

修士論文 2004年度(平成16年度)

再構成可能位置センサ群のためのロケーションモデル
管理機構

慶應義塾大学 政策・メディア研究科

村上朝一

tomo@sfc.wide.ad.jp

修士論文要旨 2004 年度 (平成 16 年度)

再構成可能位置センサ群のためのロケーションモデル 管理機構

論文要旨

本論文は、再構成可能位置センサ環境に対応する位置管理機構, SOL について述べる。現状における室内用位置情報システムは、天井や床といった固定された建造物の一部に取り付けられていることが多い。しかし近い将来、これらの位置情報システムがテーブルや棚といった可動物に設置された状態で利用されることも十分に予測される。このような動的な位置情報システムは、利用者がより柔軟にユビキタス環境を創出する手助けとなる反面、移動するたびにその位置情報を再入力するという作業を必要とする。

従来のシステムにおいては、センサの移動が検知されず、結果として測定・利用される位置情報に重大な誤差が含まれることも少なくなかった。SOL による自動・自律的位置情報管理システムは、この問題の解決策として考案された。本機構の導入により、ユビキタス環境における継続的なアプリケーション利用が可能となる。

本論文においては、位置情報に関する先行研究の概観を踏まえ、新しいロケーションモデル管理機構の必要性が検討されている。関連する概念や用語が定義された後、SOL の利用が想定される環境やアプリケーション、ならびに本システムに求められる機能要件が提示される。次に、本位置情報管理機構の設計および実装を述べた後、本システムを定性・定量の両側面から評価する。最後に、本研究の問題点と今後の課題を考察する。

キーワード：

1 ロケーションモデル 2 位置情報 3 ユビキタスコンピューティング 4 状況適応
5 アドホック位置センサ

慶應義塾大学 政策・メディア研究科
村上朝一

Abstract of Master's Thesis

Academic Year 2004

A Location Model Management System for Dynamic Sensor Environments

Abstract

In this thesis, SOL, a location model management system for dynamic sensor environments is proposed. The majority of existing indoor location systems are embedded in fixed environmental units such as ceilings or floors. Yet in the near future, those positioning systems-not the tagged objects to be located-could be mounted and used on mobile units such as tables or consoles. Such dynamic (i.e., mobile) positioning systems would allow the user to construct a more flexible ubiquitous space, yet at the same time, require highly complex manual configuration each and every time the location systems are relocated. At present, however, most of these moving sensors remain unnoticed, which could create critical errors in measurement. In order to tackle this problem, SOL is designed in such a way that it autonomously and automatically detects moving location systems and / or reconfigures the subsequent location model. As such, the proposed location model management system ensures uninterrupted / continuous tracking in the ubiquitous computing environment.

In this thesis, the status quo of the location management research is first surveyed. Based on the review of the current research, a new location model management system, SOL, is proposed. After relevant terms and concepts are defined, the thesis turns to the discussion of environments, applications, and requirements for the proposed system. Then, the design and implementation of the location model management system are presented, followed by both qualitative and quantitative evaluations of the proposed system. The thesis ends with a conclusion where limitations of this research as well as directions for future research are discussed.

Keywords:

1 ubiquitous computing 2 location model management 3 location information 4 context awareness
5 ad-hoc location sensing

Tomokazu Murakami
Graduate School of Media and Governance, Keio University

目次

第1章	序論	1
1.1	位置情報システムの現状	2
1.1.1	位置センサ多様性への対応	2
1.1.2	位置センサ移動への対応	4
1.2	問題意識と目的	5
1.2.1	問題意識	5
1.2.2	目的	6
1.3	本論文の構成	6
第2章	再構成可能位置センサ環境	7
2.1	用語定義	8
2.1.1	位置情報システム	8
2.1.2	位置センサ	9
2.1.3	位置管理機構	10
2.2	再構成可能位置センサ環境	11
2.2.1	想定環境	11
2.2.2	想定アプリケーション	12
2.2.3	位置管理機構の機能要件	13
2.3	本章のまとめ	14
第3章	再構成可能位置センサ環境に対応する位置管理機構	15
3.1	多種位置センサの管理	16
3.1.1	対象空間のモデル化	16
3.1.2	センサ検知空間のモデル化	16
3.1.3	位置情報の算出	17
3.2	センサ位置の管理	17
3.2.1	センサ位置の検知	18
3.2.2	センサ位置の推定	19
3.3	本章のまとめ	21
第4章	設計	22
4.1	全体概要	23
4.1.1	ハードウェア構成	23
4.1.2	ユースケース	23

4.1.3	ソフトウェア構成	24
4.1.4	動作概要	24
4.2	各モジュールの設計	26
4.2.1	センサモジュール	26
4.2.2	測定位置処理部	28
4.2.3	センサ位置管理部	29
4.2.4	モデル管理部	30
4.2.5	位置情報提供部	31
4.3	本章のまとめ	31
第5章	実装	32
5.1	実装概要	33
5.2	各モジュールの実装	33
5.2.1	センサモジュール	33
5.2.2	測定位置処理部	34
5.2.3	センサ位置管理部	35
5.2.4	モデル管理部	35
5.2.5	位置情報提供部	36
5.3	位置センサシュミレータ	38
5.4	本章のまとめ	38
第6章	評価	40
6.1	センサ位置推定機能の評価	41
6.1.1	想定環境	41
6.1.2	測定結果	41
6.2	関連研究との比較	42
6.2.1	位置センサの多様性に対応するシステム	44
6.2.2	センサの移動に対応するシステム	45
6.3	本章のまとめ	45
第7章	結論	46
7.1	今後の課題	47
7.1.1	多粒度位置情報の提供	47
7.1.2	多様な行動モデルを用いた評価	47
7.2	まとめ	47

目 次

1.1	多様な形状のタグ	2
3.1	センサ検知空間のモデル化	17
3.2	位置情報の算出	18
3.3	センサ位置の検知	19
3.4	センサ位置の推測	21
4.1	配置図	24
4.2	SOL のユースケース図	25
4.3	ソフトウェア構成	26
4.4	測定位置取得時のシーケンス図	27
4.5	位置情報問い合わせ時のシーケンス図	28
4.6	センサ位置推定	29
5.1	SensorModule パッケージのクラス図	34
5.2	センサ位置管理部のクラス図	35
5.3	モデル管理部のクラス図	36
5.4	クラス図	37
5.5	位置センサシミュレータ	39
6.1	センサ移動の検知率	42
6.2	推定センサ位置の誤差	43
6.3	誤差の分散	43

表 目 次

4.1	センサ登録情報	27
4.2	測定位置情報	27
5.1	SensorModule のインタフェース	34
5.2	DetectedLocation クラスのメソッド	34
5.3	LocationEvent クラスのインタフェース	37
5.4	TagEventListener インタフェースのメソッド	37
5.5	SensorEventListener インタフェースのメソッド	37
5.6	LocationEventSupplier クラスのインタフェース	37
5.7	LocationManager クラス	38
6.1	実験に使用した計算機	41
6.2	既存研究との機能比較	44

第1章 序論

本章では，本研究の背景をなす，位置情報システムの現状について説明する．次に，本研究の問題意識と，目的を述べる．

1.1 位置情報システムの現状

本研究の背景をなす，室内を想定環境とした位置情報システムの現状について述べる．近年，様々な位置特定技術を用いた位置センサが開発されている．また，位置センサが取得した位置情報を管理し，アプリケーションに提供する位置管理機構も多数開発されている．ソフトウェア開発者は，このような位置管理機構の利用により，さまざまな種類の位置利用アプリケーションを容易に実現できる．

既存の位置情報システムには，位置センサの多様性に対応するものと位置センサの移動に対応するものがある．以下，それぞれの機能を有する位置情報システムについて詳しく述べる．

1.1.1 位置センサ多様性への対応

現在，多数の位置利用アプリケーションが開発されている．アプリケーション毎に必要なとされる位置センサの形状と位置情報の粒度が異なるため，アプリケーションの多様性に対応する位置管理機構の構築には，多様な位置センサへの対応が不可欠である．

位置取得方法の要件

多くの位置センサは，机や棚といった検知対象にタグを付与し，タグ位置の検知により対象の位置を特定する．対象はアプリケーションによって異なるため，タグも，それに対応した形状のものがようになる．その結果，現在，シール状のタグ，数センチ四方のタグなど様々な形状やサイズが存在する（図 1.1）．例えば，家具や家電製品など大きな検知対象と数センチ四方のタグや，携帯品や書籍などの小さな検知対象とシール状のタグといった組み合わせがすでに存在している．

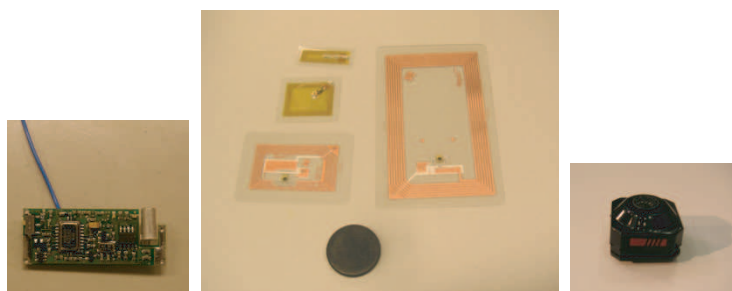


図 1.1: 多様な形状のタグ

また，必要とされる位置情報の粒度もアプリケーションにより異なる．そのため，多様なアプリケーションの実現には，多粒度の位置情報が必要である．センチメートル単位の小粒度から，メートル単位の大粒度まで様々な位置情報が利用される．例として，人の入退室管理に必要な大粒度の位置情報，商品の在庫管理に必要な中粒度の位置情報，ま

た，人の手の動作を用いたユーザインタフェースに必要な小粒度の位置情報などが挙げられる．

多様な位置取得の実現

取得できる位置情報の粒度と検知するタグの形状は，位置センサにより異なる．比較的大きなタグの位置を小粒度で検知するセンサや，比較的小さなタグの位置を大粒度で検知するセンサなどがある．既存の例として，数センチ四方のタグを利用し，センチメートル単位の位置を取得する超音波センサ [29][24] や，電波強度を用いてメートル単位の位置を取得するセンサ [7]，また，シール状のタグを利用し数十センチメートル単位の位置を取得するセンサ [25] などがある．

多様な位置利用アプリケーションの実現には多種の位置センサが必要なため，今後，位置利用アプリケーションの普及に伴い，人々の生活空間に多種の位置センサが混在するようになると考えられる．この結果ユーザは日常的に複数の位置利用アプリケーションを利用することが予想される．

センサ多様性に対応する位置管理機構

アプリケーションやタグ，センサのみならず位置センサの多様性に対応する位置管理機構も多数開発されている [10][14][11][15][23][20]．これらの位置管理機構は多種センサによって取得されるさまざまな位置情報を管理し，アプリケーションに対して位置情報に関する統一的な API を提供する．これにより，多種の位置センサを利用するアプリケーションの開発が容易になる．

しかしその一方で，センサ多様性に対応する既存の位置管理機構は，位置センサの移動に対応できない．既存の機構は位置センサが移動しないことを前提としているが，現実には位置センサが移動する場面は少なくない．位置センサの設置対象は天井，壁や床といった，通常移動することは考えにくい建造物の一部だけとは限らず，家具や家電製品といった移動可能なものを含む場合がある．この場合には，以下のような問題が発生する．

位置センサが検知対象と共に移動した際に誤差が生じる．位置センサが移動した場合，従来型のシステムにおいては，改めて位置情報を再設定しない限り，誤差を含んだ位置情報を取得し，それをアプリケーションに提供し続けることになってしまう．

この誤差を回避するには，既存システムの場合，位置センサを設置後，位置管理機構にセンサ設置場所の入力作業，あるいはそれに代わる作業が必要である．多くの場合この作業は利用者にとって負担となる．このような負担を軽減し，利便性を向上するためには，センサ設置後の設定作業を容易にする必要がある．

1.1.2 位置センサ移動への対応

すでに位置センサ移動に対応する位置情報システムが開発されている．位置センサが設置対象と共に移動した場合，位置情報システムがセンサ移動の検知や，自動再設定を行うことで，取得位置情報の誤差を小さくできる．しかし，センサ移動に対応する既存システムは，位置センサ間相対関係の検知機能を持つ単一のセンサのみを対象にしている．

位置取得方法の要件

位置センサが移動し，かつ位置管理機構がセンサの移動を認識できない場合，位置管理機構がアプリケーションに提供する位置情報に誤差が含まれる．利用する位置情報に誤差が含まれると，アプリケーションが適切に動作できない場合がある．位置センサ移動時の誤差を減らすこと，あるいは，誤差を含む可能性をアプリケーションに通知することで，アプリケーションの動作が安定し，信頼性が向上する．

位置情報システムの多くは位置センサを検知対象に取り付けた後，その設置場所を入力する必要がある．このような初期設定時の手間は，システム導入者にとって負担である．そして，システム導入に要するコストはシステム普及の妨げとなる可能性がある．本研究の主眼である位置センサ設置後の設定作業を減らすことで，位置管理機構および位置利用アプリケーションの導入コストが下がり，結果として各種アプリケーションの普及が促進されるという結果が期待できる．

動的位置センサの実現

超音波や赤外線等を送受信し，互いの位置関係を把握するセンサが開発されている[21][13]．これらは相互の位置関係を絶え間なく把握する機能を有するため，センサが移動した場合でも，移動の検知と移動後の位置計算が可能である．センサ同士の位置関係は，近接する複数のセンサとの距離計算により把握している．

既存の動的位置センサのモデルは，対象に付与されたタグを検知するタイプではなく，一様なセンサを対象に直接付与し，検知対象同士の位置関係を求めるものである．つまり，センサの移動に対応する既存の位置情報システムは，一様なセンサデバイスで構成されている．これは，対象に付与するデバイスの形状と位置情報の粒度が一様であることを意味する．また，各センサデバイスは，高い計算能力や多種の発信・受信モジュールを備えるため，サイズが大きい．そのため，以下の問題が生じる．

動的位置センサの位置管理機構

上述の動的位置センサの各タイプに対応する専用の位置管理機構が開発されている[21][13]．アプリケーション開発者はこれらの位置管理機構を利用することで，位置センサ対応型アプリケーションを開発できる．

