

修士論文 2010年度(平成22年度)

センサネットワーク基盤構築支援ツールの実現

慶應義塾大学大学院政策・メディア研究科
徳田 義幸

修士論文 2010 年度 (平成 22 年度)

センサネットワーク基盤構築支援ツールの実現

近年、無線センサノードの高機能化、小型化に伴い部屋や販売店舗、畑、キャンパス全体など屋内内外を問わずに多様な空間のスマート化が行われており、様々な実験が行われている。特に実空間の情報を取得するためにセンサネットワークが広く利用されているが、同時にセンサネットワークを活用するための基盤構築には多くの課題が残されている。特に、センサノード間の到達性確保や意図したネットワークトポロジの実現が困難である事、確実なデータの保存や利用が予測しにくい事など、基盤構築過程において手順が何度も繰り返され構築コストが肥大化してしまう問題がある。そこで、本研究ではセンサネットワーク基盤の構築をよりの確かつ容易に行う事を目的とし、構築支援ツールの提案を行う。センサネットワーク構築・運用時に生じる手順の繰り返しを回避するためには、ノード設置の構築時と、設置後の運用時に生じる問題を最小限に抑えなければならない。そのためには、実環境においてノードの配置やセンサネットワークの運用を始める前に、問題を起こす原因を発見、検証し回避する必要がある。

これらの課題を解決するアプローチとして本研究ではセンサネットワーク基盤構築支援ツール SenSONAR を提案する。SenSONAR は 3D 空間においてノード間の到達性を考慮した配置支援機能、3D 空間におけるノードの状態操作によるトポロジの検証、センサデータベースへのデータアーカイブ検証とアクセス検証という 4 つの機能群による支援を実現する。これにより、これまで実空間で実際に配置した後でなければ発見や検証ができなかった接続性確保のための配置や、適切なデータベースサーバの選定、トポロジの検証などを事前に検証する事を可能にする。

そして、基盤構築支援ツールの有効性を実証するために、SenSONOR の設計と実装を行い評価実験を行った。評価実験では、3 人のセンサネットワーク構築経験者の学生に支援ツールの有る状況と無い状況でセンサネットワーク基盤を構築してもらった。その結果、非利用時に対して SenSONOR システムを利用する事で、ノード配置時の非到達問題の回避と、センサネットワークを利用する上で容量と応答速度の観点からデータベース環境の検証を容易にする効果が見られた。これによりセンサネットワーク基盤構築時における本ツールの有用性を示す評価結果が得られた。しかし、到達範囲の推測が不十分であった事からノード配置時のトポロジ異常問題を回避する効果が不十分であった事や、応答速度の検証では効果は得られたもののユーザの負荷が高かった事から、今後は距離や障害物以外にも高低差とノード設置場所の考慮や、応答速度提示手法の改善に取り組む。

キーワード：

ユビキタスコンピューティング、センサネットワーク、シミュレーション、可視化

慶應義塾大学 政策・メディア研究科
徳田 義幸

Abstract of Master's Thesis Academic Year 2010

A Development Tool for Sensor Network Infrastructure

In recent years, various smart spaces are created and developed in indoor or outdoor. Various experiments are conducted for supporting many kinds of applications as sensor-node gets smaller and more functional. Especially, people use wireless sensor networks often when they need real space information. However, many problems are left in developing a sensor network infrastructure for getting actual space information by using wireless sensor networks. For example, developing a wireless sensor network infrastructure cost too much time by correcting various wireless communication problems and data gathering problems repeatedly. This research aims at developing wireless sensor network infrastructure more accurately and easily. A new supporting tool called, SenSONAR is introduced for developing infrastructure of wireless sensor network.

To overcome the tedious and repetitive tasks, we need to reduce the deployment costs by using SenSONAR. The key functions are placing sensor-node, editing configuration of sensor-nodes, and avoiding difficulty of storage. To realize these functions, SenSONAR provides the following features: Arrangement support function to consider the attainment between nodes in 3D space, Verification of topology on sensor network in 3D space, Verification of sensor data reading and writing on a database by bench mark function. By this tool, we can select a data base server, adjust node position and check network topology before placing nodes in actual space.

In this thesis, we focus on overcoming the repetitive tasks and implement SenSONAR system. We also have a experimental evaluations in the space which including outdoor and indoor space to demonstrate the effectiveness of the system.

Keywords

Ubiquitous Computing , Sensor Network , Simulation , Visualization

**Graduate School of Media and Governance
Keio University
Yoshiyuki Tokuda**

目次

第 1 章	序論	1
1.1	本研究の背景	2
1.2	本研究の目的と概要	2
1.3	論文構成	3
第 2 章	センサネットワーク基盤の構築	4
2.1	センサネットワーク基盤の役割と構造	5
2.1.1	センサネットワークの応用事例	5
2.1.2	センサネットワークのデータ管理手法	6
2.1.3	本研究におけるセンサネットワーク基盤の定義	8
2.2	センサネットワーク基盤構築支援の必要性	9
2.2.1	センサネットワーク基盤構築の負荷	9
2.2.2	センサネットワーク基盤構築手順の繰り返し	10
2.3	問題意識	13
2.3.1	ノード間到達性問題	13
2.3.2	トポロジ異常問題	14
2.3.3	センサデータアーカイブ問題	15
2.3.4	アーカイブデータアクセス問題	15
2.4	機能要件	15
2.5	まとめ	17
第 3 章	3次元空間におけるセンサノードを用いた事前検証によるセンサネットワーク基盤構築支援	18
3.1	本研究のアプローチ SenSONAR	19
3.2	3D 空間におけるセンサネットワーク基盤の構築・検証	19
3.2.1	3D 空間における到達範囲の可視化と配置支援	20
3.2.2	3D 空間内におけるセンサノードの状態操作機能	21
3.2.3	仮想センサネットワークを用いたデータ保存検証機能	22
3.2.4	擬似アプリケーションを用いたデータアクセス速度検証機能	23

3.3	関連研究	24
3.3.1	センサシミュレータ	24
3.3.2	ユビキタスシミュレータ	24
3.3.3	ノード配置支援ツール	25
3.4	まとめ	26
第4章	設計	28
4.1	SenSONAR の概要	29
4.2	想定環境	29
4.3	SenSONAR 利用の流れ	29
4.4	ハードウェア構成	30
4.5	ソフトウェア構成	31
4.5.1	GUI モジュール	31
4.5.2	ノード生成モジュール	32
4.5.3	ネットワークシミュレータモジュール	33
4.5.4	データベース検証モジュール	33
4.6	まとめ	35
第5章	実装	36
5.1	実装環境	37
5.2	モジュールの実装	37
5.2.1	GUI モジュール	38
5.2.2	ノード生成モジュール	41
5.2.3	ネットワークシミュレータモジュール	41
5.2.4	データベース検証モジュール	44
5.3	まとめ	45
第6章	評価	46
6.1	構築支援ツール評価実験	47
6.1.1	実験環境	47
6.1.2	実験手順	47
6.1.3	結果	49
6.1.4	構築実験結果考察	50
6.2	ユーザビリティ評価	53
6.2.1	結果	53
6.2.2	考察	53
6.3	まとめ	54

第7章	結論	55
7.1	結論	56
7.2	今後の課題	56
7.2.1	ハイブリッドシミュレーションの実現	57
7.2.2	多様なプロトコルへの対応	57
7.2.3	操作インタフェースの簡略化と携帯端末への移植	57
	参考文献	59

目次

2.1	領域規模と管理主体に基づくセンサネットワークの分類	6
2.2	各ノードがもつ役割	7
2.3	センサネットワーク基盤概要	8
2.4	センサネットワーク基盤構築における基本手順	10
2.5	センサネットワーク基盤構築フロー	11
2.6	ノード間到達性問題	13
2.7	トポロジ異常問題	14
3.1	ツール導入によるセンサネットワーク基盤構築における基本手順の変化	20
3.2	SenSONAR のシミュレーション領域	21
3.3	島と橋モデルによる配置支援機能	22
3.4	データベース応答時間の構成要素	23
3.5	Solalium 利用画面	25
3.6	UbiREAL 利用画面	25
3.7	moteView 画面	26
3.8	3dwsn 画面	27
4.1	ハードウェア構成図	31
4.2	ソフトウェア構成図	32
4.3	保存容量から見た限界利用期間の提示	34
4.4	データベースの応答時間から見た限界利用期間の提示	35
5.1	SenSONAR システム利用画面	38
5.2	センサノード設置の流れ	39
5.3	ノード情報の操作例	40
5.4	検証タブ画面	41
5.5	到達範囲計測に用いる頂点	42
5.6	簡易プロトコル実装仕様	44
6.1	支援ツール評価実験環境	48

6.2	アーカイブ検証機能において提示された総データ量と実際のデータ量の誤差	51
6.3	ツール利用によってユーザ推測したクエリ応答時間と実際のクエリ応答時間の誤差	51

表目次

4.1	センサノード情報の構成	33
4.2	センサネットワーク情報の構成	33
5.1	プロトタイプ実装環境	37
5.2	センサデータベースサーバ仕様	37
5.3	受信時の平均 RSSI(dBm) と到達パケット数	43
5.4	受信時の平均 RSSI 減衰率と壁の数	43
6.1	実験結果	49
6.2	ツール非使用時の構築作業に関するアンケート結果	49
6.3	ツール使用時の構築作業に関するアンケート結果	50
6.4	データベース検証作業に関するアンケート結果	50
6.5	ツールに関するユーザビリティアンケート結果	53

第 1 章

序論

本章では，本研究の意義と本論文の内容構成について述べる．

1.1 本研究の背景

近年，センサノードの小型化，低価格化が進み計算処理能力やネットワーク通信機能を持つセンサノードの普及が進んでいる．こうしたセンサノードの進化と計算機の小型化に伴い，様々な場所に埋め込まれた計算機やセンサノードがユーザを支援するユビキタス情報環境が注目され，その環境を実現するための様々な取り組みが行われている．[1] 中でも，無線小型センサノードを用いて温度や照度，物の位置等の実空間の情報を取得し利用するセンサネットワークの研究が活発に行われており，センサネットワークから集められた実空間の情報をもとに，ソフトウェアが空間やユーザの状況を把握し，自律的に振る舞いを変えるコンテキストウェアシステムが多く実現されている [2][3]．こうしたコンテキストウェアシステムにおいて利用されるセンサネットワークを構築するためには，事前に部屋などの対象空間に埋め込まれたセンサを用いるだけでなく，物や空間に後から無線小型センサノードを設置する事でセンサネットワークを構築し，簡易的に実空間情報を取得するインフラを作る取り組みが行われている [4][5][6]．このように，センサノードの高機能化によりセンサネットワークは様々な環境で構築，利用されている．

日々生活する場面において，実空間の情報を利用したい場面は数多く考えられる．例えば，農家のグリーンハウスにおいてグリーンハウス内の日照時間や温度分布を取得する事で栽培環境を最適化する例，展示会場等で人がより多く集まっている時間帯を検出し効果的なタイミングでのデモ展示を行う例，販売店舗において購入された商品だけでなく手にとって検討された商品の情報を集める事で効果的なマーケティングを行いたい例などが上げられる．ユーザの情報や実空間の情報を取得する手法として，RFID やカメラを用いる手法など様々な手法があるが，取得したい情報に応じて手法を選択する事が一般的であり，ユーザにデバイスを持たせたくない場合やプライバシーへの配慮の必要性など，様々な環境でセンサネットワークを利用する必要がでてくる．しかし，自室や教室、キャンパスといった多様な環境においてコンテキストウェアサービスを利用するためのセンサネットワークを構築・運用するには，ノード間の到達性確保や，冗長性を考慮した配置，データが確実に保存される事の検証など，多様な条件を満たさなければならない．しかし，設計支援ツールが不十分であるため，構築者がそうした数多くの条件をクリアするまでには事前の設計と配置と修正を繰り返し，何度もトライアンドエラーを繰り返さなければならない背景がある．

1.2 本研究の目的と概要

本研究ではセンサデータを活用する上で必須基盤となるセンサネットワークとセンサデータサーバの2つを合わせてセンサネットワーク基盤として定義し，その構築支援ツールを提案する．センサネットワーク基盤構築作業においては，構築前には予想する事が困難なノードの到達可能範囲の縮小やデータ保存時のエラーなどの障害により，構築手順繰り返しが何度も発生してしまう事が問題となっている．本研究の目的はセンサネットワーク基盤構築時における手順の繰り返しを減少させるために，構築後に生じる問題の回避支援を行う事である．

本論文では、センサネットワーク基盤構築支援ツール SenSONAR(A Support Tool for Deployment of Sensor Network Infrastructure) の実装と評価を行う。SenSONAR ではセンサネットワーク基盤構築時の手順の繰り返し問題に着目し、3次元仮想空間におけるセンサネットワーク基盤の構築時・利用時における検証機能を提供する。SenSONAR は大きく4つの機能からなり、3次元空間における仮想センサネットワーク構築機能、センサノード状態操作機能、仮想センサデータを用いたデータアーカイブ検証機能、擬似アプリケーションを用いたデータアクセス検証機能という機能群を用いてセンサネットワーク基盤構築の支援を実現する。

1.3 論文構成

本論文の構成は次の通りである。第2章でセンサネットワーク基盤の定義と考察から、構築時における問題点を示しセンサネットワーク基盤構築支援の必要性を明確にする。第3章では、第2章から得られた問題意識をもとに、支援実現のための要件を整理し、それを実現する手法の提案と既存研究との比較を行う。第4章では提案する SenSONAR システムの設計について述べ、第5章では実装と動作概要について説明する。第6章では SenSONAR システムの評価について述べ、最後に第7章で結論を示し本論文をまとめる。

第 2 章

センサネットワーク基盤の構築

本章ではまず，センサネットワーク基盤の定義と役割の考察を行い，センサネットワーク基盤が持つ特性を明確にする．その後，センサネットワーク構築時の手順を整理し，手順の繰り返しが発生してしまいユーザの構築負荷が増大する問題を明確にし，センサネットワーク基盤構築を円滑にする意義を示す．そして，得られた問題意識から構築手順の繰り返しを減少させるための要件を述べる．

2.1 センサネットワーク基盤の役割と構造

実空間情報を用いるスマートサービスや、コンテキストウェアシステムが実現されるために、システムは利用者や空間に関する情報収集を行っている。これまで、利用者に関する情報はインターネットのアクセス履歴やコンピュータの操作履歴など、仮想空間の中の情報だけであった。しかし、近年の無線センサノードの小型化や無線通信技術の発達により、実空間の情報収集が行われるようになった。例えば、GPS や RFID 等を用いて利用者の位置情報を取得する手法などが挙げられるが、中でもセンサネットワークを用いた情報収集が行われるようになった。本節では、このセンサネットワークの役割と特性からセンサネットワークを利用するために必要な環境をまとめ、センサネットワーク基盤と定義する。

2.1.1 センサネットワークの応用事例

センサネットワークが多様な環境において利用されるという事は、センサネットワークにも様々な規模や管理主体があり、収集したデータを利用するための機構が必要となる。ここでは、センサネットワークの応用事例を領域規模と管理主体に基づき分類し、本研究において扱うセンサネットワークを定義する。

センサネットワークの役割

センサネットワークは実空間の情報を収集する手段として注目され、様々なサービスや研究に利用されている。センサネットワークが情報収集の手段として利用される主な理由として、2つの理由が挙げられる。1つ目は小型無線センサノードがマルチホップ通信を行い、データを相互に通信しあう事で空間のスケールに制約を受けずにセンシングが可能な事。2つ目には、配線やマルチホップ通信などを行えば、森林での火災検知、建築物の耐震構造解析や独居老人の見守りシステムなど、従来では長期的もしくはリアルタイムに取得する事が難しかった情報を、無線センサネットワークでは取得する事が可能となる。これらのメリットを活かす事でセンサネットワークは様々な規模のサービスの実現や研究に貢献している。

センサネットワークの規模に基づく分類

センサネットワークの規模は様々であり、本研究ではセンサネットワークの規模を構築する領域の広さに基づき、屋内狭域空間、屋内広域空間、屋外狭域空間、屋外広域空間の4つに分類した。また、センサネットワークの構築を行う立場も様々であり、自治体や国が社会インフラする例や企業がプラントモニタリングや物流監視など専門性の高いソリューションサービス等を提供している。中でも、個人や小規模なグループが自分たちでセンサノードを配置しサービスを構築する例や、部屋や廊下、キャンパスなど様々な環境にセンサを配置し研究のためのテストベッド構築が多く行われている。本研究においては、こうした個人やグループによって構築されるワイヤレスセン

サネットワークを対象領域とする．また個人・グループでは実現されないような屋外広域空間はのぞく．この対象とするセンサネットワークの領域規模と管理主体を図 2.1 の点線で囲まれた領域で示す．

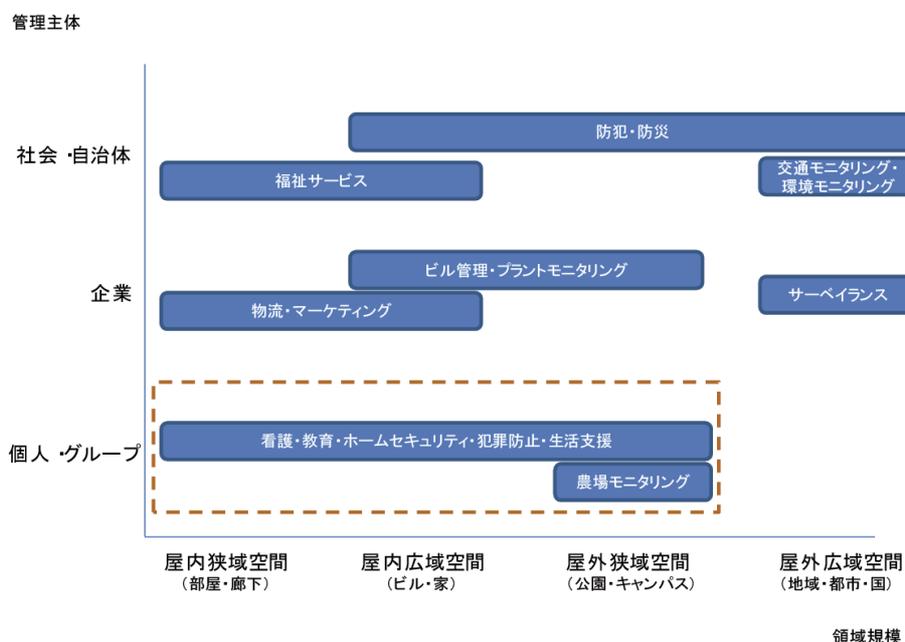


図 2.1 領域規模と管理主体に基づくセンサネットワークの分類

2.1.2 センサネットワークのデータ管理手法

屋内空間や屋外狭域空間においてセンサデータを収集・利用するためには，センサネットワークの構築とそのデータ管理サーバが必要となる．そのため，構築者はセンサネットワークの特性を把握する必要がある．センサネットワークは各ノードが特定の機能もしくは複数の機能を持つ事で，データの取得，転送，共有を実現しており，主に以下のセンサノード，シンクノード，中継ノードという3つのノードから構築される．図 2.2 に3つのノードの関係性を示す．

