

筑波大学大学院博士課程

システム情報工学研究科特定課題研究報告書

施設内での経路案内を可能にする
ナビゲーションシステムの開発
－施設内の道情報を管理する機能と
最適な経路を探索するアルゴリズムの開発－

千田 隼人

修士（工学）

（コンピュータサイエンス専攻）

指導教員 田中 二郎

2015年3月

概要

近年、スマートフォン向けの歩行者ナビゲーションシステムがユーザ数を伸ばしてきている。これまでにリリースされた同システムの傾向を見てみると、サービス提供エリアが広い一方で大規模施設の内部には対応しきれていないものと、特定の施設のニーズに合わせたために汎用性が無いものの2種類がある。そこで本プロジェクトでは、施設内情報の検索や経路案内の機能を持ち、導入できる施設を限定しないスマートフォン向けの歩行者ナビゲーションシステムの開発を行った。著者は経路案内機能の基盤となる道情報管理機能と経路探索機能の開発を担当した。経路探索機能の開発に際しては、最適な推奨経路を求めるアルゴリズムを考案し、実装して評価実験を行った。その結果、進路の分かりやすい経路を推奨経路として得られることが確認できた。本システムをリリースすることにより、多くの施設において来場者に対するサービスレベルを従前より大きく向上させられることが期待できる。

目次

第1章	はじめに	1
第2章	開発背景とプロジェクト概要	5
2.1	歩行者ナビゲーションシステムの現状	5
2.2	顧客らが抱える課題と要望	6
2.3	課題の解決策	6
2.4	想定する利用者	6
2.5	学内ガイドを可能にするナビゲーションシステムの開発	7
2.5.1	システム構成	7
2.5.2	歩行者ナビゲーションアプリ	7
2.5.3	データベース	9
2.5.4	管理アプリケーション	11
2.5.5	前システムの課題と解決策	12
2.6	施設内経路探索と施設検索を可能にするナビゲーションシステムの開発	13
2.6.1	機能	13
2.6.2	システム構成	14
2.6.3	開発体制とスケジュール	16
第3章	施設内の道情報を管理する機能の開発	19
3.1	道情報について	19
3.2	設計	20
3.2.1	データ保持部	20
3.2.2	管理用ユーザインタフェース部	21
3.3	実装	22
3.3.1	データ保持部	22
3.3.2	管理用ユーザインタフェース部	25
3.4	動作例	29
第4章	経路を探索する機能	32
4.1	探索できる経路の種類	32
4.2	経路探索アルゴリズムの設計	33
4.2.1	推奨経路探索時のエッジの優先度を決定づける属性の選定	33

4.2.2	主要路か否かの判断基準	34
4.2.3	アルゴリズム	35
4.2.4	割増率 a_1, a_2 の求め方	35
4.3	実装	37
4.3.1	歩行者ナビゲーションアプリとサーバとの間の通信方式	37
4.3.2	経路探索用ライブラリの利用	38
4.4	筑波大学を対象とした経路探索の実行例	39
4.4.1	割増率 a_1, a_2 の算出	39
4.4.2	探索結果の例	40
第5章	評価実験	42
5.1	目的	42
5.2	方法	42
5.3	結果	45
5.3.1	実験参加者の行動に関する観察項目	45
5.3.2	アンケート	45
5.4	議論	48
5.4.1	実験参加者の行動に関する観察項目	48
5.4.2	アンケート	48
第6章	おわりに	50
	謝辞	52
	参考文献	53
付録A	環境構築手順書	55
A.1	サーバ構築手順書	56
A.2	道情報データベース環境構築手順書	66
付録B	管理画面設計資料	70
B.1	画面遷移図	71
B.2	画面定義書	72
付録C	割増率算出過程で得られたデータ	91
C.1	a_1, a_2 を 0.3 刻みで変化させた結果	92
C.2	a_1, a_2 を 0.1 刻みで変化させた結果	97
付録D	評価実験資料	102
D.1	実験計画書	103
D.2	実験参加者行動記録用紙	105

D.3 アンケート	106
付録E マニュアル	107

目次

2.1	前システムの構成図	8
2.2	地図によるナビゲーション	9
2.3	ARによるナビゲーション	9
2.4	大学の構成物の論理構造	10
2.5	階層関係を表すデータベース設計の概念図	10
2.6	本システムの構成図	15
2.7	マスタスケジュール	17
2.8	道情報管理機能と経路探索機能の開発スケジュール	18
3.1	ノードの種類	21
3.2	道情報データベースのER図	23
3.3	道情報の例	24
3.4	道情報管理画面のレイアウト	26
3.5	建物類を指すマーカ	27
3.6	ノード操作ウィンドウ	27
3.7	エッジ操作ウィンドウ	27
3.8	動作制御プログラムの状態遷移図	28
3.9	道情報の初期状態	29
3.10	カーブノード削除後の道情報	30
3.11	カーブノード削除の処理フロー	31
4.1	経路探索の処理フロー	36
4.2	経路探索の通信APIより返却されるJSONの構造	38
4.3	経路探索のSQL	39
4.4	桐葉橋近辺から本部棟までの推奨経路	41
4.5	桐葉橋近辺から本部棟までの最短経路	41
4.6	合宿所バス停から開学記念館までの推奨経路	41
4.7	合宿所バス停から開学記念館までの最短経路	41
5.1	経路1(推奨経路)	43
5.2	経路1(最短経路)	43
5.3	経路2(推奨経路)	43

5.4	経路2 (最短経路)	43
5.5	経路3 (推奨経路)	44
5.6	経路3 (最短経路)	44
5.7	質問項目 (1) への回答の集計結果	46
5.8	質問項目 (2) への回答の集計結果	47

表目次

2.1	動作環境の特徴の比較	8
2.2	検索ログとして取得する情報と取得目的	11
2.3	開発スコープと開発項目	15
2.4	開発メンバと担当スコープ	16
2.5	開発項目の細分化	16
3.1	道情報データベース内で主要度, 舗装有無, 種別を表す値	23
3.2	ノードの種類番号	23
3.3	ノードテーブルのデータの例	24
3.4	エッジテーブルのデータの例	25
3.5	ノードの種類と表示色の対応関係	26
3.6	動作制御プログラムの描画状態一覧	28
3.7	道情報データベース (ノードテーブル) の初期状態	29
3.8	道情報データベース (エッジテーブル) の初期状態	29
3.9	カーブノード削除後の道情報データベース (ノードテーブル)	30
3.10	カーブノード削除後の道情報データベース (エッジテーブル)	30
4.1	エッジの属性候補の評価結果	34
4.2	経路探索の通信 API の引数	38
4.3	経路の種類番号	38
5.1	実験参加者の割当て	44
5.2	歩行所要時間の平均 (分)	45
5.3	歩行距離 (単位: m)	46
5.4	5 秒以上立ち止まった回数の平均	46
5.5	進路を誤った回数の平均	46

第1章 はじめに

本稿は、筑波大学大学院の高度 IT 人材育成のための実践的ソフトウェア開発専修プログラムにおける特定課題研究として実施した「施設内での経路案内を可能にするナビゲーションシステム」の開発について述べたものである。

我が国では 2011 年以降よりスマートフォンの普及が急速に進み、2013 年末における世帯保有率は 62.6%に達している [1]。それに伴ってスマートフォン向けアプリケーションの市場規模も大きく広がってきており、有償無償を問わずスマートフォン向けの有用なアプリケーションが世界中の開発者の手によって数多く開発され、我々の日常生活の様々なシーンで活用されている。歩行者ナビゲーションシステムも、歩行時に手軽に使用できるようにするためスマートフォン向けのアプリケーションとして開発されたものが多数存在しており、多くのユーザに利用されている。

これまでにリリースされた歩行者ナビゲーションシステムは次の 2 種類に大別される。

1. 公道や公共交通機関を利用した移動の際に使用されることを想定したもの
2. ある特定の施設の内部といった限定的な範囲内で使用されることを想定したもの

1 の種類のシステムとしては「NAVITIME」[2]や「Google マップ」[3]が著名である。これらのシステムは徒歩または自家用車による公道上の移動や公共交通機関の利用時における経路案内に対応している反面、大学やレジャー施設などといった大規模施設の内部には対応しきれていない。

2 の種類のシステムとしてはテーマパークの地図や情報を提供する「攻略ナビー東京ディズニーランド」[4]や「ユニバーサル・スタジオ・ジャパン 公式ガイドアプリ」[5]など、各施設の来場者のニーズに適合したものが公開されている。これらのシステムは対象とする施設内の経路案内に加え、施設内の各種情報をきめ細かく提供できるという特徴がある。しかし、施設ごとに専用のシステムが個別に開発されており、様々な施設に導入できる汎用性の高いシステムは存在していない。

このような社会的背景の中、本プロジェクトの顧客である筑波大学システム情報系情報工学域の和田耕一教授並びに山際伸一准教授（以下、顧客ら）は広大な筑波大学の来客者の案内方法に苦慮しており、また自身も用務先の建物が学内のどこにあるか分からない場合があるという問題を抱えている。そこで、顧客らからの要望を受け、次の特徴を持つ新たな歩行者ナビゲーションシステムを開発するプロジェクトを 2013 年度に実施した。

1. 大学内の建物や部屋、催し物などの場所を詳細に提供できる

2. 様々な大学への導入に柔軟に対応できる

このシステムはスマートフォンで使用する歩行者ナビゲーションアプリケーション（以下、歩行者ナビゲーションアプリ）、Web ブラウザ経由で使用する学内情報の管理アプリケーションおよびデータベースにより構成されている。

ところが、このシステムを対象とした評価実験やプロジェクト内における議論の結果、次の1から3に挙げる課題が浮上した。また、顧客らより新たに4の要望を受けた。

1. サービス利用者向けアプリケーションが **Android** にのみ対応している

サービス利用者向けアプリケーションが対応するプラットフォームは **Android** のみであった。ところが、2014年3月末時点での我が国におけるスマートフォン向けプラットフォームの普及率は **Android** が 57.1%、**iOS** が 41.8%と、**iOS** の普及率が **Android** に次ぐ大きさとなっている [6]。従って、**Android** 向けのアプリケーションのみをリリースしても、全体の半数弱のスマートフォンユーザは本システムのサービスを利用できない。

2. 学内情報の検索に使用できるワードは目当ての情報の正式名称のみである

学内情報の検索を行う機能は実装されているものの、目当ての情報を得るには検索キーワードとして入力する文字列がデータベースに登録されている名称と完全一致または部分一致している必要があった。しかし実際には来学し本システムを利用する者（以下、サービス利用者）が目当ての情報の略称や通称しか知らない場合も想定され、このような場合に欲しい情報を迅速に得られないという問題が生じる。

3. 目的地までの経路案内が行われない

サービス利用者に指定された建物や設備（以下、建物類）への案内を行う場合、目的地の場所と方角のみがスマートフォンの画面に表示される仕様となっており、現在地から目的地までの歩行経路は案内されなかった。しかし、目的地の場所や方角のみならず目的地までの詳細な歩行経路を案内しなければ歩行者ナビゲーションシステムとしては不十分であると言える。

4. 大学以外の施設に対応させる

大学以外の施設にも導入できる仕様にしてほしいとの要望が顧客らより新たに出された。しかし、大学の論理構造を参考にモデリングを実施していたため、データベースのスキーマが他の施設に対応できないものとなっていた。データベースは学内情報を階層構造で保持する設計としているが、当初は大学のみでの使用を想定していたため、階層数を変更できないスキーマとなっていた。大学以外の施設の情報を持するには階層数をそれぞれの施設に応じて柔軟に変更可能でなければならない。

そこで以上の課題と要望に対応するための新たなプロジェクトを2014年4月より発足させ、次の方策で課題の解決に当たることとした。

1. iOS 向け歩行者ナビゲーションアプリの開発とクラス設計の統一

Android 向け歩行者ナビゲーションアプリに加えて、iOS 向けのを新規開発する。Android と iOS ではクラス構造の設計ルールや API(Application Programming Interface) に差異があるため、Android 向けアプリケーションと iOS 向けアプリケーションの双方でクラス構造を可能な限り統一する。これによって歩行者ナビゲーションアプリのバージョンアップの作業が容易になり、高い保守性が得られることが期待できる。

2. 施設内情報へのタグ付けによる関連キーワードを用いた検索への対応

データベースに登録されている情報に略称や通称、関連する単語等をタグとして付与し、それを正式名称に加えて検索対象とすることにより、関連キーワードを用いた検索に対応する。

3. 施設内の道情報を管理する機能、経路案内機能および道情報自動登録ツールの開発

施設内の道情報を登録・管理する機能および道情報を用いてサーバ側で経路探索を行う機能を開発する。また、歩行者ナビゲーションアプリ側では地図と AR(Augmented Reality) を用いてサービス利用者に経路案内を提供する機能を開発する。さらに、手間のかかる道情報の登録作業を簡易化するための「道情報自動登録ツール」を開発する。

4. 施設内情報の階層数を柔軟に変更可能とするためのデータベーススキーマの再設計

本システムを導入する施設の構造に応じて施設内情報の階層数を柔軟に変更できるようにデータベースのスキーマを再設計する。また、それと同時に本システムの運用を行う施設の担当者（以下、施設管理者）向けの施設内情報管理機能を再設計後のスキーマに対応させる。

本システムをリリースすることにより、施設管理者は、施設の来場者に対するサービスレベルを従前より大きく向上させられることが期待できる。また、サービス利用者は、本システムの歩行者ナビゲーションアプリを自身のスマートフォンにインストールするのみで様々な施設において施設内情報の提供や施設内の経路案内といったサービスを単一のアプリケーションにより手軽に利用できるというメリットがある。

著者は2014年4月より本プロジェクトに参画し、先述の課題解決方策の3に係る役割として、次の内容の開発を担当した。

1. 施設内の道情報データベースの構築および管理アプリケーションの開発

経路案内に必要な施設内の道情報を保持するデータベースを構築し、そのデータを Web ブラウザ経由で編集するための管理アプリケーションを開発する。このデータベースに保持されているデータを基に経路案内が行われる。

2. 経路を探索する API の開発

本システムの経路案内機能では、歩行者ナビゲーションアプリからの問合せに応じてサーバ側で経路の探索が行われる。その際に歩行者ナビゲーションアプリ側から呼び出す API および API 内部における経路探索プログラムを開発する。

以上の開発を行うことにより，本システムのサーバにおいて施設内の道情報の保持とそれを利用した経路探索が可能となる．これは歩行者ナビゲーションシステムがサービス利用者に経路案内を提供する際に必要となるものである．

本稿は本章を含めて全7章と付録から成る．以降，第2章では本システムの開発背景と本プロジェクトの概要を述べる．第3章以降は著者が開発を担当する内容についての章である．第3章では施設内の道情報データベースの構築および管理アプリケーションの開発について述べる．第4章では経路を探索するAPIの開発と，APIの内部に実装している最適な経路（以下，推奨経路）を探索するアルゴリズムについて述べる．第5章では経路探索に関する評価実験について述べる．第6章ではまとめと本システムの今後の発展について述べる．また，付録としてサーバの構築に係る手順書の一式，設計資料，推奨経路探索時にエッジの仮想距離の計算に使用するパラメータを決定する実験の結果，著者が実施した評価実験の資料，本システムのマニュアルを添付する．

第2章 開発背景とプロジェクト概要

2.1 歩行者ナビゲーションシステムの現状

2011年以降におけるスマートフォンの普及に伴い、スマートフォンで利用できる様々な歩行者ナビゲーションシステムが開発されてきた。これらは土地勘の無い街を歩行する場合や大型施設を効率良く利用する場合に有用なツールとして多くのユーザに活用されている。現在普及している歩行者ナビゲーションシステムは、次の2種類に分けられる。

1. 公道や公共交通機関を利用した移動の際に使用されることを想定したもの。
2. ある特定の施設の内部といった限定的な範囲内で使用されることを想定したもの。

1の種類のシステムとしては、「NAVITIME」[2]や「Google マップ」[3]が特に著名である。これらのシステムの強みは、サービス提供地域の広大さと情報量の多さであると言える。NAVITIMEは日本国内でのサービス提供を主軸とし、海外にもサービス提供範囲を広げてきている。また、Google マップは一部の国を除く世界中の地域の情報を得ることができる。提供されている情報も多彩であり、地図をもとにした経路案内をはじめ、公共交通機関の乗換案内、道路交通情報や各地の天気情報までもが1つのシステムを用いることで手に入れられる。その反面、大学やレジャー施設などといった大型施設の内部の細かい道には対応できていない場合が多い。筑波大学を例に挙げると、学内の歩行者および自転車専用の通路の情報は、1の種類のシステムには収録されていない。また、建物の情報にも一部しか対応できていない。そのため、学内の建物間を歩いて移動する場合にはこれらのシステムを利用できず、キャンパスマップのみを頼りに歩行しなければならないという問題がある。

2の種類のシステムとしては、テーマパークや大型レジャー施設の地図や情報を提供する「攻略ナビー東京ディズニーランド」[4]や「ユニバーサル・スタジオ・ジャパン 公式ガイドアプリ」[5]などが存在する。これらのシステムはそれぞれの対応する施設の来場者のニーズに適合した、詳細な施設内情報の提供ができることが強みである。施設内の地図をもとにしたアトラクションや化粧室などの位置の案内、待ち時間などの情報を容易に手に入れられる。しかし、特定の施設に特化せず、様々な施設に導入し得る汎用性の高いシステムは未だ存在していないため、現状では施設の担当者または個人が個別に開発、運用を行っている。

2.2 顧客らが抱える課題と要望

本プロジェクトの顧客である和田耕一教授ならびに山際伸一准教授は、筑波大学に勤務する教員である。筑波大学は国立大学の中でも特に広大な敷地面積を有する大学の1つであり、学内の道の構成も複雑なため、初めて本学を訪れる者は学内で道に迷いやすい傾向がある。しかし、歩行者ナビゲーションシステムとして広く普及しているNAVITIMEやGoogleマップなどは学内の歩行者および自転車専用通路や建物の情報を収録しておらず、学内の移動の際に活用することはできない。従って、学内で目的地の場所とそこに至る経路を知るには大学が公開しているキャンパスマップを参照する必要がある、欲しい情報を迅速に情報を得にくいという問題がある。

このため、顧客らは本学の来学者に対する学内の案内方法に苦慮しており、自身も学内の用務先の建物の場所が分からない場合があるという問題を抱えている。顧客らが担当している授業を受講する学生も、初回の授業日には講義室の位置が分からずに遅刻してしまう場合がある。また、筑波大学に限らず他の大学でも同様の課題を抱えていると考えられる。

そこで顧客らは本プロジェクトに対し、学内の情報を迅速に検索でき、目的地への案内を提供する機能を持つ歩行者ナビゲーションシステムを開発することを要求している。また、本システムは筑波大学のみならず他の大学にも導入可能であるよう設計することを求めている。

2.3 課題の解決策

先述の課題および顧客らからの要望を受け、本プロジェクトでは次の方針で課題の解決を図ることとした。

1. 学内情報を管理する機能

サービス利用者による検索の対象となる学内情報を施設管理者が管理するための機能を開発する。

2. 学内情報を検索する機能

本システムに登録されている学内情報を歩行者ナビゲーションアプリで検索し情報を得る機能を開発する。

3. 目的地へ案内する機能

目的地の位置と現在地からの方角を示すことによりサービス利用者を目的地へ案内する機能を開発する。

2.4 想定する利用者

本システムの利用者は、学内の情報を管理する「施設管理者」および歩行者ナビゲーションアプリを利用する「サービス利用者」である。

施設管理者は学内の授業や教室，厚生施設などを管理する役割を担う人物であり，本システムの学内情報データベースへのデータの登録や更新作業を行う立場の者である．大学の施設部などの部門の担当者を想定している．施設管理者は本システムを自身の施設に導入することにより，施設の来場者にきめ細かい案内サービスを提供できるようになる．

サービス利用者は歩行者ナビゲーションアプリを利用して情報の検索や目的地までの案内などのサービスを受ける者である．特に本システムのメインターゲットとして想定している者は，大学の新生や来客などといった土地勘の無い者である．これらの者が本システムを利用することにより，施設外の一般公道などで使用可能な歩行者ナビゲーションアプリに類する案内サービスを学内でも無料で受けられる．

2.5 学内ガイドを可能にするナビゲーションシステムの開発

2.2 で述べた顧客らの要求を受け，平成 25 年度の PBL 型システム開発 A・B にて，大学で使用されることを想定した歩行者ナビゲーションシステム（以下，前システム）が開発された．本節では前システムの概要について述べ，前システムの設計や仕様に関する問題点とその解決策について議論する．