しかし従来のシステムでは，比較的大きな単一種のデバイスを検知対象に付与する必

要があり，多種の位置センサを利用する場合より，実現可能アプリケーションの種類が限られる．また，位置管理機構が他センサの利用を想定していないため，その場合は，さらに別の位置管理機構が必要である．加えて，単純に複数の管理機構を併用するだけでは，センサ設置条件に制約が生じる．

このような問題点に加え，位置情報システムが一種類のセンサで構成されており，複数粒度の位置情報を取得できないという制限も存在する．そのため，多種の位置センサを利用し，複数粒度の位置情報を取得する場合と比較し，実現できるアプリケーションの種類が限定される．センサが取得する位置情報の粒度が十分に小さい場合は粒度を大きく変換する機能を実現すれば，多粒度の位置情報を提供できる．しかし，単純な粒度変更だけで解決できない問題も残る．

1.2 問題意識と目的

今後想定される，複雑かつ動的な位置センサ環境において多様なアプリケーションを実現する際，既存位置管理機構では解決できない問題がある．本研究の目的は多粒度の位置情報に対応し，かつ，センサ移動時の誤差を減らす位置管理機構の構築である．以下に既存の位置情報システムに見られる問題点を整理し，本研究の目的と意義を述べる．

1.2.1 問題意識

現在，同一空間内で同時利用可能な位置利用アプリケーションが多数開発されており，一人のユーザが複数アプリケーションを同時に利用する可能性がある．また，位置センサを家具，家電製品やその他のモバイルデバイスに設置するアプリケーションも開発されている．家具や家電製品は移動する可能性があり，モバイルデバイスは移動を前提としている．これは今後，同一空間内に多様な位置センサが混在し，それらのすべて，あるいは一部が移動する環境が想定されることを意味する．

位置センサの多様性に対応するシステムの問題点

センサの移動を検知できず，取得位置情報に誤差を生じる可能性がある．結果として，アプリケーションの信頼性を損なう危険性がある．また，センサ設置時にその設置場所に関する位置管理機構の設定が必要である．この場合，位置センサに関する専門知識を持つ開発者や管理者がシステムの導入・維持を行うのであれば労力コスト以外の問題はない．しかし，家庭や職場などにおいて，専門知識を持たないユーザがシステムを導入・維持する場合，この設定の必要性は深刻な問題となりうる．

位置センサの移動に対応するシステムの問題点

位置情報粒度とタグ形状の多様性に対応できないため、位置利用アプリケーションのセンサに対する多様な要求に応えることができない。一種類の比較的大きなデバイスを検知対象に付与する必要があるが、物理的制約から小さな検知対象に付与することは難しい。また、デバイスが比較的高価なため、付与できる個数にコスト的制約も生じる。さらに現状では単粒度への対応しか実現されてないため、複数粒度の位置情報を処理できない。

1.2.2 目的

本研究の目的は、多様な位置センサが存在しかつそれらが移動する環境で利用可能な、統一的な位置管理の実現である。これにより、アプリケーション開発者は複種の位置センサを利用するアプリケーションを容易に開発できるようになる。また、センサ移動の検知・対応が必要なアプリケーションを、高価な移動対応センサを使わずに実現できる。同様に、一般ユーザもセンサの移動を気にかけることなく、さまざまなアプリケーションを同時に利用できるようになる。

センサ多様性への対応

多種の位置センサから多粒度の位置情報を取得し、アプリケーションに提供する。また、異種センサ間の位置関係を把握し、異種センサで検知した対象同士の位置関係を計算する。これにより、多種の位置センサを同時に利用するアプリケーションを開発できる。

センサ移動への対応

センサの位置を検知する専用のデバイスを用いることなく、センサの位置を把握できる。これにより、センサの移動が生じた際に、それを検知し、センサ移動が招いた取得位置情報の誤差を減らす。また、センサ設置後の設定作業を簡易にできる。

1.3 本論文の構成

2章で位置センサの種類と移動を詳細に分析し、3章で位置センサの多様性と移動に対応する位置管理機構の実現手法を説明する。4章で本システムの設計について、5章で実装について述べる。6章で本システムの定性的・定量的評価を行い、7章で結論を述べる。

第2章 再構成可能位置センサ環境

本章ではまず，本稿で使用する用語を定義する．位置情報システムを構成する位置センサと位置管理機構，位置センサを構成するセンサとタグを定義する．さらに，位置管理機構が取り扱う，ロケーションモデルと位置情報を定義する．

次に，本研究が想定する，室内空間に多種の位置センサが混在する環境について述べる．また，多種の位置センサの，多様な対象への設置を必要とする，想定アプリケーションについて述べる．最後に，想定アプリケーションを実現する位置管理機構の機能要件を述べる．

2.1 用語定義

本稿で使用する用語を定義する．まず，位置情報システムを構成する位置センサおよび位置管理機構について述べる．次に，位置センサを構成するセンサとタグを定義する．最後に，位置管理機構が取り扱う，ロケーションモデルと位置情報について述べる．

2.1.1 位置情報システム

人や物などの位置を特定し，位置情報をアプリケーションに提供するシステムを位置情報システムと呼ぶ．位置情報システムは，多くの位置利用アプリケーションが一般に必要とする機能を，簡易な API を通じて提供する．位置情報システムの利用により，位置利用アプリケーションの開発コストを低く抑えることが可能となる．現在，このような位置情報システムが多数開発されている．例として，Cricket[24]，ActiveBats[29] や Active Badge[28][27] などが挙げられる．

位置情報システムは位置センサと位置管理機構の 2 モジュールで構成されている．位置センサは人や物などの位置を測定するデバイスである．位置管理機構は，複数の位置センサが求めた測定位置をネットワークを介して取得し，管理する．位置センサと位置管理機構はどちらも多数開発されている．

位置センサ

位置センサは，超音波，電波や赤外線などを利用し，人や物の位置を測定するデバイスである．超音波，電波や赤外線などを送受信するハードウェアと，音波の到達時間や電波強度などを処理し，測定位置として他の機器に送信するソフトウェアで構成される．

現在，多種の位置センサが開発されているが，これらは物理的形狀，取得できる測定位置の種類や粒度などがそれぞれ異なる．位置センサの例として IS600[17]，RF-Code[25] などが挙げられる．IS600 は超音波を利用し，細かい粒度の位置を測定する．RF-Code は電波を利用し，粗い粒度の位置を測定する．

位置管理機構

位置管理機構は，複数の位置センサが取得した測定位置を集中的に管理し，アプリケーションに提供するソフトウェアである．通常，PC や WS 上で動作し，位置センサからネットワーク経由で測定位置を受け取る．また，位置センサから受け取った測定位置に加え，現実世界を座標空間やツリー構造などで表現したロケーションモデルや，位置センサが設置された場所に関する情報などを管理する．

現在，多種の位置管理機構が開発されており，想定環境，利用可能位置センサや対象アプリケーションなどがそれぞれ異なる．位置管理機構の例として，Location Stack[15] や Geocast[10] などが挙げられる．Location Stack は屋内を想定環境とし，多様な位置センサによって取得した測定位置を，対象の存在可能性に変換してアプリケーションに提供す

る。Geocast は屋内外を想定環境とし、多種の位置センサによって取得した測定位置を階層的に管理し、アプリケーションに提供する。

2.1.2 位置センサ

本稿では、位置センサの構成デバイスを 2 種類（センサ、タグ）に分けて考察する。多くの室内用位置センサは、超音波、電波や赤外線などを 2 種のデバイス間で送受信し、両者間の距離を計測する。2 種のデバイスは通常、電源の有無やネットワーク接続インタフェースの有無などにより、大きさと重さが異なる。多くの場合、形状が大きく、重量が重いデバイス（センサ）を建造物や家具に設置し、もう一方のデバイス（タグ）を検知対象である人や物に取り付ける。

ソフトウェアが位置センサから取得できる測定位置は、センサの種類によって様々である。タグとの距離、センサを基準としたタグの座標、検出範囲内にタグが存在するか否かの情報などがその主なものである。座標を利用する場合、複数のハードウェアモジュールとタグ間の距離を計測し、ハードウェアモジュール間の位置関係を基準として計算する。一方、検出範囲内のタグの情報を利用する場合、センサから一定の距離以内にあるタグの情報を提供する。

センサ

建造物や家具などに設置され、タグを検知するデバイスである。通常、タグが発信した超音波、電波や赤外線を受信する（センサが発信し、タグが受信する場合もある）。音波や電磁波の到達時間や強度を測定し、検知対象に付与されたタグの相対位置を計測する。また、センサに対するタグの相対位置と、センサの設置位置を合わせて利用することで、ある空間内におけるタグの位置や、タグ間の位置関係などを計算できる。そのため、多くの位置管理機構は、センサの設置位置を保持している。

センサは、静的な対象だけでなく、動的な対象に設置する場合もある。例えば、家具への設置や、イベント会場における可動物への一時的設置などである [12][18]。この場合、センサが設置対象と共に移動する可能性がある。センサが移動した場合、ある空間内におけるタグの位置や、他タグとの位置関係を位置管理機構が正しく計算するためには、センサの新しい位置を把握する必要がある。

タグ

位置の検知対象に付与され、センサに検知されるデバイスである。超音波、電波や赤外線などを、センサに向けて発信する。多くの場合、割り当てられた識別子を音波や電磁波に載せて発信する機能のみを持ち、形状が小さく、重量が軽い。そのため、容易に人に持たせたり、様々な物に付与したりできる。

タグの形状は様々である。例えば、超音波を利用する IS600 のタグは約 1 センチ四方の

立方体であり，電波を利用する RFID では数ミリ四方の小さなタグや，シール状の薄いタグがある．また，タグの形状と，取得可能な測定位置の種類は共に，位置測定技術によって制約される．使用するタグの形状は，アプリケーションが要求する位置情報の種類と，検知対象を考慮して決定される．

2.1.3 位置管理機構

位置管理機構はロケーションモデルを管理し，位置情報をアプリケーションに提供する．ロケーションモデルとは，実世界の空間構造を記述した情報である．また，位置情報とは，位置センサが取得した測定位置を，ロケーションモデルをもとに変換した情報である．対象アプリケーションにより，位置管理機構が利用するロケーションモデルと提供する位置情報の種類が異なる．

例として，ある部屋 X に RFID センサ A が設置されている状況を挙げ，測定位置，ロケーションモデルと位置情報の関係を説明する．RFID センサは「検出範囲内にタグ α が存在する」という測定位置を位置管理機構に提供する．また，ロケーションモデルには「センサ A の検知範囲は部屋 X である」という情報が含まれている．そこで，位置管理機構は，測定位置とロケーションモデルをもとに，「タグ α が部屋 X に存在する」という位置情報をアプリケーションに提供する．

ロケーションモデル

実世界の空間構造を記述した情報である．ロケーションモデルには，いくつかの種類がある [34][9]．空間を座標系によって表現するメトリックモデル，識別子を付けた空間の包含関係をツリー構造で表現するトポロジカルモデルや，両者を利用するハイブリッドモデル [10] などがある．アプリケーションの要求する位置情報の粒度が細かい場合は，メトリックモデルを利用する場合が多く，粒度が粗い場合はトポロジカルモデルを利用することが多い．

また，ロケーションモデルには，センサの設置位置に関する情報が含まれる．センサの設置位置は，位置管理機構が測定位置を位置情報に変換する際に必要である．センサの設置位置は，システム管理者かユーザが位置管理機構に入力する場合が多い．しかし，センサ設置位置の入力作業を必要としない位置管理機構も存在する [21][22]．

位置情報

位置管理機構がアプリケーションに提供する，人や物の位置を示す情報である．メトリックモデルを利用する位置管理機構が提供する位置情報は座標であり，トポロジカルモデルを利用する位置管理機構が提供する位置情報は空間の識別子である．必要な位置情報の種類はアプリケーションにより異なる．例えば，ユーザの動作を認識するユーザインタフェースや，コンテキスト利用アプリケーションなどは，座標表現による詳細な位置情報

を必要とする場合が多い．また，ユーザの移動を認識する入退室管理システムや，物の場所を認識する物品管理システムなどは，空間の識別子による粗い粒度の位置情報を必要とする場合が多い．

また，位置情報には誤差が含まれる場合がある．位置情報に含まれる誤差は２種類ある．一方は，測定位置に含まれる，センサによる位置測定誤差である．もう一方は，測定位置を位置情報に変換する際に生じる，ロケーションモデルのセンサ設置位置の設定誤差である．誤差を多く含む位置情報は，位置利用アプリケーションの信頼性を損なうため，誤差の削減は重要な課題である．前者の誤差を減らすものとして，複数の位置センサによる測定位置を利用するタイプの位置管理機構が存在する [31]．

2.2 再構成可能位置センサ環境

本節ではまず，本研究の想定環境として，想定する空間と設置する位置センサについて述べる．次に，想定アプリケーション環境として，多種のアプリケーションが同時に利用される環境について述べる．その上で，想定アプリケーション環境の実現に必要な，位置管理機構の機能要件を述べる．

2.2.1 想定環境

本研究の想定する環境について述べる．情報科学の進歩により，今後様々な空間でユビキタスコンピューティング環境が実現すると考えられる [30]．すなわち，多種多様な，環境情報取得センサ（位置センサ，マイク，カメラ，温度・湿度センサや照度センサなど）やアクチュエータ（ディスプレイ，スピーカやネットワーク制御可能な家電機器など）が，屋内外を問わず設置されると考えられる．

本研究は，屋内に多様な位置センサが設置されている環境を想定する．現在，企業や大学の研究施設に，ユビキタスコンピューティング環境の実現に向けた実証実験空間が構築されている [26][1][3][19][8]．このような，多種の環境情報取得センサとアクチュエータを備えた室内空間は今後，研究施設だけでなく，一般に普及すると考えられる．

室内空間

本研究は，家庭，オフィス，教育施設や商業施設などの室内空間を対象とする．この室内空間に，ネットワーク制御可能なアクチュエータが多数設置されていると想定する．このような室内空間においては，アプリケーションが多数のアクチュエータを制御し，様々な状況で人々の活動を支援することができる．