センサノード

センサノードの役割はデータを取得し，シンクノードへ送信する事．また，他のノードから受け取ったデータをシンクノードへ転送する事．中継ノードの役割も兼ねる．

シンクノード

センサノードから収集したデータを，ネットワークを介してデータを保管するデータベースへ転送する．シンクノードは複数存在する事が可能である．

中継ノード

センサノード，シンクノード間の距離が遠く，電波が到達できない場合にデータのマルチ

ホップ転送のみを担当する．

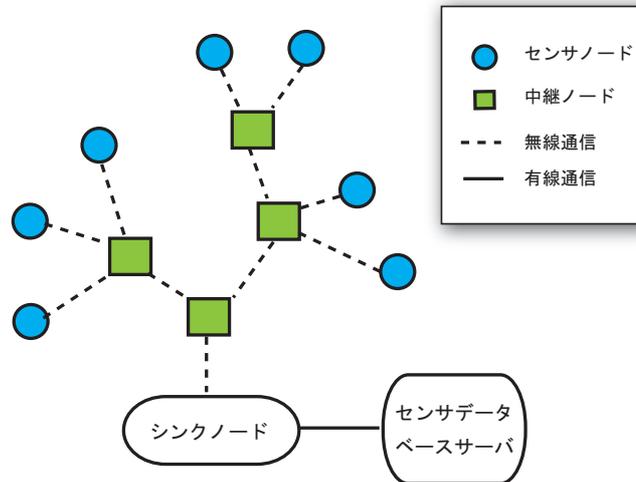


図 2.2 各ノードがもつ役割

アプリケーションがセンサネットワークからデータを取得するためには、蓄積されたデータやストリームデータへのアクセスが必要となる．そのため、上述したようにセンサネットワークには必ずデータの保存と利用の2つの機能を持つデータ管理機構が必要とされる．そのデータの管理手法には主に2つの手法が存在する．データを各センサノード内に分散して管理する分散管理型と、データをサーバのデータベースなどに集約する集中管理型である．[7][8]

以下にそれらの2つのデータ管理手法の特性を述べ、本研究において対象とするタイプの定義を行う．

分散管理型

データをセンサネットワーク内の複数のノードに分散し管理する．要求に応じて、センサデータを収集するためのプロトコルがセンサノードに実装されている事が求められると同時に、データを保管するための保存領域や高度な演算処理能力が必要となる．しかし、センサノードのみを用いてセンサネットワーク基盤を構築する事が可能となる．分散管理型には、シンクノードが必ずしも存在せず直接センサノードへクエリを送る事が可能となる．

集中管理型

集中管理型では、分散管理型とは反してデータを一元管理する．集中管理型のセンサネットワークは必要に応じてセンサノード、シンクノード、中継ノードを組み合わせ構築される．データの流れとして、センサノードから取得されたデータがセンサノード、中継ノードを経由し、シンクノードへたどり着きシンクノードがノード間だけでなくサーバへのネットワーク接続性を持つ事で、データベースへデータを集約させる．

以上の事を踏まえると、集中管理は構築・管理が容易な分、負荷がデータベースサーバへ集中する事がわかる。また、分散管理型は自律的に稼働する事で負荷を分散するメリットを持っていると言えるが、センサノードがもつ実装上の機能的制約を多く受ける事となる。しかし、センサノードがもつ機能的制約はセンサノード利用に直接影響を与える。例えば家の中全体に廉価で小型なセンサをばらまくといったケースの場合、センサノードのコストを抑えるためにマルチホップ機能を持たない非力なセンサノードを利用する事が考えられる。また、取得したいデータが多岐にわたる場合、取得データに応じて複数種類のセンサノードを利用する場合などがあげられる。しかし、システムを運用するためにセンサネットワークを構築する場合、多くの場合は取得したいデータを取得できる事、コスト的に実現できる事が最優先される事になる。そのため、こうした状況では分散管理型を採用する事は困難であり、集中管理型によってセンサネットワークのデータを管理していく事となる。こうした観点から、さまざまなシチュエーションにおいて構築可能である集中管理型はより多くの場面で求められる事になるといえる。そこで本研究では、この集中管理型のセンサネットワークを対象とする。

2.1.3 本研究におけるセンサネットワーク基盤の定義

上述したように、集中管理型のセンサネットワークを応用したアプリケーションにおいては必ずデータを取得、転送するためのセンサネットワークとそこから収集したデータを保管・共有するためのデータ管理機構が必要となる。本研究ではこのセンサネットワークとデータ管理機構を合わせてセンサネットワーク基盤と定義する。本研究におけるセンサネットワーク基盤の概要図を図 2.3 に示す。

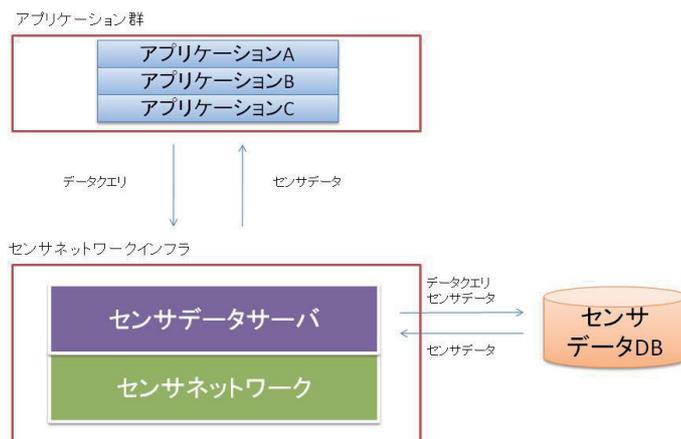


図 2.3 センサネットワーク基盤概要

2.2 センサネットワーク基盤構築支援の必要性

センサ情報を用いるシステムを構築する上で、センサネットワーク基盤構築に作業時間を多くさく事は好ましくない。本節では、センサネットワーク基盤の構築手順を 1) 設計フェーズ、2) 構築フェーズ、3) 運用フェーズの多きく 3 つのフェーズに分類する。そして、各フェーズの手順を考察し、手順が繰り返してしまうケースを明確にする。また、それにより構築作業の負荷が増加してしまう事を述べ、構築作業支援の必要性を示す。

2.2.1 センサネットワーク基盤構築の負荷

センサネットワーク基盤構築は大きく分けて、設計、構築、運用という 3 つのフェーズに分ける事が出来る。はじめに、こういったセンサノードやデータ管理サーバを用いて基盤全体を構築するか、どこにノードを配置するかといった設計を行い、それにしたがってノードの配置やデータベースサーバの構築を行う。そして構築されたセンサネットワークからデータを保存、利用し運用していく。しかし、これらのフェーズは順に 1 度ずつ行えばよいわけでない。実際にセンサネットワーク基盤を構築した事例からこれらのフェーズの関係性を示す。

センサネットワーク基盤構築事例

Mebius-Sensor プロジェクト [9] において、微気象観測を目的としキャンパス内の温度や湿度、土壌湿度といった情報を収集するセンサノード 16 個をキャンパス内に設置した。センサノードはクロスボウ社の eko[10] を利用し、マルチホップでデータを転送しサーバへ集約させる集中管理型でセンサネットワーク基盤を構築した。しかし、配置した後もセンサノードの電波到達性が得られず、何度もセンサノード配置場所へ向かい設置位置を修正する必要があった。また、各のノードの接続性が得られた後も特定のノードに転送ルートが集約し、全てのデータを収集できなかったためトポロジを修正するために再度センサノードを移動させる必要があった。

センサネットワーク基盤構築手順

上述した事例からもわかるように、センサネットワーク基盤構築において設計時には予測しえなかった問題が構築後に発生し、手順を何度も繰り返す必要がある場合がある。この際に、発生した問題に応じては作業者の移動やサーバの再構築など高い負荷が構築者にかかる事になる。また、これはノードの数やノードの設置範囲に比例して肥大化していく事になる。以下の図 2.4 に基本的な構築フェーズの関係を示す。

手順の繰り返しが発生しなければ設計フェーズ、構築フェーズという手順を追うのみでセンサネットワーク基盤構築を終える事ができるが、構築時や運用時に上述したような問題が発生した場合、図におけるルート 3 をたどり手順を戻さなければならない。こうした手順の繰り返しは構築者の作業負担を増大させてしまう問題を持つと言える。

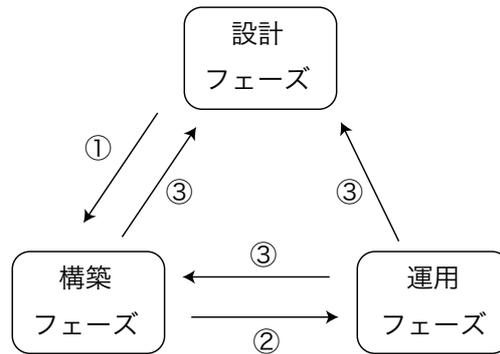


図 2.4 センサネットワーク基盤構築における基本手順

2.2.2 センサネットワーク基盤構築手順の繰り返し

本項では、センサネットワーク基盤を構築するための一連の手順をより詳細にモデル化し、センサネットワーク構築作業の概要を示す。前項で述べた基本手順をより詳細にした作業手順を図 2.5 にまとめ、フェーズごとに各手順の概要を説明する。

設計フェーズ

設計フェーズは、センサノード選定、センサネットワーク基盤全体設計、センサネットワークトポロジデザインの3つの作業から構成されている。以下に各手順内容を詳述する。

センサノード選定

想定するアプリケーションやセンサノードを設置する空間の特徴を踏まえて、センサノードの選択を行う。

センサネットワーク基盤全体設計

利用するセンサノードが決定したのち、センサデータサーバ構築用いるハードウェアの選定やデータベースに用いる DBMS の選定を行う。

センサネットワークトポロジデザイン

基盤全体の設計が済んだ後に、物理的なセンサノードの配置位置と、シンクノードまでのデータの流れを設計する。物理的な配置では、コンテキスト取得可能位置への配置と、ノード間接続が可能な位置への配置が必須要件となる。また、状況に接続冗長性や仕事量の均一化などを踏まえた配置を行う必要がある。物理的な配置位置決定には、仕様書に記載される有効到達範囲を参考に設置が行われるのが一般的である。

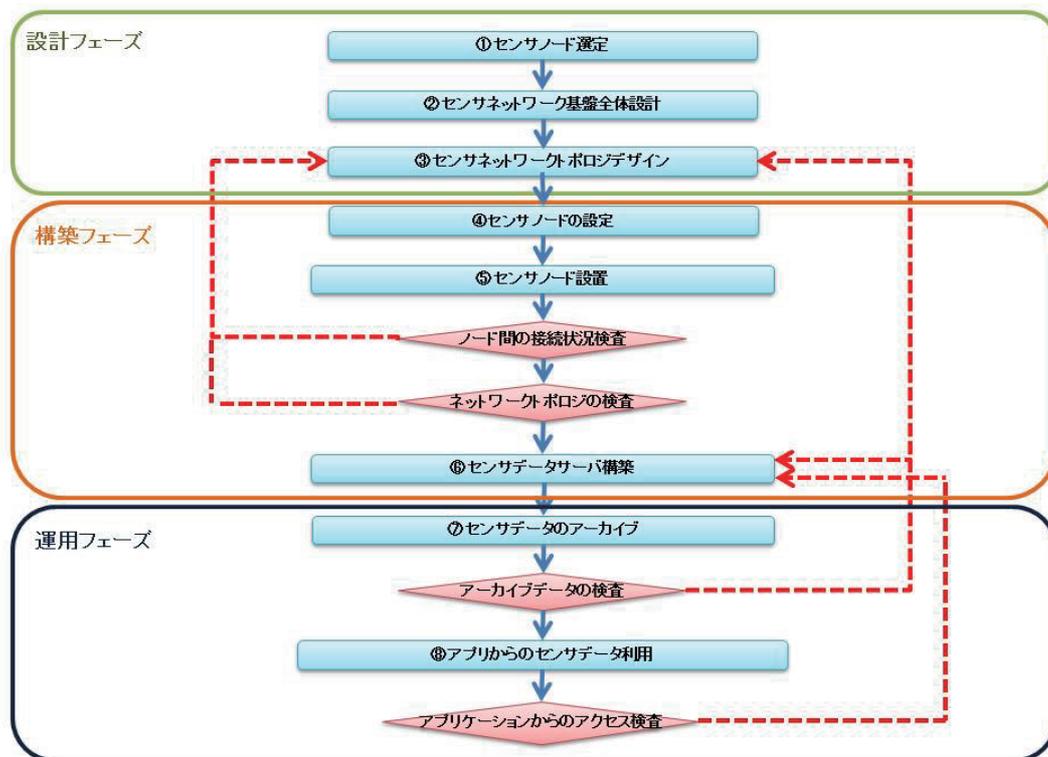


図 2.5 センサネットワーク基盤構築フロー

構築フェーズ

構築フェーズは、センサノードの設定、センサノード設置、ノード間の接続状況検査、ネットワークトポロジの検査と4つの作業から構成されている。2つの検査作業を解決できなければ、運用フェーズに進む事はできない。以下に各手順内容と順序の関係性を詳述する。

センサノードの設定

センサノードの配置の前には取得した情報に応じて、センサノードのサンプリングレートの設定や、ノードへのプログラムのインストールを行う必要がある。

センサノード設置

設計したトポロジにしたがってセンサノードを実空間に配置していく。

ノード間の接続状況検査

配置したセンサノードからデータが取得できているかを確認する。配置したセンサノード同士の到達性が確保されなければセンサデータをシンクノードまで届ける事だけでなく、遠隔においてセンサノードの状況を把握する事すら困難になってしまうためである。もしも到達範囲外に設置してしまっていた場合、トポロジの再設計とノードの再配置を行わなければな

らない。

ネットワークトポロジの検査

接続性が確保されていたとしても、ユーザによってはトポロジに対して追加要素を求める場合がある。例えば接続冗長化や仕事量の均一化などがあげられる。実際にセンサデータを収集し、そこからトポロジが任意の条件を達成できているかを検証する必要がある。もし実現できていなかった場合は、接続検査と同じように、トポロジの再設計、ノードの再配置やノードの再設定を行わなければならない。

センサデータサーバ構築

センサネットワークの2つの検査が終了したのちに、センサデータを蓄積し利用するためのデータベースサーバを構築する。集約管理型ではサーバ内のDBMSにデータを集約する例が多い。

運用フェーズ

運用フェーズは、センサデータのアーカイブ、アーカイブデータの検査、アプリケーションからのセンサデータ利用、アプリケーションからのアクセス検査という4つの手順で構成されている。以下に各手順内容と順序の関係性を詳述する。

センサデータのアーカイブ

データベースを構築した後に、センサネットワークからデータを流し実際にデータベースへ保存する。

アーカイブデータの検査

保存されたデータに欠損が無いか確認する。もし欠損があった場合はトポロジの再設計や、ノードの再配置、再設定、データベースサーバの再構築などを行わなければならない。

アプリケーションからのセンサデータ利用

保存されたセンサデータをアプリケーションから実際にアクセスして利用する。

アプリケーションからのアクセス検査

アプリケーションからデータへアクセスする際に、要求結果の欠損や遅延が無いか確認する。もしも遅延などが確認された場合は、データベースサーバの再構築を行わなければならない。

以上の説明で述べたように、センサネットワーク基盤構築作業においては、ノードの接続状況、ネットワークトポロジ、データアーカイブ、データアクセスの4つの検査項目があり、その検査項目において問題が発生した場合、作業を繰り返さなければいけない事がわかる。

2.3 問題意識

前節において、構築手順が循環してしまう4つ確認項目とその原因を明らかにした。本節では、この4つの問題意識を整理する。

2.3.1 ノード間到達性問題

上述したシナリオのように、センサネットワークトポロジをデザインした際に、仕様書を参考に配置を決定し実際に配置したものの、いざデータが取得できているかを確認するとうまく受信できていなかったというケースが発生する。このようにシステム構築者にとっては現在の構築環境では最初のデザインの段階でノード間到達性を確保する事は困難である。こうした状況に陥ると図 2.6 のように、トポロジの中間位置においてノードの到達性が確保できなかった場合、そのノード以降のノード全ての到達性が失われ情報を得る事が不可能となってしまう。その結果、センサノードの設定や位置を修正するためには設置した場所へ移動し、再設置を行うという手順を到達性が確保できるまで繰り返す事になり、作業負荷が大きくなってしまう。

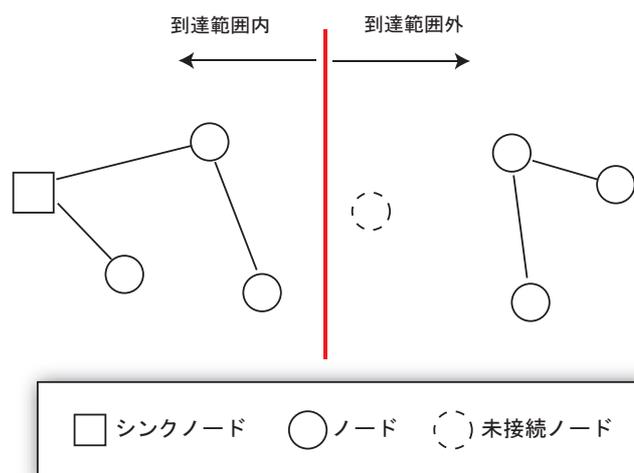


図 2.6 ノード間到達性問題

こうした設計時においてノード間の到達性が確保できない原因は大きく分け2つ挙げられる。1) 設計時に仕様情報だけでは実際に配置した際に起きる電波到達範囲変化の推測が困難 2) 実際に配置環境に行ってみると、設置方法や設置位置の都合から位置を動かさないと行けない場合などが出てきてトポロジ通りにおけない。この2つの問題の解決には従来であれば、実際にセンサノードを配置してセンサノードの到達性を逐次確認していなければならないが、設計時にこの問題を予測し防ぐ事は困難である。