2.5.1 システム構成

前システムの構成図を図 2.1 に示す．

データの管理や検索などの処理はサーバ内で行い，ユーザはクライアントである歩行者ナビゲーションアプリや Web アプリケーションを介して本システムを利用する．サービス利用者がスマートフォンにインストールされた歩行者ナビゲーションアプリを用いて学内情報の検索操作を行うと（図 2.1(1)），歩行者ナビゲーションアプリが検索 API に対して学内情報の検索を要求し（図 2.1(2)），検索 API がデータベースに問合せを行う（図 2.1(3)）．その後，検索要求の伝達ルートを遡ってサービス利用者に検索結果が提供される（図 2.1(4)-(6)）．

2.5.2 歩行者ナビゲーションアプリ

サービス利用者へのサービス提供にはスマートフォンを使用する．スマートフォンは 2011 年以降急速に普及してきていることに加え，携帯性に優れており，キャリアのサービス提供地域内なら場所を選ばず手軽にインターネットに接続できるという長所がある．また，地磁気や GPS(Global Positioning System) のセンサが標準で搭載されており，これらのセンサより得られたデータをアプリケーションが自由に応用できる．したがって本システムのサービス提供用の機器としてはスマートフォンが適切であると判断した．

歩行者ナビゲーションアプリの対応プラットフォームには Android を選定した．スマートフォンのプラットフォームの中でも特にユーザが多いものは Android と iOS であり，また PhoneGap と呼ばれるクロスプラットフォームのフレームワークも存在する．これらの動作環境の特徴を

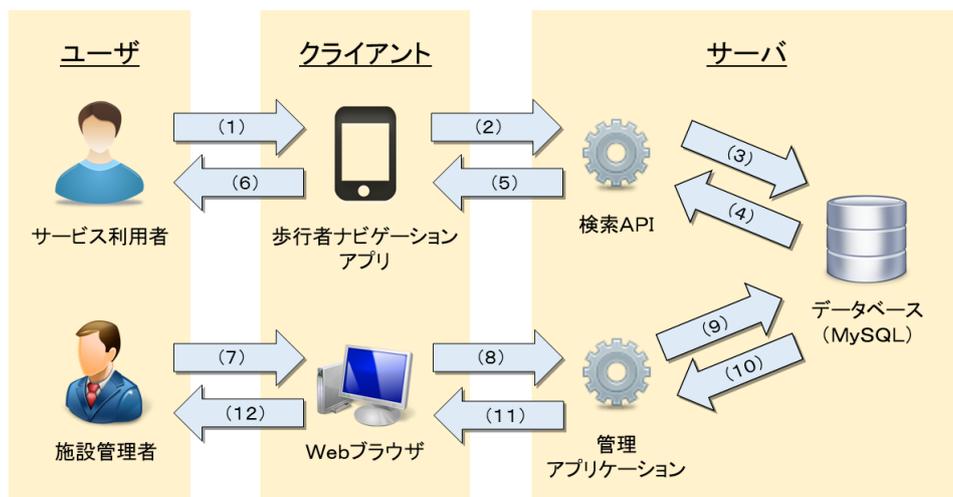


図 2.1: 前システムの構成図

比較したものを表 2.1 に示す。PhoneGap の動作速度は他の二者と比較して非常に低速であり、実用的ではないため採用しないこととした。Android は iOS と異なり開発環境が無償で手に入れることが大きな利点として挙げられる。本プロジェクトは 4 名の体制で実施したため、4 名分の iOS 向けの開発環境を購入するとコストが大きくなってしまふ。また、Android は開発担当者が既に技術知識を保有しており、すぐに開発に着手できるというメリットがある。一方、Android には機種毎に使用できる機能の制約が存在するが、この問題は歩行者ナビゲーションアプリの対応機種をあらかじめ限定することによって解決可能である。以上より、Android を歩行者ナビゲーションシステムの対応プラットフォームとすることが妥当であると判断した。

表 2.1: 動作環境の特徴の比較

	開発環境の価格	開発者学習コスト	プラットフォーム非依存性	機種毎の個別制約の有無	動作速度
Android	○ (無償)	○ (小)	× (無)	× (有)	○ (高速)
iOS	× (有償)	× (大)	× (無)	○ (無)	○ (高速)
PhoneGap	○ (無償)	× (大)	○ (有)	× (有)	× (低速)

歩行者ナビゲーションアプリに実装した機能は次の通りである。

1. 施設切り替え機能

本システムは複数の大学での使用を想定しているため、現在自身がいる大学を本機能によって切り替える。

2. 検索機能

検索ワードを用いて大学内の情報を検索すると、カテゴリ（建物，教室，授業）毎に結果が表示される。また，検索結果には現在地からの距離と方向が併記される。

3. ナビゲーション機能

現在地から目的地まで，地図およびARによりサービス利用者を案内する。地図による案内の様子を図 2.2 に，AR による案内の様子を図 2.3 に示す。地図画面では，目的地を示すマーカと，現在地から目的地までを結ぶ直線が表示される。端末を立てると地図画面から AR 画面に切り替わり，目的地を示すマーカと現在地からの方向が表示される。

4. 履歴機能とブックマーク機能

過去に調べたことのある情報を再度検索したい場合に対応するため，検索履歴やブックマークにより欲しい情報を迅速に得られるようにする。

これらの機能により，大学内の情報と目的地までの案内の提供を実現している。



図 2.2: 地図によるナビゲーション

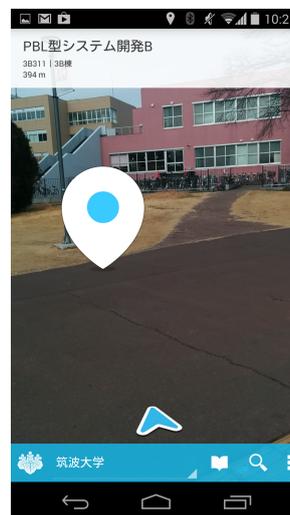


図 2.3: AR によるナビゲーション

2.5.3 データベース

検索の対象となる学内情報をデータベースに保持するに当たり，学内情報の構造化を行う必要がある。そこで，複数の大学のキャンパスマップを調査し，大学の構成物の論理構造を検討した。その結果，大学は概ね図 2.4 のような階層構造で表現できることが分かった。この階層構造の中でサービス利用者が検索する可能性のある要素は，建物以下の層であり，エリアやキャンパスは範囲が非常に大きいため検索の対象とする必要は無いと考えられる。したがって，データベースに収録するデータは建物以下の層とすることとした。この階層関係を表すためのデータベース設計の概念図を図 2.5 に示す。

目的地までの案内を行うには建物、部屋、授業が開講されている場所の位置情報が必要となる。そこで、位置情報を持つ親レコードを参照する子レコードは親レコードの位置情報を継承できる設計とした。なお、子レコードに別な位置情報を記録すれば、経路案内では子レコードの位置情報を基に案内が行われる。

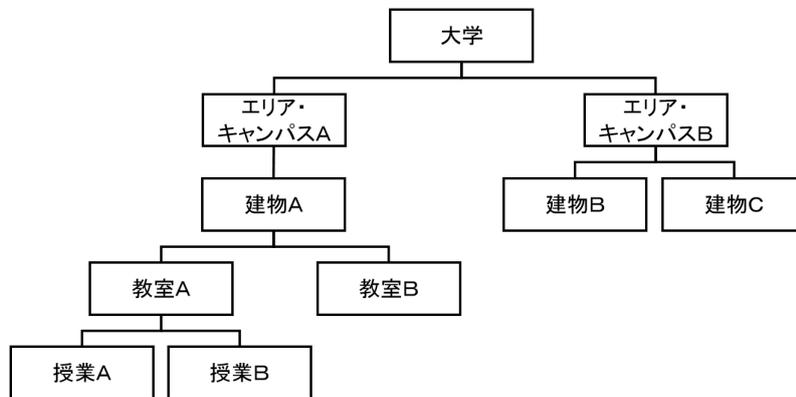


図 2.4: 大学の構成物の論理構造



図 2.5: 階層関係を表すデータベース設計の概念図

また、本システムは1つのアプリケーションをスマートフォンにインストールすることによって複数の大学で使用できる仕様とする要件がある。したがって学内情報に加えて次の情報をデータベースに保持する。

1. 大学情報

本システムの対応大学の一覧を保持するテーブルを作成する。また、学内情報の各レコードを本テーブルのレコードと対応付ける。

2. アカウント情報

本システムに登録されているデータの管理は大学の施設管理者が行う。そのため、施設管理者に個別にアカウントを付与する。また、1つの大学に異なる職掌を持つ複数の施設管理者が存在する場合も考えられるため、学内情報へのアクセス制限を学内情報の階層毎に設定できるようにする。

以上の情報をデータベースに保持することにより、複数の大学の情報を1つのデータベースに混在させることができるため、単一のサーバにより複数の大学に同時にサービスを提供することが可能になる。

さらに、本システムの今後の改善の参考にするためのデータとして、検索ログを蓄積する。取得する情報と取得目的は表 2.2 の通りであり、このデータを用いることによって客観的なデータを参考とした改善策の立案が可能となる。

表 2.2: 検索ログとして取得する情報と取得目的

取得情報	取得目的
検索ワード	サービス利用者が検索しているワードの傾向を把握する。
IP アドレス	サービス利用者の個々人と利用ログのレコードを対応付ける。
検索日時	本システムが利用されている時間帯を把握する。
大学	大学毎の本システムの利用頻度を把握する。
OS	今後これらの情報を用いた検討を行う場合に備えて取得する。
デバイス名	
カメラの有無	
解像度	

2.5.4 管理アプリケーション

データベースが保持しているデータの管理アプリケーションについて述べる。管理アプリケーションは Web アプリケーションとして開発し、次の機能を実装した。

1. アカウント管理

施設管理者のアカウント情報であるログイン名、パスワード、管理権を持つ大学と学内情報の管理範囲の編集を行う。

2. 大学管理

本システムのサービス提供対象としている大学一覧の編集を行う。

3. 学内情報管理

学内の建物、教室、授業の情報を編集する。各レコードには位置情報を付加でき、位置情報を持つレコードおよびその子レコードの位置を歩行者ナビゲーションアプリにより案内できる。また、学内情報が記録された電子ファイルのインポートも可能であり、電子ファイルがあらかじめ用意されている場合は学内情報の登録作業が簡単に行える。

4. 検索ログ閲覧

2.5.3 で述べた検索ログの一覧を閲覧する。本システムの課題点を検討する際に用いる。

施設管理者はこれらの機能を Web ブラウザにより利用でき、学内情報の登録や変更の必要が生じた場合は速やかに対応可能となる。

2.5.5 前システムの課題と解決策

前システムを対象として実施した評価実験の結果明らかとなった課題とその解決策について述べる。

1. 歩行者ナビゲーションアプリの対応プラットフォームが **Android** のみであること

前システムの歩行者ナビゲーションアプリの対応プラットフォームは **Android** のみである。一方、スマートフォン向けのプラットフォームの普及率は **iOS** が **Android** に次いで高く、多数の **iOS** ユーザが本サービスを利用できない状況となっている。そこで本サービスのユーザ数を拡大させるため **iOS** 向けのアプリケーションを開発する。

2. 学内情報を保持するデータベーススキーマに柔軟性が無いこと

前システムでは学内情報の構造は建物の中に教室が包含され、さらにその中に授業が包含されるという3階層であることを前提としてデータベーススキーマを定義した。したがって、これとは異なる構造でデータを登録できず、柔軟性に欠けるという問題がある。そこで学内情報の階層数を大学によって柔軟に変更できるようデータベーススキーマを変更する。また、これにより大学以外の施設にも本システムを導入しやすくなるというメリットも生まれる。

3. 学内情報の被検索語が正式名称のみであること

前システムのデータベースには学内情報の正式名称が登録されており、通称や略称は登録されていない。そのため学内情報を検索する際にサービス利用者が指定する検索ワードがデータベースに登録されているデータの正式名称と完全一致または部分一致していなければ目当ての情報を検索結果として得られないという問題がある。ところが、前システムの評価実験を行い検索ログの内容を確認したところ、正式名称ではなく通称や略称による検索数が多いことが判明した。また、サービス利用者が目当ての学内情報の正式名称を必ず知っているとは限らず、関連するワードを用いた検索にも対応することが望まれる。

4. 目的地までの経路案内機能が無いこと

前システムは目的地への案内機能として、目的地の位置と方角を地図および **AR** により示す仕様であった。しかし、評価実験を行った結果、目的地までの詳細な経路を示すことを求める意見が多く得られた。一般に大学は敷地面積が比較的広く、学内で道に迷いやすい傾向があり、詳細な経路案内機能を実装することによって歩行者ナビゲーションシステムとしての利便性を高められる。

以上の問題を解決することにより、本システムの導入に当たり問題となるデータベースの柔軟性の改善と利便性の向上につながる。

2.6 施設内経路探索と施設検索を可能にするナビゲーションシステムの開発

前システムの評価実験の結果や顧客らとの議論を経て、顧客らから新たに本システムを大学以外の大型施設に導入できるよう改善することと、判明した課題を解決することの要望が出された。そこで、2014年度の研究開発プロジェクトにてこれらの要望に対応するプロジェクトを実施した。なお、著者は本プロジェクトの発足時より開発に参画した。

2.6.1 機能

本プロジェクトで新たに開発する機能は、施設内情報の管理、施設内情報の検索、経路案内、道情報の自動登録である。以降、これらの機能の概要を述べる。

(1) 施設内情報の管理

本機能は前システムにおける学内情報管理機能を大学以外の施設の内部の情報の管理に対応できるよう改善したものである。本機能による管理対象となる情報は次の通りである。

1. 施設内情報

施設内情報は前システムにおける建物、教室、授業の情報に当たるもので、大学以外の施設におけるこれらの情報に類する情報を施設内情報と定義する。また、施設内情報は検索機能による検索の対象となる情報である。本機能では施設内情報を階層構造で管理し、管理画面ではツリービューを用いて階層構造を明確に表現する。

2. カテゴリ情報

前システムでは学内情報の各階層の階層名がそれぞれ建物、教室、授業という固有のものに定められており、変更ができない仕様となっていた。ところが今回本システムを大学以外の施設に導入することや階層数を柔軟に変更できる仕様に変更するため、任意の階層名を設定できる必要がある。そこで各階層の名称をカテゴリ情報として登録できる仕様に変更し、施設内情報を任意のポリシーに基づいて分類可能にする。

3. 道情報

経路案内を実現するため、本システムに新たに施設内の道情報を保持させる必要がある。そこで、道情報を保持するデータベースと道情報の管理アプリケーションを開発する。道情報のデータベースと管理アプリケーションの詳細は第3章にて述べる。

施設内情報として以上の情報を本システムで管理することにより、顧客らより与えられた要望を実現可能となる。

(2) 施設内情報の検索

本機能は前システムにおける学内情報の検索機能を施設内情報の検索に適応させたものである。施設内情報に関連する単語をタグ情報としてそれぞれの施設内情報に付与することに

よって、前システムで問題となっていた検索ワードの制限を解消し、目当ての情報の通称や略称によっても検索結果を得られるように改善する。これにより検索機能の利便性が大きく向上することが期待できる。

(3) 経路案内

データベースに登録された道情報に基づいて現在地から目的地までの詳細な経路を案内する機能である。目的地を検索すると、目的地までの歩行経路が地図上に表示される。また、AR表示に切り替えるとカメラ画像の上に歩行経路がARにより表示される。本機能で案内する経路は推奨経路、最短経路、推奨経路（階段無し）、最短経路（階段無し）の4種類である。これらの経路の詳細および探索方法については第4章にて述べる。

(4) 道情報の自動登録

道情報の登録の際には、敷地内を実際に歩きながら通路構成を調査し、管理アプリケーションを用いて手作業で登録作業を行う必要がある。したがって施設の敷地面積が広いほど道情報の登録には時間がかかり、誤りが発生する確率も上がると考えられる。そこで道情報の登録作業の手間を緩和するため、道情報を自動的に生成しデータベースに登録する機能を開発する。施設内の通路構成の調査の際にGPSログを収集するツール（以下、GPSロガー）を携帯し、歩行中に得られたGPSログをもとに道情報を生成する。

2.6.2 システム構成

本システムの構成図を図2.6に示す。

クライアントには歩行者ナビゲーションアプリと道情報の自動登録のためのGPSロガーがある。また、管理アプリケーションはWebブラウザを経由して利用する。歩行者ナビゲーションアプリはAndroid向けとiOS向けの2種類が存在する。

サーバは管理アプリケーション、歩行者ナビゲーションとの通信API、道情報自動登録機能、データベースにより構成される。DBMS(DataBase Management System)はMySQLを使用することを原則とするが、道情報データベースのDBMSはpgRoutingと呼ばれるPostgreSQLのみで使用可能な経路探索エンジンを使用するためPostgreSQLとする。

本プロジェクトにおける開発スコープと開発項目の一覧を表2.3に示す。表中のスコープIDは図2.6の構成要素の末尾に記載されているIDと対応している。

S1では施設内情報管理機能を実現するため、施設内情報データベースおよびその管理アプリケーションを開発する。S2での開発物は経路案内機能を実現するためのものである。道情報データベースとその管理アプリケーション、および道情報を用いて経路を探索するための通信APIを開発する。S3ではAndroidおよびiOS向けの歩行者ナビゲーションアプリを開発する。S4では施設内情報検索機能を実現するため、新たにタグ・類語辞典データベースとその管理アプリケーションを開発する。また、歩行者ナビゲーションアプリからサーバへ検索要求を行うための通信APIを開発する。S5では道情報の自動登録を可能とするため、スマートフォンにインストールするGPSロガーと、その情報を基に道情報を生成し道情報データベースに登録するための道情報自動登録機能を開発する。

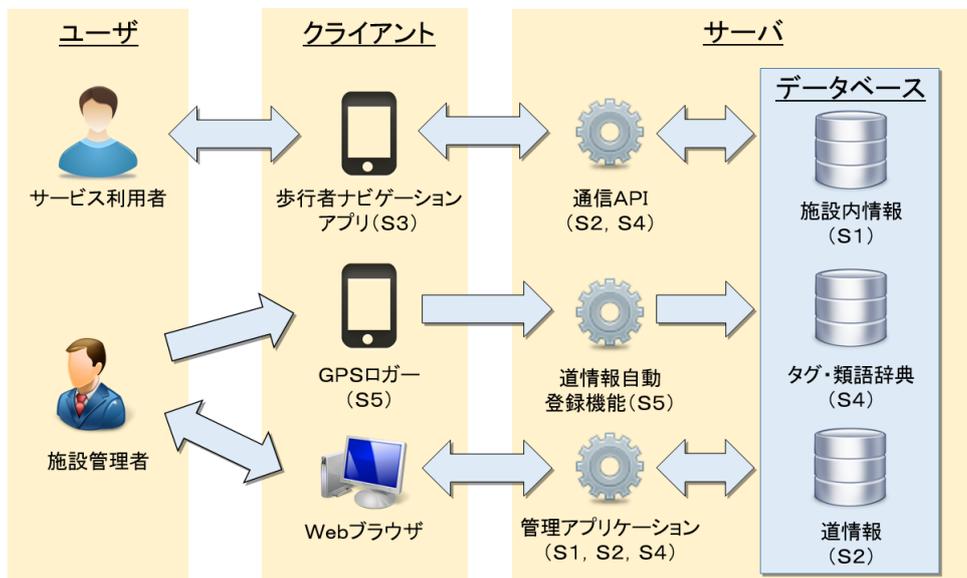


図 2.6: 本システムの構成図

表 2.3: 開発スコープと開発項目

スコープ ID	開発スコープ	開発項目
S1	施設内情報管理機能	施設内情報データベース
		管理アプリケーション（アカウント，アクセス制限，施設，カテゴリ，施設内情報）
S2	道情報管理機能 経路探索機能	道情報データベース
		管理アプリケーション（道情報）
		通信 API（経路探索）
S3	歩行者ナビゲーションアプリ	歩行者ナビゲーションアプリ（Android，iOS）
S4	施設内情報検索機能	タグ・類語辞典データベース
		管理アプリケーション（ログ閲覧）
		通信 API（検索 API，目的地設定 API）
S5	道情報自動登録ツール	GPS ロガー
		道情報自動登録機能

2.6.3 開発体制とスケジュール

本プロジェクトの開発メンバと担当スコープを表 2.4 に示す。本プロジェクトは、顧客兼課題担当教員である和田耕一教授ならびに山際伸一准教授の指導のもと、畑中裕太をリーダーとする 4 名のチームで実施する。

