

タウンマネジメントシステムによる 地図情報と連携した物品取得サービスの実現

田中堅三, 倉爪亮, 辻徳生, 長谷川勉 (九大)

A Method of Robot Service to Take Objects in Cooperation with Map and Object Information in Town Management System

Kenzo TANAKA, Ryo KURAZUME, Tokuo TSUJI, Tsutomu HASEGAWA (Kyushu University)

Abstract: Town Management System (TMS) is a data management system of a structured environment for realizing various robot services consisting of several databases and a software library for data access. This paper introduces how to design a robot service for taking a desired object placed in distant locations by cooperating map and object databases in TMS.

1. はじめに

我々はこれまで、サービスロボットが作業を行う環境側に多様なセンサ群を分散配置し、ロボットに搭載したセンサと分散センサ群を連携させ運用する「環境構造情報化」の研究を推進してきた [1-3]。その中で、分散センサから得られる膨大な環境情報や、環境内で作業する複数の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、RT Middleware 化や介護施設を想定した実証実験などを通して、TMS の機能強化と機能検証を行ってきた [4, 5]。TMS は、地図情報データベースや物品情報データベースなど複数のデータベースから構成され、それらのデータベースを組み合わせて情報を管理している。例えば、物品情報データベースは、物品の位置座標や収納庫 ID を保持しているが、その位置座標は定量地図情報データベースで管理されている 3 次元環境構造内の座標値とリンクしている。また、収納庫 ID は、位相地図情報データベースにおける収納庫ノードの ID と一致する。

本報告では、上述したサービスロボットのための環境情報構造化データベースである TMS を用い、保持している地図情報と物品情報を相互に連携して、ロボットによる物品取得サービスを実現した例を紹介する。

2. タウンマネジメントシステム, TMS

2.1 タウンマネジメントシステムの概要

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情報は、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) に大別できる。

TMS は, この「体」, 「物」, 「場」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。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。TMS で管理された環境で運用されるサービスロボットは, 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, 「体」, 「物」, 「場」

の情報を取得する。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。また, API はロボットからの問い合わせに応じて TMS が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 TMS 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。イベントは来客や手招きなど環境内で起こる様々な事象を抽象化したものであり, ユーザ側の必要に応じて自由に定義できる。

2.2 データベースの種類

TMS を構成しているデータベースの一部を以下に示す。

- 定量地図データベース
- 位相地図データベース
- 座標系データベース
- 物品情報データベース
- 移動体データベース
- ロボットデータベース
- イベントデータベース
- JAN コードデータベース

TMS は, これらのデータベースを組み合わせて情報を管理している。

3. TMS の地図・物品情報データベース

本章では, TMS の管理する地図情報データベースと物品情報データベースについて説明する。

3.1 地図情報データベース構造

TMS が管理する地図には, 壁の位置などの具体的な座標値を表す定量地図と, 部屋間の関係をグラフ構造で示した位相地図がある。以下, それぞれの地図のためのデータベース構造を示す。

3.1.1 定量地図情報データベース

定量地図情報データベースを Table 1 に示す。定量地図情報データベースでは, coordid で, 関連する位相地図毎に

Table 1 Metric map class

項目名	項目内容
mapid	定量地図 ID
coordid	座標系 ID (原点位置)
scale	縮尺 (scale)
type	種類 (建物, 階, 部屋, 収納庫)
coord_x(yz)1	地図左上起点 x(yz) 座標
coord_x(yz)2	地図右下終点 x(yz) 座標
content	バイナリデータ
grid_x_width	グリッド x 幅
grid_y_width	グリッド y 幅
grid_x_counts	x 方向のグリッド数
grid_y_counts	y 方向のグリッド数
datasize	1グリッドあたりのデータサイズ
graphmapid	位相地図 ID

固有の座標系における, 定量地図の原点座標を保持している. type はその定量地図が示す空間のラベル(建物, 階, 部屋など)を保持するためのものであり, また content において, 倉爪らの手法 [6] 等で得られた 実際の定量地図データを, グリッドの高さを示す Multi-layer DEM に変換して保持している. さらに, graphmapid で関連する位相地図 ID を保持しており, 定量地図から位相地図の参照が可能となっている.