これはあらゆる無線センサノードに起きうる問題であるが、中でもソーラーバッテリーを内蔵する

センサノードに多く見られる例として電力の確保のためにパケット送信周期が固定されているケースなどが存在する。もしくは、設置範囲が一部屋など狭い範囲であれば少ない負荷で済むが、例えばキャンパス全体のような広い屋内外の空間に構築するようなケースの場合、この性質は問題をより深刻にする。

2.3.2 トポロジ異常問題

構築フェーズにおいてもう一つの問題として構築前の設計の段階で理想的なトポロジを実現する困難さから、再設計や再配置の手順の繰り返しへ入るケースがある。例えば、全てのノードが到達範囲内にあったとしても特定の中継ノードへ転送による負荷が集中し、転送パケットのパケットロスが発生してしまう例や、中継ノードのバッテリーが切れてしまう例などが挙げられる。図 2.7 のようにノード N2 にパケット転送負荷が集中に以上が発生した場合に、冗長経路である N3 から N1 への経路が得られない場合などがあげられる。こうしたトポロジにおける異常が発生するとセンサネットワークが機能しなくなる問題を考慮すると、センサネットワークを利用する環境を構築するためには到達性の確保のみでは不十分であると言える。問題が発生してしまうとノード異常によって電源や転送の問題から到達性を失い、遠隔による設定変更などは行えず、再度設置場所へ向かいトポロジの再設計やノードの再配置を行わなければいけなくなってしまう。こうしたトポロジ異常問題が構築時になって発生してしまう理由として 1) 到達性問題を解決し、初めて検証が可能なため、実際にセンサネットワークを構築しないと検証する事が出来ない 2) プロトコルへの理解不足やデプロイ時間を考慮して位置を調整してトポロジを調整する事が多い という2つが挙げられる。

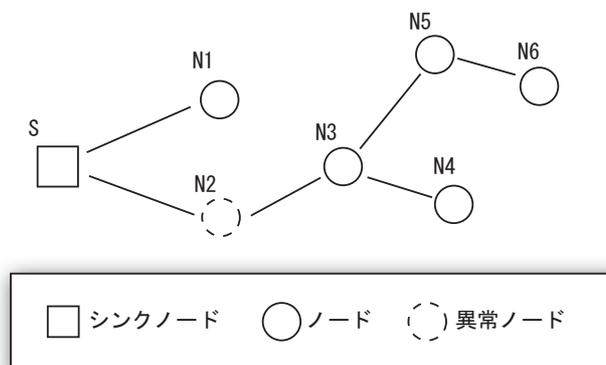


図 2.7 トポロジ異常問題

1 つ目の原因はトポロジ設計通りにノードが配置され、接続到達性が確保されたネットワークへデータを流さなければ検証する事が困難である。つまり、冗長性やノード毎の転送量の均一化などのトポロジ設計が実現されているかの検査は、構築された実センサネットワークを用いなければ検証を行えないという事である。また 2 つ目の原因には、トポロジに問題が発生した際に解決策とし

てはノードの配置位置を調整する方法と、転送プロトコルを変更してインストールしなおす必要がある。しかし、ルーティングプロトコルの修正を行うためにはノード全てに変更を反映させないといけない手間や、そもそも構築者の全てが後者の手法を選択できるわけではないという理由から、ノードの配置位置を操作して修正するケースが多い。

2.3.3 センサデータアーカイブ問題

上述したノード間接続問題と、トポロジ異常問題の2つを解決し、センサネットワークを理想通りに構築できたとしても、取得されたデータがセンサデータベースに格納されるまでに問題が発生するケースがある。例えば各ノードのサンプリングレートが速すぎてノード間のデータ転送や、データベースへの書き込みが間に合っていない例や、特定ノードの転送量が増えすぎてパケットがどこかで欠落してしまう場合などがあげられる。また、利用開始直後に起きうる問題だけでなく、長期的に利用をしていく上で、想定していた利用期間の途中でデータ保存領域が埋まってしまう場合なども考えられる。こうした原因で問題が発生してしまった場合、センサデータサーバのデータベースシステムの変更や、トポロジ再設計とノード再配置によるデータ量の操作などの手順の繰り返しを行わなければいけなくなってしまう。これらの状況に共通する要因として、センサネットワークを構築しなければトポロジ内のデータの量や流れを推測する事が困難である事があげられる。

2.3.4 アーカイブデータアクセス問題

運用時において手順の循環を招くもう一つの問題として、保存されているデータへアプリケーションがアクセスする際に正しいデータを取得できないケースが想定できる。例えば、アプリケーションの要求仕様としては100msec以内にクエリ結果が取得したいのに対し、データベースサーバの応答には150msecかかってしまう場合が挙げられる。運用開始時は要求仕様時間内で結果を返せていたが、保存領域が増えるにつれて応答時間が遅れていくといった状況も考えられる。こうした事態が発生してしまう原因として、設計時のセンサネットワークの構成情報は手順の循環の中で再設計と再配置を繰り返し変更する可能性が高く、運用開始時のセンサネットワークの情報をもとにしなければデータ利用頻度の高いアプリケーションでは、データの格納と利用の速度が釣り合わない可能性がある。もしくはそもそもアプリケーションが要求するクエリ内容への応答結果を現在のデータベースサーバ環境では実現できない可能性がある。

2.4 機能要件

本節では前節で述べた問題意識を踏まえ、手順の繰り返しを減少させセンサネットワーク基盤構築支援を実現するための要件を述べる。本論文における機能要件として設置環境を踏まえたトポロジ設計支援と、実配置前のトポロジ検証、実配置前のセンサデータのアーカイブ、アクセス検証の4つを挙げる。以下に詳述する。

設置環境における影響を考慮したトポロジ設計支援

はじめの設計時にノード間到達性の確保が困難な原因は、1) 仕様情報だけでは実際に配置した際に起きる電波到達範囲変化の推測が困難 2) 実際に配置環境に行くと、設置方法や設置位置の都合から位置を動かさないと行けない場合などが出てきてトポロジ通りにおけないという2つである。つまり、配置環境への情報が不足している事が挙げられる。そのため、到達性確保の問題を解決するためには、この2つの原因に基づき配置環境の情報を設計時に提供する事が必要となる。しかし、情報提供のためには手順が一つ増える事が懸念されるため、情報提示は少ない手順で効率的に行わなければならない。この事から本研究においては設置環境を踏まえたトポロジ設計支援として以下の2つの要件を挙げる。

1. 実空間の状況を踏まえたノード到達範囲の提示
2. 1. をもとにしたノード位置決定支援

はじめに、トポロジが設計通りに機能しない原因として障害物や距離による接続到達性を推測しユーザに伝える事で、より高精度な設計が可能となる。また、その情報を基にした配置支援を行う事で実空間の状況と電波状況の2つを踏まえたトポロジ設計が可能となり、接続性問題を回避することが可能となる。これにより構築前に実空間でそのまま動く同じ状況を経験できるといえる。実空間に近い環境での構築経験をユーザに対して提供する事で、より高精度な設計の支援が可能になるといえる。そして、中間ノード位置決定の手間を減らす必要がある。つまり、実空間の状況を踏まえたトポロジ設計支援では、実際に設置する環境をそのまま再現した検証を実配置を行わずに実現する必要がある

実配置前のトポロジ検証

トポロジ異常問題が発生してしまう原因として、1) 到達性問題を解決したうえで初めて検証が可能なので、実際にセンサネットワークを構築しないと検証する事が出来ない 2) プロトコルへの理解不足やデプロイメント時間を考慮して位置を調整してトポロジを調整する事が多いという問題がある。この事から、設計の段階でトポロジに異常がないか検証する手法の必要性は明らかであり、実現のためには以下の2つの要件が挙げられる。

1. 容易なトポロジ状況の把握
2. センサノードの状態やデータフローの操作による検証を実際の設置作業を行わずに行える事、実際に設置した時と同程度の検証が行える事。

配置前のセンサデータアーカイブ検証

センサデータアーカイブ問題が発生してしまう原因は、センサネットワークを構築しなければトポロジ内のデータの量や流れを推測する事が困難である事が挙げられる。そのため、この問題を回避するためには、設計時にその推測を実現する情報提示が行われる必要がある。この事から

1. 実際に実空間で構築されるものに近いトポロジから得られたデータを用いて、センサデータベースへの書き込み検証を行う必要がある。
2. ユーザが想定する利用情報の取得

この2つの要件の理由として、センサデータの保存検証を行うためにはこれから実空間において構築する予定であるセンサネットワークに関する情報が必要となる。またユーザが想定とするデータベースの利用情報が同時に取得できなければ、想定する利用に耐えうる環境なのかを検証する事は出来ない。

配置前のアーカイブデータアクセス検証

アーカイブデータアクセス問題の発生を回避するためには、事前にその傾向の検証を行う必要がある。そのためにはセンサデータのサンプルが格納されたデータベースと、データへアクセスするアプリケーションがあれば検証が可能である。しかし、データベースへ格納された総データ量などを正しく踏まえて検査を行うためには、実際にセンサネットワークに流れる同程度のデータ量が必要である。しかし、上述した問題と同じように実際にセンサを配置したのちでは支援効果がえられないため、実際に配置するまえに構築するセンサネットワークのデータをデータベースへの格納し、アプリケーションからアクセス可能な速度を検証する必要がある。

配置前のアーカイブデータアクセス検証においては、アプリケーションからデータベースサーバへアクセスした際に、どのようなクエリの場合どの程度の応答時間がかかるか、また利用期間に応じてそれはどう変化していくか等をユーザへ示す事で、検証機能を提供する。そのため、自由にクエリ内容の調整、利用期間の設定が必要になる。

2.5 まとめ

本章では、まずセンサネットワークの分類を行い本研究において対象とする領域を明確にした。次に、センサネットワーク構築フローの整理を行い、構築フローにおける作業繰り返しを示し、構築作業削減の必要性と問題意識を述べた。そして、それらの問題意識から機能要件を導き出した。次章では、機能要件に対するアプローチとして SenSONAR システムの概要を示す。

第 3 章

3次元空間におけるセンサノードを用いた事前検証によるセンサネットワーク基盤構築支援

本章では，第 2 章で示したセンサネットワーク基盤構築時における 4 つの問題を解決するセンサネットワーク基盤構築支援ツール SenSONAR の概要とその機能群について述べる．その後関連研究との比較から SenSONAR の支援ツールとしての優位性を示す．

3.1 本研究のアプローチ SenSONAR

本研究では、センサネットワーク基盤構築支援を実現する事を目的とした、3D 空間におけるセンサネットワーク基盤構築・検証システム SenSONAR を提案する。本節では、SenSONAR の概要と、前章から得られた機能要件の考察に基づくアプローチを述べた後、既存研究との比較を行う。

3.2 3D 空間におけるセンサネットワーク基盤の構築・検証

SenSONAR システムは 3D 空間におけるセンサネットワークの構築支援機能とデータベース検証機能を提供する事で、実空間におけるセンサネットワーク基盤構築作業における手順の繰り返しの回避支援を実現する。SenSONAR システムの機能群は、以下のように構築機能とデータベース検証機能の 2 つに分類される。

構築支援機能

- 3D 空間における到達範囲の可視化と配置支援
- 3D 空間内におけるセンサノードの状態操作機能

データベース検証機能

- 仮想センサネットワークを用いたデータ保存検証機能
- 擬似アプリケーションを用いたデータアクセス速度検証機能

これらの機能群から構成される本ツールを利用する事で、第 2 章において示した、設計フェーズと構築フェーズ、運用フェーズの 3 つのフェーズが循環してしまう既存のセンサネットワーク基盤構築のフローに対して、シミュレーションフェーズを設計フェーズの後に追加する事となる。それにより既存のフローの問題であった手順の循環の回避支援を実現する。このツール導入によるフローの変化を図 3.1 に示す。

本ツールを用いる事で、ユーザはシミュレーションフェーズにおいて手順の繰り返しの起きないセンサネットワーク基盤の設計を行う事が可能になる。本ツールがシミュレーションを行う領域を第 2 章において示した、手順のセンサネットワーク基盤構築フローの図を再度用いて示す。図 3.2 においてシャドウがかかった領域が本ツールがシミュレート可能な領域となる。ユーザはまずはじめに、センサノード配置を想定している空間の 3D モデル内において、設計支援機能のもとセンサノード設置作業を行う。そして、配置したセンサノードの状態や位置、取得データの設定を操作する事で、トポロジ検証を行う。そしてセンサネットワークを 3D 空間内において構築した後に、そこから得られたデータを用いてデータベースへの保存テストを行い、データベースサーバ利用に耐えられるものかを検証する。最後に保存されたデータへ擬似アプリケーションからのアクセスを行う事で、想定する利用が可能な環境がどうかを検証する。そして、これらの工程を終えた後に、

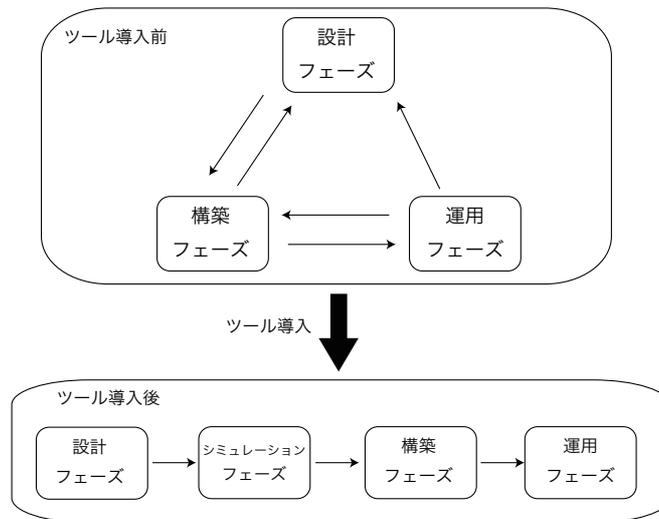


図 3.1 ツール導入によるセンサネットワーク基盤構築における基本手順の変化

実際にセンサノードの配置を行う。これらの機能をツール内で利用する事で、実配置を行う前により高精度な設計を行う事が可能となり、作業コストが肥大化する原因である実空間における構築手順繰り返しの回避支援を行う。本節では、本研究におけるアプローチである各機能群を順に詳述する。

3.2.1 3D 空間における到達範囲の可視化と配置支援

SenSONAR では設置環境を踏まえたトポロジ設計支援を実現する手法として、3D 空間における到達範囲の可視化と配置支援機能を提供する。その中で、ノード到達範囲の提示と中間ノード配置支援機能を提供する事で、実空間の設置環境と近い状況でのノード間到達性を持つセンサネットワークの構築が可能となる。3D 空間でのセンサネットワーク構築機能には以下の機能群が含まれる

- [a] 到達範囲の算出と可視化
- [b] 3D 空間へのセンサノードの設置
- [c] 島と橋モデルによる配置支援機能

機能 [a] では実空間においてノード到達性が確保されない原因であるノード間の距離と設置環境の障害物の影響による到達範囲の減少を算出する。次に得られた到達範囲をユーザに対して地図上にマッピングし可視化して提示する。また、機能 [b] では 3D 空間内に対してノードの設置を行う。しかし、要件にもあるように 3D 空間内での作業を基盤構築作業に追加させる事で基本的に手順が増えてしまう事を考えるとこれらを簡潔かつ容易に行われる必要がある。そのため手順を簡略化する機能が必要であり機能 [c] の配置支援機能が提供される。島と橋モデルによる配置支援では、トポロジ設計の手間をより軽減させるために、センサノードを配置したのちに中間ノードの配置支援

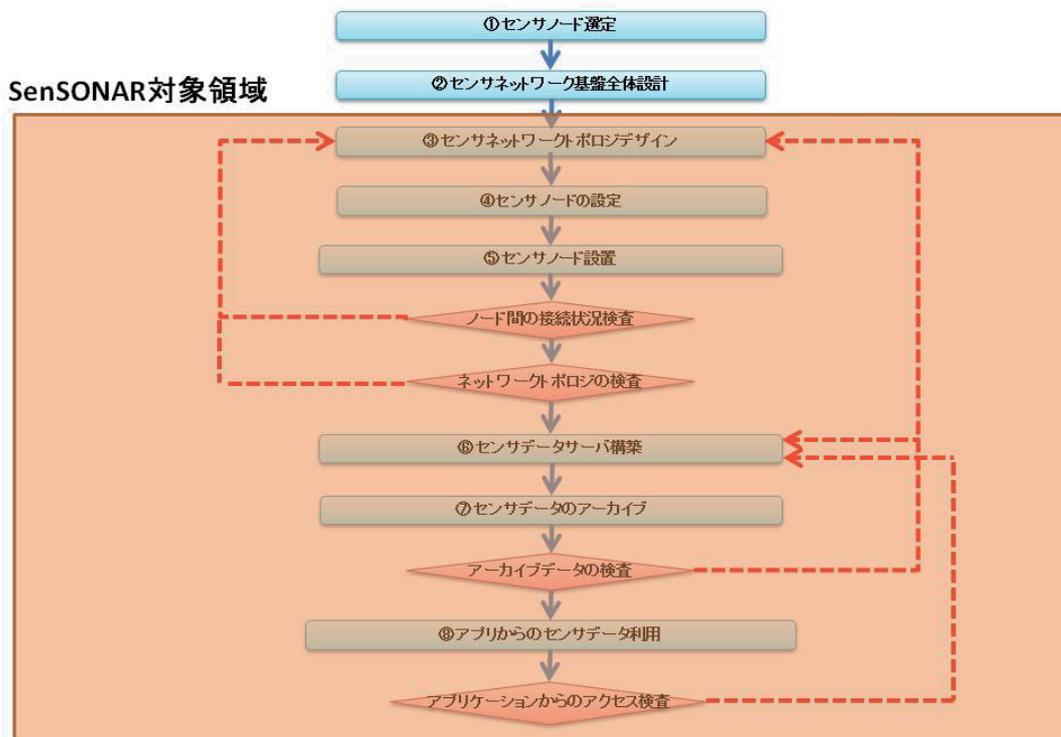


図 3.2 SenSONAR のシミュレーション領域

を実現する．センサノードを配置したのち，センサノードの集合を島とし，島と島をつなげる事を入力する事でそれを実現する．その際に橋の冗長度を設定する事で，冗長度を考慮した配置が実現される．冗長性とはあるノードからシンクノードまでの到達経路が複数存在する性質を指す．そこで，本モデルにおける冗長度とは，ある島から島まで経路ノードに重複の無い経路が2通り以上存在する度合いを示す．例えば，ノード重複の無い経路が2通りあれば冗長度1，経路が3通りあれば冗長度2と定義する．以下の図 3.3 において，配置支援機能の概要を示す．