表 2.4: 開発メンバと担当スコープ

開発メンバ	担当スコープ ID
畑中裕太 (リーダー)	S1, S4
千田隼人	S2
芳賀隼人	S1, S5
萬成亮太	S3

本プロジェクトのマススケジュールを図 2.7 に示す。スコープ毎に要件定義、開発、テストの順に作業を進めていくことを基本とし、データベースおよびその管理アプリケーションを新規に用意したものについてはテストの際にデータ登録も行う。開発作業を 11 月までに終了し、11 月から 12 月にかけてシステムの評価実験を行う。なお、本プロジェクトのタスク管理には Redmine[7] を用いた。

著者の担当である道情報管理機能と経路探索機能の開発項目を細分化したものを表 2.5 に示す。また、道情報管理機能と経路探索機能の開発スケジュールを図 2.8 に示す。最初に道情報管理機能を開発し、その後経路探索機能を開発するという順序を進める。道情報管理機能は、道情報データベースの構築、ノード登録・編集機能、エッジ登録・編集機能の順に開発する。開発とテストの終了後、評価実験で用いる筑波大学の道情報を登録する。

表 2.5: 開発項目の細分化

分類	開発項目 ID	開発内容
道情報管理機能	F1	道情報データベース
	F2	ノード登録・編集機能
	F3	エッジ登録・編集機能
経路探索機能	F4	通信 API

年	2014											2015
月	3	4	5	6	7	8	9	10	11	12	1	
行事					▲				▲	▲		
					中間発表会(7/2)			中間発表会(11/7)			国際学会(12/10)	
S1	要件定義	■										
	開発		■									
	テスト・データ登録				■							
S2	要件定義	■										
	開発		■					■				
	テスト・データ登録							■				
S3	要件定義					■						
	開発						■					
	テスト							■				
S4	要件定義					■						
	開発						■					
	テスト							■				
S5	要件定義					■						
	開発						■					
	テスト・データ登録							■				
評価実験									■			
特定課題研究 報告書執筆										■		

夏休み

図 2.7: マスタスケジュール

年		2014									
月		3	4	5	6	7	8	9	10	11	
要件定義		■					夏 休 み				
開発	F1		■								
	F2			■							
	F3				■						
	F4							■			
テスト									■		
データ登録										■	

図 2.8: 道情報管理機能と経路探索機能の開発スケジュール

第3章 施設内の道情報を管理する機能の開発

本章では経路案内に用いるための施設内の道情報を管理する機能の開発について述べる。本機能は「データ保持部」および「管理用ユーザインタフェース部」により構成されている。データ保持部は道情報を保持しておくためのデータベースである。管理用ユーザインタフェースは、施設管理者が道情報データベースのデータを編集するために使用する Web アプリケーションである。

3.1 道情報について

膨大な地図情報を保持し高速に経路探索を行うシステムとして最も身近なもの1つにカーナビゲーションシステムがある。カーナビゲーションシステムでは、経路案内に使用するための情報として地図画像と道路ネットワーク情報を保持している。

地図画像はカーナビゲーションシステムのユーザに対して、周囲の道路の形状を視覚的に分かりやすく表示することを目的としたものである。ところが、人間が地図画像を見ると道路同士の接続状態の他に道幅や建物の形状なども一目で読み取れるものの、地図画像はあくまでも単なる画像情報に過ぎないため、コンピュータでの直接の計算処理には向いていない。

一方、道路ネットワーク情報は一般にグラフを用いて表現される [8]。グラフはノードとエッジのみにより構成される非常に単純な構造を持つデータであり、小さいサイズのデータで道路ネットワークの形状を正確かつ容易に表現できる。また、グラフ理論と呼ばれる学問分野においてグラフの解析を行うための種々の計算アルゴリズムが確立されており、その中には経路探索のアルゴリズムも存在している [9]。経路探索のアルゴリズムは目的地までの案内経路を決定する際にも応用できるため、グラフはシステム内部における道路ネットワーク情報の表現方法として適していると言える。

よって、本システムでは施設内の道路ネットワーク情報をグラフデータとして保持し、経路案内に利用する。なお、本稿では道路ネットワーク情報のことを道情報と呼称することとし、道情報を保持するデータベースの名称を道情報データベースと定義する。

管理アプリケーションを用いた道情報データベースの編集や歩行者ナビゲーションアプリにおいてサービス利用者に対して経路案内を行う際に使用する地図画像は、Google マップから提供されているものを Google Maps API [10] により取得して用いる。Google マップは多くのユーザによって日常生活における様々な場面で利用されているサービスであり、地図の表示を伴う多くの既存の Web サービスでも Google マップの地図を Web ページ内に組み込むことによって利用されている。したがって、地図画像に Google マップの地図を利用することに

よって、多くのユーザに違和感を与えないユーザインタフェースを実現できる。また、地図画像の表示のみならず地図画像上へのマーカーや線の描画も容易に行えるため、本システムでの利用に最適であると判断し、採用することとした。

3.2 設計

3.2.1 データ保持部

道情報のグラフの構成要素であるノードは、施設出入口、建物出入口、交差点、カーブの4種類に分けられる(図3.1)。経路案内を行うには、道路の形状および経路案内の目的地に指定できる場所の情報が必要となる。道路の形状を表すために用いるノードが交差点およびカーブである。また、目的地に指定できる場所を示すノードが建物出入口である。なお、本プロジェクトでは本システムが施設の内部のみで利用されるという前提のもとで設計を行ったが、今後は施設の外部から内部までの一貫した経路案内を行う仕様に発展させられる可能性がある。その際は、現在地から施設出入口までの経路を Google Maps API により求め、施設出入口から施設内部の目的地までの経路を本システムの経路探索機能を用いて計算する方式をとることとなる。このため、施設の内外の境界である施設出入口の位置をあらかじめ道情報データベースに登録しておくことにより、今後の本システムの発展に容易に対応できる。よって、施設出入口という種類のノードを設ける。

建物出入口ノードは経路案内の目的地として指定できる建物類の出入口を示すノードである。建物類の中には入口の場所が分かりにくいものが存在するため、目的地の建物類の付近までの経路案内のみに留めず、建物類の出入口の位置までの経路案内を提供することが望ましい。そこで、建物出入口ノードを建物類の全ての出入口の位置に配置することにより、目的地とする建物類の出入口までサービス利用者を案内することを可能にする。

交差点ノードは複数の道が交差する地点に配置する。また、カーブノードは曲線状の道の上に適宜配置してエッジで接続することにより、グラフの形状を実際の道路形状に合わせることに使用する。本システムの歩行者ナビゲーションアプリで経路探索を実行すると、その結果の経路が地図上に表示されるとともにARによる進行方向の案内が行われる。これらの機能を実現するため、カーブノードを用いてグラフの形を実際の道路形状に対応させる。

経路探索を行う際にはそれぞれのエッジが表す道の特徴が用いられる。そのため、エッジが表す道の特徴をエッジの属性として付加する。エッジには経路探索時に使用する属性として、距離、主要度(主要路か否か)、舗装有無、種別(平坦路または階段)を付加する。距離は経路探索のベースとなる値であり、要求された経路種別に応じてこれに進路の分かりやすさを示す主要度と、歩きやすさを示す舗装有無および種別を加味して経路を求める。エッジの属性の詳細については4.2.1にて述べる。以上のノードとエッジにより形作られたグラフが道情報として道情報データベースに保持される。



図 3.1: ノードの種類

3.2.2 管理用ユーザインタフェース部

本節では道情報の管理アプリケーションの画面設計について述べる。本画面は施設管理者が施設内の道情報を表すグラフを編集し、管理するための画面である。マウス操作のみで容易にグラフの編集を行うため、次の操作方法により編集できるように設計する。

1. ノードの配置とエッジによるノード同士の接続

あらかじめノードの種類を選択し、地図上の任意の箇所を左クリックすると、選択されている種類のノードがクリックされた箇所に配置される。建物出入口ノードが選択されている場合は、この操作を行うと地図上の全ての建物類にマーカーが表示される。その後、建物類を指すマーカーのいずれかを左クリックすると、配置した建物出入口ノードがその建物類の出入口として道情報データベースに登録される。ノード同士をエッジで接続する場合は、接続したい2つのノードを順番に左クリックする。

2. ノードの移動

ノードの位置は通常時には固定し、ノードに対して直接ドラッグ操作などを行ってもノードを移動することはできないようにする。これは地図画像をスライドする際に誤ってノードの位置を動かしてしまうことを防ぐためである。ノードの移動をする際は、移動対象のノードを右クリックして表示される操作ウィンドウ（以下、ノード操作ウィンドウ）から位置の固定を解除することにより1回のみドラッグ操作による移動を許可する。ドラッグ操作を終えると、移動先の位置にノードを再び固定する。

3. ノードおよびエッジの削除

ノードの削除はノード操作ウィンドウより行う。この操作により、当該ノードおよびそれに接続するエッジがまとめて削除される。ただし、エッジが2本接続しているカーブノードを削除すると、当該ノードに隣接するノード同士が短絡される。このような仕様にする事で、曲線状のグラフに対する編集操作を行っているときに誤ってグラフの接続関係を切断してしまうことを防ぐ。エッジの削除はエッジを右クリックして表示される操作ウィンドウ（以下、エッジ操作ウィンドウ）より行う。

4. エッジ上へのカーブノードの追加

グラフの描画を行っている途中でグラフの形状の微調整を行う必要が生じる場合がある。その際に既存のノードを移動させるのみでは思い通りの形状に調整できない場合のため、エッジの途中にカーブノードを追加できる仕様にする事によって、グラフを一通り描画した後でもグラフの形状を柔軟に変更可能にする。エッジの途中へのカーブノードの追加はエッジ操作ウィンドウより行う。

5. エッジの属性の設定

エッジの属性の設定はエッジ操作ウィンドウより行う。ここで属性を設定すると、同時に道情報データベースも更新される。

なお、画面のデザインについては 3.3.2 で述べる。

3.3 実装

3.3.1 データ保持部

本システムでは PostgreSQL のみで使用可能な pgRouting[11] と呼ばれる経路探索エンジンを使用する。そのため、道情報データベースの DBMS には PostgreSQL を使用する。pgRouting は SQL を通してデータベースの道情報に対する経路探索が行えるという特徴がある。また、エッジの重みを SQL による経路の問合せと同時に動的に計算できる。従って、エッジの重みの計算回数が必要最小限に抑えられるため、pgRouting を用いずに経路探索を行う場合より経路探索が効率良く行えるというメリットがある。

道情報データベースの ER 図を図 3.2 に示す。道情報データベースはノードデータを持つテーブルとエッジデータを持つテーブルの2つにより構成される。ノードデータとエッジデータにはそれぞれ識別用の ID が付与される。また、施設 ID を各レコードに付加する。これは1つのテーブルに複数の施設のデータを混在させた場合に、どのレコードがどの施設のデータであるかを識別する際に用いるものである。

エッジデータは端ノード ID(1) と端ノード ID(2) にエッジの両端のノードの ID を指定することによって、ノードとエッジの接続関係を示している。また、経路探索で使用するエッジ

の属性（距離，主要度，舗装有無，種別）を付加する．なお，主要度，舗装有無，種別は0または1で表す（表 3.1）．

ノードデータには種類，経緯度の情報を属性として付加する．ノードの種類番号は表 3.2 の通りである．経路探索の際にはエッジの長さを用いるため，エッジの長さの計算にノードの経緯度の情報が必要となる．また，経路探索は現在地から最も近い場所に配置されているノードをスタート地点として利用するため，スタート地点とするノードを決定する際にもノードの経緯度が使用される．これに加え，ノードの種類が建物出入口である場合，そのノードが出入口を示している建物類の ID も属性の1つとして付加する．なお，建物出入口以外のノードの場合は建物類 ID は不要なため NULL となる．

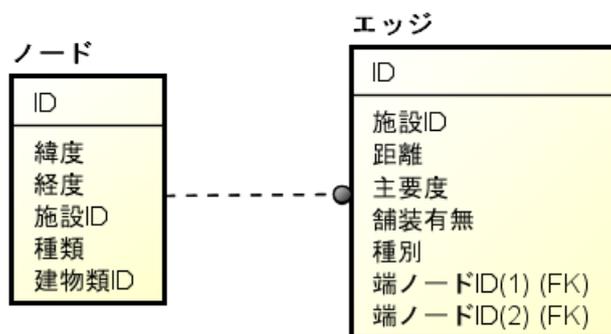


図 3.2: 道情報データベースの ER 図

表 3.1: 道情報データベース内で主要度，舗装有無，種別を表す値

値	主要度	舗装有無	種別
0	非主要路	未舗装路	平坦路
1	主要路	舗装路	階段

表 3.2: ノードの種類番号

種類番号	種類
1	施設出入口
2	交差点
3	カーブ
4	建物出入口

例えば図 3.3 のような道情報を登録すると，道情報データベースのテーブル内のデータは表 3.3，表 3.4 の通りとなる．

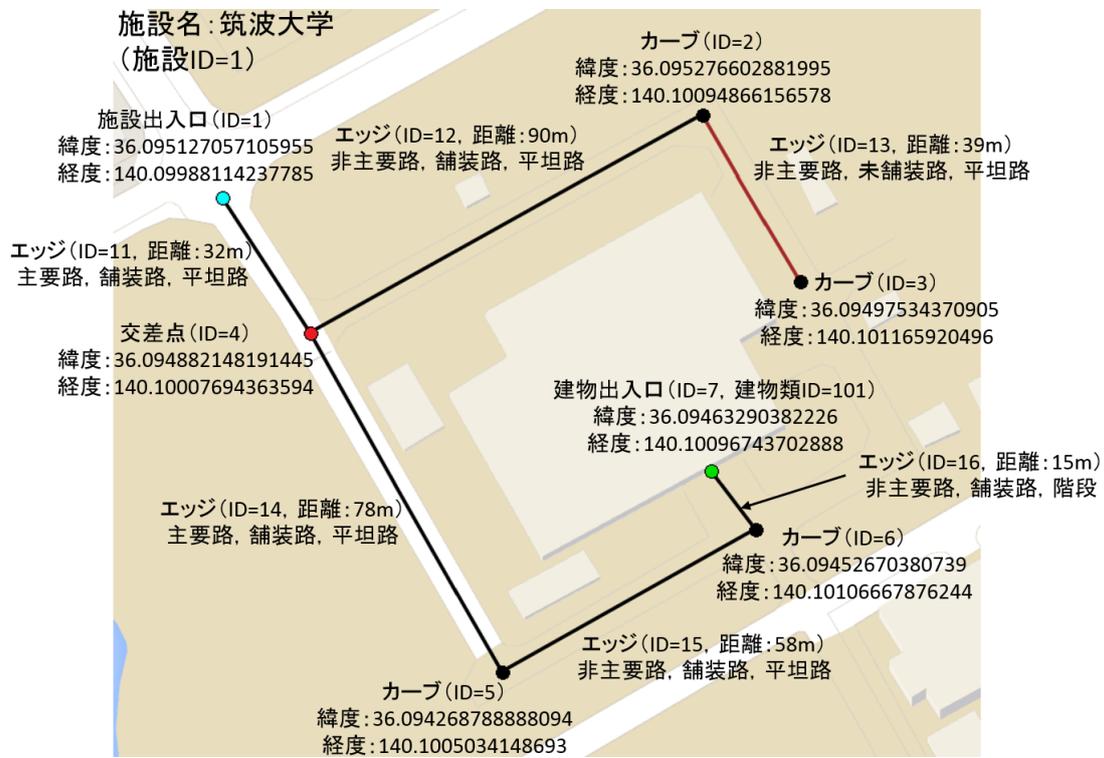


図 3.3: 道情報の例

表 3.3: ノードテーブルのデータの例

ID	緯度	経度	施設 ID	種類	建物類 ID
1	36.095127057105955	140.09988114237785	1	1	NULL
2	36.095276602881995	140.10094866156578	1	3	NULL
3	36.09497534370905	140.101165920496	1	3	NULL
4	36.094882148191445	140.10007694363594	1	2	NULL
5	36.094268788888094	140.1005034148693	1	3	NULL
6	36.09452670380739	140.10106667876244	1	3	NULL
7	36.09463290382226	140.10096743702888	1	4	101

表 3.4: エッジテーブルのデータの例

ID	施設 ID	距離	主要度	舗装有無	種別	端ノード ID(1)	端ノード ID(2)
11	1	32	1	1	0	1	4
12	1	90	0	1	0	4	2
13	1	39	0	0	0	2	3
14	1	78	1	1	0	4	5
15	1	58	0	1	0	5	6
16	1	15	0	1	1	6	7

3.3.2 管理用ユーザインタフェース部

管理用ユーザインタフェースの画面レイアウトを図 3.4 に示す。道情報の編集は地図上にグラフを描画することによって行うため、グラフ描画用のキャンバスとなる地図を画面中央に配置する（グラフ編集部）。また、グラフ編集の際には4種類のノードを使い分けるため、配置するノードの種類を変更するためのラジオボタンをグラフ編集部の下に配置する（ノード選択部）。さらに、地図上にはグラフの構成要素の種類を色分けによって表示するため、色と構成要素の種類を示す凡例を表示する（凡例部）。

凡例部には、グラフ編集部に表示されるノードの表示色とノードの種類、およびエッジの属性とエッジの表示色の対応関係が表示される。道情報を編集する際は4種類のノードが1つの地図上に混在することとなるため、ノードの種類を容易に見分けられるよう表 3.5 に示す種類毎にノードを色分けして表示する。同様にエッジも舗装路と未舗装路の区別がつくよう、舗装路は黒色、未舗装路は茶色で表示する。

ノード選択部では、道情報に追加するノードの種類をラジオボタンを用いて選択する。

建物出入口ノードの配置の際には、図 3.5 のように全ての建物類の上にマーカーが表示され、これらのうちいずれかのマーカーをクリックすると、そのマーカーの指す建物類がノードに関連付けられる。

ノードとエッジの操作ウィンドウのデザインをそれぞれ図 3.6 および図 3.7 に示す。ノード操作ウィンドウではノードの削除と移動を行える。エッジ操作ウィンドウではエッジの長さの確認やエッジの属性の編集、エッジの削除、エッジの途中へのノードの追加ができる。エッジの長さはエッジの描画や接続しているノードの移動を行ったタイミングで自動計算される。エッジの属性の編集はチェックボックスの操作によって行う。

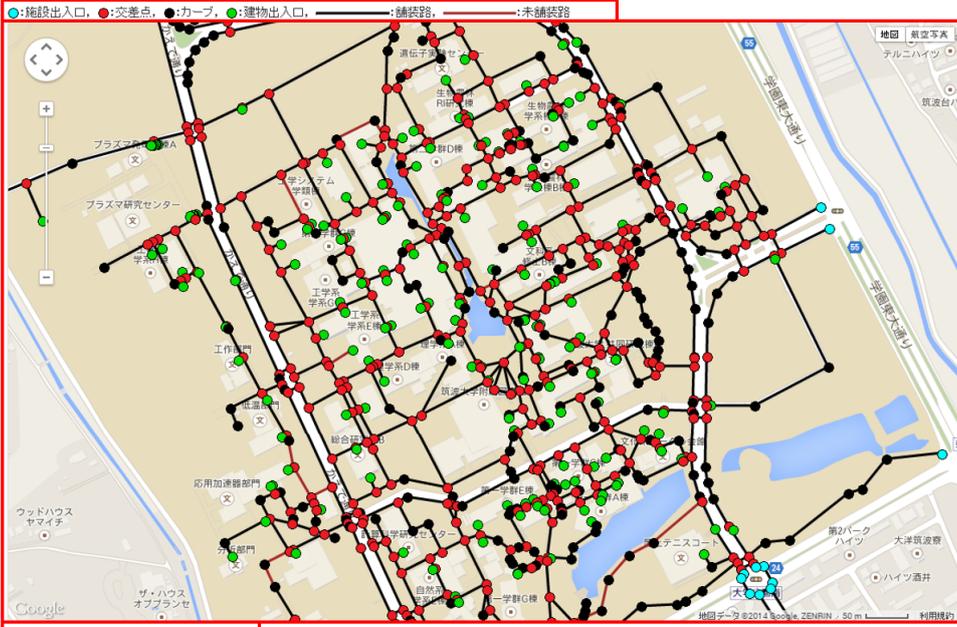
管理用ユーザインタフェースは JavaScript と Ruby on Rails により実現する。

JavaScript は道情報管理画面のグラフ編集部の動作制御およびデータ整理を担当する。施設管理者が行うグラフ編集部での編集操作に基づき、ノードやエッジの描画、操作ウィンドウの表示などを行うとともに、道情報に変更が生じた場合はデータの整理と Ruby on Rails へのデータベースの更新依頼を行う。Ruby on Rails へのデータベースの更新依頼は、道情報の変更を伴う操作があった際に Ajax を用いてリアルタイムに行われる。

