている。また、サービスロボットは無線 LAN により、光学式モーショントラッキングシステムは有線 LAN により ROS-TMS に接続されており、光学式モーショントラッキングシステムからは周期 100 [Hz] ですべてのマーカの三次元位置データが TMS に送信されている。

これらの実験により、構築したソフトウェアプラットフォーム ROS-TMS は、今回設定したシナリオに沿った「物品取り寄せタスク」を複数のサービスロボットやセンサ、アプリケーションソフトウェア、ユーザインタフェース、情報ネットワーク、シミュレータを用いて実現でき、ROS を用いた分散ネットワークの特徴とともに、第 2 章で示した情報構造化環境のためのソフトウェアプラットフォームの要求仕様 (1) (7) および (10) (11) を満たしている。また、構築したハードウェアプラットフォーム Big Sensor Box は、サービスロボットも含めて今回設定したシナリオに沿って「物品取り寄せタスク」を実行するのに必要なセンサ、ロボット、データベース、通信インフラを有しており、第 2 章で示した情報構造化環境のためのハードウェアプラットフォームの要求仕様 (1) (4) を満たしていることが確認された。

また、「物品取り寄せタスク」と同じサブタスク (Move) を用い、異なるアプリケーションを実現した例として、Fig. 18 に kobuki により見回りタスクを行わせた例を示す。見回りタスクは、環境内に倒れている人を見つけるタスクであり、部屋

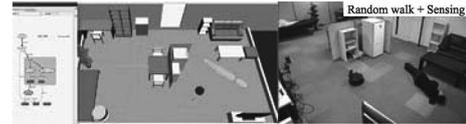


Fig. 18 Patrol task performed by kobuki

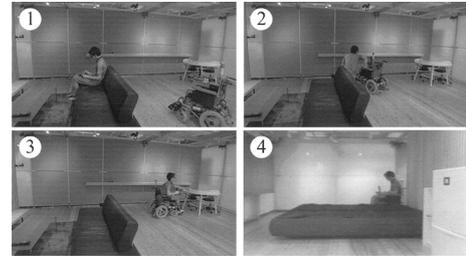


Fig. 19 Service experiment performed by wheel-chair robot

の中央まで移動 (Move)、ランダムウォーク (Random walk)、RGB-D カメラによる障害物検出 (Sensing) のサブタスクからなる。ランダムウォークと RGB-D カメラによる障害物検出は、SMACH において並列に実行される。人の大きさのセグメントが検出されると、タスクは abort 命令を出して停止し、ロボットはその場で停止すると同時に警告音を発し、異常を伝える。これにより、4.3 項で設計したタスクスケジューラモジュールにより、異なるタスクで同じサブタスクが繰り返し利用でき、サブタスクの再利用性が高められたことが確認できた。

6.2 車いす型ロボットを用いた要介護者の自動搬送サービス実験

開発した情報構造化環境のためのソフトウェアプラットフォーム ROS-TMS 4.0 とハードウェアプラットフォーム Big Sensor Box では、特に介護施設での利用が有力なアプリケーションであると考えられる。そこで介護施設での利用を想定し、開発した車いす型ロボットを用いて

- (1) Big Sensor Box に設置した光学式トラッキングシステムにより、車いす型ロボットが安全に自動操縦により移動できるか
- (2) 居住者の生体情報をモニタリングし、データベースに格納できるか

の 2 点について実験を行った。想定したシナリオは、居住者がソファから円卓まで移動し、ペットボトルを取り、ベッドへ移動するものである。

具体的には以下のシーケンスからなる。

- (1) 手元のタブレット PC を用いて自動操縦でソファまでロボットを呼ぶ
- (2) 車いすに移乗
- (3) ジョイスティックを用いた手動操縦で円卓まで移動
- (4) ペットボトルを取った後、タブレット PC を用いて自動操縦でベッドへ移動

Fig. 19 に車椅子ロボットの動作を示す。また、Fig. 20 に生体センサにより計測された搭乗者の心拍 (hr)、脳波センサから取得した瞑想度 (med)、集中度 (att) を Choreonoid に表示している様子を示す。このように、移動中に搭乗者の生体情報を

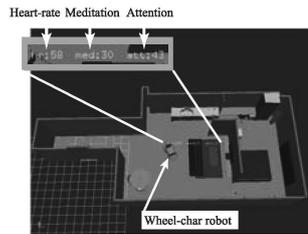


Fig. 20 Vital data displayed on Choreonoid

計測し、ROS-TMS に逐次格納しながら、自動、手動による搬送サービスが実行できている。

以上の実験を通して、構築したソフトウェアプラットフォーム ROS-TMS とハードウェアプラットフォーム Big Sensor Box は、今回設定したシナリオに沿って車いす型ロボットにより要介護者に自動搬送サービスが提供でき、本サービスに対して第 2 章で示した情報構造化環境のための要求仕様を満たしていることが確認された。