これらの機能を用いる事で実際にセンサノードを配置する前に，3D 空間内でより実際の環境に即した状況での仮想センサネットワーク構築環境を提供する．

3.2.2 3D 空間内におけるセンサノードの状態操作機能

SenSONAR では，実配置を行う前の設計時においてトポロジ検証を行う手段として 3D 空間内に構築されたセンサノードの状態を操作する事で，センサネットワーク内の異常などを再現し，トポロジ検証機能を実現する．センサノードの状態として設定可能な項目として，取得するデータ量，転送パケット総量，サンプリングレート，電源状態を利用する．データ量とサンプリングレートを調節する事でノード単体の生成パケット量や，経路にある中間ノードの転送パケット量を確認

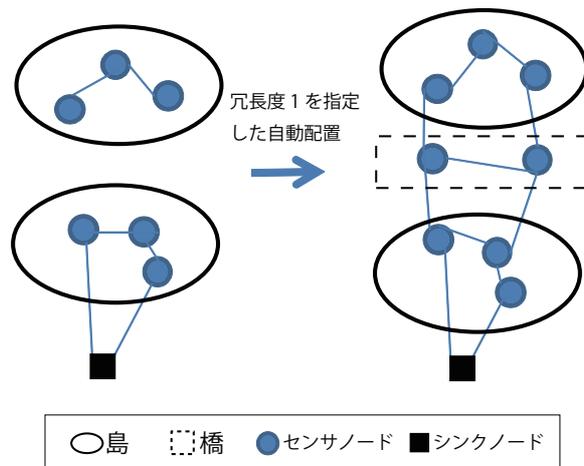


図 3.3 島と橋モデルによる配置支援機能

する事が可能になり，特定ノードへの負荷の集中の検証を行う事が可能となる．また，電源状態の操作が可能になれば，異常ノード発生時の状況の再現が可能であり，特定のノードに異常が発生した場合に可視化された情報からシンクノードまでの冗長経路が利用されているか等の確認が容易となる．こうしたノードの状態操作とノードの接続状況の可視化を利用する事で，トポロジが様々な状況においてユーザが想定した通りに動くかどうかを検証する事が可能になる．

3.2.3 仮想センサネットワークを用いたデータ保存検証機能

SenSONAR では配置前のセンサデータアーカイブの検証を実現するために，想定利用期間と 3D 空間に構築されたセンサネットワークを用いて，保存可能なデータ総量の検証と保存可能な速度の検証の 2 つの機能を提供する．

SenSONAR では配置前のセンサデータアーカイブの検証を実現するために，想定する利用期間と 3D 空間に構築されたセンサネットワークを用いて，保存可能なデータ総量の検証とクエリ実行間隔の検証の 2 機能を提供する．まず保存可能なデータ総量の検証では，構築したセンサネットワークを想定する利用期間利用し続けた場合，どれだけ保存領域が必要になるのかを算出し現在のデータベースの記憶領域情報と合わせてユーザへ提示する．それにより，現在のデータベース環境で構築したセンサネットワークの限界利用期間の把握を保存領域の観点から実現する．クエリの実行間隔の検証では，構築したセンサネットワークのサンプリングレートを保存クエリ実行頻度とし，シンクノードに集約したデータを保存した際に，保存が完了するまでの時間とデータの欠損の有無を提示する．それにより，構築したセンサネットワークのデータを現在想定する利用方法で正しく保存できるかどうかの検証機能を提供する．

2 つの機能を実現するためには，センサネットワークにおけるデータ総量を算出する必要があり，SenSONAR システムでは以下の方法を用いてノードひとつあたりのデータ総量とネットワーク全体のデータ総量を算出する．また，想定利用期間についてはユーザが想定する情報であり，シ

システムはユーザからの入力をもとに利用する。

ノードが n 個ある場合 i 番目のノードが取得するデータの総量 sd は、ノードがセンスしたデータ量 d 、サンプリングレート sr を用いる以下の式であらわされる。

$$sd_i = \sum_{i=1}^n d_i * sr_i \quad (3.1)$$

また、センサネットワークにおけるデータ総量 SD はノードごとのデータ総量 sd 、ネットワークにおけるノード数 n を用いる以下の式であらわされる。

$$SD = \sum_{j=1}^n sd_j \quad (3.2)$$

3.2.4 擬似アプリケーションを用いたデータアクセス速度検証機能

SenSONAR では配置前のアーカイブデータアクセス検証を実現する手段として、擬似アプリケーションによるアクセス可能速度検証を提供する。データベースは集約型のデータベースとして最も一般的に利用される SQL が利用される DBMS を想定とする。現在利用するセンサネットワーク基盤が実用に耐えうるものである事を検証するために、データベースからの所要応答時間を計測する事で想定時間内に正しい結果が得られるか否かを検証する。データベースからの応答時間 Ta はデータベースサーバへ要求が到達する時間 $Treq$ 、DBMS が処理に要する時間 $Tproc$ 、データベースサーバからアプリケーションへ応答が到達する時間 $Trep$ を用いて以下の様にあらわされる。それぞれの関係を図 3.4 に示す。

$$Ta = Treq + Tproc + Trep \quad (3.3)$$

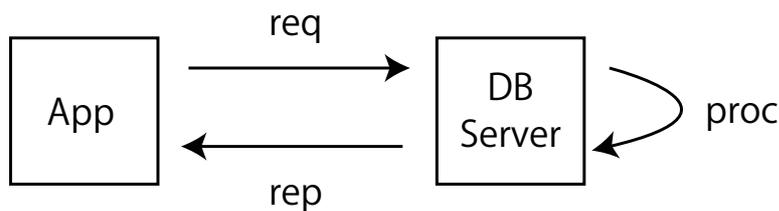


図 3.4 データベース応答時間の構成要素

$Treq$ と $Trep$ の 2 つはネットワーク環境に依存する事になるが、 $Tproc$ は主にデータベースサーバとなる計算機の仕様影響を静的に受ける事になるが、それに加え動的な影響要素として、保存容量の増加に伴い $Tproc$ が増加する事を想定すると、記憶容量やクエリの複雑化に応じた $Tproc$ の変化が考えられる。そのため擬似アプリケーションでは、 Ta が要求仕様よりも増加してしまう境界条件をユーザに対して提示する。

応答時間の遅延もしくは応答内容の欠損が生じるまで、保存されるデータ総量の増加とクエリ実行間隔の短縮を繰り返し応答要求が実現されない要求速度、保存可能容量を求め、それにより、構築者は実際に運用を始める前にデータベースサーバ環境が運用時の利用に適切であるかを検証する事が可能となる。

3.3 関連研究

センサネットワーク基盤構築における支援手法の研究は様々に行われている。以下にセンサネットワーク基盤構築において利用可能な支援ツールをセンサネットワークシミュレータ、ユビキタスシミュレータ、ノード配置支援ツールの3つに分類し関連研究を述べる。

3.3.1 センサシミュレータ

センサシミュレータツールでは、仮想的にセンサノードを用いる事で実デバイスを用いずに挙動の検証を行う事やノード間の通信を検証する事に用いられている。以下にセンサシミュレータの研究、ツール事例について述べる。

NS2

NS2[11] は、大規模なネットワークの試験に用いられるネットワークシミュレータであり、センサネットワークにも応用可能である。対応可能なプロトコルも多くノード間の通信状況を評価する機能は充実している。そのため無線ネットワークのプロトコルの評価には有効性が高いが、構築作業において利用するにはネットワーク構成やリンク情報の定義によるシナリオファイルの生成など複雑な手順が多く、即効性が必要なシミュレーションには利用が難しい。

Solaliumu

Solalium[12] は Oracle 社が開発したセンサノードのエミュレーションソフトウェアである。同社が販売する SunSPOT をエミュレータとして起動し、仮想センサノードを生成、プログラムのデプロイや実空間で起動する SunSPOT[13] との通信機能を提供する。仮想的なセンサノードと実空間のセンサノードの連携を用いる事で、センサノード内のプログラム挙動の検証などには非常に有効である。しかし、ノード間の接続確立などが前提となっている事からプログラム作成支援が主な目的となり、センサネットワーク基盤構築支援という観点においては実用性が低い。

3.3.2 ユビキタスシミュレータ

ユビキタス情報環境において、コンテキストウェアアプリケーションの開発には、プログラムの挙動を検証するために様々な空間の変化を実際に起こして検証しなければならない。そのため、空間の変化を実空間で起こす手間を除くために仮想的な環境で再現を行うユビキタスシミュレータの研究がおこなわれている。[14][15] その機能群の中にセンサネットワークの状態の可視化なども

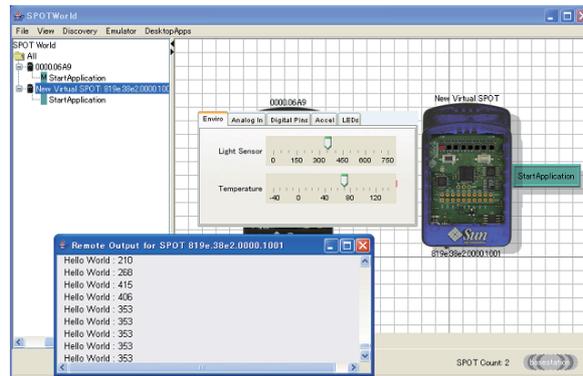


図 3.5 Solalium 利用画面

含まれる．以下にユビキタスシミュレータの研究事例について述べる．

UbiREAL

UbiREAL[16] ユビキタスアプリケーションの動作の正しさを現実空間でテストする事は困難であるという問題意識から，ユビキタスアプリケーションの動作を確認するためのシミュレータの役割を持つ．UbiREAL では主に 3D 空間におけるセンサノードやアクチュエータの配置と，センサデータからのコンテキストの生成，それにもとづくデバイス動作の可視化機能を持っている．そのため，より実空間の環境に即した 3D 空間でセンサノードの配置試験を行う事が可能だが，センサネットワーク構築時の配置位置の支援などは行われなためトポロジ設計の支援機能は持たない．また，アプリケーションの実装の正しさを検証する事を目的としており，センサデータの保存時への考慮は行われておらず，センサネットワーク基盤構築を支援に対しては要件を満たしきることが出来ない．



図 3.6 UbiREAL 利用画面

3.3.3 ノード配置支援ツール

センサノードの配置位置決定の支援ツールは，ビューワの様子にノードを配置した後に位置関係を表示するものや，配置前に利用方法と空間の情報をもとに最適な配置位置を支援する方法などがあげられる．以下にそれぞれの事例について述べる．

moteView

moteView[17] は，crossbow 社が提供するセンサノードモニタリングシステムである．センサノードデータや配置位置を視覚的に把握する事ができる．しかし配置位置の管理は，地図などの画像上に平面的なノードの配置しか行えないためノードの位置関係を正確に把握する機能は無く，設置位置とデータのバインドを行うために利用されている．また，センサネットワークのトポロジや仕事量などを可視化する機能を持つが，実際にノードを配置しないとそれらの機能を利用する事はできない．そのため一度配置し構築したセンサネットワークの管理には有効であるが，センサネットワーク基盤構築の支援において求められる配置前の検証を行う事が出来ない．

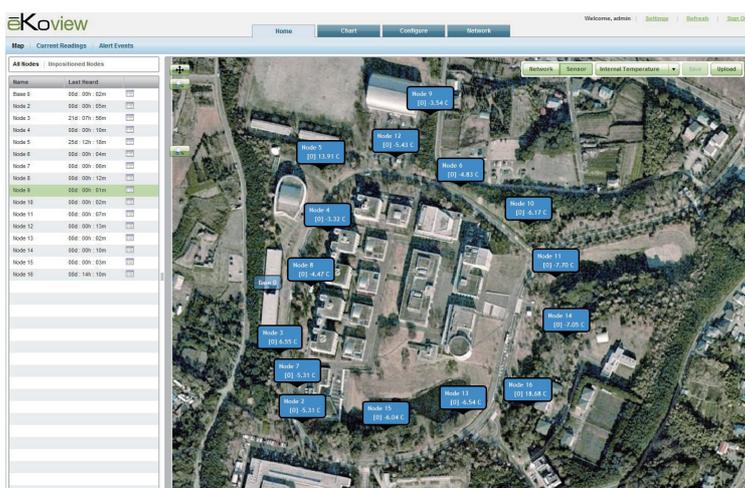


図 3.7 moteView 画面

Deployment Planning Tool for Indoor 3D-WSNs

Kouakou ら [18] は 3 次元仮想空間においてセンサの検出領域を最大化する際のセンサノードコストを最小に抑えるための最適配置支援ツールを提案している．設置環境を考慮したトポロジ設計支援という点においては，ユーザの負荷無く実現がされている．しかし，空間全体を検知する配置の想定や，センサノード数の最小化など要件が定められている事から，事なった配置形態のユーザ要求には答える事が出来ない．また，センサノードの最適配置のみが目的でありセンサネットワーク基盤の構築支援という観点においては，データの利用時や保存時への考慮がなされていない．



図 3.8 3dwsn 画面

3.4 まとめ

本章では、センサネットワーク基盤構築時の手順ループを回避するための要件の考察から、本研究で提案した SenSONAR システムの概要を利用の流れを用いて詳述した。次章では本章で示したアプローチに基づき SenSONAR システムの設計を行う。

第 4 章

設計

本章では，センサネットワーク基盤構築支援システム SenSONAR の設計について述べる．SenSONAR の機能を実現する設計の全体像をハードウェア，ソフトウェアの両面から示す．まずはじめに SenSONAR が想定する環境を明確にし，ハードウェアについては SenSONAR の動作に必要な構成を述べ，ソフトウェアに関してはモジュールに分割しそれぞれの役割について詳述する．

4.1 SenSONAR の概要

SenSONAR システムは、センサネットワーク基盤構築時における手順の繰り返し減少を目的とし、3D 空間におけるセンサネットワーク構築支援機能とデータベースサーバ検証機能を用いる事で、センサネットワーク基盤構築支援を実現する。SenSONAR では設計後にその設計内容をシミュレーションする環境をユーザに提供する。その際に 3D 空間において設置環境に即した 3D モデルを用いる事と、その空間情報を用いて到達範囲の推測を行う事で、より実空間に近い環境でのシミュレーションが可能となる。本節では、SenSONAR システムの設計について、想定環境を踏まえ、ハードウェアとソフトウェアの観点から説明する。

4.2 想定環境

本システムは、3D 空間におけるセンサネットワーク基盤構築機能とデータベースサーバ検証機能を用いる事で、配置環境を考慮した支援を行う。そのため配置環境が再現された 3D モデルが必須となる。近年、動画像解析による手法、全方位カメラからの静止画像解析による手法など様々な手法による屋内外環境の自動的な 3D モデル化の研究が行われている [19][20]。また、家具購入の際に平面の間取り情報に応じて室内空間の 3D モデルを作成してくれるサービスなども普及してきている [21]。本研究ではそうした空間の 3D モデル手法やサービスを用いて、既に配置環境の 3D モデルが利用可能な状況を想定する。また、3D モデルのフォーマットとして、より多くの 3D モデリングソフトウェアで入出力が可能であり、多様なライブラリから扱う事が可能な形式である必要性を考慮し、SenSONAR システムでは一般的に利用されている.obj 形式のモデルデータ利用を想定する。

4.3 SenSONAR 利用の流れ

1. ツールの利用準備

構築ツールを利用する構築者は、2つの準備を行う。1つはどのようなアプリケーションのためにセンサネットワーク基盤を構築し利用するのかを決定し、ツールの検証機能を利用するためにアプリケーションの仕様情報を整理しておく。2つめはアクセス可能なセンサデータベースサーバを構築し、データベースを作成しておく必要がある。

2. 3D 空間内でのノード設置

ツール開始時に 3D 空間で参考にする電界強度の基準を入力すると、それに基づいた到達範囲が 3D 空間において可視化される。3D 空間内において可視化された到達範囲、中継ノード配置支援機能を活用してシンクノード、センサノードを配置し、ノード間の接続情報確認する事で、トポロジがうまく構築されているかを確認する。

3. トポロジの検証

ノードを 3D 空間に配置した後，ノード状態操作機能を用いてノード毎のセンシングデータ，サンプリングレート，電源状態を編集，操作する．それらの操作によりトポロジ情報が更新され，ノードの packets 転送量や冗長性ルートなどの確認を行う事で，理想的なセンサネットワークトポロジの構築を進める．

4. データベースの検証

センサネットワークトポロジを構築した後は，データベースの検証機能を利用して，データベースとの連動性を検証する．想定する利用期間の入力による構築したセンサネットワークの利用可能期間の確認や，データベースアクセスを想定するクエリと実行速度を入力する事で，データへアクセス可能速度の確認を行う．

5. 問題点の修正

ユーザは自分の目指すセンサネットワークトポロジの構築とデータベース利用の検証が行えるまで，2-4 を自由な順番で繰り返す．データベース検証において問題があれば，ユーザはそれを踏まえて DBMS の変更やデータベースサーバの仕様変更を行う．