現在，ネットワーク制御可能なアクチュエータが多数入手可能である．例として，ＡＶ機器，生活家電 [4][2] が既に市販されている．また，ネットワーク接続機能を持たない機器も万能リモコン [5] を介して制御できる．ネットワーク接続機能と計算処理能力を持った家具 [33][12][18]，家具を組み立てるマテリアル [16] などにもすでに存在している．

多種位置センサ

室内空間に、多種の位置センサが複数設置されている環境を想定する。また、これらのセンサが、ネットワークに接続されている環境を想定する。このような環境で、計算機器は、位置センサが求めた測定位置を、ネットワークを介して取得し、様々な位置利用アプリケーションを実現できる。

前述の通り、現在、超音波や電波などを利用する、多種の位置センサが入手可能である。前者の例として、IS600、超音波3次元タグ[32]が挙げられる。また、後者の例としては、RF-Codeや(ISO10536, ISO14443, ISO15693に準拠する)多種のRFリーダが入手可能である。また、ワイヤレスデータ通信に利用する、無線LANモジュールや赤外線通信モジュールも位置センサとして利用可能である[7]。

2.2.2 想定アプリケーション

本研究の想定するアプリケーションについて述べる。現在、屋内で使用する位置利用アプリケーションが多数開発されている。例として、ユーザの動作認識により機器を制御するアプリケーション、ユーザの移動履歴を用いて行動予測を行うアプリケーションや、物の現在位置を把握する物品管理アプリケーションなどが挙げられる。これらの位置利用アプリケーションは、それぞれ必要とする位置情報の種類が異なるため、そのニーズに対応する位置センサを必要とする。

本研究は、多種の位置センサを同時に利用するアプリケーションを想定する。また、建造物だけでなく、家電製品や家具、人などにもセンサを設置する状況を想定する。すなわち、位置の移動がほぼ無いに等しい前者に加え、高頻度での移動が予想される後者にもセンサが設置されている空間である。前述したように、タグの形状と大きさは位置センサによって異なる。したがって、人や物など、様々な対象の位置情報を利用するためには、多種の位置センサを利用し、多様な対象にセンサを設置する必要がある。

多種位置センサの同時利用

位置センサの利用コストが低下し、一般に普及すると、多種の位置センサを同時に利用する状況が発生することが容易に予想される。まず、同一空間において、複数の位置利用アプリケーションを利用する場合が考えられる。また、同一アプリケーションが、複数種類の位置センサを同時に利用する場合も十分想定可能である。

前者の例として、ユーザの入退室とジェスチャを認識する、機器制御アプリケーションが挙げられる。ユーザの入退室という情報に関しては粒度の粗い位置センサを用い、ジェスチャの認識には粒度の細かい位置センサを用いる。また、後者の例として、ユーザと物との距離を監視する、盗難・忘れ物防止アプリケーションが挙げられる。多くの場合、人を検知する位置センサと、物を検知する位置センサは異なるため、複数種類の位置センサを同時に利用する必要性が生じる。

センサ設置対象の多様性

センサを設置する対象は、アプリケーションによりさまざまである。壁や天井などの静的（移動不可能）な対象に設置する場合と、家電製品や家具などの動的（移動可能）な対象に設置する場合がある [6]。このような例として、人を検知するためのセンサを天井に設置し、物を検知するためのセンサを棚に設置する状況が考えられる。

家電製品や家具などの動的な設置対象は、センサ設置後、アプリケーション運用中に移動する可能性がある。部屋の模様替えの際などに起こりうる状況である。また、ユーザによる移動を前提に作られた、キャスト付きの家具（棚や机）を利用する場合にも、同様の状況が発生する。このように、動的な対象に設置したセンサは、日常的に移動する可能性がある。

2.2.3 位置管理機構の機能要件

多種の位置センサが混在し、かつ、センサが移動する環境を、再構成可能位置センサ環境と呼ぶ。再構成可能位置センサ環境では、多数のタグの位置を管理し、簡易な API を介してアプリケーションに位置情報を提供する、位置管理機構が必要とされる。位置管理機構の利用により、ロケーションモデルの管理や位置センサに依存する制御をアプリケーションごとに実現するコストが抑えられ、安価での開発が可能となる。

本節では、再構成可能位置センサ環境に対応する位置管理機構に求められる機能要件を述べる。想定環境において、異種タグ間の相対位置をアプリケーションに提供するためには、多種位置センサ管理機能と、センサ位置管理機能が必要である。以下、それぞれの機能について詳述する。

多種位置センサ管理機能

さまざまなセンサから取得する、多種の測定位置を管理し、位置情報としてアプリケーションに提供する機能である。測定位置の種類には、検知範囲内にあるタグの識別子、タグまでの距離や、タグの座標などがある。位置管理機構は、これら異種の測定位置を統一的に管理し、相互の比較や変換を行う必要がある。

多種位置センサを統一的に管理するためには、測定位置の種類に依存しないロケーションモデルを利用する必要がある。また、各センサが取得した測定位置を、それぞれのロケーションモデルに適応した位置情報に変換する必要も生じる。位置管理機構は、これらの機能を兼ね備えることにより、異種位置センサ間の相対位置の計算を可能にする。

センサ位置管理機能

センサの位置を管理し、移動に対応する機能である。センサが設置対象と共に移動した場合、実際の位置と、ロケーションモデル上の位置に不整合が生じる。その結果、ロケーションモデルを利用して計算した位置情報に誤差を生じることになる。この誤差は、アプ

リケーションの信頼性を損なう重大な問題を引き起こしかねない．このような不具合を防ぐため，センサ移動後も継続的に，アプリケーションに正確な位置情報を提供する必要がある．

センサ位置を管理するためには，センサの位置を正確に検知する必要がある．また，検知したセンサ位置を正しく反映するような方法で，ロケーションモデルを更新する必要がある．位置管理機構は，これらの機能により，センサが移動した場合に，測定位置から位置情報への変換に伴う誤差を減らすことができる．

2.3 本章のまとめ

本章ではまず，位置情報システムを構成する位置センサと位置管理機構，位置センサを構成するセンサとタグを定義した．さらに，位置管理機構が取り扱う，ロケーションモデルと位置情報を定義した．また，本研究が想定する，室内空間に多種の位置センサが混在する環境について述べた．その上で，多種のセンサを，多様な対象に設置する必要があるアプリケーションと，アプリケーションの実現に必要な位置管理機構の機能要件を述べた．

次章では，再構成可能位置センサ環境に対応する位置管理機構，SOL(Space mOdel management for dynamic Location sensor environment) の概要を述べる．

第3章 再構成可能位置センサ環境に対応する位置管理機構

本章では，再構成可能位置センサ環境に対応する位置管理機構である，SOLの位置管理手法について述べる．まず，本システムの多種位置センサ管理機能について説明する．次に，センサ位置管理機能について考察する．

3.1 多種位置センサの管理

SOL は、同一空間内に多種の動的（すなわち、移動可能な）位置センサが存在する環境に対応する位置管理機構である。本節においては、SOL の多種位置センサへの対応手法に関して論を進める。はじめに、本システムが採用する、座標表現のロケーションモデルについて述べる。次に、センサ検知空間のモデル化について考察する。最後に、ロケーションモデルと位置センサによる測定位置をもとにした位置情報の計算手法について詳述する。

本システムは、空間を単一の座標系で表現するロケーションモデルを使用する。また、各位置センサの検出範囲も座標系でモデル化し、センサが取得した測定位置を、ひとつの座標系にマップする。これにより、多様な位置センサの同時利用が可能となる。また、異種センサのタグを付与した検出対象同士の相対位置の計算を実現することができる。

3.1.1 対象空間のモデル化

本システムは、アプリケーションが対象とする空間（以下「対象空間」と呼ぶ）を、ひとつの座標系としてモデル化し、管理する。このモデルを対象空間モデルと呼ぶ。タグの位置情報は、対象空間モデル上の座標点として、アプリケーションに提供される。ただし、測定位置の粒度が粗い場合は、座標点ではなく、座標領域としてアプリケーションに提供する。

粒度の粗い位置情報を必要とするアプリケーションにとって、座標表現（メトリックモデル）の位置情報より、識別子表現（トポロジカルモデル）の位置情報の方が扱いやすい場合がある。しかし、メトリックモデルは、座標領域を指定し識別子を付けることで、トポロジカルモデルに変換することが可能である。したがって、本システムの内部ではメトリックモデルにより対象空間を管理しているが、トポロジカルモデルへの変換も可能である。

3.1.2 センサ検知空間のモデル化

本システムは、各センサの検知範囲内の空間（以下「検知空間」と呼ぶ）を、座標空間としてモデル化し、管理する。位置センサは、検出範囲内に存在するタグの位置を、センサに対する相対位置として取得する。本システムは、各センサの検出範囲を、対象空間と同様に、座標系としてモデル化して利用する（図 3.1）。このモデルを検知空間モデルと呼ぶ。この検知空間モデルの座標系は、センサの物理的な中心を原点とする。

SOL は、位置センサごとの検知空間モデルを保持し、センサから受け取った測定位置を、検知空間モデルを利用して座標領域に変換する。例えば、検出範囲内にあるタグの識別子を取得する位置センサの場合、測定位置を、検出範囲全体の座標領域に変換してアプリケーションに提供する。また、タグとの距離を求めるタイプのセンサの場合には、測定位置を、原点を中心とした球の表面上の領域に変換する。さらに、タグの（センサに対す

る) 相対座標を取得する位置センサの場合, 測定位置を, 座標上の特定の領域 (通常, 座標上の任意の点) に変換する。

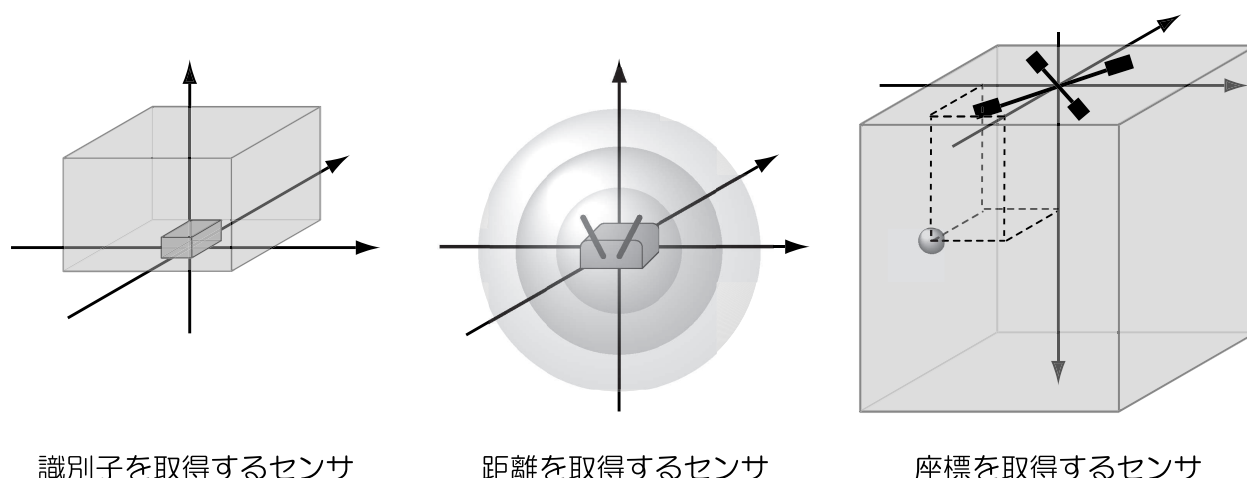


図 3.1: センサ検知空間のモデル化

3.1.3 位置情報の算出

本システムは, 位置情報を, 対象空間モデル上の座標領域としてアプリケーションに提供する。まず, センサが取得したタグの測定位置を, 検知空間モデル上の領域に変換する。次に, 検知空間モデル上の領域を, 対象空間モデルにおけるセンサの座標をもとに, 対象空間モデル上の領域に変換する。タグの存在する座標領域を, 以上の手順に従って求め, この情報をアプリケーションに提供する。この算出手順を図 3.2 に示す。

このような座標領域の提供に加え, 本システムは, 任意のタグが特定の座標領域に存在する確率という情報もアプリケーションに提供することができる。存在確率は, 1 を, 理論上タグが存在する可能性のある座標領域 (対象空間) の広さで割った数値である。位置情報の存在確率による表現は, アプリケーションの信頼性向上に貢献する。これは, タグの位置情報単独ではなく, その情報が正確である確率という情報と共に提供される方が, アプリケーション運用時の誤差の減少を助けるためである。この誤差の減少は, 検知対象が存在する可能性のある領域が広い場合に, より重要である。

3.2 センサ位置の管理

検知空間モデルにおけるタグの存在領域を, 対象空間モデルに変換する際, センサの位置が必要となる。センサの位置とは, 対象空間モデル上の, センサの存在領域のことを指す。2.1.3 で述べたように, センサが移動し, その実際の位置と, ロケーションモデル上の位置に齟齬が生じた場合, 算出した位置情報に誤差が発生することになる。センサが動的

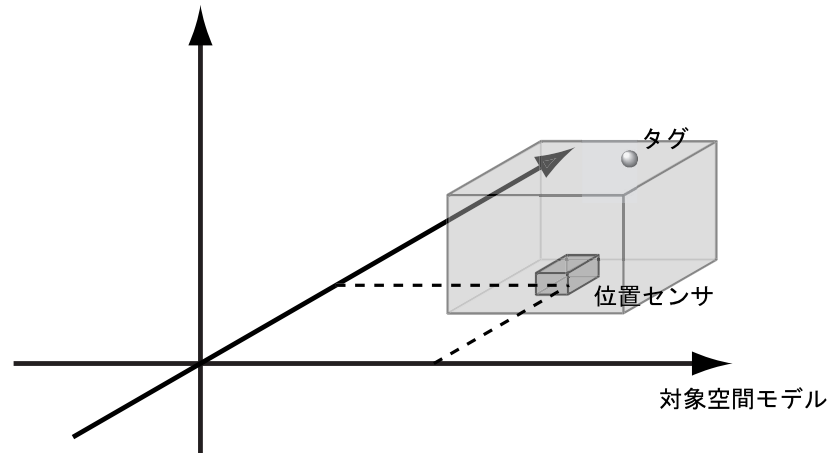


図 3.2: 位置情報の算出

対象に設置されている場合，このような位置情報の誤差が生じる可能性が飛躍的に増大するため，センサ位置の正確な管理が，位置情報管理機構に求められる重要な機能となる．

本システムは，センサの位置を動的に管理する．これはすなわち，センサが移動した場合でも，その移動前後の位置情報に含まれる誤差を最低限に抑えることが可能となることを意味する．動的な管理には，移動したセンサを他センサからの情報を利用して検知する方法や，タグから得られた情報を利用して，移動したセンサの位置を推定する方法などがある．このような方法を用いて更新されたロケーションモデル上においては，位置情報の変換に付随して生じる誤差を軽減することが可能となる．結果として，本システムは，複数のセンサが動的対象に設置された環境において，継続的な位置情報の提供を実現する．すなわち，センサが移動しても，通常その度に行う再設定が不要となり，これに伴う人的・時間的コストを省くと共に，一切の中断なくアプリケーションを利用することが可能となるのである．