グラフ編集部ではマウス操作による編集作業が行われることにより複数の描画状態が生じ

↓凡例部

ART



←ノード選択部 ↑グラフ編集部

図 3.4: 道情報管理画面のレイアウト

表 3.5: ノードの種類と表示色の対応関係

種類	表示色
施設出入口	水色
建物出入口	緑色
交差点	赤色
カーブ	黒色



図 3.5: 建物類を指すマーカ



図 3.6: ノード操作ウィンドウ

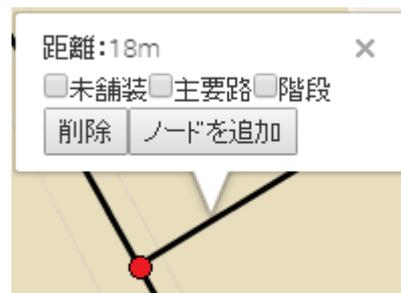


図 3.7: エッジ操作ウィンドウ

るため、本画面の動作制御プログラムは状態パターンを用いて実装する。これにより動作制御プログラムの可読性が向上するため、不具合の混入を防止できるとともに改修作業も容易になる。描画状態としては表 3.6 に挙げるものが存在し、図 3.8 の条件によって各状態間を遷移する。

Ruby on Rails は本画面のビューを呼び出す他、JavaScript からデータベースへのアクセスを行うための窓口としても機能する。道情報データベースに対するデータの追加、削除、更新、検索を行うためのメソッドが定義されており、JavaScript はこれらのメソッドを実行することによって道情報データベースへのアクセスを行う。

表 3.6: 動作制御プログラムの描画状態一覧

ID	状態名	内容
ST1	通常状態	描画操作が行われていない、または完了した状態。
ST2	建物類関連付け待ち状態	建物出入口ノードに建物類を関連付けるため図 3.5 のように建物類を指すマーカを表示し、建物類が指定されることを待機している状態。
ST3	ノード移動中状態	ノードの操作ウィンドウで「移動」ボタンがクリックされた後、ノードがドラッグ操作により移動されることを待機している状態。
ST4	エッジ作成中状態	ノード間をエッジで接続するための一方のノードの指定が完了し、もう一方のノードが指定されることを待機している状態。

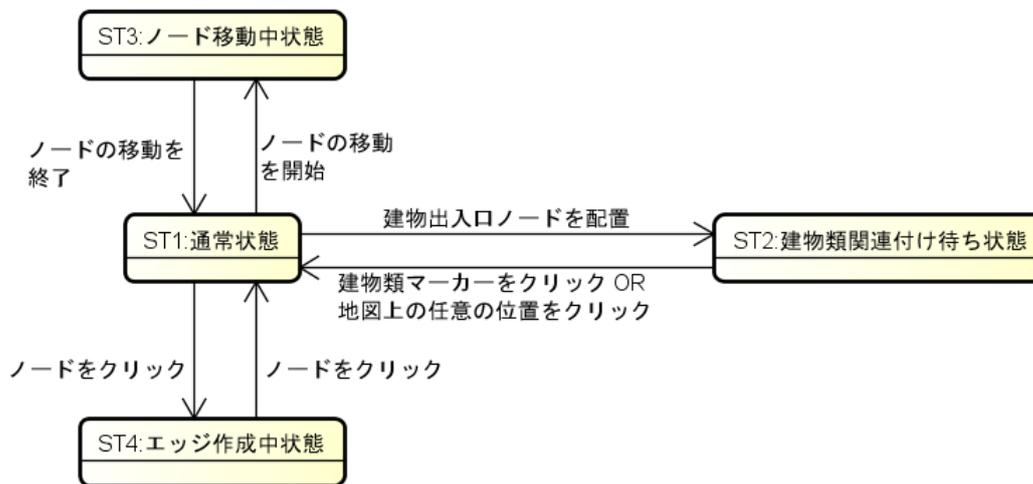


図 3.8: 動作制御プログラムの状態遷移図

3.4 動作例

本節では、カーブノードの削除を行う場合を例として本機能の動作機序を述べる。初期状態として図 3.9 に示す道情報がある場合を想定する。このとき、道情報データベースには表 3.7 および表 3.8 に示すデータが記録されている。ただし、道情報データベースのデータは本節での解説に必要な部分のみ抜粋している。

このときカーブノード (ID=12) に対して削除操作を行うと、3.2.2 で述べた通りカーブノードの両側に存在するノード同士 (ID=12, 13) が短絡され、図 3.10 に示す状態に変化する。また、道情報データベースのデータは表 3.9 および表 3.10 に示す状態に更新される。

この操作を行ったときの処理フローを図 3.11 に示す。ブラウザでカーブノードの削除操作が行われると、JavaScript より Ruby on Rails に対してカーブノードに接続するエッジと隣接するノードの ID を問合せる (図 3.11 (1.1, 1.2))。これにより得られた情報を用いて、カーブノードの両隣に隣接するノード同士を JavaScript 内で接続する (図 3.11 (1.3))。その後、道情報データベースを Ruby on Rails 経由で更新し (図 3.11 (1.4, 1.5, 1.6, 1.7))、ブラウザに結果を表示する。

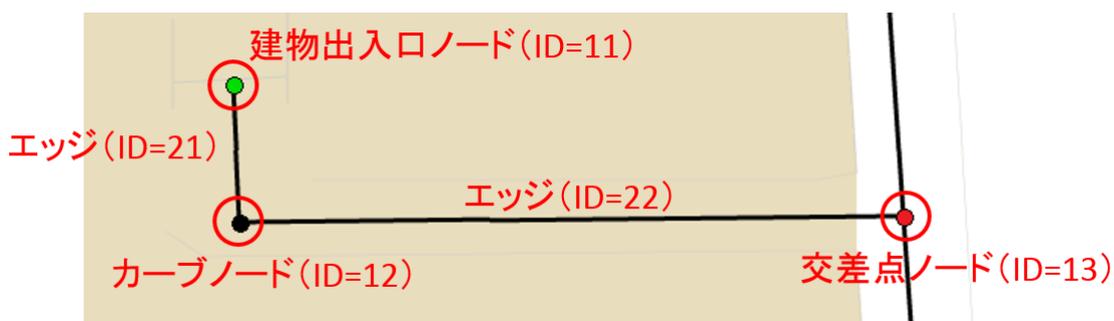


図 3.9: 道情報の初期状態

表 3.7: 道情報データベース (ノードテーブル) の初期状態

ID	ノードの種類番号	緯度	経度
11	4 (建物出入口)	36.10073591253271	140.10131880640984
12	3 (カーブ)	36.100686067435504	140.10132148861885
13	2 (交差点)	36.100688234614296	140.1016165316105

表 3.8: 道情報データベース (エッジテーブル) の初期状態

ID	端ノード ID(1)	端ノード ID(2)
21	11	12
22	12	13

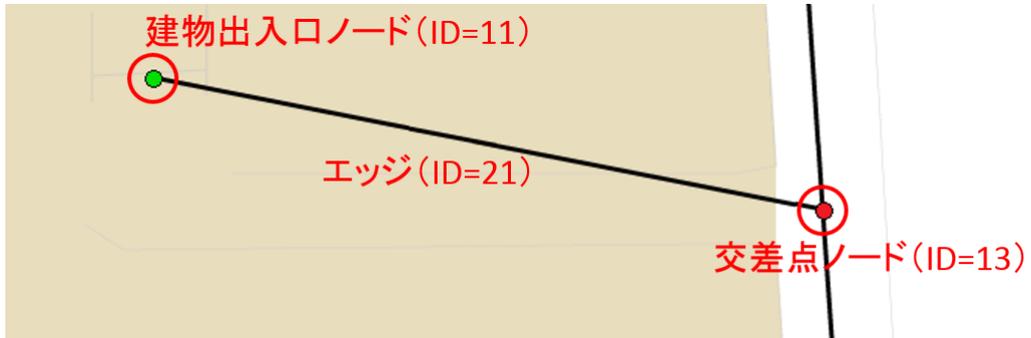


図 3.10: カーブノード削除後の道情報

表 3.9: カーブノード削除後の道情報データベース (ノードテーブル)

ID	種類	緯度	経度
11	4 (建物出入口)	36.10073591253271	140.10131880640984
13	2 (交差点)	36.100688234614296	140.1016165316105

表 3.10: カーブノード削除後の道情報データベース (エッジテーブル)

ID	端ノード ID(1)	端ノード ID(2)
23	11	13

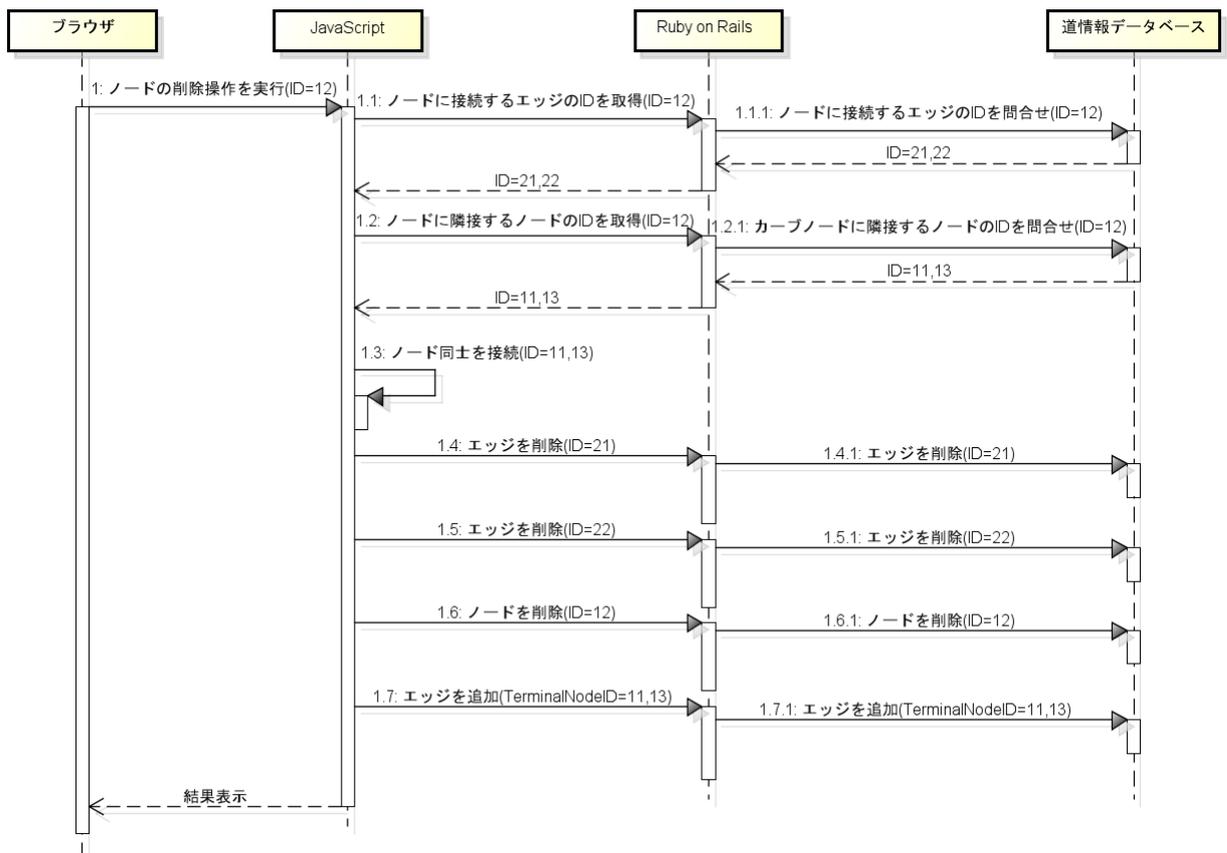


図 3.11: カーブノード削除の処理フロー

第4章 経路を探索する機能

本章では経路を探索する機能について述べる。経路探索は道情報データベースへのアクセスが必要となるため、歩行者ナビゲーションアプリからの要求に応じてサーバ内にて計算を行う。歩行者ナビゲーションアプリからサーバへの経路探索の要求は、サーバに実装した通信 API を使用する。現在地の経緯度、目的地の建物類の ID、経路の種類などといった経路探索に必要な情報を引数として歩行者ナビゲーションアプリが通信 API を実行すると、サーバ内で経路探索が行われ、その結果が返される。

4.1 探索できる経路の種類

本システムの経路案内機能の利用者として想定している人物は、初めて施設に訪れた者と施設内の通路構成をある程度知っている者である。

初めて施設に訪れた者は施設内の通路構成を全く知らないため、歩行距離に歩きやすさと分かりやすさを加味して求めた経路を提示することが望ましいと考えられる。また、カーナビゲーションシステムと同様に歩行者ナビゲーションシステムにおいても最短経路に加えてサービス利用者の希望に応じた経路案内を提供することが求められる [12]。通路構成の複雑な施設では、目的地までの経路が複数通りある場合が多い。例えば、目的地までの経路の候補に次の2通りがある場合、歩きやすく分かりやすい1の経路を提示する方が適切である。

1. 歩行者の往来が多い主要な道を主に歩行する経路。
2. 狭くて分かりにくい道や未舗装路の道を通る経路。

歩行中に道順について不安を感じたり、歩きにくい道を通ることの無いよう、本システムでは1の経路を推奨経路として提示する。

施設内の通路構成をある程度知っている者には目的地まで効率良く移動する経路を提示すべきと考えられる。最短経路は現在地から目的地までの距離のみで決定する経路であるため、先述の2のような経路となる場合がある。しかし、推奨経路より数分程度所要時間を短縮できるため、施設内の通路構成をある程度把握している者が普段訪れない建物類まで歩行する場合などに有用である。

以上の2つの種類の経路のそれぞれに、身体的な理由により階段の歩行が困難な者に対応するため階段を通らない経路を探索するオプションを選択できるようにする。サービス利用者は健常者のみとは限らず、階段の歩行が難しい者が存在することも想定される。そこで、階

段無しのオプションを選択可能とすることで、スロープなどを探して施設内を右往左往することなく円滑に目的地まで移動できるようになる。

以上より、本システムでは推奨経路、最短経路、推奨経路（階段無し）、最短経路（階段無し）の4種類の経路を探索できる仕様とする。

4.2 経路探索アルゴリズムの設計

4.2.1 推奨経路探索時のエッジの優先度を決定づける属性の選定

推奨経路の探索では、一定の条件を満たすエッジが優先的に経路として選択される仕様とする。そのため、優先度を決定づけるためのエッジの属性を選定する。

推奨経路は総距離に加えて進路の分かりやすさと歩きやすさを重視する経路である。従って、この2つの要素を適切に表す属性を検討する。進路の分かりやすさの指標となるものには、主要度および案内板有無が挙げられる。主要な道は付近に目印となるものが多い傾向があり、地図と照らし合わせた際に主要でない道よりも付近の地理が分かりやすい。また、案内板が設置されている道では案内板を参照することによって方角や周囲の建物類などが容易に把握できる。歩きやすさの指標となるものには舗装有無、傾斜有無、道幅が挙げられる。未舗装の道は靴を汚しやすく、降雨時などには特に歩みにくくなる。傾斜のある道は歩行中の疲労度合いを高める原因となり得る。また、道幅が狭い道は自転車とのすれ違い時に接触する危険が高い。特に筑波大学を例にとると、構内を多数の自転車が走行しており、道幅が狭いところでは危険を感じる場面が多くある。

一方、属性の数が多くなると、それに伴い道情報の登録作業にかかる時間や誤りの混入確率も増大してしまう。そこで、エッジの属性として何を採用するか検討するため、エッジの属性として考えられる候補を必要性および登録の容易さの観点で各属性候補を4段階で相対的に評価した（表4.1）。評価値が大きいほど必要性が高く、登録が容易であることを示している。

エッジの長さは経路探索のベースとなるデータであり、必要性は最も高い。また、エッジの登録と同時に自動計算が可能であるため、エッジの長さの登録に別途コストは発生しない。主要度および舗装有無は施設内での道の位置づけを示す重要な指標となるため、エッジの長さに次いで必要性が高いと言える。登録者が施設の通路構成を大まかに把握している者であれば主要度の判断は比較的容易に行える。舗装有無は登録にあたり現地調査が必要となるが、一般に未舗装の道の本数は限られており、調査が済めば登録は速やかに完了できることが見込まれる。

案内板有無は道の分かりやすさを判断する指標として有効である [13]。しかし、案内板は主要路近辺に多く設置されている傾向があるため、主要度をエッジの属性として採用するのであれば案内板有無をエッジの属性として別途登録する必要性は低いと考えた。傾斜は歩行時に大きな障害になるケースが少ないため、必要性を低く評価した。道幅は車両とは異なり歩行者にとって重要な要素でないため、最も必要性が低いと判断した。また、傾斜および

道幅は現地調査に大きなコストがかかるためエッジの属性として用いることは現実的ではなく、特に傾斜は計測が容易でない。

以上より、本システムでは推奨経路の探索に用いるエッジの属性として、道の長さ、主要度、舗装有無の3つを採用することとした。また、階段を通らない経路を探索するため、階段有無の属性もこれらに加えることとした。

表 4.1: エッジの属性候補の評価結果

道の属性候補	必要性	登録の容易さ	合計点
道の長さ	4	4	8
主要度	3	3	6
舗装有無	3	2	5
案内板有無	2	2	4
傾斜有無	2	1	3
道幅	1	1	2

4.2.2 主要路か否かの判断基準

道情報の初回登録は、入力するデータの量が多いため複数人で行うことが想定される。その際、エッジの主要度も各登録者が個別に登録する必要があるが、登録者によって主要度の判断に違いが生じないように、基本となる判断基準を設ける。

一般公道を例とすると、国道や都道府県道、市町村道が主要な道に該当する。特に国道は道路法にその基準が明確に定められている [14]。そこで、同法における国道の基準を参考に、主要度の判断基準を次のように定める。

次の条件 A, B について $A \wedge \neg B$ が真である場合、主要路と見なす。

条件 A 次のいずれかに該当すること。

1. 施設を縦断する道路
2. 施設を横断する道路
3. 施設を循環する道路
4. 公式な施設出入口から 1~3 のいずれかを満たす道路までを接続する道路

条件 B 次のいずれかに該当すること。

1. 未舗装または階段の道路
2. 施設発行の公式な地図に道路として表示されていない道

4.2.3 アルゴリズム

(1) 全体の処理フロー

経路探索の処理フローの全体図を図4.1に示す。歩行者ナビゲーションアプリより経路探索が要求されると、サービス利用者の現在地に最も近い位置に配置されているノードから目的地の建物類に関連付けられている建物出入口ノードまでの経路を探索する。目的地の建物類に関連付けられている建物出入口ノードが複数存在する場合は、該当する全てのノードまでの経路を探索し、最も総距離の短い経路を採用する。続いて探索結果の経路の総距離を求める。ここで求めた総距離の値は歩行者ナビゲーションアプリの画面に表示される。最後に探索結果の経路をJSON形式のデータに整理し、歩行者ナビゲーションアプリに返却する。

(2) 経路探索の方法

経路探索に使用するアルゴリズムはDijkstra法を基本とする。最短経路の探索を行う場合はエッジの長さをそのまま重みとして用いる。推奨経路の探索を行う場合は、エッジの重みとしてエッジの長さ属性をもとに計算した仮想距離を使用する方法が有用である[15]。そこで本システムでは主要路でない場合および舗装されていない場合に、当該エッジの長さを一定の割増率により割り増した値を仮想距離として用いる。これにより、経路探索時における主要路や舗装路の優先度が上がり、結果として分かりやすく歩きやすい道が選択されることになる。

エッジの長さを l 、主要路でない場合の割増率を a_1 、未舗装の場合の割増率を a_2 とすると、エッジ e_i の仮想距離 w_i は式(4.1)により求められる。

$$w_i = \begin{cases} l & \text{(主要路かつ舗装路)} \\ l(1 + a_1) & \text{(非主要路かつ舗装路)} \\ l(1 + a_1 + a_2) & \text{(非主要路かつ未舗装路)} \end{cases} \quad (4.1)$$

階段無しの経路を探索する場合は、階段属性が設定されているエッジの重みを非常に大きい値に置き換える。なお、本システムではこの値を 10^6 [m]としている。これにより、目的地に階段を通らずに到達できる経路が存在する場合は、その経路が探索結果として求められる。