6. 実空間でのノード配置

最後に，3D 空間内に構築されたノードの配置情報をもとに実際のノード配置を行う．

4.4 ハードウェア構成

SenSONAR システムのハードウェア構成について説明する．ハードウェアは図 4.1 で示すようにユーザ PC，ディスプレイ，センサデータサーバから構成されるユーザ PC は，各々のユーザが利用するコンピュータであり，SenSONAR システムが動作している．ユーザはディスプレイに出力される SenSONAR システム画面を通して 3D 空間内でのセンサノードの配置や確認，データの検証を行う．こうして 3D 空間内においてセンサネットワークを構築した後，構築したセンサネットワークの情報をを用いてセンサデータベースサーバへデータ保存とデータアクセスの検証を行う．センサデータベースサーバには仮想空間のセンサノードのデータ DBMS が稼働している想定とする．また，このデータベースサーバはセンサネットワーク基盤構築の際にユーザが自由に選ぶものであり，SenSONAR システムの実装対象には含まない．ユーザ PC からセンサデータベースサーバへは実際の利用を想定したネットワーク環境で接続される事が望ましいが，ここでは IP ネットワークを用いて接続している．



図 4.1 ハードウェア構成図

4.5 ソフトウェア構成

上述した SenSONAR システムを構成する要素である各モジュールについて詳述する。ユーザが SenSONAR システムを利用する流れは以下ようになる。手順 1,2,3 で 3D 空間においてセンサネットワーク構築を行い、手順 4, 5 において構築したネットワークのデータを用いてデータベース運用検証を行う。これによって得られた情報をもとにユーザは実空間におけるセンサネットワークの構築を行う事となる。

SenSONAR のソフトウェアは、図 4.2 に示すように GUI モジュール、ノード生成モジュール、ネットワークシミュレータモジュール、データベース検証モジュールの 4 つから構成されている。利用者は GUI モジュールを介してシステムを利用し、GUI モジュールから得られた入力内容に応じて、ノード生成モジュールもしくはデータベース検証モジュールが呼び出される。まずノード生成モジュールでは、ユーザが GUI の仮想空間においてセンサの設置位置を指定する事でノードが生成され、生成されたノードの情報がネットワークシミュレータモジュールへと渡され、到達範囲の情報やノード間の接続情報が生成される。そしてネットワークシミュレータモジュールから得られた仮想ノードの接続情報は GUI モジュールへ送られ、情報提示部の更新が行われる。GUI において検証モジュールを呼び出した場合は、この接続情報とユーザが入力する利用想定に関する情報をもとにデータベースの検証が行われる。各モジュールの詳細について次項以降に述べる。

4.5.1 GUI モジュール

GUI モジュールは SenSONAR のユーザインタフェースを提供し、利用者の PC 上で動作する。ユーザからの操作入力受付とその操作結果の出力を行う。

ユーザの操作要求は設置に関する要求と、検証に関する要求にわかれる。そのため、センサ設置操作受付部、検証操作受付部、情報提示部の 3 つのモジュールから構成されている

センサ設置入力受付部は、まず始めに可視化範囲を決定する電界強度の入力に呼び出される、その後ユーザが 3 D 空間へセンサを設置する場合に操作を行う毎に呼び出される。センサの設置には直接座標を指定する場合と、エリアを指定した島と橋モデルに基づいた配置支援の 2 つの方法が

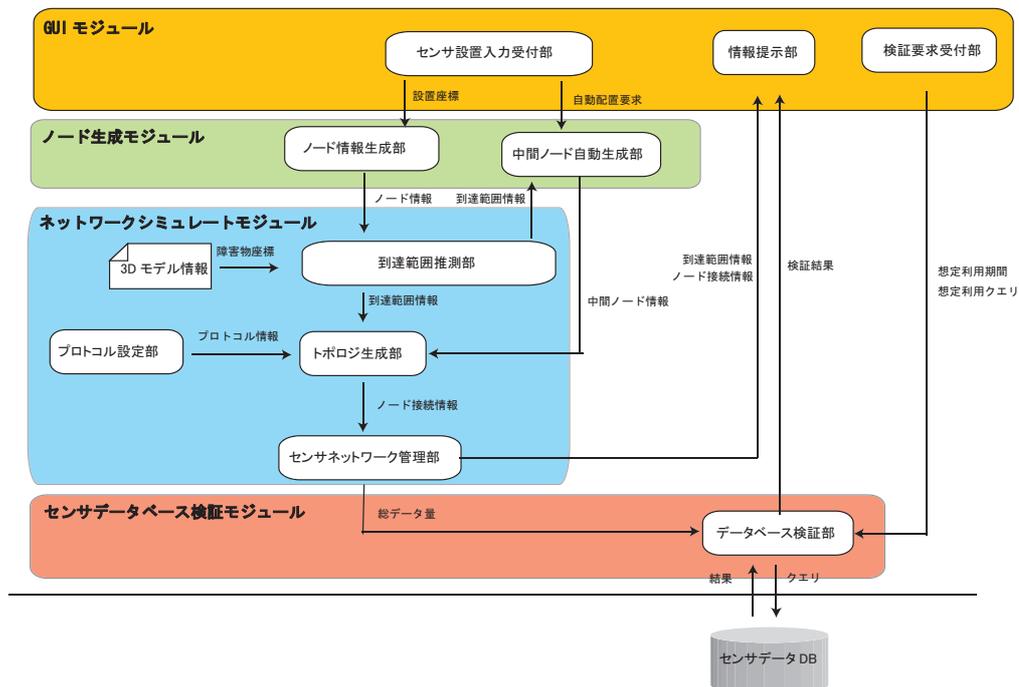


図 4.2 ソフトウェア構成図

らユーザは選択を行う。そのため直接設置する座標もしくは、配置支援を行う範囲の座標がノード生成部へとおくられる。

検証要求受付部では、データベースへのデータアーカイブの検証と、データアクセスの検証が行われる。その検証に利用するための情報として、ユーザから想定している利用期間を取得しデータベース検証部へと渡す。

情報提示部では、センサ設置入力によって設置されたセンサノードの状態が 3D 空間において表示され、検証の検証結果が可視化される。また 3D 空間においてはユーザは空間を自由に移動し表示される情報を更新する事が可能である。

4.5.2 ノード生成モジュール

ノード生成モジュールはノード情報生成部と、中間ノード自動生成部の 2 つから構成されている。両部は GUI から受け取ったノードの設置操作内容に基づいて、ノード情報を生成する。設置座標のみが送られてきた場合、ノード情報生成部が呼び出され指定された座標情報を持つセンサノード情報を生成する。センサノード情報は表 4.1 の構成を持つ。

エリアを示す範囲の座標と冗長度情報が送られてきた場合、中間ノード自動生成部が呼び出され、エリアとエリアを繋げる中間ノードとしてノードを生成する。中間ノードを生成する際には、ネットワークシミュレータモジュールの到達範囲推測部から現在設置されているノードの到達範囲

ノード ID	電源情報	ノードタイプ	親ノード ID	ベースまでのホップ数
パケット転送料	利用センサ	サンプリングレート	座標情報	到達範囲情報

表 4.1 センサノード情報の構成

を取得し、それに基づきエリアとエリアを繋げる中間ノードを生成する。

4.5.3 ネットワークシミュレータモジュール

センサネットワークシミュレータモジュールでは、到達範囲推測部、トポロジ生成部、プロトコル設定部とセンサネットワーク管理部の4つから構成される。

まず到達範囲推測部はノード生成モジュールからノード情報を取得し、それに基づきそれらのノードが持つ到達範囲の推測を行う。到達範囲の推測は距離による減衰と、壁などの障害物による減衰の2つをもとに電解強度の減衰を算出する。その際に障害物の情報を3Dモデルファイルから取得し利用する。各ノードの到達範囲を算出した後、それをトポロジ生成部に渡す。

トポロジ生成部では、到達範囲推測部から取得した各ノードの到達範囲とプロトコル設定部から取得したプロトコル情報を素に、各ノードの接続情報を生成する。接続情報はノードの接続関係とデータの転送量を含む。

プロトコル設定部では、トポロジ生成部がノード間の接続情報を生成するためのルールを生成する。プロトコル設定部では今回簡易的なシンクノードまでのホップ数を最小に抑える Depth First の転送プロトコル情報を設定する。しかし、プロトコル設定部を独立させる事で、外部の保存ファイルを読み込む事が可能であり、AODV など無線データ転送におけるプロトコルを読み込む事が可能な設計となっている。

最後にセンサネットワーク管理部においては、トポロジ生成部から取得したノード間の接続情報とデータ転送量を管理し、情報提示部へと送る。また、データベース検証モジュールから要求を受けた際にも、センサネットワークの情報をデータベース検証部へと送る。生成されたノード間の接続情報は複数のモジュールから参照されるため、生成部とは別に管理部を作り多様なモジュールからの参照を可能にした。

センサネットワーク全体としては以下の表 4.2 の構成を持つ。

総ノード数	シンクノード到達ノード数	総データ量
-------	--------------	-------

表 4.2 センサネットワーク情報の構成

4.5.4 データベース検証モジュール

データベース検証モジュールはデータベース検証部で構成される。データベース検証部はセンサネットワーク管理部から取得したセンサネットワークの情報と、検証要求受付部から取得した想定

利用情報を用いてデータベースアーカイブとデータベースアクセスの2つの検証を行い、得られた結果を情報提示部へと送る。以下にデータアーカイブ検証時と、データアクセス検証時の振る舞いについて述べる。

データアーカイブ検証

アーカイブ検証では、2つの検証情報をユーザへ提供する。ひとつは、現在のデータ保存間隔でデータを正しく格納できているか、ふたつめは現在のセンサネットワークデータを保存しつづけるとどれだけの期間利用する事ができるかという情報である。そのため GUI モジュールから取得するユーザの想定利用情報には、想定利用期間、サンプリングレートあたりのセンサネットワークデータ総量、データ保存間隔が含まれる。これらの情報を用いる事で、図 4.3 に示す様にグラフによる簡潔な情報提示を行う。これにより、ユーザは構築したセンサネットワークのデータ総量情報と、データベースサーバの保存容量に基づいた限界利用期間を把握する事が可能となる。

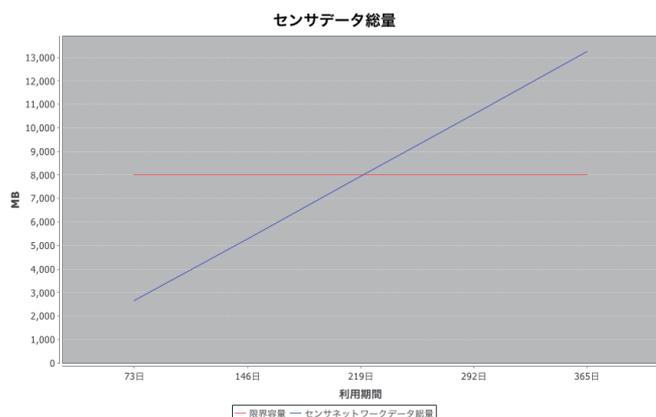


図 4.3 保存容量から見た限界利用期間の提示

データアクセス検証

データアクセス検証においては、データベースサーバの応答時間から見た限界利用期間を提示する。構築者が想定するアプリケーションがデータベースへクエリを出した際に、どの程度の応答時間がかかるのかを期間に応じてグラフ化する。しかし、実際にセンシングしたデータが無ければ、クエリへの応答処理時間を厳密に出すのは困難である。その理由として、データベーステーブルへ格納されるデータ量、クエリ内容、クエリ実行結果の情報量などの要因によって処理時間は変動するものであり、実際にセンシングしたデータが無ければ、クエリ実行結果の情報量などを特定する事は困難である。

そこで SenSONAR では、構築したセンサネットワークのサンプリングレートに基づき、疑似データを生成する。疑似データに対して、コンテキストウェアシステムにおいて利用が想定されるシンプルなクエリを定め、それらの応答処理時間をグラフ化して提示する事で、アプリケーション

ンのクエリが利用可能な内容か否か判断する支援を行う。可視化する応答時間は6種類の内容から選択する。図4.4の例では帰ってくるクエリ1から3にかけてそれぞれのクエリ結果が1件、全体データの0.01% 全体データの1% となる様にクエリ文を生成し応答時間の違いを可視化している。また、その3種類とは別にクエリ文において検索条件が1つ、2つ、3つとなるクエリを生成し同じように可視化を行う。このように、クエリによる検索処理量と得られる情報量にもとづくクエリごとの応答時間をベンチマークとしてグラフで見る事で、ユーザは自分が想定するアプリケーションが利用するクエリが、データベースにおけるどの程度の割合のデータを取得するクエリかを比較検討する事でおおよその自分のクエリの実行結果を推測する事が可能となる。

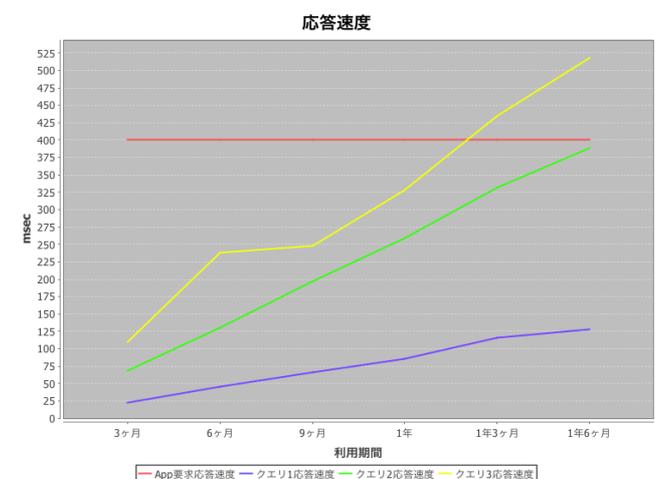


図 4.4 データベースの応答時間から見た限界利用期間の提示

4.6 まとめ

本章では、SenSONAR システムの利用の手順について述べ、ついて設計について全体の構成とモジュールごとの機能、処理の流れを述べた。次章では、本章で行った設計に基づき SenSONAR システムの実装について述べる。

第 5 章

実装

本章では，第 4 章で述べた SenSNOAR の設計に基づく実装についてハードウェアとソフトウェアの両面から述べる．ソフトウェアにおいては各モジュールの実装内容を示す．

5.1 実装環境

プロトタイプ実装では，SenSONAR システムの開発と利用に用いる計算機と，実空間に配置したセンサデータを管理するセンサデータベースサーバの2つを用いた．システム開発環境には MacBookPro を利用した．また Java 実行環境にはサンマイクロシステム株式会社が開発した VM を用いた．また，3D 空間を表現する 3D グラフィックライブラリには jMonkeyEngine[22] を利用した．また実空間において実際のノードを設置する評価実験にはオラクル社の SunSPOT[13] を用いた．プロトタイプ実装環境のハードウェア仕様とソフトウェア仕様を表 5.1 に示す．また，センサデータベースサーバとして IBM Think PadX31 を利用し，DBMS として MySQL を用いた．センサデータベースサーバの仕様を表 5.2 に示す．

CPU	2.8Ghz Intel Core 2 Duo
メモリ	8GB
OS	MaxOS10.6.5
JavaVM	J2SDK 1.6.0
3D グラフィックライブラリ	jMonkeyEngine
センサノード	SunSPOT
SunSPOT squawkVM	SunSPOT API 5.0

表 5.1 プロトタイプ実装環境

CPU	1700MHz Intel Pentium M processor
メモリ	1GB
OS	Ubuntu10.04
DBMA	MySQL5.1

表 5.2 センサデータベースサーバ仕様

5.2 モジュールの実装

本節では，第 4 章で述べたソフトウェア構成に基づき GUI モジュール，ノード生成モジュール，ネットワークシミュレートモジュール，センサデータベース検証モジュールの4つのモジュールの実装方法について述べる．本ソフトウェアは jMonkeyEngine3.0 を用いて 3D 表現を行うため全てのモジュールの実装を Java を用いて行った．

5.2.1 GUI モジュール

GUI モジュールの役割は，仮想センサに関する操作と検証要求を受け付ける事である．SenSONAR は 3D モデル操作画面，センサノード一覧画面，トポロジ確認画面の 3 つの画面から構成される．SenSONAR システムの画面全体図を以下の図 5.1 に示す．また，それぞれの画面が持つ役割を以下に詳述する．

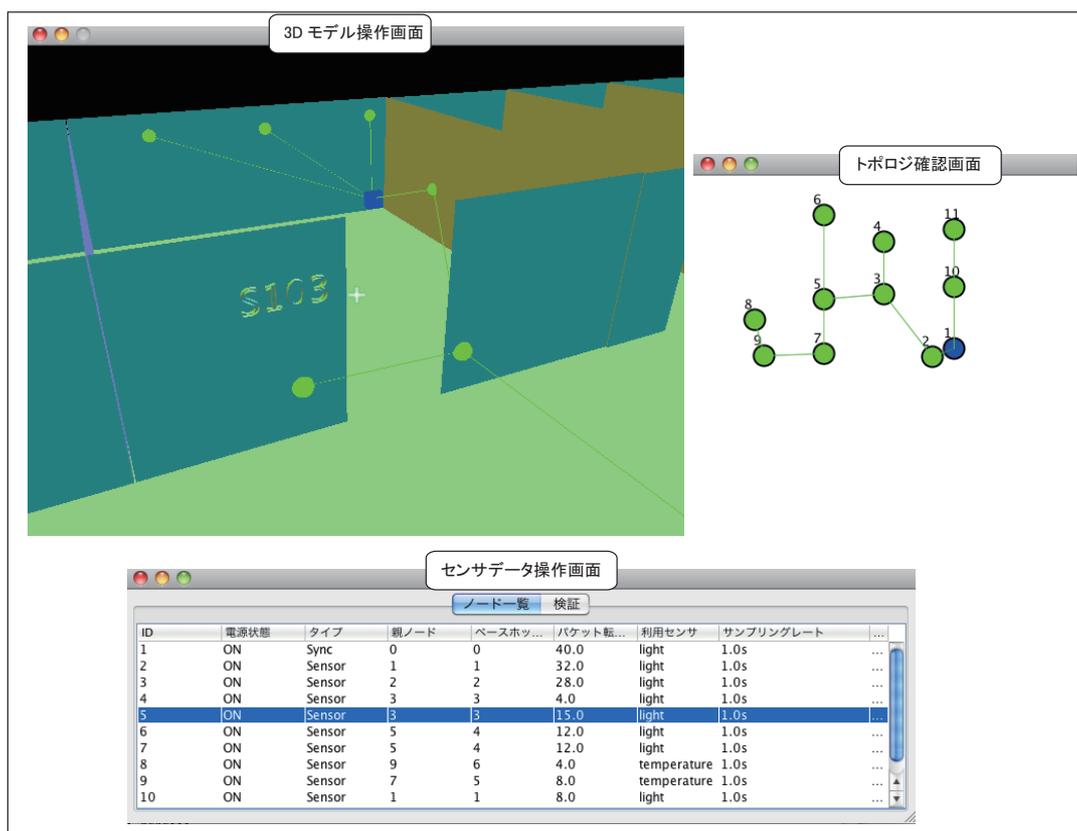


図 5.1 SenSONAR システム利用画面

3D モデル操作画面

ユーザは 3D モデル操作画面を用いて設置環境における任意の場所へのノード設置を実現する．実装は jMonkeyEngine3.0 を用いて行った．3D モデル操作画面における操作は，マウスとキーボードを用いて行われる．キーボードを用いてセンサノード設置と視点の上下左右の移動を行い，マウスを用いて視点の角度の調節を行う．センサノード設置の際には，画面中央に映る「+」が設置箇所を示すマークとなり，ユーザはマークを設置箇所へ移動させ設置ボタンを押す事で設置作業を行う．また一度ノードが設置されるとノードの到達範囲が可視化される，ユーザは次のノード配置にはこの可視化された情報を利用する事になる．配置されたノード間に到達性が存在する場合は