3.2.1 センサ位置の検知

動的対象に設置したセンサを，他センサによって検知する方法である．動的対象に設置したセンサの位置を，静的対象に設置したセンサによって取得し，対象空間モデルにマップする．このような手法によるセンサ位置の検知方法を図 3.3 に示す．

この方法は，センサを，静的対象と動的対象の両者に設置することで実現可能となる．例として，天井に超音波センサを設置し，棚や机などに RFID センサを設置する場合が挙げられる．この場合，棚や机などの可動物に設置した RFID に，超音波センサのタグを付与することで，RFID のセンサ位置を検知する．

このような，他センサによる検知方法は，再構成可能位置センサ環境に対する単純な解決策である．しかしその一方で，現実場面での諸制約により，このような解決が可能でない状況も十分に想定できる．例えば，検知対象と天井のあいだの距離が 10 メートルを超

える空間（吹き抜けのロビーや階段）においては，天井に設置すべきセンサが，対象物に取り付けられたタグの発する超音波を検知できなくなってしまう．あるいは，短期間の展示などの目的で，商業見本市会場を一時的に利用する場合など，物理的あるいは美観上の制約により，天井にセンサを取り付けることが不可能な状況が発生することもある．さらに，建造物へのセンサ設置が可能な条件下であっても，一時的利用のための設置と撤去に伴うコストという観点から，このような解決方法が最善とはならない場面も予想される．このような場合においては，次項で述べる推定によるセンサ位置算出の方が，より現実的な対応となりうる．

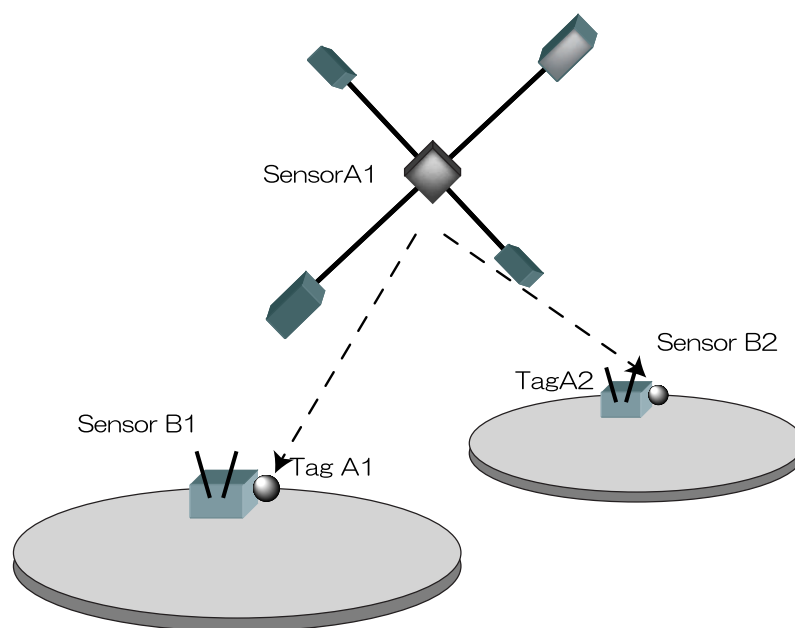


図 3.3: センサ位置の検知

3.2.2 センサ位置の推定

タグの測定位置から，移動したセンサの位置を間接的に算出する方法である．この場合，既存のシステムにおいては，センサの移動はタグの移動として誤認される．しかし，本システムは，複数のセンサに検知されている任意のタグの測定位置を利用し，移動したセンサの位置を推定する．このような手法によるセンサ位置の推定方法を図 3.4 において説明する．

図中の $A \cdot B$ はともに，座標の測定位置を取得してアプリケーションに提供するタイプの位置センサである．まず，ある時系列上の点（瞬間） $T1$ において，これら 2 つのセンサが，同一のタグ α の位置を求める．センサ A は，タグ α の位置を $(x1, y1, z1)$ と測定し，

センサ B は (x_2, y_2, z_2) と測定する．この 2 つの情報から，瞬間 T_1 におけるセンサ A とセンサ B の相対位置関係は， $(x_1 - x_2, y_1 - y_2, z_1 - z_2)$ と表現することができる．

次の瞬間 (T_2)，センサ B が示すタグ α の位置が (X_2', Y_2', Z_2') と変化する．この変化は複数の解釈が可能である．一つには，センサ $A \cdot B$ はともに不動で，タグ α が動いた結果，このような変化が生じたとみなすことができる．この捉え方は，センサの移動を想定しない，あるいは移動に対応できない，旧来の位置情報システムにおいて主流となっている．

もう一つの解釈は，動いたのはタグ α ではなく，センサ B であるとする考え方である．上記の既存システムにおいては，センサは不動とする解釈に支配されるため，このような状況を把握することができず誤差が生じる．つまり，移動したセンサの実際の位置と，ロケーションモデル上の位置に齟齬が生じ，算出した位置情報に誤差が発生することになる．このような誤差はアプリケーション運用上の障害となるため，本システムにおいては，その回避を目的とし，以下のような手順をとる．

センサ B によるタグ α の測定位置に変化が生じた場合，「タグ α が動いた」という結論に達する前に，同じタグ α のセンサ A による測定位置との比較を行う．この場合，センサ A の測定位置にも変化が生じている場合には，タグ α が動いたという解釈が妥当と考えられる．一方で，センサ A による測定位置には変化がない場合，すなわち，センサ A は依然としてタグ α の位置を (x_1, y_1, z_1) と報告してくる場合，動いたのはタグ α ではなく，センサ B の方であると結論付ける．ここからさらに，センサ B が移動した結果，センサ $A \cdot B$ の相対位置関係が， $(x_1 - x_2', y_1 - y_2', z_1 - z_2')$ に変化したと推定することも可能である．本システムは，これらの情報を，必要に応じアプリケーションに提供する．

このタグの測定位置を利用した動的センサ位置の推定という手法が成立するためには，少なくとも 2 つの条件が必要である．第一に，該当するセンサが取得する測定位置の粒度が細かくなくてはならない．粒度が粗いと，センサ位置の推定が発散してしまうためである．第 2 の条件としては，該当するセンサ同士の検知空間が重複する領域に，最低 1 つのタグが存在していることである．この手法では，任意のタグが複数のセンサに検知された際の位置情報の一致・不一致を基準に，移動したのはセンサかタグかという判断を行う．そのため，重複領域にタグが 1 つもない場合には，そこから得られる情報をもとにセンサの移動を検知することができなくなってしまう．したがって，対象空間内に存在するセンサとタグの数が不十分な場合には，この手法の実現が困難になることが予想される．

しかし一方で，センサとタグの数さえ確保できれば，前項で述べたような物理的あるいは美観上の制約から他センサによる位置検知が難しい状況下であっても，この手法により再構成可能位置センサ環境への対応を図ることができる．また，今後開発が進み，センサやタグが一層低価格で利用可能となれば，一時的設置や撤去に要する人件費などよりも格段に低コストで，複数種類の動的センサが混在する環境における，多様なアプリケーションの実現が可能となる．

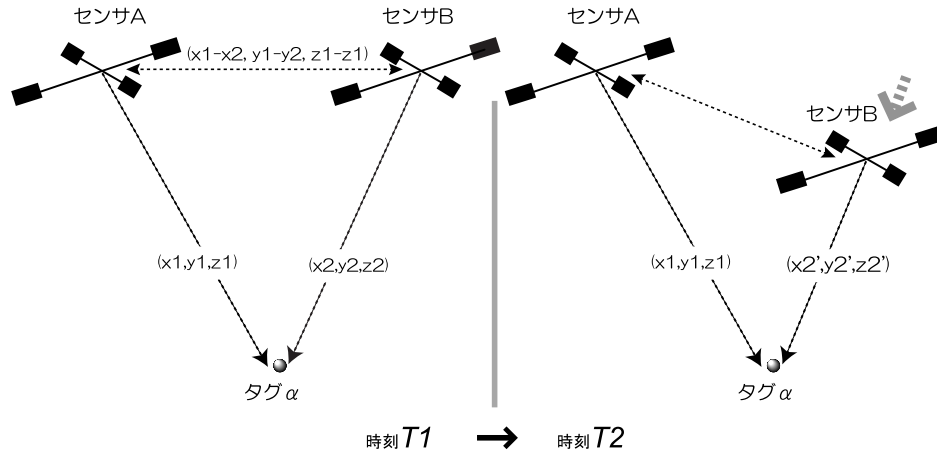


図 3.4: センサ位置の推測

3.3 本章のまとめ

本章では，再構成可能位置センサ環境に対応する位置管理機構，SOL の多種位置センサ管理機能とセンサ位置管理機能について述べた．まず，多種位置センサ管理機能を実現する，対象環境のモデル化，センサ検知空間のモデル化および位置情報の算出方法を概説した．次に，このようなセンサ位置管理機能を実現する，センサ位置の検知とセンサ位置の推定について述べた．

次章では，再構成可能センサ環境対応型位置管理機構，SOL の具体的な設計について述べる．

第4章 設計

本章では，再構成可能位置センサ環境に対応する位置管理機構である，SOLの設計について述べる．まず，ハードウェア構成，ソフトウェア構成と全体の動作手順について説明する．次に，各ソフトウェアモジュールの設計について説明する．

4.1 全体概要

本節では、SOL の設計概要として、ハードウェア構成、ユースケース、ソフトウェア構成と、全体の動作概要を述べる。

4.1.1 ハードウェア構成

本システムは、位置センサとサーバマシンで構成する。また、本システムを利用するアプリケーションは、アプリケーションホスト上で動作する。本システムの配置図を図 4.1 に示す。以下、それぞれの構成要素について詳細に述べる。

- 位置センサ

原則として、ネットワーク接続機能を持つ、位置センサとする。対象空間に設置し、ネットワークに接続する。またネットワーク接続機能を持たない位置センサ場合には、PC や小型ネットワークノードを接続して利用する。

- サーバマシン

PC や WS などの計算機器である。ひとつの対象空間に、1 台のサーバマシンを設置する。また、位置センサやアプリケーションホストとの通信のため、ネットワークに接続する。

- アプリケーションホスト

本システムを利用するアプリケーションが動作するホストである。ネットワークに接続し、サーバマシンと通信する。アプリケーションによって、PC や WS などの計算機、あるいは、携帯電話や PDA などのモバイル端末の場合がある。

4.1.2 ユースケース

本システムのユースケースを記述する。位置センサ上で動作する、ソフトウェアモジュール、アプリケーション、およびシステム管理者がそれぞれ、本システムに対して行う動作を以下に述べる。また、本システムのユースケース図を図 4.2 に示す。

- 位置センサ主体のユースケース

位置センサは本システムに対して、測定位置を送信する。

- アプリケーション主体のユースケース

アプリケーションは本システムに対して、位置情報を問い合わせ、また、位置イベントリスナを登録する。

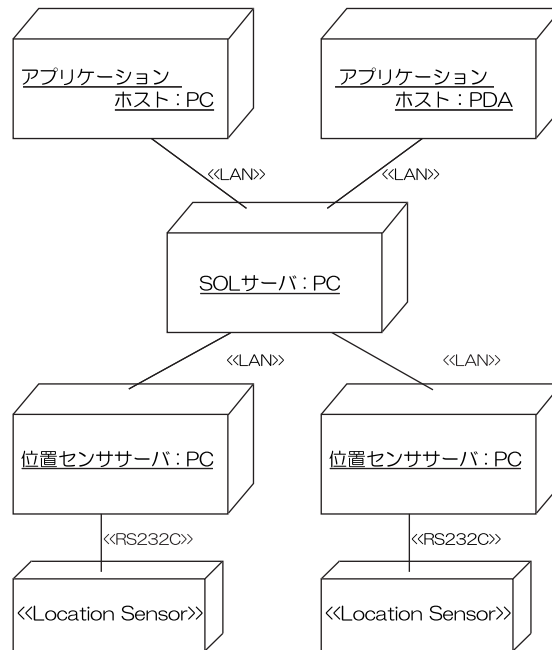


図 4.1: 配置図

- システム管理者主体のユースケース

システム管理者は本システムに対して、対象空間モデルと使用する位置センサの検知空間モデルを登録する。また、位置センサ設置時の位置を登録する。

4.1.3 ソフトウェア構成

本システムは大きく4つのモジュールで構成されている。測定位置処理部は、位置センサから測定位置を取得し、検知空間モデル上の座標に変換する。センサ位置管理部は、測定位置処理部から、測定位置を取得し、センサの位置を計算する。モデル管理部は、対象空間モデルと検知空間モデルを保持し、測定位置を位置情報に変換する。また、位置情報提供部はアプリケーションに対して位置情報を提供する。

本システムのソフトウェア構成を図4.3に示す。それぞれの構成要素については、4.2に詳細を述べる。

4.1.4 動作概要

本システム全体の動作概要について述べる。まず、測定位置を位置センサから取得した際の動作手順について述べる。次に、アプリケーションから、位置情報の問い合わせを受け取った際の動作手順について述べる。