4.2.4 割増率 a_1, a_2 の求め方

本システムにおける最適な推奨経路の条件を「極端な回り道をせず、なるべく長い区間で主要路および舗装路を歩行すること」と定義し、この条件になるべく近い経路を探索できるよう割増率を求める。

以降、割増率を求める手順を述べる。なお、最適な割増率は本システムを運用する施設によって異なる。

(1) 道情報の登録

あらかじめ施設の道情報を登録する。割増率は登録済みの道情報をもとに求める。

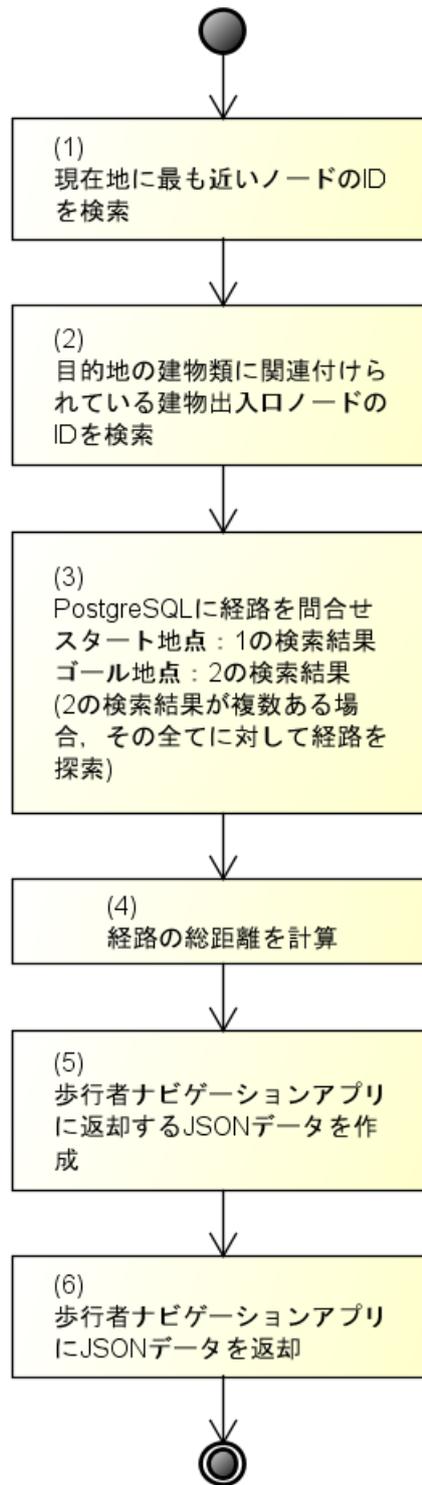


図 4.1: 経路探索の処理フロー

(2) 全ての建物出入口ノード間における経路の探索

全ての建物出入口ノード間における推奨経路と最短経路を探索する。推奨経路の探索では $0.1 \leq a_1 \leq 3.0, 0.1 \leq a_2 \leq 3.0$ の範囲で a_1, a_2 の全ての組み合わせについてエッジの仮想距離を求め、経路を探索する。探索したそれぞれの経路について、総距離、総距離に占める主要路でないエッジの長さの割合（以下、非主要路率）、総距離に占める未舗装路の長さ（以下、未舗装路率）の割合を計算する。

(3) サマリデータの生成

(2) で得られたデータを用いて a_1, a_2 の組み合わせ毎に次の値を計算する。これらの値を総称してサマリデータと呼ぶ。

1. 非主要路率の平均値
2. 未舗装路率の平均値
3. 推奨経路と最短経路の総距離の差の平均値
4. 推奨経路と最短経路の総距離の差の標準偏差

(4) 割増率の決定

推奨経路の非主要路率および未舗装路率は割増率が大きくなるとともに減少し、最終的には一定の値に収束する傾向がある。一方、推奨経路と最短経路の総距離の差は割増率が大きくなるとともに増加していく。したがって、非主要路率および未舗装路率が収束値に近づいており、かつ推奨経路と最短経路の総距離の差が必要以上に大きくならないような a_1, a_2 を求める。なお、筑波大学を対象とした割増率の算出の過程は 4.4.1 で述べる。

4.3 実装

4.3.1 歩行者ナビゲーションアプリとサーバとの間の通信方式

歩行者ナビゲーションアプリからサーバに対する経路探索の要求は、サーバに Ruby on Rails で実装されている通信 API を介して行う。通信 API は Ruby on Rails のコントローラのメソッドとして定義されており、これに対して歩行者ナビゲーションアプリから HTTP リクエストを送信する。HTTP リクエストの送信の際は、経路探索に必要な情報を引数として渡す。HTTP リクエスト送信時の URL は次の通りである。

```
http://(サーバドメイン名)/api/routes/search?(引数名)=(値)&...
```

引数の一覧を表 4.2 に示す。version 引数は歩行者ナビゲーションアプリ側が想定している API のバージョンとサーバの実際の API のバージョンが合っていることを確認し、予期せぬ不具合の発生を防止するためのものである。バージョンの照合を終えると、from_lat 引数およ

表 4.2: 経路探索の通信 API の引数

引数名	内容
version	API のバージョン番号
facility_id	施設 ID
type	探索を要求する経路の種類番号 (表 4.3 参照)
from_lat	現在地の緯度 (10 進法表記)
from_lng	現在地の経度 (10 進法表記)
to_entry_id	目的地の建物類の ID

表 4.3: 経路の種類番号

種類番号	経路の種類
1	推奨経路
2	最短経路
3	推奨経路 (階段無し)
4	最短経路 (階段無し)

び from_lng 引数で指定された位置に最も近いノードから to_entry_id 引数が指す建物類の建物出入口ノードまでの経路が、type 引数で指定された経路の種類に従って探索される。

探索結果は JSON(JavaScript Object Notation) 形式のデータとして歩行者ナビゲーションアプリに返却される。返却する JSON の構造は図 4.2 の通りとなっており、総距離とスタート地点からゴール地点までに通過するノードの経緯度の一覧が返却される。

```

{
  "alert": <<クライアントに伝えるメッセージ (無い場合は null) >>,
  "length": <<総距離 (単位: m) >>,
  "coordinates": [
    { "lat": <<緯度>>, "lng": <<経度>> },
    ...
  ]
}

```

図 4.2: 経路探索の通信 API より返却される JSON の構造

4.3.2 経路探索用ライブラリの利用

経路探索には 3.3.1 でも述べた通り pgRouting と呼ばれる経路探索エンジンを利用する。pgRouting には SQL を介して経路探索を行うための関数がいくつか定義されており、本機能では

Dijkstra 法による経路探索を行う `pgr_dijkstra` 関数を用いる。例として割増率 $a_1 = 1.5, a_2 = 1.6$ とし、スタート地点のノード ID を 100、ゴール地点のノード ID を 200 として `pgr_dijkstra` 関数を用いて推奨経路の経路探索を実行する場合の SQL を図 4.3 に示す。edges テーブルの distance カラムがエッジの長さであり、9 行目に仮想距離を計算する式が記述されている。通信 API のプログラム中にこのような SQL を自動的に生成するコードが記述されており、歩行者ナビゲーションアプリからの要求内容に応じてこのような SQL を用いて経路探索を行う。

```
1: select
2:     seq, id1 as node, id2 as edge, cost
3: from
4:     pgr_dijkstra(
5:         'select
6:             id,
7:             cast(startnode_id as int4) as source,
8:             cast(endnode_id as int4) as target,
9:             cast((distance * (1 + 1.5 + 1.6)) as float8) as cost
10:         from edges',
11:         100, 200, false, false)
```

図 4.3: 経路探索の SQL

4.4 筑波大学を対象とした経路探索の実行例

本節では筑波大学を例に、割増率 a_1, a_2 の算出プロセスとそれを用いた探索結果を示す。

4.4.1 割増率 a_1, a_2 の算出

4.2.4 で述べた方法に従い、割増率 a_1, a_2 を求める。 a_1, a_2 の増加に伴う非主要路率と未舗装路率の変化率が小さくなる領域を絞りこむため、 a_1, a_2 を 0.3 から 3.0 まで 0.3 刻みで増加させた結果を付録 C.1 に示す。 $1.2 \leq a_1 \leq 2.1, 1.2 \leq a_2 \leq 1.8$ の範囲内で非主要路率と未舗装路率の変化が収まっている様子が確認できる。

そこで、この範囲内で a_1, a_2 を 0.1 刻みで増加させ、より詳細に非主要路率と未舗装路率の変化を調べた結果を付録 C.2 に示す。非主要路率の変化率は $a_1 = 1.6$ を超えると小さくなることが確認できる。また、未舗装路率は $a_2 = 1.5$ を超えるとほぼ変化しないことが分かる。一方、 a_1, a_2 が大きくなるにつれて最短経路に対する推奨経路の距離比が大きくなっていくため、 a_1, a_2 はむやみに大きい値にするべきではない。したがって、非主要路率と未舗装路率の変化率が小さくなる時の値である $a_1 = 1.6, a_2 = 1.5$ が割増率として妥当であると判断できる。

なお、本学の学生 10 名に対して、推奨経路を歩行した場合に最短経路より何分までなら所要時間が延びることを許容できるか聞き取りにより調査したところ、平均 5.4 分までなら許容できるとの結果が得られた。したがって、人間の歩行速度を 5km/h と仮定すると推奨経路と最短経路の距離差は約 450m までであれば許容される。一方、 $a_1 = 1.6, a_2 = 1.5$ としたときの推奨経路と最短経路の距離差の平均は約 74m、標準偏差は約 104m である。よって、 $a_1 = 1.6, a_2 = 1.5$ のときの推奨経路と最短経路の距離差は許容範囲内に十分収まっていると言える。

以上より、筑波大学での推奨経路探索の際の仮想距離を求めるための割増率は $a_1 = 1.6, a_2 = 1.5$ を用いる。

4.4.2 探索結果の例

先述のアルゴリズムを用いて筑波大学の経路探索を行った結果の例を示す。

図 4.4 と図 4.5 に桐葉橋近辺から本部棟までの推奨経路と最短経路の探索結果を示す。最短経路は松美上池沿いの未舗装路を通行することに対し、推奨経路はほぼ主要路のみを通行する経路となっている。

一方、合宿所バス停から開学記念館への経路を探索すると、最短経路（図 4.7）はほぼ直線的に目的地へ向かう経路となるが、推奨経路（図 4.6）は大学を循環する道路を使用して大きく迂回する経路となっており、約 350m の距離差が生じてしまっている。これは推奨経路を探索する際に先述の割増率を用いてエッジの仮想距離を計算した結果、目的地まで直線的に向かう経路の仮想距離が大きくなってしまったことが原因である。このように、推奨経路の総距離が最短経路より大幅に長くなってしまう場合もある。



図 4.4: 桐葉橋近辺から本部棟までの推奨経路 図 4.5: 桐葉橋近辺から本部棟までの最短経路



図 4.6: 合宿所バス停から開学記念館までの推奨経路 図 4.7: 合宿所バス停から開学記念館までの最短経路

第5章 評価実験

本章では経路を探索する機能の評価実験について述べる。

5.1 目的

本実験の目的は次の2つである。

1. 推奨経路の妥当性の確認

推奨経路が最短経路と比較して進路の分かりやすい経路であるかを確認する。また、歩行の所要時間が妥当であるかを確認する。

2. 本システムが提供する最短経路案内の有用性の確認

目的地まで本システムを利用して最短経路を歩行した場合の歩行効率を確認する。

5.2 方法

本実験は12名の実験参加者(A-L)の協力を得て筑波大学構内で実施する。実験参加者に本システムまたは筑波大学のキャンパスマップ(紙媒体)を用いて所定の経路を歩行してもらい、その所要時間と実験参加者の様子を観察する。歩行する経路は次の通りである。また、被験者は表5.1に示す通り使用するツール毎に歩行する経路を変更して割当てることにより、前に同じ経路を歩行したことが次の歩行経路の実験結果に影響することを防ぐ。

経路 1: 合宿所バス停→開学記念館

推奨経路：964m (図 5.1)，最短経路：612m (図 5.2)

経路 2: 大学会館前交差点→本部棟

推奨経路：751m (図 5.3)，最短経路：652m (図 5.4)

経路 3: 本部棟→自然系学系 E 棟

推奨経路：667m (図 5.5)，最短経路：623m (図 5.6)

本実験で評価の対象とする事項は本システムのナビゲーション性能であり、特に案内経路の評価を行う必要がある。そこで、歩行者ナビゲーションシステムの評価軸として挙げられる項目 [16] のうち、次のものを実験参加者の歩行中における行動より測定することとした。



図 5.1: 経路 1 (推奨経路)



図 5.2: 経路 1 (最短経路)



図 5.3: 経路 2 (推奨経路)



図 5.4: 経路 2 (最短経路)



図 5.5: 経路 3 (推奨経路)



図 5.6: 経路 3 (最短経路)

表 5.1: 実験参加者の割当て

	経路 1	経路 2	経路 3
推奨経路	A, B, C, D	E, F, G, H	I, J, K, L
最短経路	I, J, K, L	A, B, C, D	E, F, G, H
キャンパスマップ	E, F, G, H	I, J, K, L	A, B, C, D

1. 歩行所要時間

スタート地点からゴール地点までの歩行にかかる所要時間を測定する。

2. 歩行距離

キャンパスマップを用いて歩行した距離を測定する。

3. 5秒以上立ち止まった回数

実験参加者に、進路が分からなくなり本システムの地図画面またはキャンパスマップを確認するときは必ず立ち止まるよう指示し、これらの確認のために5秒以上立ち止まった回数を記録する。

4. 進路を誤った回数

本システムを利用時の歩行中に、本システムが示す進路と異なる方向に進んだ回数を記録する。また、歩行終了後に進路を誤った原因を実験参加者からの聞き取りにより確認する。

また、実験終了後に実験参加者にアンケートを記入してもらい、歩行時に本システムを利用しての感想、問題点、改善案などを確認する。なお、本アンケートでは歩行者ナビゲーションアプリの仕様に関する質問項目も設けてあるが、本稿では経路探索に関する質問項目についてのみ言及する。本実験の詳細な手順書、実験参加者行動記録用紙、アンケートの様式を付録Dに添付する。

5.3 結果

5.3.1 実験参加者の行動に関する観察項目

実験参加者の行動に関する観察項目の結果を表5.2～5.5に示す。

表 5.2: 歩行所要時間の平均 (分)

	経路 1	経路 2	経路 3
推奨経路	12.3	10.8	10.0
最短経路	11.0	9.8	10.8
キャンパスマップ	10.0	10.3	12.3

5.3.2 アンケート

アンケートの質問項目のうち、経路探索に関する質問とその回答について述べる。

表 5.3: 歩行距離 (単位 : m)

	経路 1	経路 2	経路 3
推奨経路	974	751	694
最短経路	622	652	665
キャンパスマップ (平均)	724.8	749.3	915.0

表 5.4: 5 秒以上立ち止まった回数の平均

	経路 1	経路 2	経路 3
推奨経路	1.3	2.0	2.5
最短経路	2.3	3.5	5.3
キャンパスマップ	2.0	0.8	3.8

表 5.5: 進路を誤った回数の平均

	経路 1	経路 2	経路 3
推奨経路	0.0	0.5	0.75
最短経路	1.0	1.0	0.5

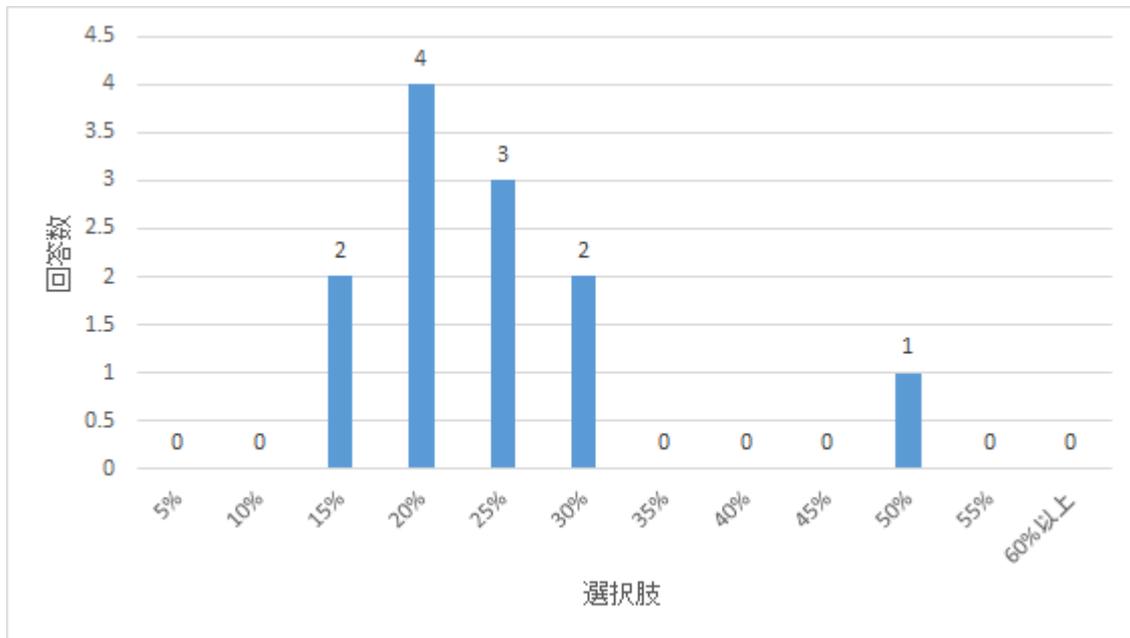


図 5.7: 質問項目 (1) への回答の集計結果

(1) 推奨経路の総距離は最短経路の何%までなら増えても構わないと思いますか？

本質問項目への回答の集計結果を図 5.7 に示す。

(2) 進路を確認する際にどのようなものを目印にして歩きましたか？(複数回答可能)

本質問項目への回答の集計結果を図 5.8 に示す。

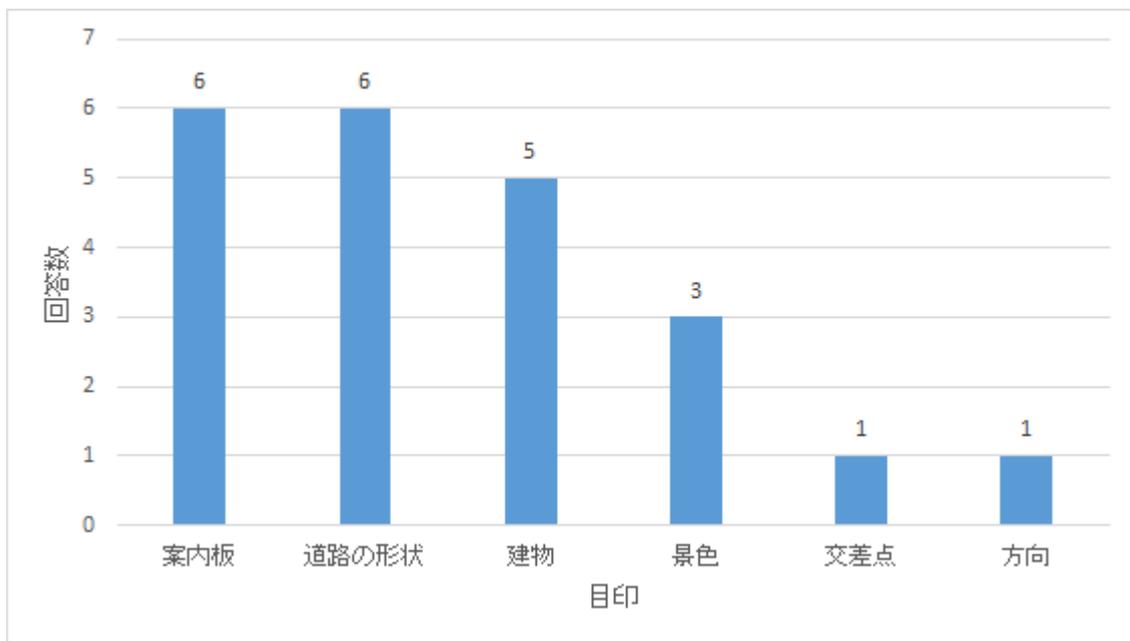


図 5.8: 質問項目 (2) への回答の集計結果

(3) 推奨経路, 最短経路, 推奨経路 (階段無し), 最短経路 (階段無し) の他に提供してほしい経路の種類はありますか？(複数回答可能)

本質問項目への回答として得られた経路の種類は次の通りである。

1. 指定された地点を経由する経路
2. 主要な道路のみを経由する経路
3. 自分の記憶や感覚に合った経路
4. 過去に自分が通ったことのある道が優先される経路
5. 同じ目的地に向かう場合に一般によく用いられる経路
6. 最も短い時間で目的地に到達できる経路