ノード同士が直線で結ばれ、接続関係を容易に把握する事が可能となる。これらのノード配置の流れを以下の図 5.2 に示す。

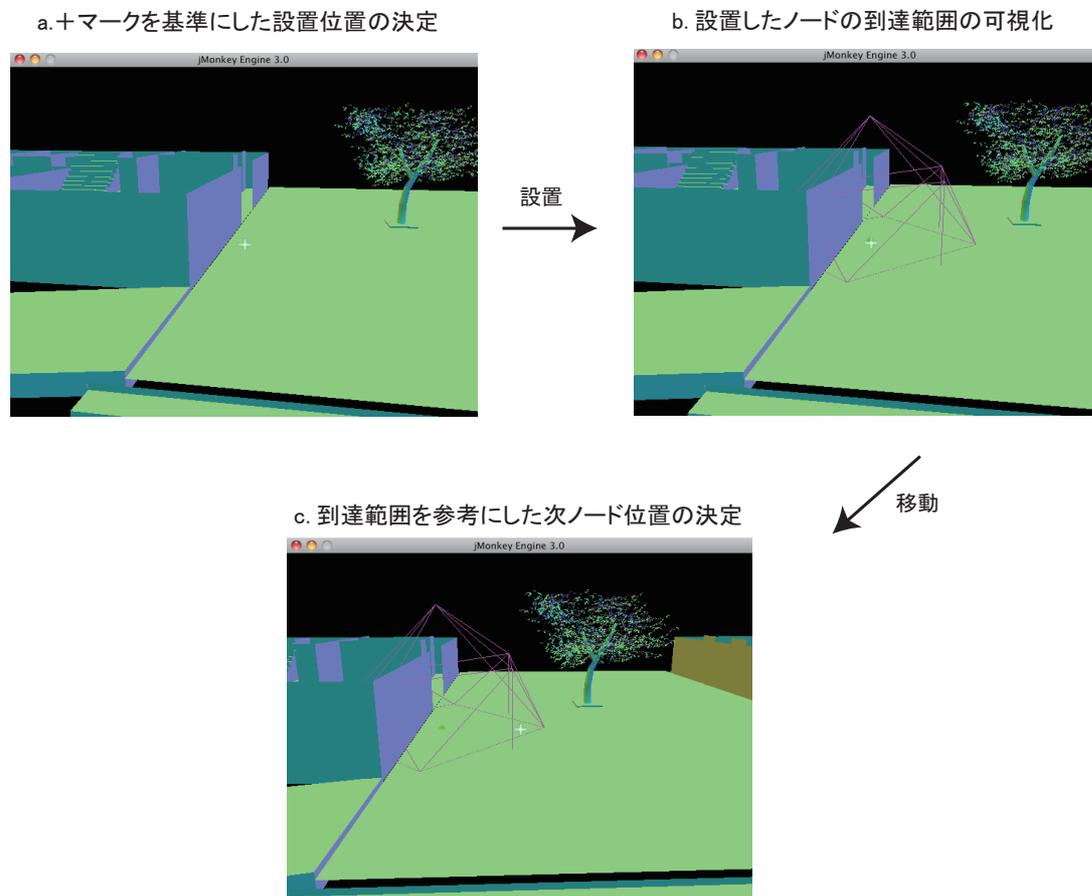


図 5.2 センサノード設置の流れ

トポロジ確認画面

トポロジ確認画面では、センサノード間の接続関係とセンサノード設置時の到達範囲を 2 次元で可視化し表示する。3D モデルでは全体像の把握が困難なため、センサネットワークトポロジ全体を容易に確認するための役割を持つ。3D モデルに設置されたセンサノードの座標情報を x 座標と y 座標のみ利用し平面化している。表示されるトポロジ情報は、3D モデル操作画面やデータ操作画面と連動しており、3D モデル操作画面やセンサデータ操作画面で情報が更新されると自動的にネットワークトポロジ確認画面も最新の情報へと更新される。

センサデータ操作画面

データ操作画面はデータ一覧タブと検証タブのふたつのタブビューで構成されており、それぞれ配置したセンサノードのデータ操作と、構築したセンサネットワークのデータを用いたデータベ-

スとの連動検証の役割を持つ。

- データ一覧タブ

データ一覧タブでは、配置したセンサノードが持つ情報を表形式で表示する。本タブではノードが持つ各情報の設定やノードの削除が可能となる。ノード配置時にはノード情報の利用センサやサンプリングレートが暫定的に入力されるが、それぞれのセンサノードの役割を変えたい時には表を直接編集する事でノード毎の設定を変更する事ができる。その他に、電源状態の操作によるトポロジ変化の確認などを実現する。以下の図 5.3 にノード情報操作時の例として特定のノードの電源状態を OFF にした際の、データ操作画面とトポロジ画面を示す。以下の例では、もともと 7 番ノードの親である 8 番が電源を切った事で、7 番がルートを変更し 3 番を親ノードに変更した事を確認している。

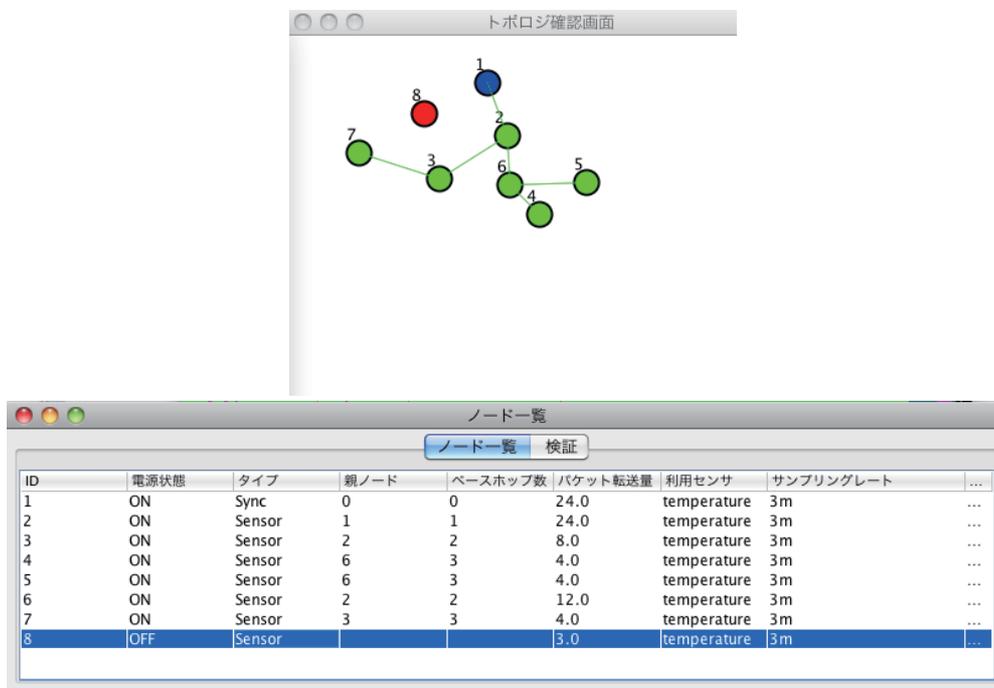


図 5.3 ノード情報の操作例

- 検証タブ

検証タブでは、構築したセンサネットワークの全体の情報確認と、データベース連動時の検証を行う。検証ではデータアーカイブ検証とアクセス検証機能を提供する。データアーカイブ検証時は、ユーザが想定する利用期間を入力するとシンクノードひとつあたりのデータ、クエリ実行間隔が自動的に入力され、データアクセス検証時は要求速度を入力し、検証ボタンを押す事で、これらの情報がデータベース検証部に送られ、結果を取得し検証タブに結果として入力された内容の実現可否を表示する。以下の図 5.4 に検証タブの画面構成を示す。

総ノード数:	到達性確保ノード数:	総データ量:
利用期間	シンクノード総データ量	クエリ実行間隔
<input type="text"/> 日	<input type="text"/>	<input type="text"/> msec
		想定データベース容量(CB)
		<input type="text"/>
		アーカイブ検証
想定クエリ	要求速度	
<input type="text"/>	<input type="text"/> msec	
		アクセス検証

図 5.4 検証タブ画面

5.2.2 ノード生成モジュール

ノード生成モジュールでは、GUI モジュールから得られたユーザの要求に基づき、任意の位置へのノード生成と中間ノードの自動生成がおこなわれる。任意の位置へのノード生成では GUI モジュールから空間における座標情報を取得し、座標情報と 4 章の設計で示したノード情報を生成する。また、GUI モジュールから任意の座標点ではなく範囲の座標、つまり島の範囲座標が送られてきた場合、中間ノードの自動生成を行う。自動配置ではまず、島の端にあり相手の島へ最も近いノードを接続として定める。接続ノードは指定された冗長性 + 1 の個数生成される。その後は以下の手順を行うよう実装した。到達範囲の推測は x 軸、 y 軸の 2 次元で行い、 z 軸の高さは一定の高さで配置を行うものとした。

1. 互いの島の接続ノードの到達範囲情報を取得する
2. 1 の結果に基づき到達範囲内にノードを設置する
3. 新しく配置されたノードを含むように最寄りの島の範囲を拡大し、追加されたノードを接続ノードとする
4. 互いの島の範囲が重なるまで 1-3 を繰り返す

5.2.3 ネットワークシミュレータモジュール

ネットワークシミュレータモジュールは到達範囲推測部、プロトコル設定部、トポロジ生成部、センサネットワーク管理部からなるが主に到達範囲推測部とトポロジ生成部の実装について述べる。

到達範囲推測部

到達範囲推測は主に、距離減衰と障害物減衰の 2 つを考慮し 3 次元上での到達範囲を推測する。その手順は大きく分けて以下の 2 つの手順をとる。まず到達範囲を 3 次元空間で推測するために、計測範囲を定義する。そして計測範囲において到達性の推測を行い、到達範囲を算出する。以下の手順を順に詳述していく。

- 到達範囲推測方法

到達範囲を3次元情報で算出するために、まず計測範囲の頂点を決定する。到達範囲の計測は球形にくまなく行う事が理想的であるが、演算量が増えてしまう事から本実装では球体を14の頂点で簡略化した形を用いる。ノード設置位置を原点とし、計測範囲を図5.5にしめす。配置位置を0位置としx軸、y軸、z軸上の正負方向に向あわせて6点、配置位置を0位置とした立方体頂点の8点の合計14つの頂点が計測範囲をかたどる頂点とする。

この計測範囲の頂点の距離は、利用するセンサノードが本来通信可能な限界到達距離である。そして原点から各頂点へ向かった線分上で、距離と障害物による影響を踏まえ、到達可能な頂点を線分上に設定する。これを14個分繰り返す事で、全ての頂点の座標が距離と障害物を踏まえ推測された到達可能な座標に更新される。それらの頂点を直線で結びユーザへ到達範囲として提示する。

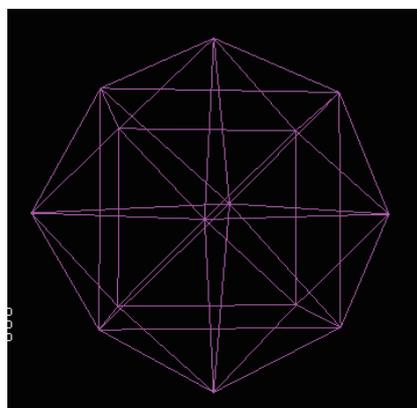


図 5.5 到達範囲計測に用いる頂点

- 原点から頂点への線分上での到達可能な距離の算出

到達範囲を算出するためには、上述した14つの頂点までの距離を、距離による減衰と障害物による減衰を反映させた座標に変更しなければならない。距離と障害物による減衰を考慮した到達距離の推測には、利用するセンサノードの通信可能な範囲を明確にする必要がある。また、推測方法を特定するためにもRSSI(受信電波強度)と距離の関係、RSSIと障害物の関係性を明らかにする必要がある。そこで実装を行う前に予備実験を行った。予備実験では、通信ノード間の距離毎の受信時RSSI測定と、通信ノード間の障害物によるRSSIの減衰量を測定を行った。始めに距離別のRSSI測定予備実験ではデルタ館の1階廊下においてSunSPOTを1m, 3m, 5m, 10m, 15m, 20mの間隔で置き、それぞれの1000パケットを送信しRSSIと到達パケット数を測定した。測定時の出力は-3dBmとした。以下の表5.3に実験結果を示す。

以上の実験結果から、15mも20mも共に到達性が得られたが、RSSIにはわずかな差分しかない事や、20mではパケット到達率が約100分の1である事を考慮すると、送信頻度が

	1m	3m	5m	10m	15m	20m
RSSI	-11.97	-24.90	-31.93	-36.71	-45.42	-46.63
到達パケット数	1000	992	948	971	567	91

表 5.3 受信時の平均 RSSI(dBm) と到達パケット数

数分ほどの通信時には利用が困難であるといえる．そのため，本実装では SunSPOT の通信可能な限界距離を 15m，RSSI を -45 と定義した．

また，実験結果から RSSI と距離が反比例している事が読み取れる，そこで RSSI の値から近似直線を取り，RSSI とノード間の距離 $L(m)$ の関係を以下の様に定義した．

$$P = 40 * \frac{1}{L} - 45 \quad (5.1)$$

次に障害物によってどれだけ RSSI が減衰するのかを測定するために，今回は大きく影響を与える要因である壁を対象とし，ノード間の距離が 10m でありノード間の直線上に壁がある事で発生する RSSI の変化を測定した．本実験環境においては，壁はガラスのみを利用した．以下の表 5.4 に実験結果を示す．

壁の数	0 枚	1 枚	2 枚	3 枚
RSSI 減衰率	0%	7%	17%	33%

表 5.4 受信時の平均 RSSI 減衰率と壁の数

この結果から，本実装では壁の枚数と減衰率の関係を以下の様に定義した．減衰率 R と壁の枚数 W の関係を以下の様に定義した．

$$R = (13W - 7)/100 \quad (5.2)$$

これらの関係性を踏まえると，距離と障害物の 2 つの要因を考慮し推測する RSSI である Prw はノード間の距離 L ，障害物の数 W ，距離減衰を踏まえた RSSI を表す P ，障害物による RSSI 減衰率 R を用いて以下のように表す事が出来る．SenSONAR システムでは 3D モデルデータから壁の枚数 W を取得し，推測 RSSI の Prw が限界 RSSI である -45 を上回る値を距離 L を探索する事で，通信可能な距離の推測を行う．この条件をみたく距離 L の探索は，SunSPOT の限界距離である 15m から 1m 減少させて値を確認していく事で実現する．

$$Prw = P * (1 - R) = (40 * \frac{1}{L} - 45) * (1 - (13W - 7)/100) \quad (5.3)$$

トポロジ生成部

トポロジ生成部では、プロトコル情報をもとにノード間の接続情報を生成する。本実装では、簡易ルーティングプロトコルを実装し用いた。簡易プロトコルでは breadth-fast プロトコルとして、各ノードがシングルホップを利用し、シンクノードに到達するまでのホップ数が少なくなるよう経路をとるよう実装した。各ノードはシンクノードまでのホップ数と近隣ノードまでの距離、自己の到達範囲情報を保持しており、それを用いてルーティングを行う。以下の図 5.6 に簡易プロトコルの実装内容を示す。

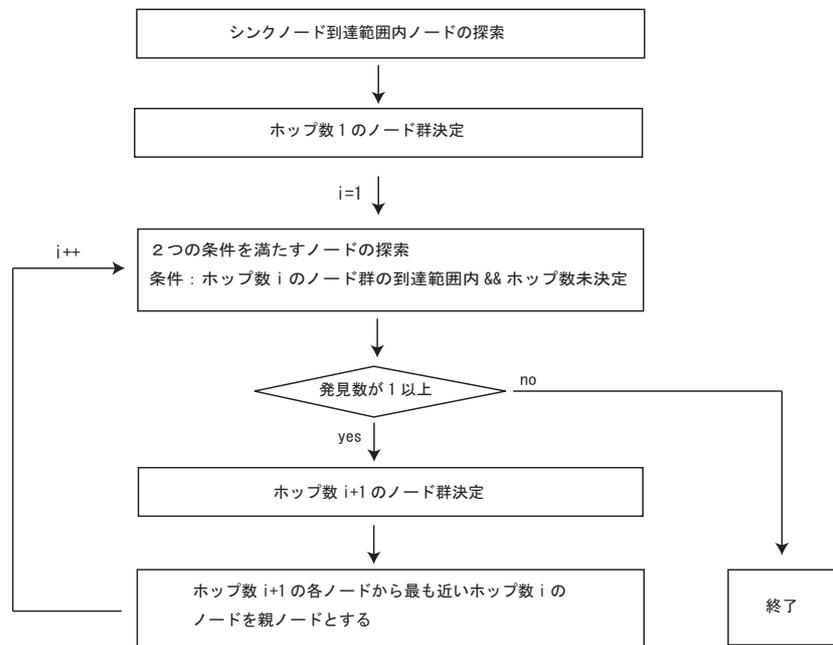


図 5.6 簡易プロトコル実装仕様

5.2.4 データベース検証モジュール

データベース検証モジュールでは、データベース検証部がセンサネットワークの情報とユーザからの想定利用情報をもとに、検証結果を生成し情報提示部へと送る。本実装ではデータベースにおけるデータの管理は単一のテーブルで行うものとした。