3.1.2 位相地図

位相地図情報データベースを Table 2 に示す. 位相地図情報データベースでは, grid_mapid として関連する複数の定量地図 ID を保持しており, 位相地図から定量地図の参照が可能となっている. また, 隣り合う位相地図の ID は, linked_graphmapid で保持しており, さらに建物, 階, 部屋などの階層構造を表すために hgraphmapid に上位の位相地図 ID を保持する.

3.2 物品情報データベース構造

物品情報を管理するための物品情報データベースの一部を Table 3 に示す. 物品情報には, タグ ID や名前だけでなく物品の位置座標も保存されている. 位置座標の値は, 関連する定量地図座標系内の 3 次元座標で示される. また, 物品の収納されている収納庫やデスクなどの位相地図 ID を additional に保存しており, 物品情報から収納されたノードの参照が可能となっている.

4. 物品情報と地図情報の連携

TMS が管理する物品情報と地図情報は相互に連携している. 例えば, 物品情報は物品の置かれているノードの位相地図 ID を保持している. 物品取得サービスを行う際, この位相地図のノードを目的地とし経路計画を行う.

Table 2 Topological map class

項目名	項目内容
graphmapid	位相地図 ID
type	種類 (建物, 階, 部屋, 収納庫)
status	状態
hgraphmapid	上位の位相地図 ID
coordid	座標 ID
grid_mapid	関連する定量地図 ID
linked_graphmapid	リンクしている位相地図 ID (建物, 階, 部屋, 収納庫)
x(yz)_graphmapid	リンクしている位相地図 ID の x(yz) 座標
belong_graphmapid	所属しているグラフ地図 ID (建物, フロア, 部屋, 収納庫)
belong_tagid	所属しているタグ ID
belong_visionid	所属しているビジョン ID

Table 3 Object class

項目名	項目内容
tagid	タグ ID
name	名前
type	種類
coordinate_x(y,z)	x(y,z) 位置座標
size	サイズ
additonal	付加情報 (所属空間)

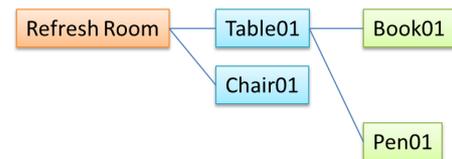


Fig.1 階層構造

4.1 階層構造

一般に物品の置かれる状況は以下のような状態である.

- 物品が収納庫やデスクに置かれている
- その収納庫やデスクが部屋に置かれている

このような状況に置かれた物品と環境構造の繋がりには, Fig.1 のような階層構造となる. この階層を表現するために, 各ノードには上位または下位のノード ID を格納できるようにしている.