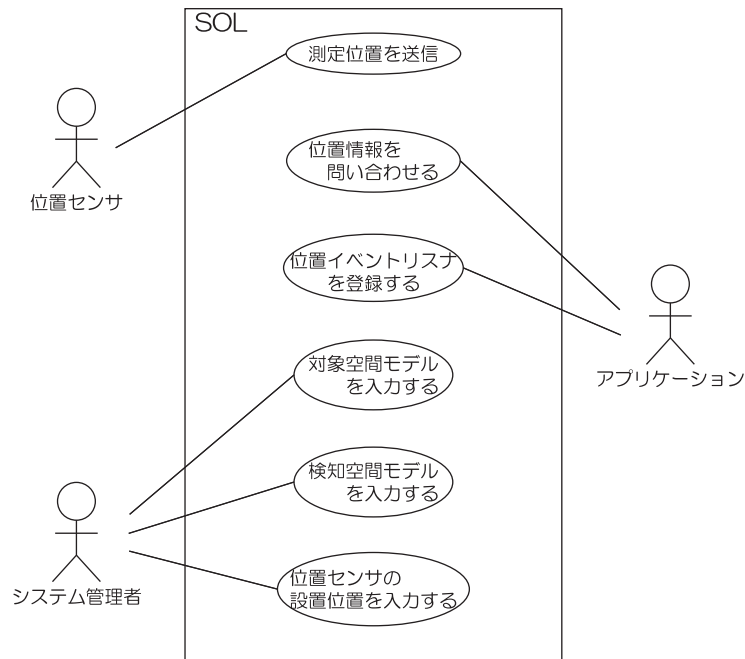


図 4.2: SOL のユースケース図

測定位置の処理

測定位置を位置センサから取得した際の各部の動作手順を示す．また，シーケンス図を図 4.4 に示す．

1. 測定位置処理部は，位置センサから測定位置を取得する．
2. 測定位置処理部は，測定位置を検知空間モデル上の座標に変換する．
3. センサ位置管理部は，測定位置を受け取りセンサの位置を算出する．センサが移動した場合は対象空間モデルを更新する．
4. モデル管理部は，測定位置を受け取り，タグ位置を更新する．
5. モデル管理部は，タグ位置とセンサ位置を更新後，位置情報提供部に通知する．
6. 位置情報提供部は，登録されている位置イベントリスナに対して，タグ位置とセンサ位置の移動を通知する．

位置情報問い合わせの処理

アプリケーションから，位置情報の問い合わせを受け取った際の各部の動作手順を示す．また，シーケンス図を図 4.5 に示す．

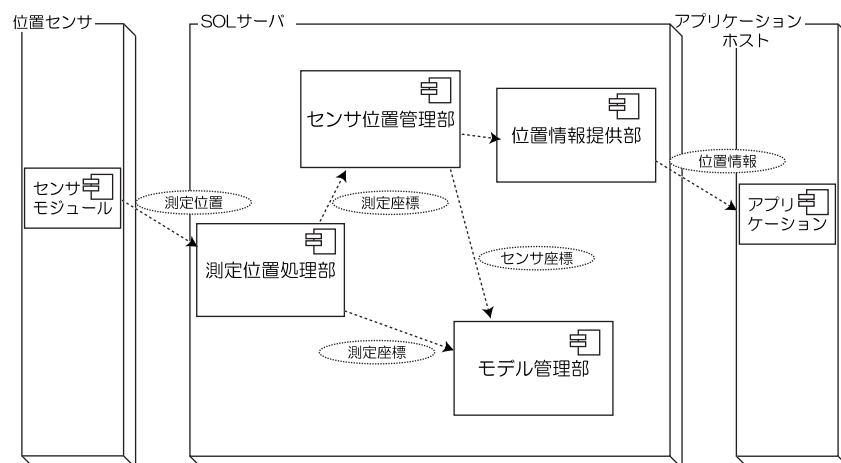


図 4.3: ソフトウェア構成

1. 位置情報提供部は、アプリケーションから位置情報の問い合わせを受け取る。
2. 位置情報提供部は、モデル管理部に対して、タグ位置やセンサ位置を問い合わせる。
3. モデル管理部は、位置情報提供部からの問い合わせに対して、タグ位置やセンサ位置を返す。
4. 位置提供部は、モデル管理部から受け取った、タグ位置やセンサ位置をアプリケーションに対して返す。

4.2 各モジュールの設計

本節では、本システムを構成する各ソフトウェアモジュールの設計を述べる。センサモジュールは、位置センサに接続された位置センササーバ上、あるいはSOLサーバ上で動作する。測定位置取得部、センサ位置管理部、モデル位置管理部と位置情報提供部はSOLサーバ上で動作する。

4.2.1 センサモジュール

測定位置処理部に測定位置を送信するモジュールである。システム開発者が、位置センサごとに実装する。センサモジュールは起動時にセンサ登録情報として、センサ識別子、位置センサ情報と初期位置をモデル管理部に送信し、登録する。センサ登録情報を表 4.1 に示す。また、位置センサ情報には、測定位置タイプ、検知空間の形状と粒度の情報が含まれる。測定位置タイプは以下の3種類がある。

- 識別子型

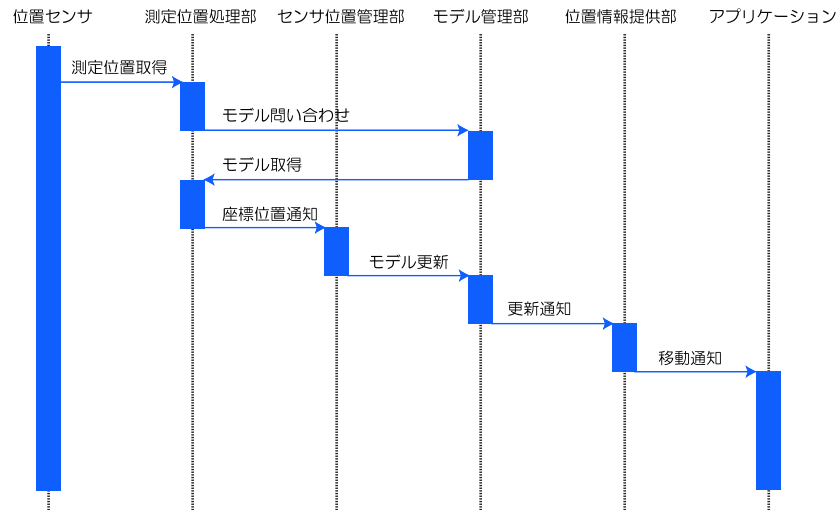


図 4.4: 測定位置取得時のシーケンス図

表 4.1: センサ登録情報

センサ識別子	位置センサ情報			初期位置
	測定位置タイプ	検知空間の形状	粒度	

パッシブ型 RFID や赤外線モジュールによる位置検知など，センサの検出範囲に存在するタグの識別子を取得する測定位置である．

- 距離型

アクティブ型 RFID や無線 LAN モジュールによる位置検知など，センサとタグの距離を取得する測定位置である．

- 座標型

超音波センサによる，センサに対するタグの相対座標を取得する測定位置である．

測定位置処理部に送信する測定位置情報は，タグを新たに検知した場合，あるいは，タグの位置が変化した場合に生成し，測定位置処理部に送信する．また，測定位置の送信時には，センサモジュールが保持する，センサ識別子を付加する．センサモジュールが測定位置処理部に送信する測定位置の情報を表 4.2 に示す．

表 4.2: 測定位置情報

センサ識別子	タグ識別子	タグ位置情報
--------	-------	--------

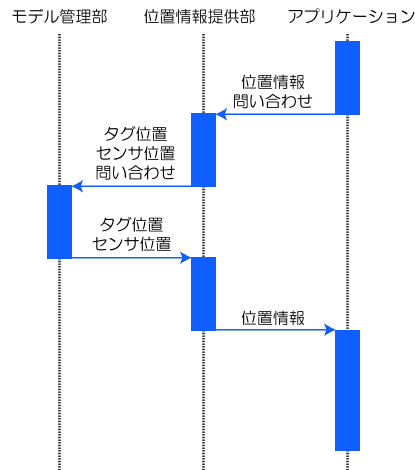


図 4.5: 位置情報問い合わせ時のシーケンス図

4.2.2 測定位置処理部

測定位置取得部は，センサモジュールから測定位置を取得し，処理する．測定位置を取得した場合，センサ識別子をもとに，位置センサ情報をモデル管理部に問い合わせ，それをもとに，測定位置を検知空間モデル上の座標に変換する．その後，変換した測定位置を，モデル管理部とセンサ位置管理部に渡す．

測定位置の座標への変換方法は，位置センサの測定位置タイプにより異なる．以下に，座標への変換方法を，測定位置タイプごとに記述する．

- 識別子型

識別子型の測定位置は，タグが検知空間内のどこに存在するかの情報を含まない．そのため，測定位置を検知空間モデル上の全領域を表現する座標に変換する．

- 距離型

距離型の測定位置は，センサとタグ間の距離を表現する．また，検知空間モデルはセンサの位置を原点とした座標空間である．そのため，測定位置を検知空間モデル上の，原点を中心とする球を表現する座標に変換する．

- 座標型

座標型の測定位置は，センサを原点とした場合のタグの座標を表現する．そのため，測定位置の座標が表現する点を，測定位置の粒度を考慮しながら，検知空間モデル上の領域に変換する．

4.2.3 センサ位置管理部

センサ位置管理部は、測定位置取得部から受け取った測定位置をもとに、センサの位置を管理する。センサ位置管理部は、測定位置を取得後、モデル管理部の保持する検知空間モデル、センサ位置とタグ位置を利用し、センサ位置の移動を判断し、センサが移動した場合はモデル管理部に通知する。また、センサの移動判断フェーズと同時に、タグの位置を更新する。

測定位置取得後、センサ位置更新までの処理の流れを、アクティビティ図として図 4.6 に示し、説明する。

センサ管理部はまず、測定位置を取得したセンサが、他のタグを検知しているか否かを調べる。他のタグを検知していなければ、そのまま次の段落に処理を移す。他のタグを検知している場合は、検知している全てのタグが一様に移動しているか否かを調べる。一様であれば、センサが移動したと判断し、センサ位置を更新する。検知している全てのタグが一様な移動をしていない場合、次段落の処理に移る。

まず、同じタグを検知している他センサの存在を調べる。ない場合は処理を終了する。ある場合は、対象空間モデルを利用し、2 センサの取得したタグ位置が一致するか否かを調べる。一致した場合、処理を終了する。一致しなかった場合、センサが移動したと判断し、モデル管理部に対して、新たなセンサ位置を通知する。

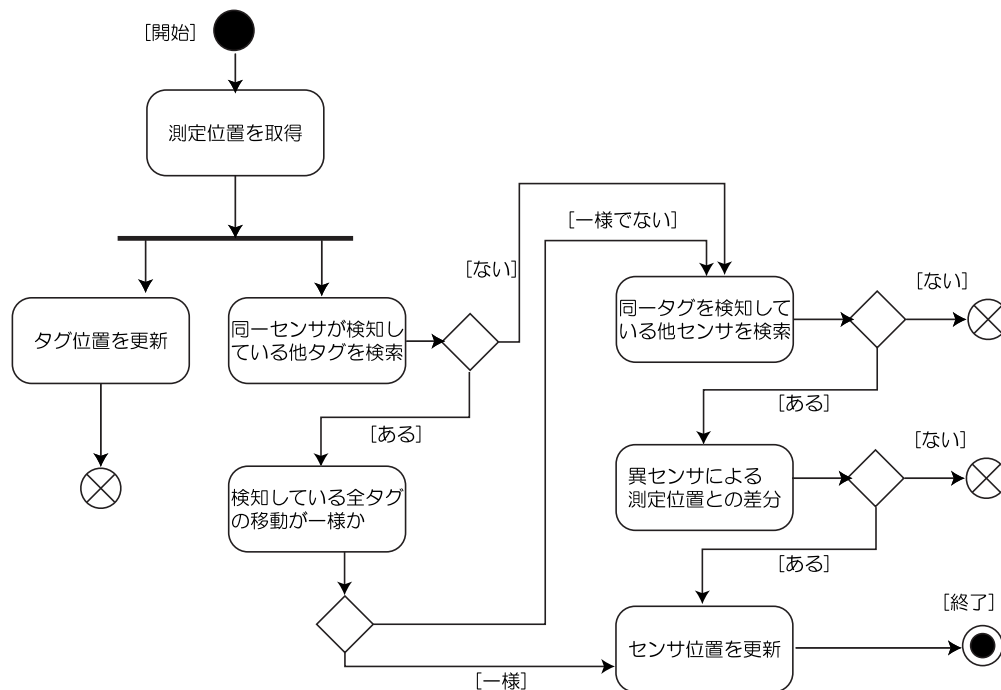


図 4.6: センサ位置推定

4.2.4 モデル管理部

モデル管理部は、対象空間モデル、検知空間モデル、センサ位置とタグ位置を管理する。これらの情報を、他モジュールからの入力をもとに更新し、また、他モジュールからの問い合わせに対して返答する。以下それぞれの情報の処理について記述する。

対象空間モデル

対象空間モデルは、本システムが対象とする空間の座標表現である。座標空間の大きさを、初期設定としてシステム管理者が入力する。本システムは、センサの位置とタグの位置を対象空間モデル上の座標として表現し、アプリケーションに提供する。

検知空間モデル

検知空間モデルは、位置センサがタグを検知する空間（検出範囲）のモデルである。モデル管理部は、センサモジュールから取得した、センサ識別子と位置センサ情報をもとに、位置センサごとの検知空間モデルを作成する。また、モデル管理部は、測定位置取得部の問い合わせに応じて、検知空間モデルの情報を渡す。

センサ位置

センサ位置は、対象空間モデル上のセンサの位置である。センサ位置の初期値は、センサモジュールから取得した、センサ登録情報をもとに設定する。システム運用開始後、センサの移動を検知した際は、センサ位置管理部からの情報をもとに、センサ位置を更新する。

モデル管理部はセンサ位置を更新した際、位置情報提供部に対し、移動したセンサの識別子と、移動後の座標を通知する。モデル管理部は、他モジュールからのセンサ位置問い合わせに対して、センサの位置を対象空間上の座標として返答する。