そのためにはネットワークのデータサイズを算出する必要があるが、本実装ではノードが取得する情報量は MySQL5.1 の仕様 [23] に基づきデータカラムの型に必要なサイズを基準とした。例えば温度を取得するセンサに float 型を利用した場合 1 データあたり 4byte の領域を必要とする。そしてデータベースの記憶可能領域の取得はデータを保存するテーブルの記憶可能領域とファイルシステムの記憶可能領域を利用する。テーブルの記憶可能量域では SHOW TABLE STATUS FROM DB 名 LIKE ' テーブル名' を実行する事で、テーブルの限界サイズを表す

Max_data_length 変数を取得し，それを記憶可能領域とした．またデータアーカイブ検証のための保存クエリ実行では，マルチスレッド化し保存クエリ送信間隔が前回のクエリ処理に影響を受けないよう独立させた．

5.3 まとめ

本章では，SenSONAR システムの実装について述べた．初めに，実装環境を述べ，次に第 4 章で述べた設計に基づき各モジュールの詳細な実装内容について述べた．次章では，SenSONAR の評価実験とその結果について述べる．

第 6 章

評価

本章では，SenSONAR の評価を行う．本システムの評価手法として，基本性能と支援手法の 2 つの評価実験を実施した．まずはじめに，SenSONAR が提供する各機能の推測精度を測定した．ついで，システムを実際にセンサネットワーク基盤の構築に利用してもう実験を行った．これらの実験概要と実験結果を示し，結果に対する考察を行う．

6.1 構築支援ツール評価実験

本節では，SenSONAR の評価を行うため，被験者に実際にセンサネットワークを構築しその利用を検証してもらう実験を行った．評価実験では，センサネットワーク基盤構築時の問題点であるノード到達性問題，トポロジ異常問題，データアーカイブ問題，アーカイブデータアクセス問題がそれぞれツール利用によってどれだけ回避されたかを評価する．以下にその実験環境，手順そして結果について延べ，得られた考察をまとめる

6.1.1 実験環境

本評価実験では，対象空間内に人感センサを用いた来客検知アプリケーションの構築を想定とし，実際にセンサノードを対象空間内に設置しセンサネットワーク基盤を構築してもらった．その際に仕様上の通信距離をユーザへ伝え、地図と定規を用いて設計をしてもらった場合と、本支援ツールを用いた場合の2ケースの実験を行った．実験環境は慶應義塾大学の大学院棟デルタ館の1階とデルタ棟周辺の屋外空間を対象とした．以下の図 6.1 に実験環境のモデルを示す．被験者は，情報機器の操作に慣れた 20 代の学生 3 名である．その全員が実空間情報を用いるシステム構築のためにマルチホップを用いた無線センサネットワークシステムを構築した経験があった．全員が SenSONAR システムの利用経験が無い．また，実際に配置するのに利用するセンサノードは SunSPOT を利用し，実装で述べた簡易プロトコルと同じものを評価実験用に SunSPOT へ実装し利用した．配置に利用する SunSPOT は組み込まれた LED の色によって，接続の有無とシンクノードまでのホップ数が表現され，被験者は実際に配置を行う際にはそれをたよりに配置を行う．

6.1.2 実験手順

実験において被験者は，実験の目的や SenSONAR の利用方法の説明を受けた後，センサネットワーク基盤の構築作業を行った．想定するアプリケーションのための実現のためにセンサネットワークを構築するというシナリオで，実際にセンサネットワークを設計，構築し，利用の検証を行うという手順を，ツールを使わない場合と使う場合で2ケース行った．また構築する際にユーザには，上述した図 6.1 に示されたシンクノードの設置箇所への設置，3 か所へのセンサノード設置が義務となっており，どこノードを追加しても良いので3箇所のセンサノードからシンクノードへデータを収集するセンサネットワークを構築するというタスクが与えられた．具体的には2つのケースではそれぞれ以下の手順で作業を行ってもらった．実験では，実際の構築作業において発生した手順の繰り返しとして，ノード配置後に到達性が得られなかった回数とトポロジに異常があった回数，そしてそのために用いたノード数を計測した．そして，データベース検証時にはツールによって提示された情報の精度の評価として，実際に環境モニタリングによって収集された実データを挿入しツールが推測した値と，実データを用いて得られた値の比較を行った．比較ではツールが算出したセンサネットワークのデータ総量と実際のデータ総量の比較，ツール利用によってユーザ

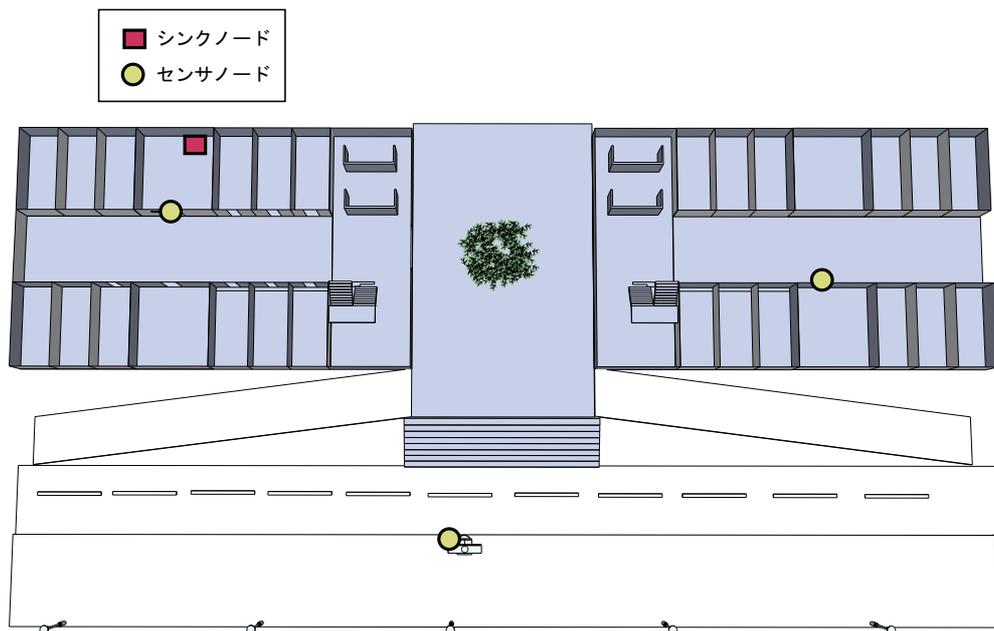


図 6.1 支援ツール評価実験環境

が求めたクエリ応答時間と実際の応答時間の 2 つの値の計測を行った。また実験終了後にツールに関するアンケートを 5 点段階評価で行った。以下に実験における 2 つのケースの手順を詳述する。これらの実験結果から、センサネットワーク基盤構築時の手順繰り返しの原因となる 4 つの問題を回避できたか否かを評価する。

1. ツール非使用ケース

- (a) 仕様情報からセンサノードの到達範囲とルーティングプロトコルを確かめる
- (b) 地図、定規、ペンを用いて位置とトポロジを設計する
- (c) 設計内容に従って実際にセンサノードを配置する
- (d) 設計したトポロジを実現できたかを確認する
- (e) センサーつあたりのデータ量とサンプリングレートをもとに、128GB のデータベース容量がある場合の限界利用期間を算出する
- (f) ダミーデータを自分で挿入し、クエリを実行し応答時間から見た限界利用期間を求める

2. ツール使用ケース

- (a) ツールを用いてセンサノードの位置とトポロジを設計する
- (b) 設計内容に従って実際にセンサノードを配置する
- (c) 設計したトポロジを実現できたかを確認する
- (d) ツールを用いて容量、応答時間の観点から限界利用期間を取得する

6.1.3 結果

実験結果として構築実験において計測した項目と、作業に関するアンケート結果から構築作業とデータベース検証作業に関する実験結果を順に述べる。構築支援機能群ではノード配置支援機能とノード状態操作機能によって、ノード到達性問題、トポロジ異常問題が回避されたかを評価する。データベース検証機能群ではデータアーカイブ検証機能とデータアクセス検証機能によって、アーカイブ問題、アクセス問題がそれぞれ回避されたかを評価する。

構築支援機能群について

表 6.1 は実際にノードを配置した後に、トポロジに異常が発生した回数、到達性に問題が発生した回数、利用した合計利用ノード数を被験者がツールを用いなかった場合と、用いた場合に分けて示している。実験結果では、ツールを非利用時も利用時もどちらも、トポロジ異常が発生しておりツール利用時、非利用時どちらも平均 3 回であり、回数に大きな変化は見られなかった。しかし、到達性問題発生数についてはツール利用時に回数が大きく減少し、被験者 B,C の例では 0 回と発生しなかった。また、利用されたノード数もツール利用によって被験者 A と被験者 C の数がそれぞれ 1 と 2 減少しており、は平均 1 ノード分減少している事がわかった。

次に表 6.2,6.3 に実験終了後にユーザへ行ったアンケート内容と、その結果を示す。表 6.2 ではツールを利用せずに紙とペンを用いて設計を行ったケースの実験結果を示し、表 6.3 ではツールを利用して設計を行ったケースの実験結果を示す。

被験者	トポロジ異常問題発生数	到達性問題発生数	利用ノード数
A(ツール非使用)	2	2	12
B(ツール非使用)	2	2	13
C(ツール非使用)	5	5	13
A(ツール使用)	1	1	11
B(ツール使用)	4	0	13
C(ツール使用)	4	0	11

表 6.1 実験結果

設問	被験者 A	被験者 B	被験者 C
ノードの位置決定は容易だった	2	3	2
トポロジの設計は容易だった	3	3	2
接続性の実現は容易だった	2	3	2
トポロジの実現は容易だった	1	3	1
データが保存されている事の検証は容易だった	2	3	2
長期利用を考慮したサーバ容量の検証は容易だった	2	1	1

表 6.2 ツール非使用時の構築作業に関するアンケート結果

設問	被験者 A	被験者 B	被験者 C
ノードの位置決定は容易だった	5	5	5
トポロジの設計は容易だった	5	5	5
接続性の実現は容易だった	4	5	4
トポロジの実現は容易だった	4	4	3
データが保存されている事の検証は容易だった	5	5	4
長期利用を考慮したサーバ容量の検証は容易だった	5	5	5

表 6.3 ツール使用時の構築作業に関するアンケート結果

データベース検証機能群について

データベース検証機能群の評価として、センサデータアーカイブ検証機能によって提示されたグラフの情報の精度の評価を行った。その結果、システムが 3D 空間に構築されたセンサネットワークの総データ量の誤差はデータベースのレコード数 1 万行あたり約 10byte である事がわかった。以下の図 6.2 にレコード数あたりのデータサイズ誤差を示す。次に、アーカイブデータアクセス検証機能によって提示されたグラフを利用する事で、ユーザが自己のクエリの応答時間推測をどの程度正確に行えた評価した。以下の図 6.3 にツールを用いてユーザが推測した応答時間と、実際にデータを挿入し同クエリを発行し測定した応答時間の誤差を示す。その結果、誤差の平均は絶対値で 79msec となった。また、誤差の範囲として -200msec から 80msec の間の誤差が発生した。また全ての被験者に共通する傾向として、誤差の値はマイナス値になる例が 9 割を占めツールを利用して推測された応答時間の推測値は、実際にデータを挿入して測定した値よりも傾向が見られた。つまり、ツールを用いて推測した応答時間は実際の応答時間よりも遅く見積もられる傾向が見られた。また、表 6.4 に示すアンケートを合わせてみるとデータアーカイブ検証機能については全てのユーザが 5 点とした事からユーザにとっても容易に正しい情報を取得出来たと言える。反面、アーカイブデータアクセス検証機能では平均 3.3 点となり、非利用時の平均 1.3 よりも高い得点ではあるがユーザにとって負荷がかかっていた事がみてとれる。

設問	被験者 A	被験者 B	被験者 C
(ツール非使用) 長期利用を考慮したサーバ容量の検証は容易だった	2	2	1
(ツール非使用) 長期利用を考慮した DB サーバ応答時間の検証は容易だった	1	2	1
(ツール使用) 長期利用を考慮したサーバ容量の検証は容易だった	5	5	4
(ツール使用) 長期利用を考慮した DB サーバ応答時間の検証は容易だった	4	3	3

表 6.4 データベース検証作業に関するアンケート結果

6.1.4 構築実験結果考察

実験結果をもとに、ツールの機能をセンサネットワーク構築支援機能とデータベース検証の 2 つに分け、それぞれの結果考察を行う。

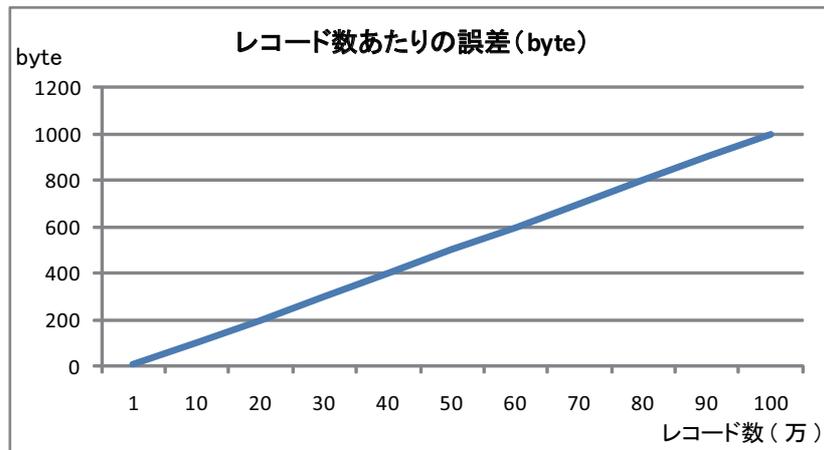


図 6.2 アーカイブ検証機能において提示された総データ量と実際のデータ量の誤差

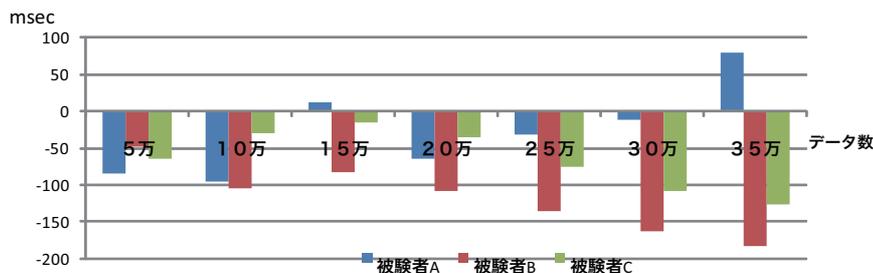


図 6.3 ツール利用によってユーザ推測したクエリ応答時間と実際のクエリ応答時間の誤差

センサネットワークの構築について

実験結果から、ツールを利用しなかった場合に対して、ツールを用いる事でノード間の非到達問題の発生回数が減少する事がわかった。その理由として実験後のインタビューでは、地図に定規とペンを用いて有効範囲を考慮しながらノード配置位置の決定する際、一定の間隔を決定し、その間隔を全てのノード間に適用している例や、感覚で距離を適宜変えている例が多い事がわかった。この事から実際の環境に適したノード配置が行えず、実際に配置を行うと近すぎる例や遠すぎる例が発生し、到達性、トポロジの両方に問題が発生しているといえる。それに対して、ツールを用いた場合、被験者はツールが提示した有効範囲に素直に従いノード位置を決定していった。その結果、実験結果でも述べたように、ノード間の非到達問題の回数と利用ノードを削減させる効果が得られた。これらの、非到達問題の減少と利用ノードの削減という結果を踏まえると、ツールを利用する効果としてノードの配置位置決定ではノード数の効率が良く、到達性確保が容易な配置が実現できたと言える。アンケートにおいて被験者全員がノード決定やトポロジ決定が容易だったと応えた事からもわかる。

また、トポロジ異常の発生回数については減少した例と増加した例があり、平均でみた場合は変化がなくこの問題を回避したという明確な結果は得られなかった。その原因として、ツール利用時にツールでは到達範囲外と推測したノードへの到達性が得られてしまった例があげられる。これは多様な場面で起きた事を考えると、現在の到達範囲の手法では確実に到達する範囲は示せていても、限界まで広げられる範囲は示せていないと言える。しかし、アンケートのトポロジ実現が容易だったという設問では、ツール未使用時は平均 2 だったのに対し、ツール使用時は平均 3.66 という結果が得られた事からツールを用いる事でトポロジ異常が発生した際の、再設計の負荷が減少している事はみられた。

しかし、ツール利用時もノード間の非到達問題が被験者 A において 1 度発生しておりすべてを 0 にする事は出来なかった。その原因として、ノードの高低差や配置位置の関係を考慮しきれなかった事が考えられる。配置位置はユーザが自由に配置を行ったため、地面に設置されるケースがあった。そうすると電波の進行方向が限定され、自分より低い場所に届きにくかったのではないかと推測される、そのため高低差の激しいエリアに関しては提示された到達範囲の情報は正しく機能していなかったと言える。この事から、今後の課題としてユーザへ提示する到達範囲の推測には現状の距離と障害物による推測のみでは不十分であり、ノード間の高低差や設置された位置による基本的な到達範囲の変形を考慮する必要があると考えられる。それにより、トポロジ以上の発生回数を削減できる事も見込まれる。