(4) その他ご意見やシステムの不具合がありましたら記入してください。

本質問項目への回答は全て歩行者ナビゲーションアプリの仕様に関する意見であり、経路探索結果に関する回答は得られなかった。

5.4 議論

5.4.1 実験参加者の行動に関する観察項目

歩行所要時間の平均（表 5.2）は、経路 1 と経路 2 においては最短経路よりも推奨経路の方が長かったものの、1 分程度の差しか生じなかった。経路 3 においては最短経路より推奨経路の方が短い結果となったものの、ほぼ誤差の範囲内であり、いずれの経路においても推奨経路および最短経路のどちらを選択しても歩行所要時間の差は無視できる範囲内にあると言える。経路 1 ではキャンパスマップを用いた場合に最も歩行時間が短くなった。キャンパスマップを用いた場合は全ての実験参加者が最短経路に似た経路を歩行し、かつ途中で進路を誤ったために引き返すことがなかったためと考えられる。最短経路を歩行した実験参加者は平均 1 回 1 度歩行した道を引き返したため、キャンパスマップを用いた場合よりやや歩行時間が長くなった。経路 3 は目的地である自然系学系 E 棟の入口の位置が分かりにくかったために、キャンパスマップを用いた場合の歩行所要時間が最も長くなった。この結果より、本システムを用いることによって目的地の建物類の出入口を素早く見つけることができると言える。

歩行距離（表 5.3）は、経路 1 と経路 2 ではキャンパスマップを用いた場合の平均が推奨経路と最短経路の間の値となったことに対し、経路 3 の平均は推奨経路および最短経路のどちらよりも長いものとなった。実験後の聞き取り調査では、多少遠回りになったとしても過去に自身が歩行したことのある道を選ぶ実験参加者がいることが明らかとなり、「本システムがあればより短い経路を安心して歩行できて良い」という意見を得た。この結果より、本システムによる最短経路の案内を用いれば効率良く目的地に到達できることが確認できた。

進路を誤った回数の平均（表 5.5）は経路 1 と経路 2 では最短経路の方が多く、推奨経路の方が進路が分かりやすいことが分かった。経路 3 では推奨経路の方が進路を誤った平均回数が多いが、進路の誤りが多い箇所は最短経路の場合にも通過するところであり、実験参加者間のわずかな個人差が影響したものと考えられる。

以上より、

1. 最短経路と比較して推奨経路の方が進路が分かりやすい。
2. 本システムの最短経路案内は目的地までの効率の良い経路を探す際に役立つ。

の 2 点が明らかとなり、本システムが提示する推奨経路の妥当性と最短経路の有用性が確認できた。

5.4.2 アンケート

質問項目 (1) では、15～30%の間に回答が集中した。最も回答数の多い選択肢は 20%であったが、この基準と表 5.3 の値を基に推奨経路の距離として許容される最大長を計算すると次の

通りとなる。

経路 1 746.4m (本システムによる探索結果の経路はこの範囲を超えている)

経路 2 782.4m (本システムによる探索結果の経路はこの範囲内に収まっている)

経路 3 798.0m (本システムによる探索結果の経路はこの範囲内に収まっている)

経路 1 の推奨経路は構内を循環する道路を用いて大きく迂回する経路となっている。大きく迂回する経路以外に周囲に主要路が存在しない場合はこのように推奨経路の距離がやや長くなり過ぎてしまうことがある。したがって、今後は道の主要度の判断基準を再検討し、可能な限り推奨経路の総距離が長くないよう工夫する必要がある。

質問項目 (2) は、経路案内時に目印として活用されているものを調査することによって今後の発展について検討することを目的に設けた質問である。本質問の回答では、案内板や道路の形状を参考に歩行するという意見が多い結果となった。本システムにおける推奨経路の探索時には主要路が優先して選択されるようになっているが、一般に主要路はそうでない道と比較して案内板が多い傾向があるため、推奨経路は目印が多く歩きやすい道であると言える。また、本システムでは AR 表示と現実の道の形を照合しながら歩くため、道路の形状を参考にしながら歩きたいというニーズも満たすことができている。建物や景色を参考にするという意見もあったが、歩行中にこれらの情報をより効果的に活用するには、AR 表示により周囲の建物や景色の案内を行う方法が考えられる。建物や景色の名称を AR で表示することによって、進路がより素早く把握できるようになることが期待できる。

質問項目 (3) の回答については、特に 5 の意見が、提示する経路の改善方法として有効であると考えられる。新たな種類の経路として設けるのではなく、推奨経路の探索時における考慮事項の 1 要素として追加することにより、より分かりやすくかつ効率の良い経路を提示できるようになると考えられる。

本実験により、本システムの経路探索機能はおおむね要件を満たしていることが確認できた。今後は推奨経路の探索方法をより洗練させるとともにアンケートの結果に基づき歩行者ナビゲーションアプリの仕様を改善することによって、さらに有用なシステムになることが期待できる。

第6章 おわりに

本プロジェクトでは施設内での経路案内を可能にするナビゲーションシステムの開発を行った。現在普及している歩行者ナビゲーションシステムには一般公道や公共交通機関を用いた経路の案内を行うもの、特定の施設専用開発されたものの2種類が存在する。前者はサービスを利用できる範囲が広いものの、施設の敷地内における詳細な経路案内ができないという問題がある。後者は来場者のニーズに合わせて施設内部の情報を詳細に提供できるが、その性質上施設毎に個別開発されており、導入する施設の種類を選ばない汎用的なシステムは存在していない。そこで本プロジェクトでは施設内の詳細な情報提供や経路案内が可能かつ汎用的な歩行者ナビゲーションシステムを開発した。

本システムの開発では、まず大学内での使用を想定したシステムを開発し、その後汎用化やシステムの改善を目的として次のことを行った。

1. iOS 向け歩行者ナビゲーションアプリの開発

大学内での使用を想定したシステムを開発した段階では Android 向けの歩行者ナビゲーションアプリのみ開発していたため、新たに iOS 向けのアプリを開発した。

2. 検索機能の拡張

あいまい検索に対応させ、検索時の利便性の向上を図った。

3. 経路案内機能の追加と道情報自動登録ツールの開発

施設の敷地内における詳細な経路案内を行う機能を開発した。また、経路案内に使用する道情報の登録作業を簡易化するため、道情報を自動登録するツールを開発した。

4. 施設内情報データベースの設計の変更

施設内情報データベースに汎用性を持たせるため、データベースの設計を変更した。

著者は先述の3に係る役割として、道情報データベースの構築と管理アプリケーションの開発および経路を探索するAPIの開発を行った。道情報データベースとその管理アプリケーションは、経路案内機能を追加するに当たり新たに必要となったものである。これらを開発することにより、施設内の詳細な道情報を本システムが保持することが可能となった。経路を探索するAPIの開発に際しては、推奨経路、最短経路およびそれらの階段を含まない経路の4種類を探索するアルゴリズムを考案し、実装した。また、経路探索結果の評価実験を行った結果、最短経路と比較して進路の分かりやすい経路を推奨経路として得られることが確認

できた。さらに、本システムの経路案内は目的地とする建物類の出入口を迅速に発見することに役立つことも確認できた。これにより汎用的な経路案内機能の基盤が整い、様々な施設において単一のシステムを用いて経路案内のサービスを提供できるようになった。

今後の展望として、推奨経路の探索にそれぞれの道が人々に利用されている頻度を考慮事項として追加することが挙げられる。現状ではエッジの長さ、主要度、舗装状態の3項目のみを用いて経路探索が行われるが、評価実験で使用した経路1の推奨経路のように、あまり現実的でない経路が提示されてしまう場合があるという課題がある。そこで、道路が人々に利用されている頻度を考慮事項として追加することによりさらに現実的な経路が選択されるように改善できると考えられる。

謝辞

本プロジェクトを遂行するに当たり、著者の指導教員である田中二郎教授には報告書や発表などについて、折に触れて有益なご意見やご指摘を賜りました。また、本プロジェクトの課題担当教員である和田耕一教授並びに山際伸一准教授には顧客役をご担当頂くとともにプロジェクト全般について終始多くの貴重なご助言を頂きました。先生方の厳しくも暖かいご指導により本プロジェクトを無事に終わられましたことに深く感謝申し上げます。

並列分散処理研究室秘書の森田美紀さん並びに石原美奈子さんには本プロジェクトに関する事務手続きなどで大変お世話になりました。

高度IT育成のための実践的ソフトウェア開発専修プログラムの同期および後輩には、本システムの評価実験にご協力いただきました。おかげさまで大変参考になる評価結果が得られました。ここに感謝の意を表します。

本プロジェクトのメンバーである畑中裕太君、芳賀隼人君、萬成亮太君には共にプロジェクトを遂行する仲間として心の支えとなってくださり、多くの場面でお世話になりました。皆さんと同じチームで活動でき、とても充実した日々を過ごすことができましたことを心より感謝申し上げます。

最後に、2年間の大学院での生活における様々な場面でご支援を頂いた家族、友人、教職員の皆様に御礼申し上げます。

参考文献

- [1] 総務省. 平成 26 年版情報通信白書.
- [2] ナビタイムジャパン. NAVITIME. <http://www.navitime.co.jp/>.
- [3] Google Inc. Google マップ. <https://www.google.co.jp/maps/>.
- [4] Google Play. 攻略ナビー東京ディズニーランド. https://play.google.com/store/apps/details?id=com.attraction_navi.tdl&hl=ja.
- [5] ユー・エス・ジェイ. ユニバーサル・スタジオ・ジャパン 公式ガイドアプリ. <http://www.usj.co.jp/app/>.
- [6] MM 総研. スマートフォン市場規模の推移・予測 (2014 年 4 月). <http://www.m2ri.jp/newsreleases/main.php?id=010120140423500>.
- [7] Jean-Philippe Lang. Redmine. <http://www.redmine.org/>.
- [8] 糸永航, 松田一朗, 米山範隆, 伊東晋. 地図画像からの道路ネットワークの自動抽出. 電子情報通信学会論文誌, Vol. J82-D-II, No. 11, pp. 1990–1999, November 1999.
- [9] 佐藤公男. グラフ理論入門ー C 言語によるプログラムと応用問題 (原理がわかる工学選書). 日刊工業新聞社, April 1994.
- [10] Google Inc. Google Maps API. <https://developers.google.com/maps/?hl=ja>.
- [11] pgRouting Community. pgRouting Project. <http://pgrouting.org>.
- [12] 松田三恵子, 杉山博史, 土井美和子. 歩行者の経路への嗜好を反映した経路生成. 電子情報通信学会論文誌, Vol. J87-A, No. 1, pp. 132–139, January 2004.
- [13] 鴛海祐太, 佐野友紀. 経路探索に用いられる情報が、歩行者の「迷い感」に与える影響. 一般社団法人日本建築学会 学術講演梗概集, E-1, 建築計画 I, 各種建物・地域施設, 設計方法, 構法計画, 人間工学, 計画基礎, pp. 853–854, July 2008.
- [14] 井上正仁, 能見善久 (編). 六法全書 平成 26 年版 I, p. 2144. 有斐閣, March 2014.

- [15] 荒井亨, 戸川望, 柳澤政生, 大附辰夫. 屋内用歩行者ナビゲーションにおける歩行者の嗜好を反映させる経路探索手法. 情報処理学会研究報告, Vol. 2006, No. 103, pp. 105–110, September 2006.
- [16] 間邊哲也, 長谷川孝明. 歩行者ナビゲーションシステムの分類学. 電子情報通信学会技術研究報告, Vol. 109, No. 325, pp. 67–72, December 2009.

付録A 環境構築手順書

A.1 サーバ構築手順書

環境構築手順書

平成26年2月5日

キャンパスARの基盤の構築

ART

文責: 芳賀隼人 版: 1.1

1	使用環境	3
2	インストール	3
2.1	Debian	3
2.2	RVM	3
2.3	nginx	4
	本体のインストール	4
	nginx の設定	4
	バーチャルホストの設定	5
2.4	unicorn	6
2.5	MySQL	6
2.6	Ruby	7
3	アプリケーションの設定	7
4	アプリケーション・サーバーの起動と終了	8
4.1	サーバー起動後の場合	8
4.2	アプリケーション・サーバーの一時的な起動	8
4.3	アプリケーション・サーバーの恒久的な起動	9
4.4	アプリケーション・サーバーの終了	9
5	データベースのバックアップと復元	10
5.1	データベースのバックアップ	10
5.2	データベースの復元	10
6	納品環境の構築	10

1 使用環境

当システムで使用するサーバは、下記の環境を設定します。

OS: Debian 6.0

Ruby パッケージ管理システム: rvm 1.20.0

Web サーバ: nginx 1.2.1

アプリケーションサーバ: unicorn 4.6.2

データベース: MySQL 5.5.30

プログラミング言語: Ruby 2.0.0p0

2 インストール

2.1 Debian

本システムの OS には Debian での実行を想定しています。下記の手順に従って、PC に Debian をインストールして下さい。

1. インストール CD-ROM を PC にセットして、CD ブートをを行います。
2. しばらく、インストーラに回答する形で進めます。
3. ルートパスワードは、絶対に忘れないようにして下さい。

2.2 RVM

RVM とは、同一のサーバに複数の Ruby 処理系をインストールし、切り替えることが出来るシステムです。今回はひとつの処理系のみを利用します。RVM をインストールすることで、いくつかの複雑なインストール手順を簡略化できます。

1. ユーザー権限で、下記のコマンドを実行します。

```
curl -L https://get.rvm.io | bash
```

2. 下記のコマンドで RVM がインストールされたことを確認します。

```
rvm -v  
rvm 1.20.0 (stable) by Wayne E. Seguin <wayneesequin@gmail.com>, Michal Papis <mpapis@gmail.com>  
> [https://rvm.io/]
```

2.3 nginx

本体のインストール

1. 下記のコマンドを実行し、管理者になります。

```
sudo su -
```

2. 下記のコマンドを実行し、nginx をインストールします。

```
aptitude install nginx-full
```

3. 下記のコマンドで nginx がインストールされたことを確認します。

```
nginx -v  
nginx version: nginx/1.4.1
```

nginx の設定

1. nginx の設定を変更するため、設定ファイルを開きます。

```
/etc/nginx/nginx.conf
```

2. ファイルを下記の内容に置き換えます。

```
user www-data;  
worker_processes 8;  
pid /run/nginx.pid;  
  
events {  
    worker_connections 768;  
}  
  
http {  
    sendfile on;  
    tcp_nopush on;  
    tcp_nodelay on;  
    keepalive_timeout 65;  
    types_hash_max_size 2048;  
  
    include /etc/nginx/mime.types;  
    default_type application/octet-stream;  
  
    access_log /var/log/nginx/access.log;  
    error_log /var/log/nginx/error.log;  
  
    gzip on;  
    gzip_disable "msie6";  
  
    include /etc/nginx/conf.d/*.conf;  
    include /etc/nginx/sites-enabled/*;  
}
```

3. 設定を反映させるため、下記のコマンドを実行し、nginx を再起動します。

```
service nginx restart
```

バーチャルホストの設定

1. サーバーにアクセスされたドメインに応じて、呼び出されるアプリケーション・サーバーを切り替える設定を行います。下記の場所にファイルを作成します。ここでは、設定ファイルを「vhost-setting」という名前にしたと仮定します。

```
/etc/nginx/sites-available/
```

2. 下記の内容を入力します。

<domain> は Web ページのドメイン名、<port> は任意のポート番号に変更する必要があります。

ここで指定したドメイン名 (<domain>) とポート番号 (<port>) は、他の作業で指定することになります。

```
upstream <domain> {
    server 0.0.0.0:<port>;
}

server {
    listen 80;
    server_name <domain>;
    root /var/www/<domain>/server/public;

    access_log /var/log/nginx/<domain>_access.log;
    error_log /var/log/nginx/<domain>_error.log;

    location / {
        if (-f $request_filename) { break; }
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $http_host;
        proxy_pass http://<domain>;
    }
}
```

例えば、ドメイン名が「campus-ar.jp」、ポート番号が「5000」の場合、設定ファイルの内容は下記のようになります。

```
upstream campus-ar.jp {
    server 0.0.0.0:5000;
}

server {
    listen 80;
    server_name campus-ar.jp;
    root /var/www/campus-ar.jp/server/public;

    access_log /var/log/nginx/campus-ar.jp_access.log;
    error_log /var/log/nginx/campus-ar.jp_error.log;

    location / {
        if (-f $request_filename) { break; }
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $http_host;
        proxy_pass http://campus-ar.jp/;
    }
}
```

- 作成したバーチャルホストの設定を利用出来るようにするため、下記のコマンドを実行します。

```
ln -s /etc/nginx/sites-available/vhost-setting /etc/nginx/sites-enabled/vhost-setting
```

- 設定を反映させるため、下記のコマンドを実行し、`nginx` を再起動します。

```
service nginx restart
```

2.4 unicorn

`unicorn` は Rails アプリケーションを動作させるアプリケーション・サーバーです。`nginx` が `unicorn` を呼び出し、`rails` アプリケーションを実行します。

- ユーザー権限で下記のコマンドを実行します。

```
gem install unicorn
```

- 下記のコマンドで `unicorn` がインストールされたことを確認します。

```
unicorn_rails -v  
unicorn_rails v4.6.2
```

2.5 MySQL

- 管理者権限で下記のコマンドを実行します。

```
aptitude install mysql-server
```

- インストール途中で `MySQL` の管理者パスワードを設定します。パスワードは絶対に忘れないようにして下さい。
- インストールが終わったら、本システム用のデータベースを作成します。下記のコマンドを実行した後、パスワードを入力して `MySQL` に接続します。

```
mysql -u root -p
```

- 下記のコマンドを実行してデータベースを作成します。
下記の例ではデータベース名に「`art`」を指定していますが、任意のデータベース名で構いません。

```
CREATE DATABASE art; quit;
```

2.6 Ruby

1. ユーザー権限で下記のコマンドを実行します。

```
rvm install 2.0.0
```

2. 下記のコマンドで Ruby がインストールされたことを確認します。

```
ruby -v  
ruby 2.0.0p0 (2013-02-24 revision 39474) [i686-linux]
```

3 アプリケーションの設定

サーバーのセットアップが済んだら、rails アプリケーションのセットアップを行います。

1. 下記の場所に rails アプリケーションを展開します。

```
/var/www/<domain>/server/
```

2. データベースの設定を記した下記のファイルを開きます。

```
/var/www/<domain>/server/config/database.yml
```

3. データベースのユーザー名とパスワードを入力します。username には「root」、password には MySQL インストール時に設定した管理者パスワードを設定します。また、アプリケーション用のデータベース作成時に、データベース名に「art」以外の名前を指定した場合は、database にその名前を指定します。

```
:  
database: art  
username: root  
password: (インストール時に設定した管理者パスワード)  
:
```

4. アプリケーションを動かすに必要なライブラリのインストール及びデータベースのセットアップをします。まず下記の場所に移動して下さい。

```
/var/www/<domain>/server/
```

5. 次にライブラリをインストールします。下記のコマンドを実行して下さい。

```
bundle install
```

6. データベースをセットアップします。下記のコマンドを実行して下さい。

```
rake db:migrate
```

7. これでアプリケーションの設定は完了です。

4 アプリケーション・サーバーの起動と終了

アプリケーション・サーバーを起動すると、実際に利用者のアクセスに対して、nginx と unicorn、unicorn と rails アプリケーションの間でやり取りが行われ、本システムが利用できるようになります。

4.1 サーバー起動後の場合

サーバーを起動した直後は、アプリケーションを動かすのに必要なデーモンが一部起動していない可能性があります。管理者権限で下記のコマンドを実行することで、必要な機能を起動します。

```
service mysql restart
service nginx restart
```

4.2 アプリケーション・サーバーの一時的な起動

1. 下記の場所に移動します。

```
/var/www/<domain>/server/
```

2. 下記のいずれかのコマンドを実行します。<port> には、nginx のバーチャルホスト設定時に指定したポート番号を指定します。-E オプションの後には、目的に応じて production か development をつけます。通常は production です。

```
unicorn_rails -p <port> -E production
unicorn_rails -p <port> -E development
```

3. 上記コマンドを実行すると、nginx からの送信を待ち続ける状態になります。この方法で起動したアプリケーション・サーバーは、ターミナルのサーバーとのセッションが切れたり、ユーザー操作によってコマンドが中断されると終了します。

4.3 アプリケーション・サーバーの恒久的な起動

1. nginx や MySQL サーバーと同様に、アプリケーション・サーバーを起動し続ける場合は、下記のコマンドを実行します。

```
unicorn_rails -p <port> -E production -D
unicorn_rails -p <port> -E development -D
```