7. 結 言

本論文では、まず情報構造化環境を構築するためのハードウェア、およびソフトウェアプラットフォームの要求仕様について考察し、これまでに筆者らが開発してきたソフトウェアプラットフォーム TMS を紹介した。その後、TMS 開発の過程で明らかとなった拡張性、柔軟性、開発効率などの問題に対処するため、オープンソースミドルウェア ROS を導入した新たな ROS-TMS を提案した。また新たに構築した ROS-TMS の実証実験施設として、ハードウェアプラットフォーム Big Sensor Box を提案した。さらに、日常生活で頻繁に起こり得る軽作業である「物品取り寄せタスク」と、介護施設での利用を想定した「車いす型ロボットによる自動搬送」に対して、開発した情報構造化環境のためのソフトウェアプラットフォーム ROS-TMS 4.0 とハードウェアプラットフォーム Big Sensor Box が、ロボットがサービスを提供するソフトウェア・ハードウェアプラットフォームとして必要な機能・性能を有しているかを、実験により検証した。

TMS から ROS-TMS へ至る一連のシステム開発を通して、筆者らが最も重視した点は、オープン化、分散化、拡張性、柔軟性である。すなわち、標準規格（デファクトスタンダード）に則った頑強かつ拡張性に優れた分散システムを採用し、必要な機能は可能であれば公開されたモジュールを用い、不可能であれば自ら作成して公開することで、ロボットコミュニティとしての知の有効活用と開発時間、コストの低減を目指すものである。従来の TMS はモノリシックなシステムであり、通信ロスや実行効率などでは有利であるが、新たな機能の追加や拡張には不向きであった。ROS-TMS はモジュール化された分散システムであり、新たな機能やデバイスの追加や拡張が容易であり、またシステムが大規模になるほど TMS に比べて開発効率の点で有利になると考えられる。

第 2 章で示した「ソフトウェアプラットフォームの要求仕様」の (8) (9) については、現段階ではまだセンサ情報の融合や高度な推論機能などが不十分であり、今後、機能の拡充を継続して行っていく予定である。

謝 辞 本研究は、国立研究開発法人科学技術振興機構の研究成果展開事業センター・オブ・イノベーション (COI) プログラムにより、助成を受けたものである。

参 考 文 献

- [1] T. Sato, Y. Nishida and H. Mizoguchi: "Robotic room: Symbiosis with human through behavior media," *Robotics and Autonomous Systems*, vol.18, No.1-2, pp.185-194, 1996.
- [2] J.-H. Lee, N. Ando and H. Hashimoto: "Design policy of intelligent space," *Proc. on 1999 IEEE International Conference on Systems, Man, and Cybernetics*, vol.3, pp.1077-1082, 1999.
- [3] A.P. Pentland: "Smart rooms," *Scientific American*, vol.274, no.4, pp.54-62, 1996.
- [4] R.A. Brooks: "The intelligent room project," *Proc. of the 2nd International Conference on Cognitive Technology (CT '97)*, CT '97, pp.271-278, 1997.
- [5] J.A. Kientz, S.N. Patel, B. Jones, Ed Price, E.D. Mynatt and G.D. Abowd: "The georgia tech aware home," *CHI '08 Extended Abstracts on Human Factors in Computing Systems, CHI EA '08*, pp.3675-3680, 2008.
- [6] S. Sugano and Y. Shirai: "Robot design and environment design—waseda robot-house project," *Proc. of International Joint Conference SICE-ICASE, 2006*, pp.I-31-I-34, 2006.
- [7] H. Noguchi, T. Mori and T. Sato: "Automatic generation and connection of program components based on rdf sensor description in network middleware," *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.2008-2014, 2006.
- [8] K.-H. Park, Z. Bien, J.-J. Lee, B.K. Kim, J.-T. Lim, J.-O. Kim, H. Lee, D.H. Stefanov, D.-J. Kim, J.-W. Jung, et al.: "Robotic smart house to assist people with movement disabilities," *Autonomous Robots*, vol.22, no.2, pp.183-198, 2007.
- [9] Y. Kato, T. Izui, Y. Tsuchiya, M. Narita, M. Ueki, Y. Murakawa and K. Okabayashi: "Rsi-cloud for integrating robot services with internet services," *Proc. of IECON 2011—37th Annual Conference on IEEE Industrial Electronics Society*, pp.2158-2163, 2011.
- [10] H. Gross, Ch. Schroeter, S. Mueller, M. Volkhardt, E. Einhorn, A. Bley, Ch. Martin, T. Langner and M. Merten: "I'll keep an eye on you: Home robot companion for elderly people with cognitive impairment," *Proc. of IEEE International Conference on Systems, Man, and Cybernetics*, pp.2481-2488, 2011.
- [11] M. Tenorth, A.C. Perzylo, R. Lafrenz and M. Beetz: "The roboearth language: Representing and exchanging knowledge about actions, objects, and environments," *Proc. of IEEE International Conference on on Robotics and Automation*, pp.1284-1289, 2012.
- [12] <http://www.ros.org/>
- [13] Emile Aarts and Reiner Wichert: *Ambient intelligence*. Springer, 2009.
- [14] <http://www.companionable.net/>
- [15] M. Chan, D. Esteve, C. Escriba and E. Campo: "A review of smart homes—present state and future challenges," *Computer Methods and Programs in Biomedicine*, vol.91, no.1, pp.55-81, 2008.
- [16] D.H. Stefanov, Z. Bien and W.-C. Bang: "The smart house for older persons and persons with physical disabilities: structure, technology arrangements, and perspectives," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol.12, no.2, pp.228-250, 2004.
- [17] K. Ashton: "That 'internet of things' thing," *RFid Journal*, vol.22, no.7, pp.97-114, 2009.
- [18] 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.
- [19] 倉爪亮, 村上剛司, 木室義彦, 家永貴史, 馬場伸一, 股中翔: "ロボッ