データベースの利用検証について

実験結果から、容量からみた検証において 1 万行あたりの誤差が 10byte という精度は被験者にとって大きな誤差ではなく、被験者は判断に十分な精度を持った情報から判断が行えたといえる。つまり、本研究において想定した環境においてはツール利用により、データアーカイブ問題にもとづく手順の繰り返しの回避効果が見られたと言える。本プロトタイプの実装においては、DBMS 環境やテーブル構造を定義しておりデータ総量の算出も MySQL の仕様情報をもとに算出を行ったためこの様な精度が得られと考えられる。しかし、同 DBMS においてもより複雑な構造をしたデータベーステーブルの考慮を行う必要が挙げられるため、今後はセンサネットワークにおいて利用されるデータベース構造の詳細な定義を行いそれを踏まえた情報提示の検証を行う必要がある。

また、応答時間からみた検証では実験結果からツール利用によってユーザが得た応答時間と実際データを挿入し測定した応答時間の誤差は平均で約—79—msec となったが、これは環境モニタリングの様なデータの時間精度が粗い例においては大きな問題となりにくいだが、人の行動検知など厳密な応答時間を求めるアプリケーションにおいては問題発生の原因となり得ると言える。また、アンケートから複数のグラフを組み合わせて自分のクエリの応答時間を推測する事はなれないと困難である事や、被験者が設定したクエリによって誤差が大きく変わってしまう問題がある事がわかった。そのため、特定のアプリケーションを想定した際には有用性が見られる効果が見られた、汎用的なセンサネットワーク基盤の構築という観点からは不十分な結果であったと言える。

6.2 ユーザビリティ評価

本節では，SenSONAR のユーザビリティ評価について述べる．ユーザビリティ評価は上述した構築実験終了後に，アンケートを用いて行った．被験者は情報機器の操作に慣れた 20 代男性 3 名であり，全員が始めて SenSONAR システムを利用した．以下アンケートより得られた結果とその考察について述べる．

6.2.1 結果

表 6.5 にユーザビリティアンケートに用いた設問とその結果を示す．アンケートでは，設問 2，5 の「効率的に作業ができたか」と「主観的満足度」についてはそれぞれ平均 5 点，平均 4.66 と高い結果を得られたのに対し，設問 1 の「使い方を習得するのは簡単だ」については平均 3.33，設問 4 の「操作方法は記憶しやすい」については 3.66 という結果に止まった．中でも被験者 B はどちらの質問に対しても 2 点としている．

設問	被験者 A	被験者 B	被験者 C
使い方を習得するのは簡単だ	5	2	3
効率的に作業ができたか	5	5	4
操作方法は記憶しやすい	5	2	4
エラー発生後の回復など、必要だと思った事が全部できる	4	3	4
主観的満足度	5	5	5

表 6.5 ツールに関するユーザビリティアンケート結果

6.2.2 考察

前項のアンケート結果から，作業の効率性やユーザの利用に関する満足度については高い結果が得られた事がわかった．中でもユーザの主観的満足度についてはリカーとの主観的満足度の中間点は 3.6 である事が知られており，3 人の被験者から 5 をえられた事は高い満足度が得られと言える．そのため，きちんと利用する事ができれば，ユーザにとって効率の良い作業が不快感無く実現されている事がわかった．しかし，使い方の学習や使い方の記憶しやすさについては高い得点を得る事ができなかった．これは，SenSONAR は複数の画面で構成されている事や，操作インターフェースがキーボードからマウスに柔軟に入れ替わる事に起因している事が考えられる．また，キーボードを利用する事でマウスによる指定やドラッグなどの様な直感的な動作ではなく記憶に基づく動作が必要とされている事が挙げられる．またインタビューにおいて再設計を行うや設置場所を参考にするためにはラップトップを持ち歩かなければならず，移動が多い場合には紙の方が良かったという答えが得られた．この事から，現在の機能を利用する事でユーザにとっては効率的に作業が出来ると感じられるが，それに対して利用方法を学ぶためには学習と慣れが必要である問題が明らかに

なった．そのため，今後は現在の機能群を維持した状態でのインタフェースの簡略化や携帯端末への移植などにより，よりユーザビリティを高められると考えられる．

6.3 まとめ

本章では，ツール利用による評価実験について示し，その評価結果から有用性の検証を行った．次章では，本研究における今後の課題の検討を行い，その後 本論文をまとめる．

第7章

結論

本研究では，センサネットワーク基盤構築支援を目的として SenSONAR の設計と実装，その評価を行った．本章では評価における考察から得られた結論と今後の課題を簡潔に述べ，本論文をまとめる．

7.1 結論

計算機の小型化や無線小型センサの普及に伴い、空間のスマート化やコンテキストウェアシステムの実現のために、センサネットワークがひろく用いられるようになった。センサネットワークを利用する事でこれまでは容易に取得できなかった実空間の情報を取得できるようになった。しかし、センサネットワーク利用するためには、何度も設計、構築、運用のフェーズを行き来し繰り返さなければならなかった。実空間で再設計を行い再配置を繰り返す事は作業コストが大きくユーザへの負荷となっていたが、設計時の情報が少なく、センサネットワーク基盤の構築支援が不十分であった。

本論文では、センサネットワーク基盤構築時の手順の繰り返しの回避を支援する、基盤構築支援ツールを提案した。提案ツールでは設置環境に即した 3D 空間を用いたセンサネットワーク構築支援機能、3D 空間で構築されたセンサネットワークを用いたデータ利用検証機能をを提供した。これまで構築者は、実際に実空間でセンサネットワークを構築しなければ設計内容の検証を行う事が出来なかったが、本ツールを利用する事で、事前に効率の良いセンサネットワークの設計とデータ利用検証を行う事が可能となった。それにより、実空間での配置後に手順を繰り返し作業コストを低減した。

プロトタイプ実装を用いて実際のセンサネットワークの設計、構築、利用検証を行う構築評価実験とユーザビリティ評価を行った。それらの評価結果から、SenSONAR を利用する事で、仕様書と地図を用いる一般的な方法に比べて、到達性問題による手順の循環を回避できる事とノード効率の良い配置を実現できる事、そして現在のデータベース環境が設計したセンサネットワークの適切な利用期間を容易に検証できる事が認められた。しかし、トポロジ異常問題の回避を実現するためには、配置環境を考慮した到達範囲の推測において、ノード間の位置関係やノードの設置場所への考慮が必要である事がわかった。次に、データベース環境の検証において容量の観点においてはユーザにとって容易かつ正確な検証機能を提供できたが、応答速度の観点においては、汎用的に利用するには誤差があり情報提示手法の改善を行う必要がある。また、ユーザビリティ評価においては、SenSONAR は操作に慣れるまでには時間がかかる事と、操作が複雑である事がユーザにとっては覚えにくい事がわかった。しかし、そうした問題点があるのに対して、利用する事で効率性の向上やユーザから高い主観的満足度を実現した。

これらの事から今後は、到達範囲推測手法や応答時間検証手法の改善を行ったうえで、より多くのセンサノードへの対応や実際にセンサノードを配置したあとの継続したツールによる支援、操作インタフェースの簡略化などに取り組む。

7.2 今後の課題

SenSONAR の実装は十分ではなく、さらなる改善が求められる。本節では、SenSONAR の設計及び実装における問題点と解決策を述べ、今後の展望とする。

7.2.1 ハイブリッドシミュレーションの実現

SenSONAR では仮想的に生成したセンサノードと実空間に設置されたセンサノードの接続を実現する必要がある。実験において、SenSONAR システムを用いて設計を行った後に実空間において配置を行った際に、再度設計を行いノードを追加するという場面や、設計構築が完了した後に新たにノードを追加し拡張したいケースなどがみられた。このことから構築作業においては、既に配置した後など、3D空間の情報と実空間の情報を融合して利用したいケースが有る事がわかったが、現在のSenSONARは実空間と仮想空間のセンサノードの接続性が無くその要求を実現する事が出来なかった。そのため、今後は仮想センサノードとリアルセンサノードを区別しながら3D空間内で扱える必要がある。それにより、実空間において配置が行われた後でも柔軟かつ効率的なセンサネットワークの構築を支援する事が可能となる。

7.2.2 多様なプロトコルへの対応

プロトタイプ実装では、簡易プロトコルを実装し3D空間でのトポロジ再現を行った。しかし、これだけでは3D空間において特定のトポロジ再現しか行えず、異なるプロトコルの多様なセンサノードへ対応するには不十分である。今後はSmartCorridorやMebiusSensorの様に多様なセンサノードによる大規模なセンサネットワークの構築が想定され、そうした環境に対応するためにもAODV、DSRやDSDVなど多様なルーティングプロトコルに対応する必要がある。

7.2.3 操作インターフェースの簡略化と携帯端末への移植

プロトタイプ実装での評価結果やアンケート結果から、SenSONARの操作方法はユーザにとって学習する事と覚える事が大変である事がわかった。主にその原因は複数のウィンドウと、キーボードとマウスを多用する操作体系にあると考えられる。また、設計したセンサネットワークの構築や管理を行っていく上で、SenSONARを用いて設計した内容を屋外で手軽に参照する方法が必要である事がインタビューからわかった。そのため、今後はタッチパネル型の携帯端末などへの移植を行い、操作インターフェースの簡略化とポータブル化の実現を目指す必要がある。

謝辞

本研究を進めるにあたり，ご指導を頂きました慶應義塾大学環境情報学部教授徳田英幸博士に深く感謝いたします。また，慶應義塾大学環境情報学部准教授高汐一紀博士，慶應義塾大学環境情報学部教授清木康博士，慶應義塾大学環境情報学部専任講師中澤仁博士には，本論文の執筆にあたって多くの御助言を賜りました事を心より感謝致します。

また，慶應義塾大学徳田，高汐，中澤研究室の諸先輩方には，お急がしい中貴重な示唆やご助言，ご指導を頂きました。ここに深い感謝の念を表します。

そして，研究の日々を共に過ごした，山本純平氏，米澤祐紀氏，小川正幹氏，金澤貴俊氏，野沢高弘氏，HORN 研究グループの皆様，瀧本拓哉氏，論文執筆にあたり多大な協力をしてくださった，生天目直哉氏，伊藤友隆氏，西條晃平氏に感謝します。最後に，研究生活だけでなく6年間の大学，大学院生活においていかなる時も支えてくれた家族に深く感謝し，謝辞と致します。

2011年2月14日

徳田 義幸

参考文献

- [1] Masaki Ogawa, Jumpei Yamoto, Yoshiyuki Tokuda, Takatoshi Kanazawa, Naoki Nakagawa, Masaki Ito, Soko AokiI, Hiroto Aida, Jin Nakazawa, Kazunori Takashio, and Hideyuki Tokuda. Smart corridor: Development of smart space for corridors and public streets. In *The 3rd International Workshop on Design and Imtegration Principles for Smart Objects (DIPSO 2009)*, 2009.
- [2] A.K.Dey. Understanding and using context. In *Personal Ubiquitous Computing,5(1):4-7, 2001*, 2001.
- [3] Ito M. Airy notes : An experiment of microclimate monitoring in shinjuku gyoen garden, networked sensing systems, 2007. In *INSS '07. Fourth International Conference, 6-8 June 2007, 260-266*, 2007.
- [4] Beigl M. μ parts : Low cost sensor networks at scale. In *Proc. 7th International Conference on Ubiquitous Computing Demonstration (UbiComp'05), Tokyo, Japan, 2005*, 2005.
- [5] Lars Erik Holmquist, Friedemann Mattern, Bernt Schiele, Petteri Alahuhta, Michael Beigl5, and Hans-W. Gellersen. Smart-its friends: A technique for users to easily establish connections between smart artefacts. In *UBICOMP 2001: UBIQUITOUS COMPUTING,Lecture Notes in Computer Science, 2001, Volume 2201/2001, 116-122,*, 2001.
- [6] 米澤拓郎, 榊原寛, 中澤仁, 高汐一紀, 徳田英幸. Spot and Snap: DIY Smart Object Serviceを実現するセンサノードと日用品の関連付けインタラクシオン, 第3章. 情報処理学会 論文誌「インタラクシオンの原理と応用」特集号, Vol48, 2007.
- [7] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. A tiny aggregation service for ad-hoc. 5th Symposium on Operating Systems Design and Implementation (OSDI '02), 2002.
- [8] Jun 'ichiYura, Jin Nakazawa, Hideyuki Tokuda. Guaranteed data storing mechanism for mobile sensor database. In *DBSJ Letters, Vol.6, No.4, March 2008, pp.5-8*, 2008.
- [9] 伊藤昌毅. センサネットワーク技術を活用した環境モニタリングの実現. 電気学会 次世代産業システム研究会, IIS-10-070, pp. 23-28, 2010.
- [10] *Crossbow Technology, Inc. eKo System. <http://xbow.com/>.*
- [11] *ns2 (network simulator 2). <http://www-nrg.ee.lbl.gov/ns/>.*

- [12] *SunMicrosystems, Inc. Solarium*. <http://http://www.sunspotworld.com/>.
- [13] Smith Randall B. Spotworld and the sun spot. In *Information Processing In Sensor Networks*, 565-566, 2007, 2007.
- [14] John J.Barton, HP Labs, and Vikram Vijayaraghavan. Ubiwise, a ubiquitous wireless infrastructure simulation environment. In *UbiComp*, 2002.
- [15] E. O'Neill, M. Klepal, D. Lewis, T. O'Donnell, D. O'Sullivan, and D. Pesch. A testbed for evaluating human interaction with ubiquitous computing environments. In *Proc. of 1st Int'l Conf. on Testbeds and Research Infrastructures for the Development of Networks and Communities*, pp. 6069, 2005.
- [16] Hiroshi Nishikawa, Shinya Yamamoto, Morihiko Tamai, Kouji Nishigaki, Tomoya Kitani, Naoki Shibata, Keiichi Yasumoto, and Minoru Ito. Ubireal: Realistic smartspace simulator for systematic testing. In *UBICOMP 2006: UBIQUITOUS COMPUTING*, 2006.
- [17] *Crossbow Technology, Inc. MoteView*. <http://xbow.com/>.
- [18] Marc T. Kouakou, Shinya Yamamoto, Keiichi Yasumoto, and Minoru Ito. Deployment planning tool for indoor 3d-wsns. In *UbiComp*, 2010.
- [19] Georgios Vouzounaras, Juan Diego Perez-Moneo Agapito, Petros Daras, and Michael G. Strintzis. Laser brush: a flexible device for 3d reconstruction of indoor scenes. In *SMVC '10 Proceedings of the 2010 ACM workshop on Surreal media and virtual cloning*, 2010.
- [20] Martin Habbeke and Leif Kobbelt. 3d reconstruction of indoor and outdoor building scenes from a single image. In *SPM '08 Proceedings of the 2008 ACM symposium on Solid and physical modeling*, 2008.
- [21] *ASKUL Corporation*. <http://www.askul.co.jp/>.
- [22] *jmonkeyengine.com. jMonkeyEngine3.0*. <http://www.jmonkeyengine.com/>.
- [23] *MySQL5.1* <http://dev.mysql.com/doc/refman/5.1/>.
- [24] Jakob Nielsen. Usability engineering. In *Morgan Kaufmann Publishers Inc., San Francisco, CA, USA*, 1993.

付録A

評価実験におけるアンケート

性別 () 年齢 () SQL利用経験 () センサネットワーク構築経験 ()

———— ツール非利用時について ————

①口設計について

- | | | | | | |
|------------------|-----|---|---|---|----|
| ・ 配置位置の設計は容易だった | いいえ | | | | はい |
| > ノードの位置決定は容易だった | 1 | 2 | 3 | 4 | 5 |
| ・ ノードの接続実現は容易だった | | | | | |
| > トポロジの設計は容易だった | 1 | 2 | 3 | 4 | 5 |

②実配置について

- | | | | | | |
|-----------------|---|---|---|---|---|
| ・ 接続性の実現は容易だった | 1 | 2 | 3 | 4 | 5 |
| ・ トポロジの実現は容易だった | 1 | 2 | 3 | 4 | 5 |

③検証について

- | | | | | | |
|--------------------------------|---|---|---|---|---|
| ・ 長期的な利用を考慮した際のサーバ容量の検証は容易だった | 1 | 2 | 3 | 4 | 5 |
| ・ 長期的な利用を考慮した際のアクセス速度の検証は容易だった | 1 | 2 | 3 | 4 | 5 |

———— ツール利用時について ————

① 設計について

- | | | | | | |
|------------------|---|---|---|---|---|
| ・ 配置位置の設計は容易だった | | | | | |
| > ノードの位置決定は容易だった | 1 | 2 | 3 | 4 | 5 |
| ・ ノードの接続実現は容易だった | | | | | |
| > トポロジの設計は容易だった | 1 | 2 | 3 | 4 | 5 |

② 実配置について

- | | | | | | |
|-----------------|---|---|---|---|---|
| ・ 接続性の実現は容易だった | 1 | 2 | 3 | 4 | 5 |
| ・ トポロジの実現は容易だった | 1 | 2 | 3 | 4 | 5 |

③ 検証について

- | | | | | | |
|---------------------------------|---|---|---|---|---|
| ・ データがサンプリングレートで保存される事の検証は容易だった | 1 | 2 | 3 | 4 | 5 |
| ・ 長期的な利用を考慮した際のサーバ容量の検証は容易だった | 1 | 2 | 3 | 4 | 5 |
| ・ 長期的な利用を考慮した際のアクセス速度の検証は容易だった | 1 | 2 | 3 | 4 | 5 |

———— ユーザビリティについて ————

- | | | | | | |
|-------------------------------|---|---|---|---|---|
| ・ 学習しやすさ、初めてでも簡単に使えるか | 1 | 2 | 3 | 4 | 5 |
| ・ 効率的に作業ができたか | 1 | 2 | 3 | 4 | 5 |
| ・ 記憶しやすさ、不定期に利用しても使いやすいかどうか | 1 | 2 | 3 | 4 | 5 |
| ・ 間違えても直したり、やりたいと思った作業が全て出来たか | 1 | 2 | 3 | 4 | 5 |
| ・ ユーザの主観満足度、楽に楽しく作業ができたか | 1 | 2 | 3 | 4 | 5 |

自由記述欄