2. 上記のように、「-D」オプションを付けてアプリケーション・サーバーを起動すると、デーモン化します。この場合、セッションが切れてもアプリケーション・サーバーが動き続けます。

4.4 アプリケーション・サーバーの終了

1. 恒久的に起動したアプリケーション・サーバーを終了する場合は、unicorn の親プロセスを kill します。以下のコマンドを実行し、アプリケーション・サーバーの親のプロセス ID を求めます。

```
ps -ax | grep "unicorn_rails master"
```

2. コマンドを実行すると、以下のような出力が得られます。

```
ps -ax | grep "unicorn_rails master"
warning: bad ps syntax, perhaps a bogus '- '?
See http://gitorious.org/procps/procps/blobs/master/Documentation/FAQ
16953 pts/4  S1+  0:04 unicorn_rails master -p 6000 -E development
18499 pts/2  S1+  0:01 unicorn_rails master -p 5001 -E development
31088 ?       S1    0:00 unicorn_rails master -p 5000 -E production -D
31105 pts/1   S+    0:00 grep unicorn_rails master
```

この内、終了したいプロセスの先頭の番号を記憶します。例えばポート番号「5000」で production として動作しているアプリケーション・サーバーを終了したい場合は、その行の先頭の「31088」を記憶します。

3. 下記のコマンドを実行してアプリケーション・サーバーを終了します。

```
kill <PID>
```

先ほどの例の場合だと、下記ようになります。

```
kill 31088
```

4. 起動中のアプリケーション・サーバーを再度リスタートアップしてみると、目的のアプリケーション・サーバーが終了していることが確認できます。

```
art:/var/www/campus-ar.jp/server# ps -ax | grep "unicorn_rails master"
warning: bad ps syntax, perhaps a bogus '- '?
See http://gitorious.org/procps/procps/blobs/master/Documentation/FAQ
16953 pts/4  S1+  0:04 unicorn_rails master -p 6000 -E development
18499 pts/2  S1+  0:01 unicorn_rails master -p 5001 -E development
31118 pts/1   S+    0:00 grep unicorn_rails master
```

5 データベースのバックアップと復元

5.1 データベースのバックアップ

下記のコマンドを実行することで、バックアップが可能になります。パスワードにはMySQLインストール時に設定した管理者パスワードを入力します。

```
mysqldump -u root -p パスワード art > artdump.sql
```

コマンドを実行すると、artdump.sql というバックアップファイルが作成されます。

5.2 データベースの復元

下記のコマンドを実行することで、バックアップファイルからデータの復元が可能になります。パスワードにはMySQLインストール時に設定した管理者パスワードを入力します。

```
mysql -u root -p パスワード art < artdump.sql
```

6 納品環境の構築

- 2~4章を参考に、アプリケーションサーバの起動と終了ができる状態にしてください。
- 納品物の server ディレクトリを下記の場所にコピーしてください。

```
/var/www/<domain>/
```

- データベースをセットアップします。下記のコマンドを実行してください。

```
rake db:migrate
```

- 納品物の環境構築は以上です。4章に従って、アプリケーションサーバを起動してください。

A.2 道情報データベース環境構築手順書

道情報DB環境構築手順書

操作手順

1. ユーザアカウント「art」でログインする。
2. PostgreSQLをインストールする。

```
$ sudo apt-get install postgresql
```

- ※ 「続行しますか [Y/n]?'と表示されたら Enter キーを押す。
- ※ PostgreSQLのインストールが完了すると、新たにユーザアカウント「postgres」が作成される。

3. ユーザアカウント「postgres」にパスワードを設定する。

```
$ sudo passwd postgres
新しいUNIXパスワードを入力してください:<<パスワードを入力>>
新しいUNIXパスワードを再入力してください:<<パスワードを入力>>
```

4. postgresql-server-dev-9.1をインストールする。

```
$ sudo apt-get install postgresql-server-dev-9.1
```

- ※ 「続行しますか [Y/n]?'と表示されたら Enter キーを押す。

5. GEOS(Geometry Engine Open Source)をインストールする。

```
$ sudo apt-get install libgeos-dev
```

- ※ 「続行しますか [Y/n]?'と表示されたら Enter キーを押す。

6. ソースコード保存用ディレクトリを作成し、そのディレクトリに移動する。

```
$ mkdir db_env_src
$ cd db_env_src
```

7. PROJ.4をインストールする。

```
$ wget http://download.osgeo.org/proj/proj-4.8.0.tar.gz
$ tar -xvzf proj-4.8.0.tar.gz
$ cd proj-4.8.0
$ ./configure
$ make
$ sudo make install
$ cd ..
$ rm -r proj-4.8.0
```

8. Flex をインストールする。

```
$ sudo apt-get install flex
```

9. xsltproc をインストールする。

```
$ sudo apt-get install xsltproc
```

10. libcunit1-ncurses-dev をインストールする。

```
$ sudo apt-get install libcunit1-ncurses-dev
```

※ 「続行しますか [Y/n]？」と表示されたら Enter キーを押す。

11. graphicsmagick-imagemagick-compat をインストールする。

```
$ sudo apt-get install graphicsmagick-imagemagick-compat
```

※ 「続行しますか [Y/n]？」と表示されたら Enter キーを押す。

12. dblatex をインストールする。

```
$ sudo apt-get install dblatex
```

※ 「続行しますか [Y/n]？」と表示されたら Enter キーを押す。

13. ldp-docbook-xsl をインストールする。

```
$ sudo apt-get install ldp-docbook-xsl
```

※ 「続行しますか [Y/n]？」と表示されたら Enter キーを押す。

14. GDAL(Geospatial Data Abstraction Library) をインストールする。

```
$ wget http://download.osgeo.org/gdal/gdal-1.9.2.tar.gz
$ tar -xvzf gdal-1.9.2.tar.gz
$ cd gdal-1.9.2
$ ./configure
$ make
$ sudo make install
$ cd ..
$ rm -r gdal-1.9.2
```

15. JSON-C をインストールする。

```
$ sudo apt-get install 'libjson0' 'libjson0-dev'
```

※ 「続行しますか [Y/n]?' と表示されたら Enter キーを押す。

16. PostGIS をインストールする。

```
$ wget http://download.osgeo.org/postgis/source/postgis-2.1.2.tar.gz
$ tar -xvzf postgis-2.1.2.tar.gz
$ cd postgis-2.1.2
$ ./configure
$ make
$ sudo make install
$ cd ..
$ rm -r postgis-2.1.2
```

17. BGL(Boost Graphic Library) をインストールする。

```
$ sudo apt-get install 'libboost.*1\.\49\.\0' 'libboost-doc'
'libboost.*-dev'
```

※ 「続行しますか [Y/n]?' と表示されたら Enter キーを押す。

18. CGAL をインストールする。

```
$ sudo apt-get install libcgall-dev
```

※ 「続行しますか [Y/n]?' と表示されたら Enter キーを押す。

19. cmake をインストールする。

```
$ sudo apt-get install cmake
```

※ 「続行しますか [Y/n]?' と表示されたら Enter キーを押す。

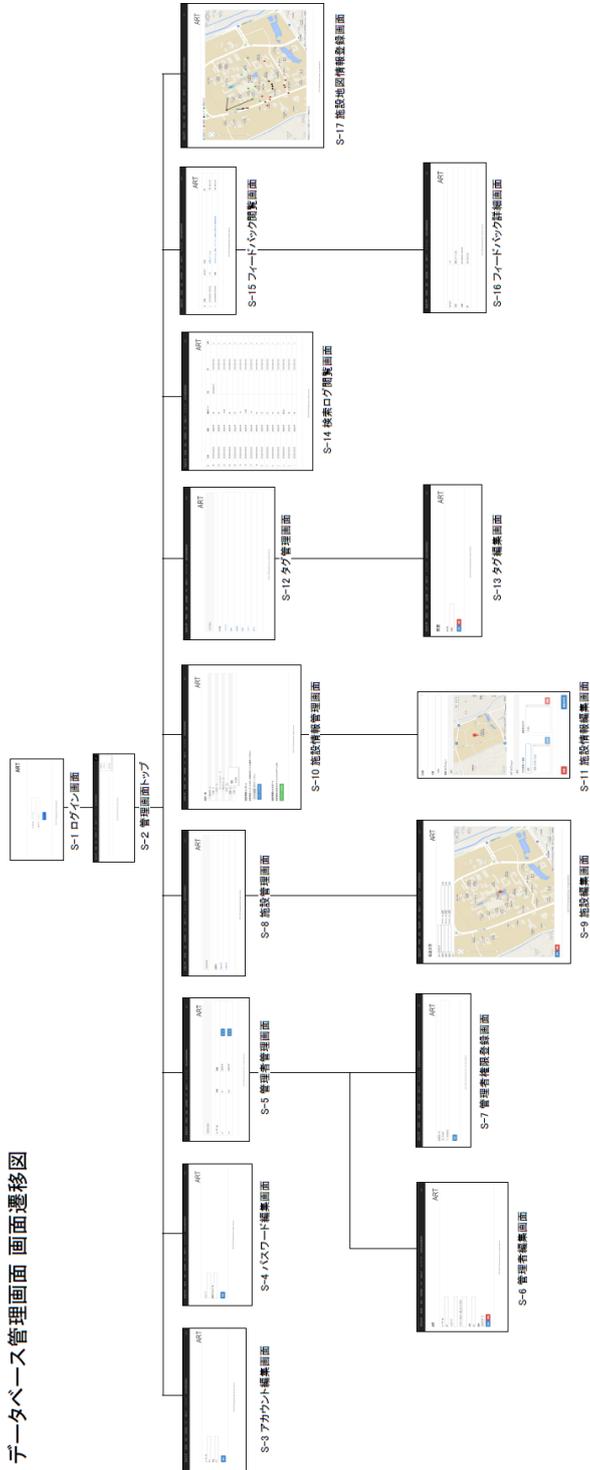
20. pgRouting をインストールする。

```
$ wget https://github.com/pgRouting/pgrouting/archive/v2.0.0.tar.gz
$ tar -xvzf v2.0.0.tar.gz
$ cd pgrouting-2.0.0
$ cmake
$ make
$ sudo make install
$ cd ..
$ rm -r pgrouting-2.0.0
```

付録B 管理画面設計資料

B.1 画面遷移図

データベース管理画面 画面遷移図

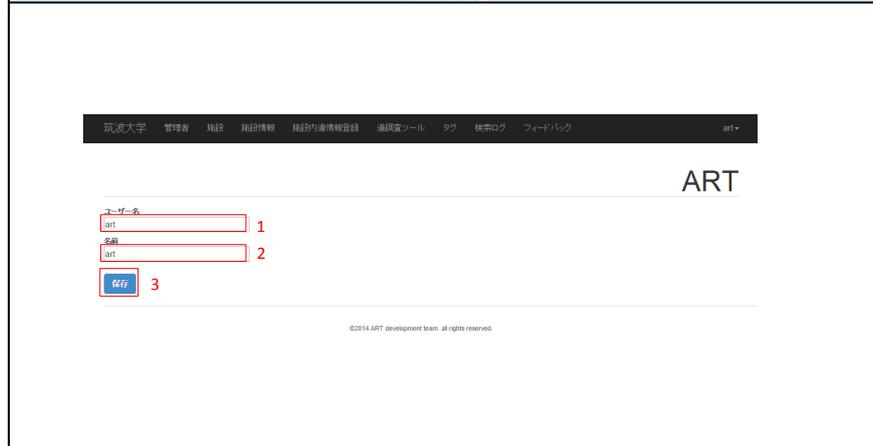


B.2 画面定義書

画面レイアウト定義					
共通情報	作成者	畑中裕太	更新者	畑中裕太	
	作成日付	2014/6/4	更新日付	2014/11/13	
書誌情報	画面ID	S-1			
	画面の名称	ログイン画面			
	概要	管理者がデータ管理システムにログインするための画面			
レイアウト図					
識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	ユーザ名	テキストボックス	空欄		ユーザ名を入力するためのテキストボックス
2	パスワード	テキストボックス	空欄	半角英数字、記号	パスワードを入力するためのテキストボックス
3	ログイン	ボタン			ログインを開始するボタン。画面ID「S-2」に遷移する。

画面レイアウト定義					
共通情報	作成者	畑中裕太	更新者	畑中裕太	
	作成日付	2014/6/4	更新日付	2014/11/13	
書誌情報	画面ID	S-2			
	画面の名称	管理画面トップ			
	概要	管理者がデータ管理システムにログインした時に表示されるトップ画面			
レイアウト図					
識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	施設名	リンク			管理者が編集できる施設名が表示される。画面ID「S-2」に遷移する。
2	管理者	リンク		「master」権限を持つ施設管理者のみ	画面ID「S-5」に遷移する。
3	施設	リンク		「master」権限を持つ施設管理者のみ	画面ID「S-8」に遷移する。
4	施設情報	リンク			画面ID「S-10」に遷移する。
5	施設内道情報	リンク			画面ID「S-17」に遷移する。
6	道調査ツール	リンク			画面ID「S-18」に遷移する。
7	タグ	リンク			画面ID「S-12」に遷移する。
8	検索ログ	リンク			画面ID「S-14」に遷移する。
9	フィードバック	リンク			画面ID「S-15」に遷移する。
10	アカウント名	コンテキストメニュー			ドロップダウンリストを表示するボタン。⑩⑪⑫のリンクがリスト化したコンテキストメニューが表示される。
11	アカウント	リンク			画面ID「S-3」に遷移する。
12	パスワード	リンク			画面ID「S-4」に遷移する。
13	ログアウト	リンク			画面ID「S-1」に遷移する。

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-3		
	画面の名称	アカウント編集画面		
	概要	ログインしているユーザがアカウント情報を編集するための画面		



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	ユーザー名	テキストボックス	新規:空欄 編集:編集するIDのユーザー名	半角英数字、記号255文字以内	ユーザー名を入力するためのテキストボックス
2	名前	テキストボックス	新規:空欄 編集:編集するIDの名前	全半角かな・漢字・英数字、記号255文字以内	名前を入力するためのテキストボックス
3	保存	ボタン			変更を保存するためのボタン

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-4		
	画面の名称	パスワード編集画面		
	概要	ログインしているユーザがパスワードを変更するための画面		

レイアウト図



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	パスワード	テキストボックス	空欄	半角英数字、記号255文字以内	パスワードを入力するためのテキストボックス
2	パスワード確認	テキストボックス	空欄	半角英数字、記号255文字以内	パスワードの再確認を行う。同じパスワードをもう一度入力するためのテキストボックス
3	保存	ボタン			保存するボタン

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-5		
	画面の名称	管理者管理画面		
	概要	データベース管理システムの管理者を管理するための画面。権限が「master」の管理者しか操作できない。		

レイアウト図



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	管理者新規登録	リンク			画面ID「S-6」に遷移する。
2	ユーザー名	リンク			登録されている管理者のユーザー名一覧が表示される。画面ID「S-6」に遷移する。
3	名前	ラベル			登録されている管理者の名前一覧が表示される。
4	キャンパス	ラベル			登録されている管理者の操作できる施設一覧が表示される。
5	管理者権限	リンクボタン			画面ID「S-7」に遷移する。

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-6		
	画面の名称	管理者編集画面		
	概要	管理者情報の新規登録や編集、削除を行う画面。		

レイアウト図

The screenshot shows a web form titled 'art' with a header 'ART'. The form contains several input fields and buttons. Red boxes and numbers 1 through 7 are overlaid on the form to identify specific components: 1 points to the 'ユーザー名' (User Name) input field containing 'art'; 2 points to the 'パスワード' (Password) input field; 3 points to the 'パスワードをもう一度入力してください' (Please re-enter your password) confirmation field; 4 points to the '名前' (Name) input field containing 'art'; 5 points to the '施設' (Facility) dropdown menu showing '筑波大学'; 6 points to the '保存' (Save) button; and 7 points to the '削除' (Delete) button. A copyright notice '©2014 ART development team. all rights reserved.' is visible at the bottom of the page.

識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	ユーザ名	テキストボックス			ユーザ名を入力するためのテキストボックス
2	パスワード	テキストボックス	空欄	半角英数字、記号255文字以内	パスワードを入力するためのテキストボックス
3	パスワード確認	テキストボックス	空欄	半角英数字、記号255文字以内	パスワードを確認するためのテキストボックス
4	名前	テキストボックス	新規: 空欄 編集: 編集するIDの名前	全半角かな・漢字・英数字、記号255文字以内	名前を入力するためのテキストボックス
5	施設	ドロップダウンリスト	IDの一番上の施設		施設を選択するためのドロップダウンリスト
6	保存	ボタン			新規登録、変更を保存するためのボタン。画面ID「S-5」に遷移する。
7	削除	ボタン			管理者を削除するためのボタン。画面ID「S-5」の識別ID②から遷移した時のみ表示される。画面ID「S-5」に遷移する。

画面レイアウト定義					
共通情報	作成者	畑中裕太	更新者	畑中裕太	
	作成日付	2014/6/4	更新日付	2014/11/13	
書誌情報	画面ID	S-7			
	画面の名称	管理者権限登録画面			
	概要	管理者の権限を登録するための画面			
レイアウト図					
識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	管理者権限	チェックボックス			権限を登録するためのチェックボックス
2	保存	ボタン			管理者権限情報の保存を行うボタン。画面ID「S-5」に遷移する

画面レイアウト定義					
共通情報	作成者	畑中裕太	更新者	畑中裕太	
	作成日付	2014/6/4	更新日付	2014/11/13	
書誌情報	画面ID	S-8			
	画面の名称	施設管理画面			
	概要	データベースに登録している施設の管理を行う画面。管理者権限が「master」の管理者しか操作できない			
レイアウト図					
識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	施設新規登録	リンク			画面ID「S-9」に遷移する
2	施設名	リンク			データベースに登録している施設一覧が表示される。画面ID「S-9」に遷移する

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-9		
	画面の名称	施設編集画面		
	概要	施設の新規登録、編集、削除を行うための画面レイアウト図		

筑波大学 ART

1 名前:筑波大学
階層:0
2 階層:1
階層:2
4 追加
6

カテゴリ名:建物 5
カテゴリ名:教室 3
カテゴリ名:授業 削除

8 9

識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	施設名	テキストボックス	新規:空欄 編集:編集するIDの施設名	全半角かな・漢字・英数字、記号 255文字以内	施設名を入力するためのテキストボックス
2	階層	テキストボックス	0から連番	編集不可	階層レベルが表示される。入力できず、一番上の階層が0で下の階層になるにつれ番号が上
3	カテゴリ名	テキストボックス	カテゴリ	全半角かな・漢字・英数字、記号 255文字以内	カテゴリ名を入力する
4	カテゴリ追加	リンク			階層とカテゴリ名のテキストボックスを追加する
5	カテゴリ削除	リンク			選択した行の階層とカテゴリ名を削除する
6	地図	Google Maps			GoogleMapを表示する
7	座標	マーカー			GoogleMap上に置くこと、置いた地点の位置座標を取得する
8	保存	ボタン			新規登録、保存を登録するためのボタン。画面ID「S-8」に遷移
9	削除	ボタン			施設を削除するためのボタン。画面ID「S-8」の識別ID②から遷移した時のみ表示される。画面ID「S-8」に遷移する。

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID			S-10
	画面の名称	施設情報管理画面		
	概要	ログインしている管理者が操作可能な施設の施設情報を管理するための画面。施設情報の階層構造の変更、施設情報のインポート・エクスポートを行うことが可能		

レイアウト図

ART

1 施設一覧

2 閉

3

4 編集

5 子を追加

施設情報インポート

施設情報をインポートします。CSV形式のファイルを選択してください。

6 ファイルを選択 | 選択されていません

7 インポート(CSV形式)

施設情報エクスポート

施設情報をCSV形式のファイルでエクスポートします。

8 エクスポート(CSV形式)

©2014 ART development team. all rights reserved.