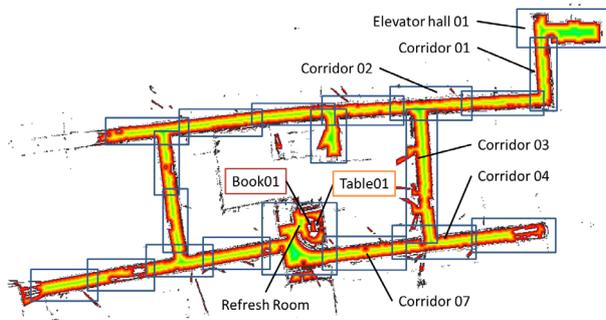


Fig.2 地図データの分割

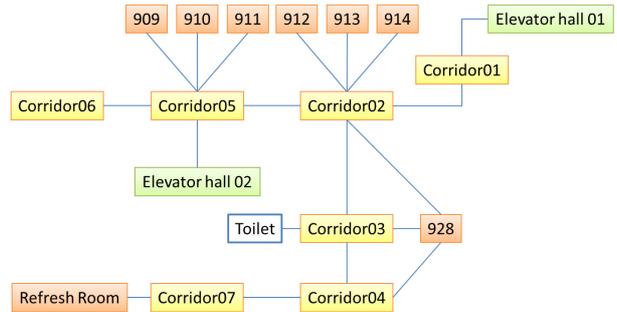


Fig.4 位相地図

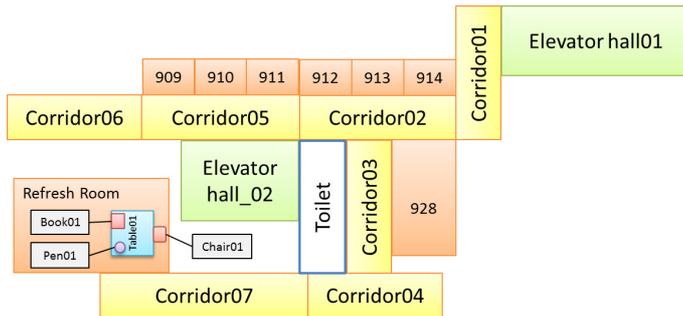


Fig.3 定量地図

Table 4 Topological map class (Refresh Room)

項目名	項目内容
graphmapid	Refresh_Room
type	部屋
status	0(0:立入可, 1:立入不可)
hgraphmapid	9floor
coordid	coord01
grid_mapid	Refresh_Room01
linked_graphmapid	Corridor07
x(yz)_graphmapid	200(200,200)
belong_graphmapid	Table01
belong_tagid	-
belong_visionid	-

5. TMS を用いた物品取得

本章では、TMS を用いた物品取得を、机上に置かれた本の取得を例に、格納する具体的なデータを用いて説明する。

5.1 地図データの分割

建物・フロアなど広大な空間を1つの定量地図データにすると、データサイズが大きくなってしまふ。ロボットのメモリや無線通信の伝送速度の制限から、地図データは出来る限り小さいことが望ましい。そこで、まず手動で地図データの分割を行う。

今回の実験では、建物のフロア(40m × 80m)のデータを用いた。Fig.2 に分割の例を示す。枠線で囲む領域を、1つの定量地図データとして登録する。全体を一つの地図データとした時、データサイズはバイナリ形式で4.57MBであった。これを19分割した時、定量地図データのサイズは、バイナリ形式で一つ当たり130~670KBであった。

5.2 データベース登録

TMS への物品情報と位相地図情報のデータベース登録について説明する。

5.2.1 地図情報の登録

例として Fig.3 のような、空間 Corridor07・Refresh Room・Table01 に置かれた物品 Book01 を考える。各空間は接続関係があり、例えば Corridor07 と Refresh Room は繋がっている。また、Refresh Room の空間内には Table01 が存在し、それらには階層関係がある。位相地図は、

このような空間の繋がりを Fig.4 のように表現したものである。経路計画には空間の連結関係情報が必要であり、この情報は位相地図の linked_graphmapid、および hgraphmapid に格納される。また隣接する位相地図への接続位置(リンク位置)は、x(yz)_graphmapid に格納されている。

Fig.1 に示すように、Book01 の格納ノードは Table01 である。これを表現するために、Table01 の位相地図情報の belong_tagid には book01 を格納している。また、Table01 の上位ノードは Refresh Room である。これを表現するために、Table01 の hgraphmapid には Refresh Room を、Refresh Room の belong_graphmapid には Table01 を格納している。

この Table01 と Refresh Room の位相地図情報を Table 4、Table 5 に示す。また、Refresh Room の定量地図情報を Table 6 に示す。

5.2.2 物品情報の登録

Book01 の登録例を Table 7 示す。Book01 を収納しているノードを示すために、additional に Table01 を格納している。

5.3 処理の流れ

以下に処理の流れを示す。

Table 5 Topological map class(Table01)

項目名	項目内容
graphmapid	Table01
type	収納庫
status	0(0:立入可,1:立入不可)
hgraphmapid	Refresh_Room
coordid	coord01
grid_mapid	-
linked_graphmapid	-
x(yz)_graphmapid	-
belong_graphmapid	-
belong_tagid	book01
belong_visionid	-