タグ位置

タグ位置は、検知空間モデル上のタグの座標である。タグ位置を測定位置取得部から取得し、検知空間上の座標を更新する。

モデル管理部はタグ位置を更新した際、位置情報提供部に対し、移動したタグの識別子と、移動後の座標を通知する。また、モデル管理部は、位置情報提供部からのタグ位置問い合わせに対して、タグの座標を、センサ位置をもとに対象空間モデル上の座標に変換し、返答する。

4.2.5 位置情報提供部

位置情報提供部は、アプリケーションに対して位置情報を提供するモジュールである。アプリケーションの位置問い合わせに対して応答し、また、タグとセンサの移動をイベントとして通知する。位置情報提供部は、位置問い合わせと、位置イベントリスナ登録のための API を用意する。

位置情報提供部は、アプリケーションからタグ位置、センサ位置の問い合わせを受け取ると、モデル管理部に問い合わせ、対象空間上の座標として返す。また、モデル管理部からセンサ位置やタグ位置の更新通知を取得した際、センサやタグの移動を示す、位置イベントを生成し、登録されているアプリケーションに通知する。

4.3 本章のまとめ

本章では SOL の設計について述べた。まず、ハードウェア構成、ユースケース、ソフトウェア構成と全体の動作手順について説明した。次に、各ソフトウェアモジュールの設計について述べた。

次章では SOL の実装の詳細について述べる。

第5章 実装

本章では，SOL のプロトタイプシステムの実装について述べる．まず，実装概要を述べ，次に，各モジュールの実装について説明する．

5.1 実装概要

SOL システムのプロトタイプと、本システムの基本性能評価に用いる、位置センサシミュレータを実装した。実装言語は、プラットフォーム非依存性と機能拡張性を確保するため、Java を使用した。開発環境として、J2SE version1.4.1_01 を用いた。

本実装では、センサモジュール、測定位置処理部、センサ位置管理部、及び、モデル管理部をそれぞれパッケージ化した。センサモジュールや位置情報提供部では、アプリケーション開発用の抽象クラスやインタフェースを実装した。また、今回の実装では、2次元座標のロケーションモデルを使用した。そのため、アプリケーションに対して提供する位置情報は、対象空間上の2次元座標である。

5.2 各モジュールの実装

本節では、SOL を構成する各モジュールの実装について述べる。各モジュールのパッケージ構成、主要なクラスとインタフェースについて述べる。センサモジュールと位置情報提供部については、API を記述する。

センサモジュールは、位置センサに接続された PC 上で動作し、測定位置をネットワーク経由で SOL サーバ上の測定位置処理部に送信する。また、SOL サーバ上では、各サブシステムを起動する。各サブシステムは動作を開始すると、センサモジュールからの測定位置を待ち受け、届けられた測定位置を用いて、位置情報算出の処理を行う。

5.2.1 センサモジュール

センサモジュールとして、SensorModule クラスを実装した。また、各種の位置センサ（識別子型、距離型、座標型）に対応するクラスを、それぞれ SensorModule クラスを継承して実装した。クラス図を図 5.1 に示す。

SensorModule クラスの子クラスを使用することで、センサ識別子、検知空間の形状、粒度と初期位置などの位置センサ情報の送信や、測定位置処理部への測定位置の送信などを行える。センサがタグの測定位置を取得した際、SensorModule クラスの tagSensed メソッドや tagMoved メソッドを呼ぶことによって、測定位置を測定位置処理部に送信できる。SensorModule クラスのコンストラクタと、主なインタフェースを表 5.1 に示す。

送信された、位置センサ情報と測定位置は、Facade クラスが受け取る。位置センサ情報を取得した場合は、モデル管理部に、新たな位置センサを登録する。また、測定位置を取得した場合は、取得時のタイムスタンプを付加し、測定位置処理部に渡す。

今回の実装では、実際の位置センサとして、識別子型位置センサである、RF-Code 社の RFCode Spider（アクティブ RFID）と、Sofel 社の ISO15693 準拠 RFID（パッシブ RFID）用センサモジュールを実装した。また、座標型の測定位置を取得する位置センサとしては、シミュレータを実装した。

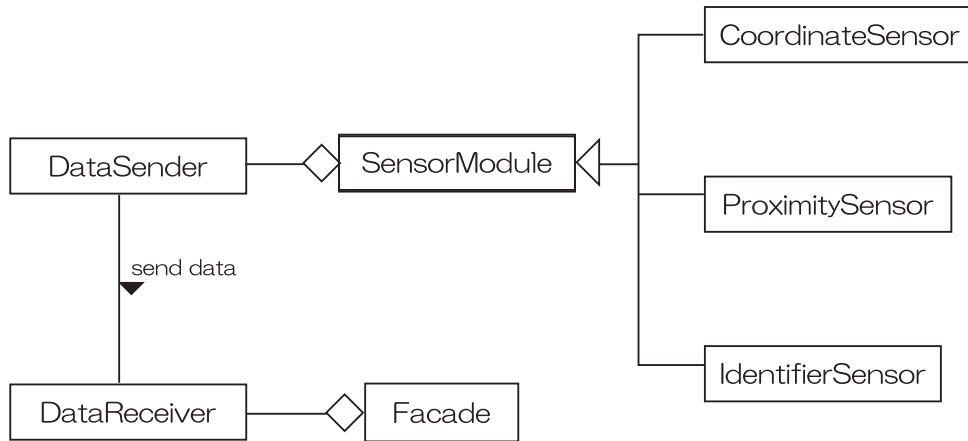


図 5.1: SensorModule パッケージのクラス図

表 5.1: SensorModule のインタフェース

SensorModule(String sensor_id, int sensing_area, int granularity)
public void setSensorLocation(int x, int y)
public void tagSensed(String tag_id, int x, int y)
public void tagMoved(String tag_id, int x, int y)
public void tagUnSensed(String tag_id)

5.2.2 測定位置処理部

測定位置処理部は、センサモジュールから取得した測定位置を、検知空間モデル上の座標に変換する。その際、測定位置に含まれるセンサ識別子をもとに、モデル管理部からセンサの座標と、検出空間モデルを取得し、利用する。変換した座標、センサ識別子とタグ識別子をもとに、DetectedLocation オブジェクトを生成し、DetectedLocation オブジェクトを格納するキューに挿入する。センサ位置管理部は、キューから DetectedLocation オブジェクトを取り出し、順次処理する。DetectedLocation クラスのメソッドを表 5.2 に示す

表 5.2: DetectedLocation クラスのメソッド

public DetectedLocation(String sensor_id, String tag_id, Coordinate Location)
public String getSensor_id()
public String getTag_id()
public Coordinate getCoordinate()
public Timestamp getTimestamp()

5.2.3 センサ位置管理部

センサ位置管理部は，測定位置情報をもとに，センサの位置を管理する．SensorLocationManager クラスは，測定位置処理部がキューに挿入した，DetectedLocation オブジェクトを順次取り出し，処理する．SensorManager クラスは，DetectedLocation オブジェクトをセンサ識別子ごとに一定時間管理し，タグの移動とセンサの移動を判別する．センサ移動の判別処理について以下，詳述する．

まず DetectedLocation オブジェクトを，タイムスタンプ順に，SensorLocationManager 内のリストに挿入する．リスト内に，ある程度 DetectedLocation オブジェクトが溜まったところで，センサ移動の判別を開始する．

まず，ひとつのセンサに検知されている複数のタグが，一様な移動をしていないか，調べる．一番古い DetectedLocation オブジェクトと同じセンサ識別子を持つ，オブジェクトをリスト内から探す．見つかった場合は，両 DetectedLocation オブジェクトのタグの移動方向と距離を計算する．一様な移動が，複数の DetectedLocation オブジェクト内で見つかった場合は，センサが移動したと判別し，モデル管理部に通知する．

次に，DetectedLocation オブジェクトのタグが，他センサに検知されているか否かを，モデル管理部の Tag オブジェクトを参照して調べる．他のセンサに検知されている場合は，MultiSensorThread を生成し，他センサからの測定位置を監視する．他センサが，タグの移動を検知しなかった場合は，DetectedLocation オブジェクトのセンサが移動したと判断し，センサの座標を更新する．

センサ位置管理部のクラス図を図 5.2 に示す．

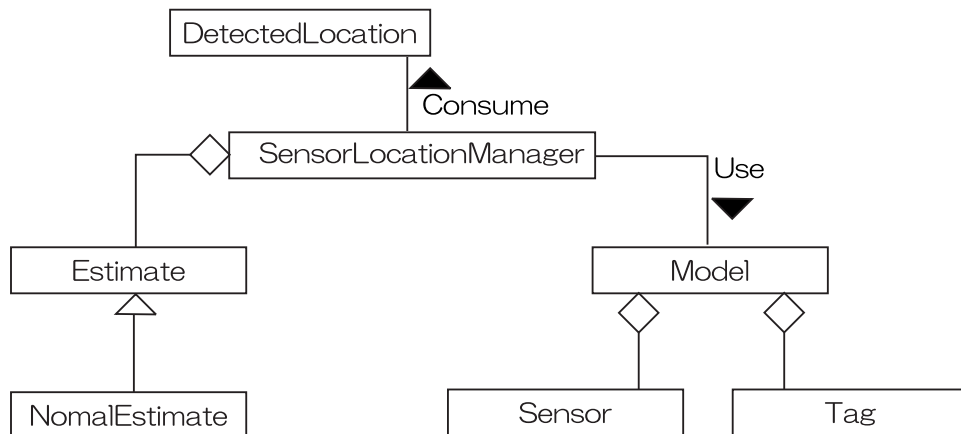


図 5.2: センサ位置管理部のクラス図

5.2.4 モデル管理部

モデル管理部は，ロケーションモデルを管理する．Model クラスは，対象空間内にあるタグとセンサを管理する．また，Model オブジェクトは，Tag クラス，Sensor クラスのオ

プロジェクトを所有する。Tag クラスは、タグの識別子、存在する座標とタグを検知しているセンサの識別子を持つ。また、Sensor クラスは、センサの識別子、存在する座標と検知しているタグの識別子を持つ。

他モジュールは、Model クラスのメソッドを用いて、存在するタグやセンサの識別子、それらの座標などを参照、更新する。また、Model クラスは `java.util.Observable` クラスを継承し、モデル情報を更新した際は、登録されている Observer に通知する。登録されている Observer には、位置情報提供部の位置イベント生産者や、モデル閲覧用のビューアなどがある。

モデル管理部のクラス図を図 5.3 に示す。

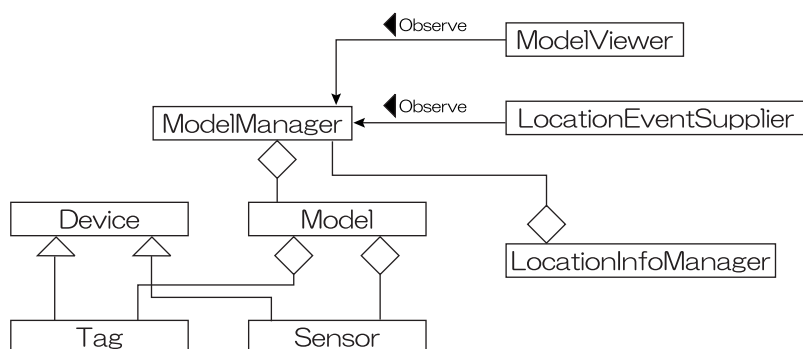


図 5.3: モデル管理部のクラス図

5.2.5 位置情報提供部

位置情報提供部は位置情報を利用するための API を、アプリケーションに提供するモジュールである。位置情報提供部のパッケージ構成を図 5.4 に示す。

本システムは、タグやセンサの移動を検知した際、位置イベントとして、`LocationEvent` オブジェクトを生成し、登録されているリスナに通知する。アプリケーションは、`TagEventListener` インタフェースや `SensorEventListener` インタフェースを実装し、`LocationEventSupplier` クラスに登録することで、位置イベントを受け取れる。タグの移動を検知した際には、`TagEvent` オブジェクトを生成し、センサの移動を検知した際には、`SensorEvent` オブジェクトを生成する。`TagEvent` クラス、`SensorEvent` クラスは両者とも、`LocationEvent` クラスを継承して作成した。

`LocationEvent` クラスのメソッドを表 5.3 に、また、`LocationEventSupplier` クラスのメソッドを表 5.6 に示す。`TagEventListener` インタフェースと `SensorEventListener` インタフェースのメソッドをそれぞれ、表 5.4 と、表 5.5 に示す。

また、アプリケーションは、`LocationManager` クラスの提供するメソッドを利用することで、タグやセンサの位置情報を取得できる。`LocationManager` クラスの提供するインタフェースを表 5.7 に示す。

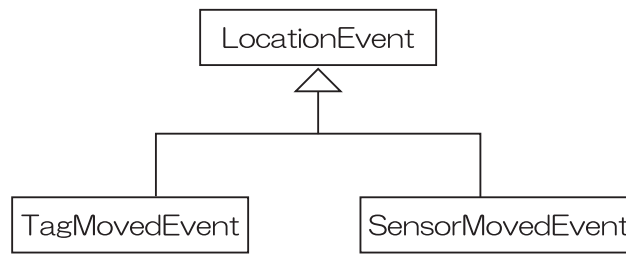


図 5.4: クラス図

表 5.3: LocationEvent クラスのインタフェース

public String getNode_id()
public Coordinate getCoordinate()

表 5.4: TagEventListener インタフェースのメソッド

public void tagMoved(TagEvent t_event)
public void tagSensed(TagEvent t_event)
public void tagUnsensed(TagEvent t_event)

表 5.5: SensorEventListener インタフェースのメソッド

public void sensorMoved(SensorEvent s_event)
--

表 5.6: LocationEventSupplier クラスのインタフェース

public void addTagEventListener(TagEventListener listener)
public boolean deleteTagEventListener(TagEventListener listener)
public void addSensorEventListener(SensorEventListener listener)
public boolean deleteSensorEventListener(SensorEventListener listener)

表 5.7: LocationManager クラス

<code>public Coordinate getSensorLocation(String sensor_id)</code>
<code>public Coordinate getTagLocaiton(String tag_id)</code>
<code>public Set getSensorSet()</code>
<code>public Set getTagSet()</code>
<code>pubilc int getDistance(String tag_id, String tag_id)</code>