- トタウンの共通プラットフォーム技術のメカニズム”, 日本ロボット学会誌, vol.26, no.5, pp.415-419, 2008.
- [20] 村上剛司, 長谷川勉, 木室義彦, 千田陽介, 家永貴史, 有田大作, 倉爪亮: “情報構造化環境における情報管理の一手法”, 日本ロボット学会誌, vol.26, no.2, pp.192-199, 2008.
- [21] http://openrtm.org/openrtm/ja/project/NEDO_Intelligent_PRJ_ID121
- [22] 辻徳生, ビョюнソク, ユンチェン, 永田晃洋, 長谷川勉, 倉爪亮, 諸岡健一, 村上剛司: “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”, 第 13 回計測自動制御学会システムインテグレーション部門講演会, pp.1H4-7, 2012.
- [23] ビョюнソク, 辻徳生, 橋口優香, 永田晃洋, 中島洗平, 倉爪亮, 長谷川勉, 諸岡健一: “情報構造化アーキテクチャの提案とサービスロボットのオンライン動作計画の実現”, 第 19 回ロボティクスシンポジウム予稿集, pp.624-630, 2014.
- [24] https://github.com/irvs/ros_tms/wiki/
- [25] M.J. Johnson, E. Guglielmelli, G.A.D. Lauro, C. Laschi, M.C. Carrozza and P. Dario: “6 giving-a-hand system: The development of a task-specific robot appliance,” *Advances in Rehabilitation Robotics*, pp.127-141, 2004.
- [26] P. Dario, E. Guglielmelli, C. Laschi and G. Teti: “Movid: a personal robot in everyday life of disabled and elderly people,” *Technology and Disability*, vol.10, no.2, pp.77-93, 1999.
- [27] R.G. Simmons: “Structured control for autonomous robots,” *IEEE Transactions on Robotics and Automation*, vol.10, no.1, pp.34-43, 1994.
- [28] R. Simmons, R. Goodwin, K.Z. Haigh, S. Koenig and J. O’Sullivan: “A layered architecture for office delivery robots,” *Proc. of the first international conference on Autonomous agents*, pp.245-252, 1997.
- [29] R.G. Simmons, R. Goodwin, K.Z. Haigh, S. Koenig, J. O’Sullivan and M.M. Veloso: “Xavier: Experience with a layered robot architecture,” *ACM Sigart Bulletin*, vol.8, no.1-4, pp.22-33, 1997.
- [30] R. Alami, R. Chatila, S. Fleury, M. Ghallab and F. Ingrand: “An architecture for autonomy,” *The International Journal of Robotics Research*, vol.17, no.4, pp.315-337, 1998.
- [31] R. Alami, S. Fleury, M. Herrb, F. Ingrand and F. Robert: “Multi-robot cooperation in the martha project,” *Robotics & Automation Magazine, IEEE*, vol.5, no.1, pp.36-47, 1998.
- [32] T. Fong, C. Kunz, L.M. Hiatt and M. Bugajska: “The human-robot interaction operating system,” *Proc. of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pp.41-48, 2006.
- [33] J.-H. Kim, I.-B. Jeong, I.-W. Park and K.-H. Lee: “Multi-layer architecture of ubiquitous robot system for integrated services,” *International Journal of Social Robotics*, vol.1, no.1, pp.19-28, 2009.
- [34] C.-L. Wu, C.-F. Liao and L.-C. Fu: “Service-oriented smart-home architecture based on osgi and mobile-agent technology,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol.37, no.2, pp.193-205, 2007.
- [35] S.K. Das, D.J. Cook, A. Battacharya, E.O. HEIERMAN III, and T.-Y. Lin: “The role of prediction algorithms in the mavhome smart home architecture,” *Wireless Communications, IEEE*, vol.9, no.6, pp.77-84, 2002.
- [36] 橋口優香, ビョюнソク, 辻徳生, 諸岡健一, 倉爪亮: “環境情報構造化プラットフォーム ros-tms におけるタスク管理機構”, 日本機械学会ロボティクスメカトロニクス講演会, pp.1A1-N01, 2015.
- [37] 吉見卓, 香月理絵, 大賀淳一郎: “環境・作業情報構造化を利用したロボット作業技術”, *東芝レビュー*, vol.64, no.1, pp.14-18, 2009.
- [38] 香月理絵, 西山学, 大賀淳一郎, 吉見卓, 菅原淳, 中本秀一: “異機種ロボット間のプログラム共通化を目指したソフトウェア構造—実装方法と東芝の双腕自律移動ロボットでの検証—”, 第 9 回システムインテグレーション部門講演会, pp.1L2-4, 2008.
- [39] J. Bohren and S. Cousins: “The smach high-level executive [ros news],” *IEEE Robotics & Automation Magazine*, vol.4, no.17, pp.18-20, 2010.
- [40] 村上剛司, 松尾一矢, 野原康伸, 長谷川勉, 倉爪亮: “知的収納庫と floor sensing system を用いた物品追跡システム”, 第 28 回日本ロボット学会学術講演会予稿集 DVD-ROM, 3P1-7, 2010.
- [41] 稲田大亮, 辻徳生, 倉爪亮, 諸岡健一: “高速 3 次元距離センサによる計測点群とのリアルタイム干渉判定に基づくオンライン回避動作の生成”, 第 33 回日本ロボット学会学術講演会予稿集 DVD-ROM, 2B2-05, 2015.
- [42] 重兼聡夫, 渡邊裕太, 稲田大亮, ビョюнソク, 辻徳生, 河村晃宏, 倉爪亮: “屋内空間 cps プラットフォーム big sensor box の開発と車いす型見守りロボットによる搬送実験”, 第 33 回日本ロボット学会学術講演会予稿集 DVD-ROM, 3H1-02, 2015.