Table 6 Metric map class (Refresh Room)

項目名	項目内容
mapid	Refresh_Room01
coordid	9floor-coord
scale	1
type	0 (0:部屋,1:廊下,2:エレベータホール)
coord_x(yz)1	0(0,0)
coord_x(yz)2	5000(5000,0)
content	Multi-layer DEM
grid_x_width	100 (mm)
grid_y_width	100 (mm)
grid_x_counts	50
grid_y_counts	50
datasize	4 (byte)
graphmapid	Refresh_Room

1. 物品名の指定し, 物品情報データベースから tagid を取得
2. tagid から物品の収納された位相地図 ID を取得
3. 位相地図情報データベース内から位相地図を検索
4. 収納ノードの種類が「部屋」でなければ, hgraphmapid から上位ノードを検索
5. 物品の置かれた部屋のノードを目的地とする大域経路計画
6. 現在地を含む定量地図の取得と局所経路計画
7. ロボットの移動
8. 次の定量地図を TMS から取得
9. 目的地に到着するまで 6~8 を繰り返す

各処理について詳細な説明を以下に示す.

Table 7 Object class

項目名	項目内容
tagid	book01
name	Book01
type	book
coordinate_x(y,z)	100(100,100)
size	200
additional	Table01

5.3.1 物品名の指定し, 物品情報データベースから tagid を取得

まず, ユーザがサービスロボットに取得して貰いたい物品名を指定する. 次に物品名を検索キーとして, 物品情報データベース内を検索し, 主キーである tagid を取得する.

5.3.2 tagid から物品の収納された位相地図 ID を取得

物品情報の additional に格納されている位相地図 ID を取得する.

5.3.3 位相地図情報データベース内から位相地図を検索

取得した位相地図 ID を元に該当する位相地図データを検索する.

5.3.4 収納ノードの種類が「部屋」でなければ, hgraphmapid から上位ノードを検索

大域経路計画には現在地と目的地の部屋の位相地図 ID を用いる. もし物品を収納しているノードの種類が「収納庫」や「デスク」など「部屋」と同じ階層のノードでなければ, その上位ノードを検索し, 「部屋」階層のノードを取得する.

5.3.5 物品の置かれた部屋のノードを目的地とする大域経路計画

ロボットの現在地と目的地の位置情報から, 位相地図を用いて経路計画を行う. 現在地は, ロボット搭載センサや外部のセンサから取得し, TMS に格納されている. 目的地は, ロボットに物品を取得させる場合, 物品位置が目的地となる. この探索には位相地図を用い, グラフ探索により目的の位相地図の ID を探索する. Table 4 の linked_graphmapid には, 接続している位相地図の ID が, また x(yz)_graphmapid には次の位相地図の卒族部の座標が格納されている. 目的地の位相地図が見つかるまでリンクを辿り, 経路を見つける. この例では Corridor03 から Book01 の置かれた Table01 までの大域経路計画を行う (Fig. 2).

例) Corridor03 Corridor04 Corridor07 Refresh Room Table01

5.3.6 現在地を含む定量地図の取得と局所経路計画

ロボットの実際の移動には, ロボットの現在地を含んだ定量地図を TMS から取得する必要がある. そこで, ロ

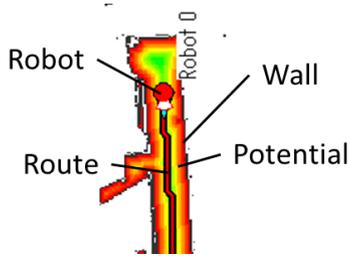


Fig.5 局所経路計画

ロボットの現在地と、定量地図の範囲を示す Table 6 の $coord_x(yz)1,2$ の座標を比較する。この座標内に現在位置が含まれていれば、その定量地図の Multi-layer DEM を取得する。もし、取得した地図に次の位相地図の接続部が含まれない場合、接続部が含まれるまで周辺の定量地図を取得する。取得した定量地図にポテンシャル場を生成し、壁から十分に離れた局所経路計画を行う (Fig.5) .