5.3 位置センサシュミレータ

本実装の基本性能評価のため，位置センサのシュミレータを実装した．本シュミレータは，タグの移動とセンサの移動をシュミレートし，2次元の座標表現による測定位置を生成する．また，生成した測定位置は，センサモジュールの実装を用いて本システムに送信する．

センサとタグの初期位置はランダムに設定でき，また，位置センサの検出範囲を自由に設定できる．また，センサとタグの，ひとつひとつをランダムに移動させられる．シュミレータの外観を図 5.5 に示す．大きな円が，センサの検出範囲を示し，円の中心にあるノードがセンサを表す．その他のノードはタグである．また，センサと線で繋がれているタグは，検知されている事を表す．

5.4 本章のまとめ

本章では，SOL のプロトタイプ実装について述べた．まず，実装概要を述べ，本システムのパッケージ構成を示した．次に，各モジュールの実装について説明した．また，センサモジュールと位置情報提供部の API について述べた．

次章では，SOL の定量的評価と定性的評価を行う．

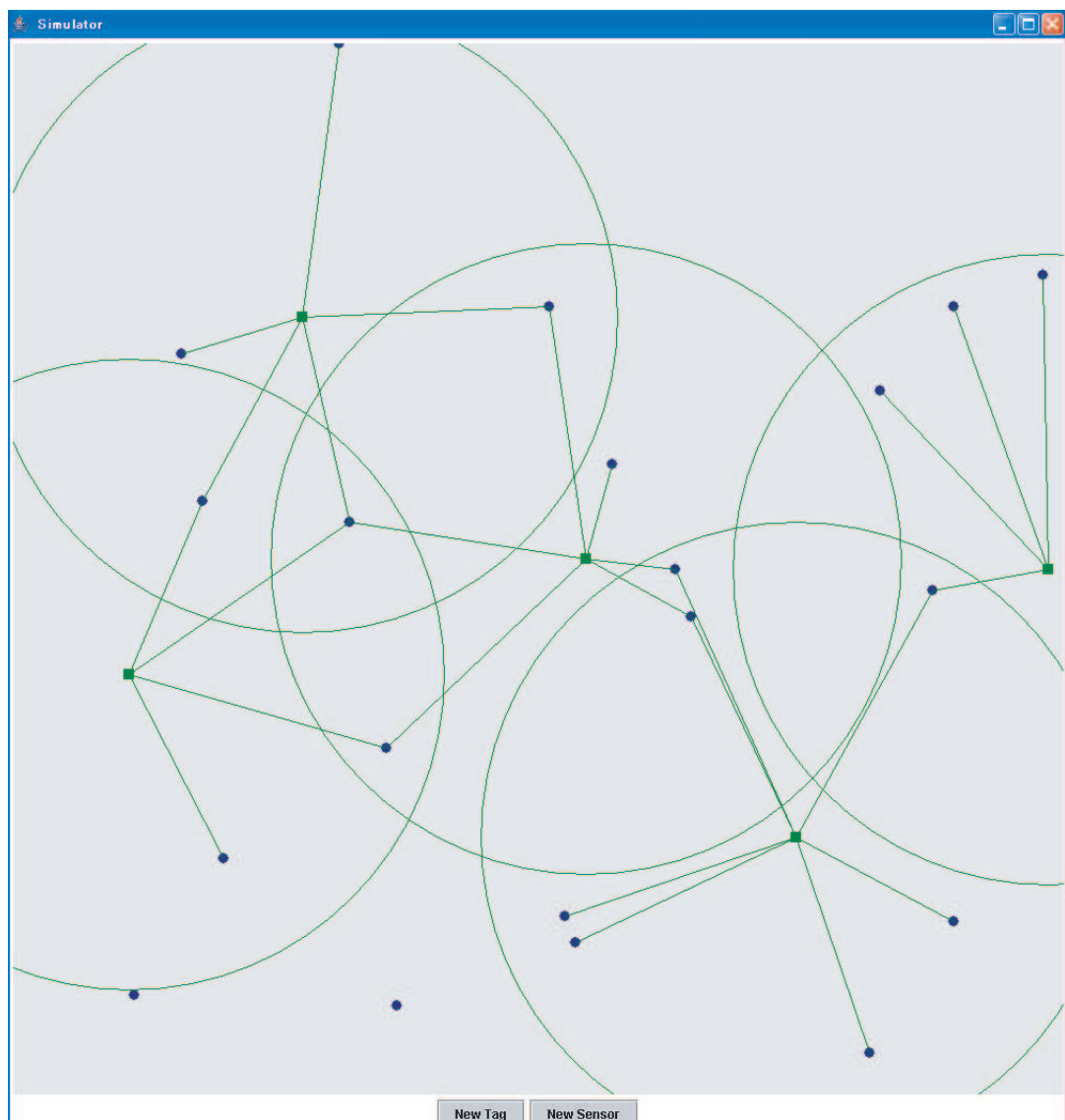


図 5.5: 位置センサシミュレータ

第6章 評価

本章では，SOL の評価について述べる．まず，本システムのセンサ位置推定機能を，定量的に評価する．次に，本システムの実現機能を既存システムと比較し，定性的に評価する．

6.1 センサ位置推定機能の評価

本節では，SOL のセンサ位置推定機能を定量的に評価する．位置推定アルゴリズムの有効性を評価するため，シュミレータを用いた測定実験を行った．測定実験では，シュミレータ上のセンサ位置と，本システムが推定したセンサ位置とを比較し，評価した．

また今回は，シュミレータと本システムを 1 台の PC 上で動作させ，測定した．今回の測定に使用した計算機の性能を表 6.1 に示す．

表 6.1: 実験に使用した計算機

CPU	AMD Athlon 1.53 GHz
メモリ	512 MB
OS	Microsoft Windows XP Professional

6.1.1 想定環境

測定実験の想定する環境について述べる．今回の測定実験では，対象空間に座標を取得する超音波センサが動的に設置されており，タグを保持したユーザが対象空間を移動する環境を想定する．想定環境におけるサンプルアプリケーションとして，ユーザ行動の解析が挙げられる．

対象空間の広さを 10 メートル四方，超音波センサの検出範囲を直径 3 メートルの円，測定位置の粒度を 5 センチメートルとした．また，対象空間内に超音波センサが 4 つ設置されていることとし，タグを持ったユーザの数を変化させて測定した．

6.1.2 測定結果

本システムのセンサ位置推定アルゴリズムが，センサの移動を正しく検知できるかを評価した．センサ移動の検知率，検知したセンサ移動の誤差と，誤差の分散についてそれぞれ，対象空間内に存在するタグの個数を変化させて測定した．以下，それぞれの測定結果について述べる．

センサ移動の検知率

本システムによる，センサ移動の検知率を図 6.1 に示す．センサ移動の検知率は，シュミレータ上で起きたセンサの移動回数と，本システムが推定したセンサの移動回数を記録して計算した．検知率は， n 回のセンサ移動が起きた場合に，検知できる移動回数を示す．

測定の結果，センサ移動の検知率は，タグの個数に比例して増加することが分かった．また，想定環境において，18 個のタグがあれば，90% のセンサ移動を検知できる．

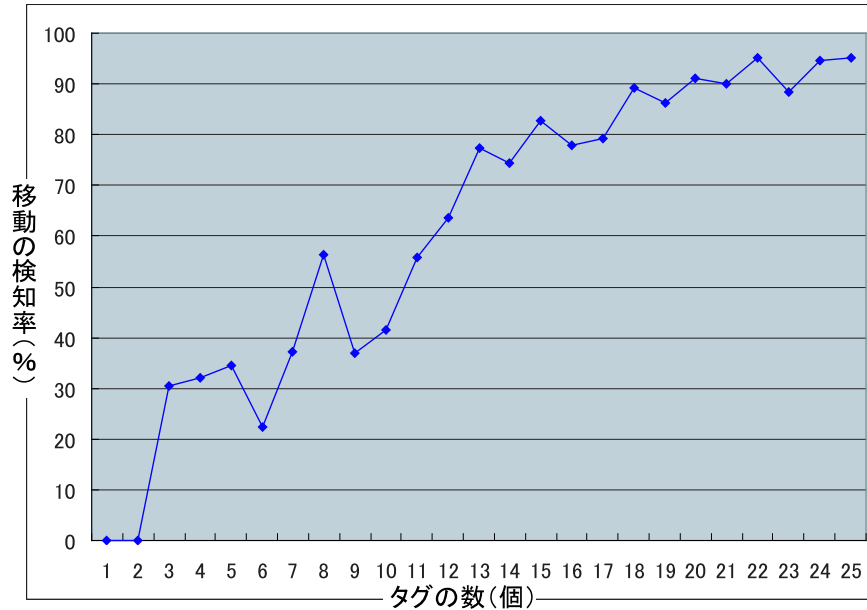


図 6.1: センサ移動の検知率

推定センサ位置の誤差

本システムが推定した，移動後のセンサ位置を，シュミレータ上のセンサ位置と比較し，その誤差を求めた．結果を図 6.2 に示す．図中の折れ線は誤差の平均値を，棒グラフは誤差の最大値を示す．

測定の結果，タグの個数が十分に多い場合，センサ位置をより正確に推定できることが分かった．想定環境においては，タグの個数が 15 個以上で誤差が安定し，60 センチメートル以下になることが分かった．

また，誤差の分散を図 6.3 に示す．誤差の分散も，タグの個数が十分に多い場合に小さくなる傾向がある．想定環境においては，タグが 15 個以上で誤差の分散が安定し，小さくなる．

6.2 関連研究との比較

本節では，定性的評価として，本システムの機能を関連研究と比較する．比較する機能は，位置センサ多様性への対応，センサ移動への対応と，多粒度位置情報の提供である．比較対象に，LocationStack[15]，Geocast[10]，DOLPHIN[21]，SpotON[13] を挙げる．以下，それぞれの機能について，各システムの対応を説明する．また，機能比較の結果を表 6.2 にまとめる．

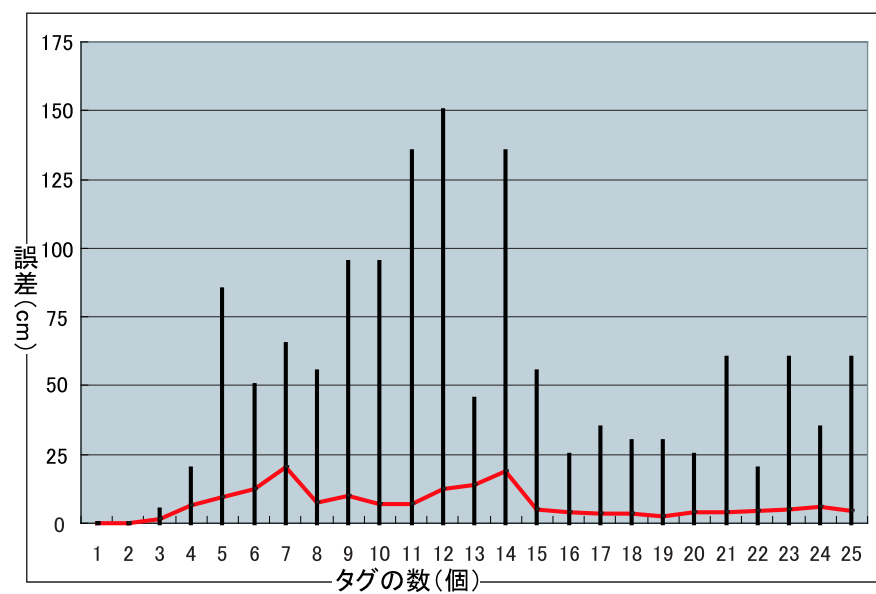


図 6.2: 推定センサ位置の誤差

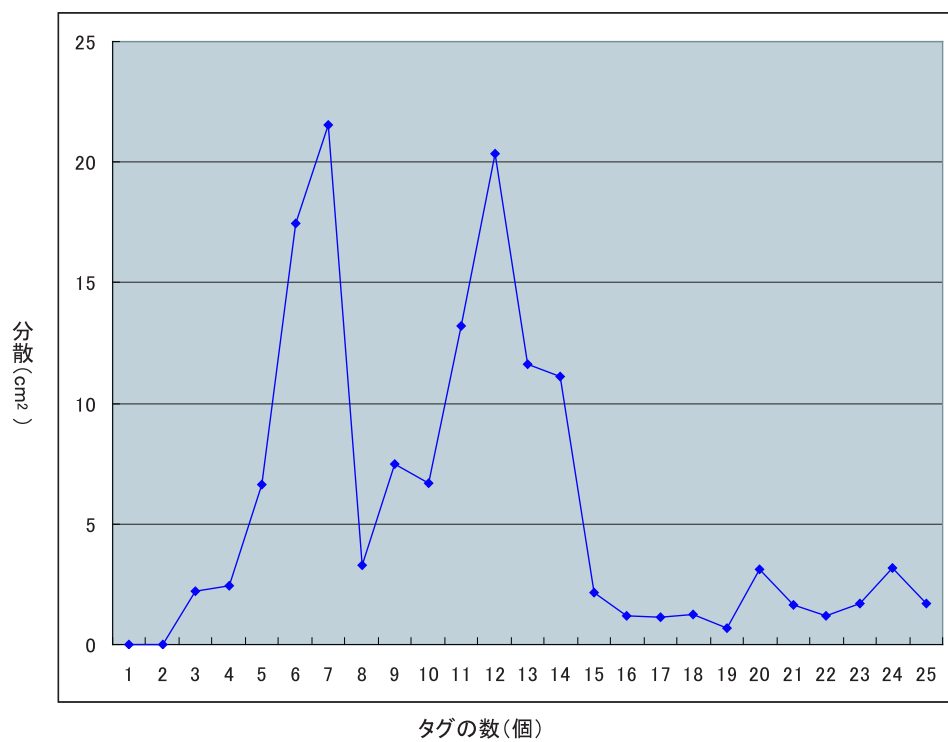


図 6.3: 誤差の分散

表 6.2: 既存研究との機能比較

位置管理機構	位置センサ多様性への対応	センサ移動への対応	多粒度位置情報の提供
Location Stack	○	× ¹	○
Geocast	○	× ¹	○
SpotON	× ²	○	× ³
DOLPHIN	× ²	○	× ³
SOL	○	○	× ³