倉爪 亮 (Ryo Kurazume)

1991 年東京工業大学機械物理学専攻修士課程修了。同年 (株) 富士通研究所入社, 1995 年東京工業大学機械宇宙学科助手, 2000 年スタンフォード大客員研究員, 同年東京大学生産技術研究所博士研究員, 2002 年九州大学システム情報科学研究所助教授, 2007 年より同教授, 2016 年より副研究院長,

現在に至る。日本ロボット学会論文賞 (1993, 2014), 日本ロボット学会功労賞 (2008), 日本機械学会ロボティクス・メカトロニクス部門学術業績賞 (2012) 等受賞。群ロボット, 歩行機械, レーザ計測, サービスロボットの研究に従事。博士 (工学)。

(日本ロボット学会正会員・フェロー)



辻 徳生 (Tokuo Tsuji)

2005 年九州大学大学院システム情報科学研究所博士後期課程修了。同年広島大学大学院工学研究科 COE 研究員, 2008 年産業技術総合研究所知能システム研究部門産総研特別研究員, 2011 年九州大学大学院 システム情報科学研究所情報知能工学部部門助教, 2016 年金沢大学理工研究域機械工学系准教授, 現在に至る。多指ハンド, 生活支援ロボット, 画像処理の研究に従事。電子情報通信学会, 日本機械学会各会員。博士 (工学)。

(日本ロボット学会正会員)



ピョ ユンソク (Pyo Yoonseok)

2009 年韓国光云大学電子情報工学部電子工学科卒業。同年韓国科学技術研究院研究員, 2016 年九州大学大学院システム情報科学府博士後期課程修了。サーボコントローラ, 環境構造化の研究に従事。博士 (工学)。



河村晃宏 (Akihiro Kawamura)

2013 年九州大学大学院システム情報科学府情報知能工学専攻博士課程修了。同年より立命館大学理工学部ロボティクス学科助教, 2015 年より九州大学システム情報科学研究所助教授, 現在に至る。多指ロボットハンド, ロボットアームの動的制御に関する研究に従事。IEEE, 計測自動制御学会等の会

員。

(日本ロボット学会正会員)