5.3.7 ロボットの移動

局所経路計画で作成した経路に沿い、TMS から他の移動体情報等も用いて、ロボットを移動させる (Fig.6(a)) .

5.3.8 次の定量地図を TMS から取得

空間の連結座標まで到着すると、隣の位相地図の情報を読み込む。Table 4 の `linked_graphmapid` を参照し、隣の位相地図の ID を取得する。この位相地図 ID を用いて、処理手順 6 ~ 8 を同様に実行する (Fig.6(b)~(d)) .

6. まとめ

サービスロボットのための環境情報構造化データベースであるタウンマネジメントシステム (Town Management System, TMS) を用い、保持している地図情報と物品情報を相互に連携してロボットによる物品取得サービスを実現する手法を示した。

謝辞 本研究の一部は文部科学省科学研究費補助金基盤研究 (B) (課題番号 23360115) の支援を受けた。

文献

- [1] 長谷川, 環境プラットフォーム「ロボットタウン」ム, 日本ロボット学会誌, Vol.26 No.05, pp.23-26, 2008.
- [2] 長谷川, 人間共生広域環境の情報構造化, 計測自動制御学会誌 計測と制御, Vol.49, No.6, pp.361-366, 2010.
- [3] 倉爪 亮, ロボット共生環境のための環境情報計測技術, 計測自動制御学会誌 計測と制御, Vol.49, No.6, pp.348-353, 2010.
- [4] 村上, 長谷川, 木室, 千田, 家永, 有田, 倉爪:”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.
- [5] 倉爪, 村上, 木室, 家永, 馬場 殷, 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

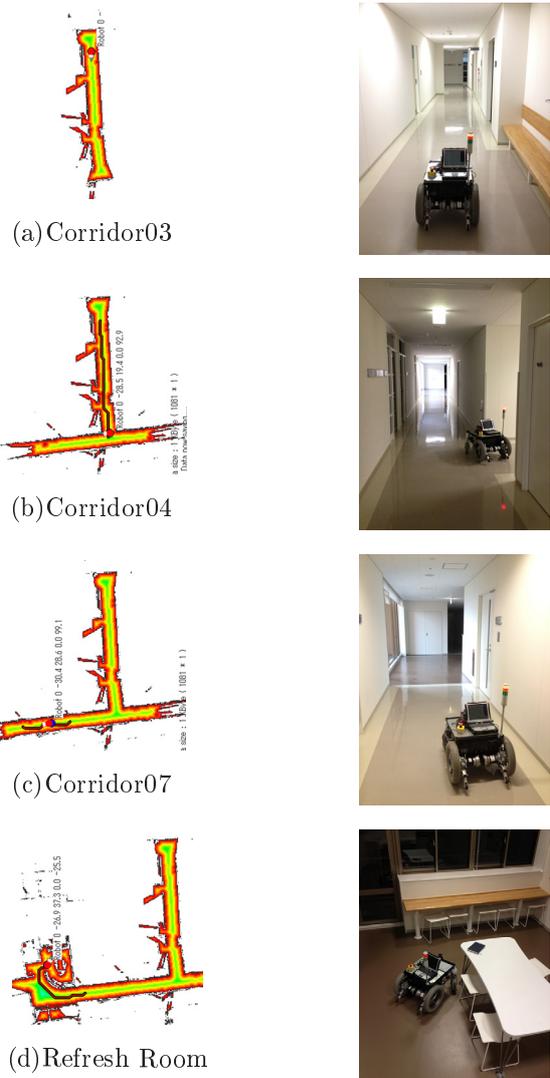


Fig.6 ロボットの移動

No.05, pp.27-31, 2008.

- [6] 倉爪, 戸畑, 村上, 長谷川:”Cps-slam の研究-大規模建造物の高精度 3 次元幾何形状レーザ計測システム-”, 日本ロボット学会誌, vol.25, no.8, pp.1234-1242, 2007.