評価：○ = 十分，× = 不十分

¹ センサの移動を想定していない

² 特別なセンサを使用している

³ 1 種類の位置情報のみを提供する

6.2.1 位置センサの多様性に対応するシステム

位置センサの多様性に対応する位置管理システムとして，Location Stack と Geocast を挙げる．両者とも，位置センサの多様性に対応し，多粒度位置情報の提供が可能である．しかし，センサの移動に対応していない．

Location Stack

Location Stack は，位置情報を階層的に管理する．これにより，異種の位置センサが取得する，測定位置情報の差異を吸収する．また，様々な粒度の位置情報をアプリケーションに提供する．さらに，位置以外の環境情報を取得するセンサを用い，抽象度の高いコンテキストをアプリケーションに提供できる．

Location Stack は位置センサの多様性に対応し，また，多粒度の位置情報をアプリケーションに提供できる．これにより，容易に，多種の位置センサを使用するアプリケーションを，開発できる．しかし，Location Stack は位置センサの移動を想定していない．そのため，再構成可能位置センサ環境には適用できない．

Geocast

Geocast は，座標系とツリー構造のロケーションモデルを合わせて使用する．2 種ロケーションモデルの同時使用により，多種の位置センサが取得した測定位置に対応する．アプリケーションに提供する位置情報の粒度も多様である．また，Geocast は，空間の移動（例：移動中の列車）に対応して，ツリー構造のロケーションモデルを更新する．これにより，ツリー構造のモデルが変化するセンサの移動，つまり，センサの他空間への移動に対応できる．

Geocast は位置センサの多様性に対応し、また、多粒度の位置情報をアプリケーションに提供する。さらに、センサの他空間への移動に対応する。しかし、再構成可能位置センサ環境で発生する、同一空間内におけるセンサの移動に、対応できない。

6.2.2 センサの移動に対応するシステム

センサの移動に対応するシステムとして、SpotON と DOLPHIN を挙げる。両者とも、センサの移動に対応するシステムといえる。しかし、専用のデバイスを利用するため、位置センサの多様性には対応できない。

SpotON

SpotON は、静的な位置検知デバイス（インフラストラクチャ）を必要としない、位置検知を実現している。SpotON は、電波強度の測定によりデバイス間の距離を計算し、相対位置を取得する。全ての構成デバイスが移動できるため、センサの移動に対応するシステムと言える。

しかし、位置検知に専用のデバイスを使用するため、位置センサの多様性には対応していない。また、単一の位置情報を提供するため、多粒度の位置情報を利用するためには、他の位置管理機構が必要である。

DOLPHIN

DOLPHIN は、センサの設置後、センサ位置の入力を必要としない位置情報システムである。DOLPHIN は、センサが設置された際、超音波を利用してセンサ間の距離を取得し、相対位置を取得する。そのため、センサの移動に対応するシステムといえる。

しかし、位置検知に専用のデバイスを使用するため、位置センサの多様性には対応していない。また、多粒度の位置情報を利用するためには、他の位置管理機構が必要である。

6.3 本章のまとめ

本章では、SOL の評価について述べた。まず、センサ位置推定の正確さを定量的に評価した。次に、定性的評価として、本システムの実現する機能を、既存システムと比較した。次章では、本研究における今後の課題を整理し、最後に本稿をまとめる。

第7章 結論

本章では，本システムにおける今後の課題について述べ，最後に本研究についてまとめる．

7.1 今後の課題

本システムにおける，今後の課題として，多粒度位置情報の提供と，多様な移動モデルを用いたシュミレーション実験を挙げる．

7.1.1 多粒度位置情報の提供

現状の実装で，本システムはアプリケーションに対して，座標表現の位置情報を提供している．しかし，アプリケーションによって，必要とする位置情報は様々である．例えば，検知対象の存在する空間の識別子や，ある位置から検知対象への距離などを必要とする場合がある．このようなアプリケーションに対して，座標表現の位置情報を，識別子や距離に変換して提供する機構が必要である．

識別子表現の位置情報については，本システムの持つ，座標表現による対象空間モデルの他に，特定の座標領域と空間識別子をマップするモデルを用意することで対応できる．

7.1.2 多様な行動モデルを用いた評価

今回のシュミレーションでは，タグとセンサを区別せず，両者とも，単純なランダムな移動をさせ，実験を行った．しかし，タグとセンサは設置対象が異なるため，実世界におけるタグとセンサの移動モデルは異なる．そのため，より現実に近い移動モデルを用いたシステム評価が必要である．

7.2 まとめ

本稿では，再構成可能位置センサ環境に対応する位置管理機構である，SOL について述べた．SOL は，位置センサの多様性と，センサの移動に対応する．これにより本システムは，多様な位置センサが，動的な対象に設置されている環境で，位置情報を継続的にアプリケーションに提供できる．

近年，位置センサの多様化・小型化が進んでいる．また，位置情報を利用する多様なアプリケーションが開発されている．今後のユビキタスコンピューティング環境では，多種の位置センサが，静的な対象だけでなく，動的な対象に設置されると考えられる．また，ひとつのアプリケーションが，多種の位置センサによって取得した位置情報を利用すると考えられる．このような位置センサ環境における，アプリケーション開発を，容易にするためには，多種の位置センサに対応し，かつ，センサの移動に対応する位置管理機構が必要である．

本システムは，アプリケーションが対象とする空間と，位置センサの検知空間を，座標によってモデル化する（対象空間モデルと検知空間モデル）．モデル化した検知空間は，センサ位置をもとに，対象空間モデルにマップし，ロケーションモデルを作成する．また，アプリケーション運用後，センサの移動を検知・推定し，ロケーションモデルを更新する．

本システムは、位置センサの検知空間と測定位置を、統一的な座標に変換することで、位置センサの多様性に対応する。また、センサの位置を他センサを用いて検知する手法と、タグを用いて推定する手法の実現により、センサの移動に対応する。本システムの利用により、再構成可能位置センサ環境における、アプリケーション開発が容易になる。

謝辞

本研究の機会を与えて下さり，絶えずご指導を賜りました慶應義塾大学政策・メディア研究科 徳田英幸教授に深く感謝致します．また，本論文の副査として貴重なご助言を下さいました慶應義塾大学政策・メディア研究科 村井純教授，同大学同研究科 高汐一紀助教授に深く感謝致します．

慶應義塾大学徳田研究室の先輩方には折りにふれ貴重な示唆や助言，指導を頂きました．特に，岩本健嗣氏，岩井将行氏には多くのお時間を頂きました．また，同研究室 ACE 研究グループの方々に大変お世話になりました．特に，出内将夫氏には，多大な協力を頂きました．ここに深い感謝の意を表します．

平成 17 年 2 月 15 日
村上朝一

参考文献

- [1] Cooltown project. Online. <http://cooltown.hp.com>.
- [2] It オープンレンジ. <http://feminity.toshiba.co.jp/feminity/series/range/index.html>.
- [3] Mit project oxygen. Online. <http://oxygen.lcs.mit.edu/>.
- [4] インターネット冷蔵庫. http://www.v-sync.co.jp/r_d/net_fridge.html.
- [5] 学習機能付マルチリモコン. <http://www.sugi-ele.co.jp/top.htm>.
- [6] Jeffrey S. Shell Thanh Pham Aadil Mamuji, Roel Vertegaal and Changuk Sohn. Auralamp: Contextual speech recognition in an eye contact sensing light appliance. In *UbiComp 2003 Adjunct Proceedings*. ACM Press, 2003.
- [7] Paramvir Bahl and Venkata N. Padmanabhan. Radar: An in-building rf-based user location and tracking system. *Proceedings of IEEE Infocom*, 2000.
- [8] Meyers B. Krumm J. Kern A. Brumitt, B. and S. Shafer. Easyliving: Technologies for intelligent environments. In *Handheld and Ubiquitous Computing*, September 2000. <http://research.microsoft.com/easyliving/>.
- [9] S. Domnitcheva. Location modeling: State of the art and challenges. In *Proceedings of the Workshop on Location Modeling for Ubiquitous Computing*, Sep 2001.
- [10] Frank Durr and Kurt Rothermel. On a location model for fine-grained geocast. In Joseph F. McCarthy Anind K. Dey, Albrecht Schmidt, editor, *Proceedings of the 5th International Conference on Ubiquitous Computing (UbiComp 2003)*, Vol. 2864 of *Lecture Notes in Computer Science*, pp. 18–35. Springer-Verlag, October 2003.
- [11] David Graumann, Walter Lara, Jeffrey Hightower, and Gaetano Borriello. Real-world implementation of the location stack: The universal location framework. In *Proceedings of the 5th IEEE Workshop on Mobile Computing Systems & Applications (WMCSA 2003)*, pp. 122–128. IEEE Computer Society Press, October 2003.
- [12] Jin Nakazawa Kenta Matsumiya Masaki Ito Masato Saito Hideyuki Tokuda, Kazunori Takashio. Sf2: Smart furniture for creating ubiquitous applications. In *IEEE Proceedings of International Workshop on Cyberspace Technologies and Societies(IWCTS2004)*, pp. 423–439, January 2004.

- [13] J. Hightower, C. Vakili, G. Borriello, and R. Want. Design and calibration of the spoton ad-hoc location sensing system, 2001.
- [14] Jeffrey Hightower and Gaetano Borriello. Particle filters for location estimation in ubiquitous computing: A case study. In Nigel Davies, Elizabeth Mynatt, and Itiro Siio, editors, *Proceedings of the Sixth International Conference on Ubiquitous Computing (UbiComp 2004)*, Vol. 3205 of *Lecture Notes in Computer Science*, pp. 88–106. Springer-Verlag, September 2004.
- [15] Jeffrey Hightower, Barry Brumitt, and Gaetano Borriello. The location stack: A layered model for location in ubiquitous computing. In *Proceedings of the 4th IEEE Workshop on Mobile Computing Systems & Applications (WMCSA 2002)*, pp. 22–28, Callicoon, NY, June 2002. IEEE Computer Society Press.
- [16] Masao Ideuchi, Daisuke Maruyama, Tomokazu Murakami, Naohiko Kohtake, Jin Nakazawa, Kazunori Takashio, and Hideyuki Tokuda. u-texture: A self-organizable material for building smart furniture. *ACM UbiComp Poster Session*, 9 2004.
- [17] InterSense. Is-600 mark 2 precision motion tracker.
<http://www.isense.com/products/prec/is600/>.
- [18] Masana Murase Hideyuki Tokuda Kenta Matsumiya, Soko Aoki. A zero-stop authentication system for sensor-based embedded real-time applications. In *Cambridge International Science Publishing Journal of Embedded Computing*, October 2004.
- [19] Robert J. Orr Gregory D. Abowd Christopher G. Atkeson Irfan A. Essa Blair MacIntyre Elizabeth Mynatt Thad E. Kidd, Cory D. The aware home: A living laboratory for ubiquitous computing research. In *the Second International Workshop on Cooperative Buildings*, October 1999. <http://www.cc.gatech.edu/fce/ahri/>.
- [20] Ulf Leonhardt and Jeff Magee. Multi-sensor location tracking. *ACM/IEEE International Conference on Mobile Computing and Networking*, pp. pages 203–214, 1998.
- [21] K. Hirasawa S. Yokoyama M. Mizumachi H. Morikawa M. Minami, Y. Fukuju and T. Aoyama. Dolphin: A practical approach for implementing a fully distributed indoor ultrasonic positioning system. In Nigel Davies, Elizabeth Mynatt, and Itiro Siio, editors, *Proceedings of the Sixth International Conference on Ubiquitous Computing (UbiComp 2004)*, Vol. 3205 of *Lecture Notes in Computer Science*, pp. 347–365. Springer-Verlag, September 2004.
- [22] D.E. Manolakis. Efficient solution and performance analysis of 3-d position estimation by trilateration. In *IEEE Trans. on Aerospace and Electronic System Vol*, Vol. 32, pp. 1239–1248, 8 1996.

- [23] Hani Naguib and George Coulouris. Location information managment. *Ubicomp*, p. p. 35, 2001.
- [24] N. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *MobiCom*, Aug 2000.
- [25] InData Systems. Rf code spider rfid system.
<http://www.indatasys.com/html/products>.
- [26] T . Okoshi, S . wakayama, Y . Sugita, S . Aoki, T . Iwamoto, J . Nakazawa, D . Furusaka, M . Iwai, A . Kusumoto, N . Harashima, J . Yura, N . Nishio, Y . Tobe, Y . Ikeda, H . Tokuda. Smart space laboratory project: Toward the next generation computing environment. In *Proceedings of IEEE International Workshop on Networked Appliances(IWNA)*, 2 2001.
- [27] R. Want and A. Hopper. Active badge and personal interactive computing objects. In *IEEE Transactions on Consumer Electronics*, February 1992.
- [28] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The active badge location system. No. 92.1, ORL, 24a Trumpington Street, Cambridge CB2 1QA, 1992.
- [29] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. In *IEEE Personnel Communications*, 4(5):42–47, October 1997.
- [30] M Weiser. The computer for the 21st century. In *Scientific American* 256(3), pp. 94–104, September 1991.
- [31] 西田佳史, 秋山賢治, 堀俊夫, 柿倉正義. 超音波 3 次元位置計測のための冗長なセンサデータを用いた高速位置推定アルゴリズム. ロボティクス・メカトロニクス講演会'03 講演論文集, 5 2003.
- [32] 西田佳史, 西谷哲史, 相澤洋志, 堀俊夫, 溝口博. ポータブルな超音波 3 次元タグ - 簡便なキャリブレーション手法 -. 第 21 回日本ロボット学会学術講演会, 9 2003.
- [33] 青木崇行, 村瀬正名, 松宮健太, 中澤仁, 西尾信彦, 高汐一紀, 徳田英幸. Smart furniture: Improvising ubiquitous hot-spot environment. *Proceedings of 情報処理学会情報家電コンピューティング研究グループ第 4 回研究会*, 第 4 巻, 11 2002.
- [34] 村上朝一, 中西健一, 桐原幸彦, 徳田英幸. ロケーションモデル作成支援機構の構築. 日本ソフトウェア科学会ソフトウェアシステム研究会 SPA サマーワークショップ SPA-SUMMER, 2002.