識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	施設情報一覧	リスト			施設情報の一覧がツリー形式で表示されるリスト。ドラッグで親子関係を変更できる
2	閉	ボタン			下の階層の施設情報の表示・非表示をするためのボタン
3	メニュー	ボタン			コンテキストメニューを表示するためのボタン
4	編集	リンク			画面ID「S-11」に遷移する
5	子を追加	リンク			画面ID「S-11」に遷移する
6	ファイル選択	ボタン			インポートするためのファイルを選択するためのボタン
7	インポート	ボタン			インポートするためのボタン
8	エクスポート	ボタン			エクスポートするためのボタン

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
	画面ID			S-11
書誌情報	画面の名称	施設情報編集画面		
	概要	施設情報の新規登録、編集、削除を行うことが可能。タグの新規登録、施設情報との関連付けも可能である。		

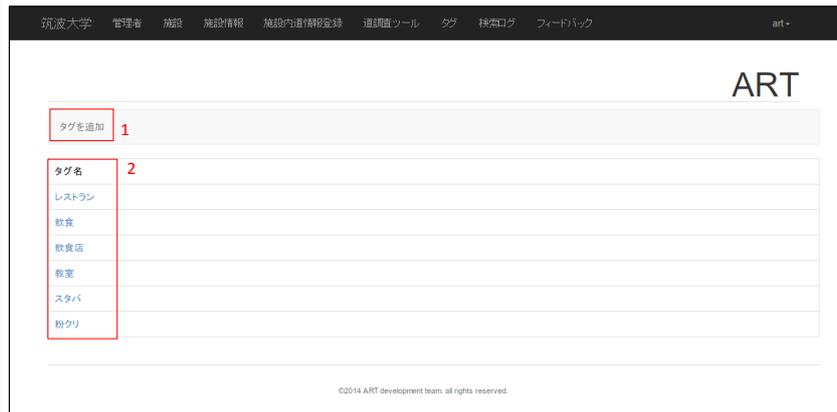
レイアウト図



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	施設情報名	テキストボックス	新規: 空欄 編集: 編集するIDの施設情報	全半角かな・漢字・英数字、記号255文字以内	施設情報名を入力するためのテキストボックス
2	場所(オプション)	リンク			位置座標を登録するためのリンク。地図とマーカーの表示・非表示
2	地図	地図			GoogleMapを表示する
2	マーカー	マーカー			GoogleMap上に置く、置いた地点の位置座標を取得する
2	タグ(オプション)	リンク			タグを編集するためのリンク。タグ登録のためのテキストボックス・リンク・ボタンやタグ削除ボタンの表示・非表示を行う。画面ID「S-10」の識別ID④から遷移した時のみ
2	検索	テキストボックス	空欄	全半角かな・漢字・英数字255文字以内	タグ名を入力するためのテキストボックス
2	タグ名	リスト			⑥で入力されたタグ名に該当するタグ名を一覧で表示する。該当するタグ名がない場合、「〇〇」を作成して追加と表示する
2	タグ追加	ボタン			タグの追加をするためのボタン。⑦を選択しなければ、押下することができない。
2	登録済タグ	リスト			登録済タグの一覧が表示される。
2	タグ削除	ボタン			タグの削除をするためのボタン。⑧を選択しなければ押下することができない。
2	施設情報削除	ボタン			施設情報の削除をするためのボタン。画面ID「S-10」の識別ID④から遷移した時のみ利用できる。画面ID「S-10」に遷移する。
2	編集保存	ボタン			施設情報の保存をするためのボタン。画面ID「S-10」に遷移する。

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-12		
	画面の名称	タグ管理画面		
	概要	ログインしている管理者が操作可能な施設に登録されているタグを管理する画面		

レイアウト図



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	タグ登録	リスト		「master」権限を持つ施設管理者のみリンク表示	画面ID「S-13」に遷移する。
2	タグ名	リンク			データベースに登録しているタグ一覧が表示される。画面ID「S-13」に遷移する。

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-13		
	画面の名称	タグ編集画面		
	概要	タグの新規登録、編集、削除を行うことが可能な画面 レイアウト図		



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	タグ名	テキストボックス	新規: 空欄 編集: 編集するIDのタグ名	全半角かな・漢字・英数字255文字以内	タグ名を入力するためのテキストボックス
2	保存	ボタン			新規登録、変更を保存するためのボタン。画面ID「S-12」に遷移
3	削除	ボタン			タグを削除するためのボタン。画面ID「S-12」の識別ID②から遷移した時のみ表示される。画面ID「S-12」に遷移する。

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-14		
	画面の名称	ログ閲覧画面		
	概要	クライアント側で検索された検索ログの一覧を確認することが可能な画面		

筑波大学 管理画 施設 施設情報 施設予約情報登録 施設予約ツール タグ 検索ログ フォードバック art

ART

- actionで絞込検索: 検索 経路検索 目的地絞
- 日付で絞込検索: 2014年 11月 12日 から 14年 11月 12日 まで
- 個体識別番号で絞込検索:
- 絞り込み

#	日時	action	検索ワード	目的地	経路の種類	OS	個体識別番号	API
140	14/11/12 12:51:27	経路検索		1A棟	推奨経路		123	2
137	14/11/10 14:19:38	検索	1				192.168.117.1	2
136	14/11/10 14:18:28	検索	あ				192.168.117.1	2
135	14/11/05 11:48:39	経路検索		1A棟			192.168.117.1	2
134	14/11/05 11:38:25	検索	あ				192.168.117.1	2
133	14/11/05 11:34:32	検索	あ				192.168.117.1	2
132	14/11/05 11:32:25	検索	スタバ				192.168.117.1	2
131	14/11/05 11:32:20	検索	1				192.168.117.1	2
130	14/11/05 11:21:07	検索	あ				192.168.117.1	2
129	14/11/05 11:19:20	検索	あ				192.168.117.1	2
128	14/10/22 09:42:06	経路検索		1A棟			192.168.117.1	2
127	14/10/22 09:39:00	検索	1				192.168.117.1	2
126	14/10/22 09:35:49	経路検索		1A棟			192.168.117.1	2
125	14/10/22 09:34:33	目的地絞定		1A棟			192.168.117.1	2
124	14/10/22 09:33:43	検索	レストラン				192.168.117.1	2
123	14/10/21 18:57:51	検索	2				192.168.117.1	2

識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	actionの絞込検索	チェックボックス			どのactionで絞込検索をするか選択するためのチェックボックス
2	日付の絞込検索	ドロップダウンリスト			いつからいつまでで絞込検索をするか選択するためのドロップダウンリスト
3	個体識別番号の絞込検索	テキストボックス			どの個体識別番号で絞込検索をするか選択するためのテキストボックス
4	絞込検索ボタン	ボタン			
5	ID	リスト			IDを表示するためのリスト
6	日時	リスト			日時を表示するためのリスト
7	APIの種類	リスト			APIの種類を表示するためのリスト
8	検索ワード	リスト			検索ワードを表示するためのリスト
9	目的地	リスト			設定された目的地を表示するためのリスト
10	経路の種類	リスト			経路の種類を表示するためのリスト
11	OS	リスト			OSを表示するためのリスト
12	個体識別番号	リスト			個体識別番号を表示するためのリスト
13	API	リスト			APIを表示するためのリスト

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-15		
	画面の名称	フィードバック閲覧画面		
	概要	クライアント側から送信されたフィードバックを閲覧するための画面		

レイアウト図



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	ID	リスト			IDを表示するためのリスト
2	日時	リスト			日時を表示するためのリスト
3	カテゴリ	リスト			カテゴリを表示するためのリスト
4	本文	リンク			本文の一部を表示するリンク。 画面ID「S-16」に遷移する
5	IPアドレス	リスト			IPアドレスを表示するためのリスト

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID			S-16
	画面の名称	フィードバック詳細画面		
	概要	フィードバックの内容の詳細を確認するための画面		



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	日時	リスト			日時を表示するためのリスト
2	カテゴリ	リスト			カテゴリを表示するためのリスト
3	本文	リスト			本文を表示するためのリスト
4	IPアドレス	リスト			IPアドレスを表示するためのリスト

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-17		
	画面の名称	施設内道情報管理画面		
	概要	施設内の経路情報を登録するための画面		



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	地図	地図			Google Mapを表示する
2	ノード種類	ラジオボタン			ノードの種類を選択するためのラジオボタン
3	ノード	マーカー			GoogleMap上を押下すると、押下した地点の位置座標を取得する。右クリックをすると、ノードの「移動」、「削除」を行うことができる。
4	エッジ	線			ノードとノードを結ぶ線。右クリックをすると、エッジの「属性の登録」、「カーブノードの追加」、「距離の表示」を行う

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID	S-18		
	画面の名称	道調査ツール登録画面		
	概要	道調査ツールで収集したGPSのログを登録するための画面		



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	ファイル選択	ボタン			道調査ツールで収集したGPSのログファイルを選択するためのボタン。
2	ファイルアップロード	ボタン			選択したファイルをアップロードするボタン。画面ID「S-19」に遷移する。

画面レイアウト定義				
共通情報	作成者	畑中裕太	更新者	畑中裕太
	作成日付	2014/6/4	更新日付	2014/11/13
書誌情報	画面ID			S-19
	画面の名称	道調査ツール編集画面		
	概要	道調査ツールで収集したGPSのログを編集し、道情報を自動登録するための画面		



識別ID	論理項目名	画面部品の種類	初期値	制限事項	画面部品の説明
1	画像編集	画像			画像の任意の位置をクリックすると、クリックした位置が太線で囲まれている場合、黒く塗りつぶされる。
2	道情報自動登録	ボタン			道情報を自動で登録する。画面ID「S-17」に遷移する。

付録C 割増率算出過程で得られたデータ

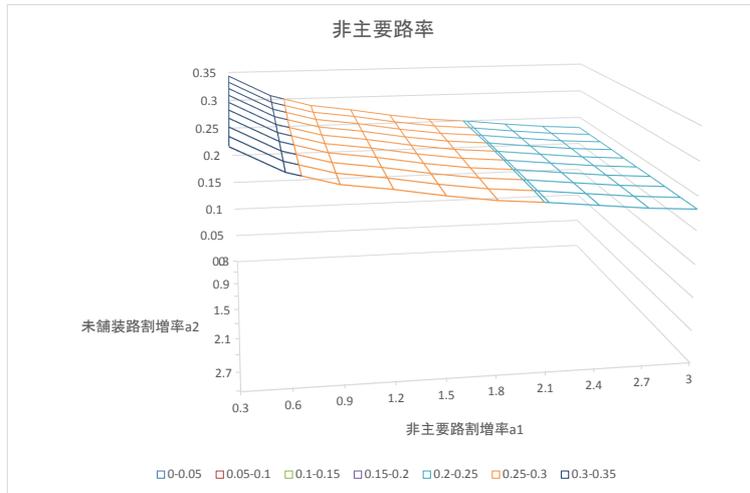
C.1 a_1, a_2 を 0.3 刻みで変化させた結果

非主要路率 (割増率0.3刻み)

		未舗装路割増率 a_2									
		0.3	0.6	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3
非 主 要 路 率	0.3	0.343521	0.343004	0.342958	0.342771	0.342733	0.342705	0.342689	0.342631	0.342592	0.342612
	0.6	0.306781	0.30618	0.306083	0.305838	0.305694	0.305795	0.305774	0.305722	0.305704	0.305683
	0.9	0.286811	0.286603	0.28635	0.286331	0.28621	0.286192	0.286163	0.286268	0.286276	0.286291
	1.2	0.276918	0.277059	0.276925	0.276859	0.276736	0.276719	0.276623	0.276588	0.276585	0.276731
	1.5	0.264953	0.26514	0.264976	0.264987	0.264902	0.264867	0.264851	0.264831	0.264826	0.264848
	1.8	0.255106	0.255464	0.255388	0.255425	0.255371	0.255335	0.255277	0.25529	0.25529	0.255312
	2.1	0.249419	0.2494	0.249497	0.249625	0.249593	0.249631	0.249596	0.24958	0.249581	0.249455
	2.4	0.241569	0.241727	0.242019	0.242007	0.242348	0.242324	0.242415	0.242426	0.242446	0.242414
	2.7	0.235084	0.235124	0.235038	0.235187	0.235316	0.235526	0.235515	0.235534	0.235547	0.235288
	3	0.230427	0.230594	0.230645	0.230794	0.23053	0.230537	0.230698	0.230792	0.230759	0.230812

● 値の差 (黄色表示: 差の大きい部分 (0.09以上))

		未舗装路割増率 a_2									
		0.3	0.6	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3
非 主 要 路 率	0.3⇒0.6	-0.03674	-0.03682	-0.03688	-0.03693	-0.03704	-0.03691	-0.03692	-0.03691	-0.03689	-0.03693
	0.6⇒0.9	-0.01997	-0.01958	-0.01973	-0.01951	-0.01948	-0.0196	-0.01961	-0.01945	-0.01943	-0.01939
	0.9⇒1.2	-0.00989	-0.00954	-0.00942	-0.00947	-0.00947	-0.00947	-0.00954	-0.00968	-0.00969	-0.00956
	1.2⇒1.5	-0.01197	-0.01192	-0.01195	-0.01187	-0.01183	-0.01185	-0.01177	-0.01176	-0.01176	-0.01188
	1.5⇒1.8	-0.00985	-0.00968	-0.00959	-0.00956	-0.00953	-0.00953	-0.00957	-0.00954	-0.00954	-0.00954
	1.8⇒2.1	-0.00569	-0.00606	-0.00589	-0.0058	-0.00578	-0.0057	-0.00568	-0.00571	-0.00571	-0.00586
	2.1⇒2.4	-0.00785	-0.00767	-0.00748	-0.00762	-0.00725	-0.00731	-0.00718	-0.00715	-0.00714	-0.00704
	2.4⇒2.7	-0.00649	-0.0066	-0.00698	-0.00682	-0.00703	-0.0068	-0.0069	-0.00689	-0.0069	-0.00713
2.7⇒3.0	-0.00466	-0.00453	-0.00439	-0.00439	-0.00479	-0.00499	-0.00482	-0.00474	-0.00479	-0.00448	



未舗装路率(割増率0.3刻み)

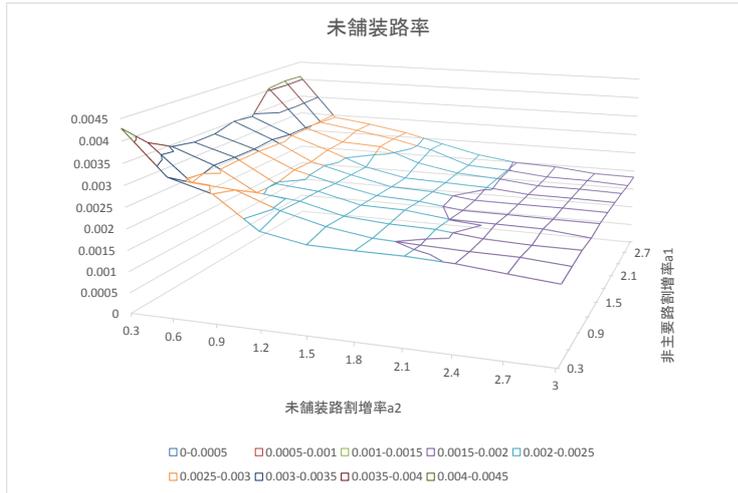
		未舗装路割増率a2									
		0.3	0.6	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3
非主要路割増率a1	0.3	0.004285	0.0032729	0.0029869	0.0022558	0.0020623	0.0020426	0.0020369	0.0019896	0.0018899	0.001796068
	0.6	0.003758	0.0029085	0.0028288	0.0024436	0.002182	0.0020227	0.0019751	0.0019413	0.0019273	0.001843625
	0.9	0.003459	0.0030858	0.0025172	0.0024042	0.0021455	0.0020646	0.0020407	0.0019299	0.0018902	0.001889012
	1.2	0.003358	0.0030587	0.0024854	0.0023892	0.0021435	0.0021206	0.0019977	0.0019869	0.0019732	0.001866367
	1.5	0.003365	0.0030517	0.0025993	0.0023351	0.0021229	0.0020721	0.0019412	0.0019372	0.0019233	0.001919727
	1.8	0.003501	0.0030878	0.0025857	0.0023224	0.0020972	0.0020601	0.0019346	0.0019289	0.0019289	0.001914473
	2.1	0.003443	0.0030203	0.0026759	0.0023003	0.00209	0.0020523	0.0020134	0.001923	0.0019177	0.001900221
	2.4	0.004047	0.0030153	0.0026879	0.0026359	0.0022658	0.0020731	0.0020065	0.0019024	0.0019015	0.001879193
	2.7	0.004101	0.0029777	0.0028167	0.0026437	0.0024161	0.0020557	0.0020035	0.0019022	0.0018886	0.001839826
	3	0.004081	0.0029382	0.0027995	0.0026216	0.0023743	0.0021312	0.001987	0.0019529	0.0018446	0.001830951

●値の差(黄色表示:差の大きい部分(0.0001以上))

		未舗装路割増率a2									
		0.3	0.6	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3
非主要路割増率a1	0.3⇒0.6	-0.00053	-0.000364	-0.000158	0.0001877	0.0001197	-1.99E-05	-6.18E-05	-4.83E-05	3.735E-05	4.75575E-05
	0.6⇒0.9	-0.0003	0.0001773	-0.000312	-3.94E-05	-3.65E-05	4.186E-05	6.57E-05	-1.13E-05	-3.71E-05	4.53868E-05
	0.9⇒1.2	-0.0001	-2.71E-05	-3.18E-05	-1.49E-05	-1.96E-06	5.6E-05	-4.29E-05	5.691E-05	8.306E-05	-2.26451E-05
	1.2⇒1.5	6.43E-06	-7E-06	0.0001139	-5.41E-05	-2.06E-05	-4.85E-05	-5.65E-05	-4.97E-05	-5E-05	5.33605E-05
	1.5⇒1.8	0.000137	3.616E-05	-1.37E-05	-1.28E-05	-2.57E-05	-1.2E-05	-6.59E-06	-8.28E-06	5.645E-06	-5.25479E-06
	1.8⇒2.1	-5.8E-05	-6.75E-05	9.027E-05	-2.21E-05	-7.21E-06	-7.84E-06	7.875E-05	-5.89E-06	-1.12E-05	-1.42515E-05
	2.1⇒2.4	0.000604	-5.02E-06	1.2E-05	0.0003356	0.0001758	2.088E-05	-6.87E-06	-2.06E-05	-1.62E-05	-2.10281E-05
	2.4⇒2.7	5.39E-05	-3.76E-05	0.0001288	7.84E-06	0.0001503	-1.74E-05	-2.98E-06	-1.94E-07	-1.29E-05	-3.93668E-05
	2.7⇒3.0	-2E-05	-3.95E-05	-1.72E-05	-2.22E-05	-4.18E-05	7.542E-05	-1.66E-05	5.067E-05	-4.4E-05	-8.87545E-06

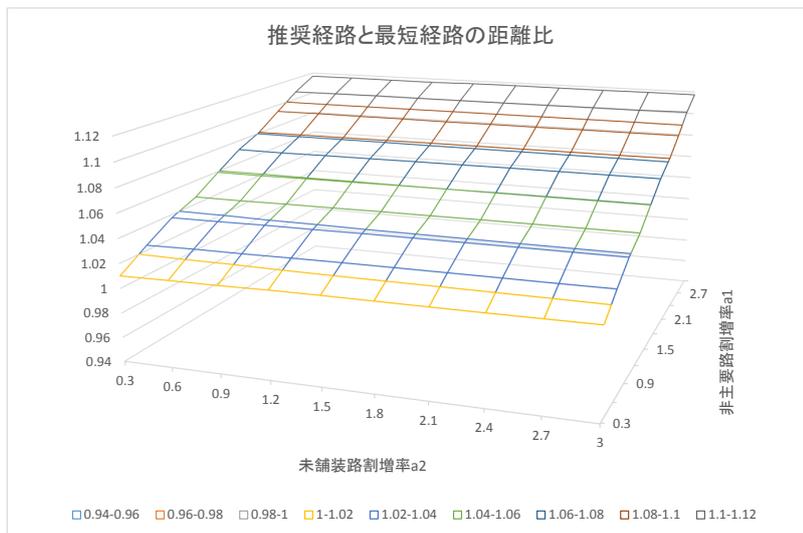
●値の差(黄色表示:差の大きい部分(0.0001以上))

		未舗装路割増率a2									
		0.3⇒0.6	0.6⇒0.9	0.9⇒1.2	1.2⇒1.5	1.5⇒1.8	1.8⇒2.1	2.1⇒2.4	2.4⇒2.7	2.7⇒3	
非主要路割増率a1	0.3	-0.00101	-0.000286	-0.000731	-0.000193	-1.97E-05	-5.73E-06	-4.73E-05	-9.96E-05	-9.39E-05	
	0.6	-0.00085	-7.98E-05	-0.000385	-0.000262	-0.000159	-4.76E-05	-3.38E-05	-1.4E-05	-8.36E-05	
	0.9	-0.00037	-0.000569	-0.000113	-0.000259	-8.09E-05	-2.39E-05	-0.000111	-3.98E-05	-1.16E-06	
	1.2	-0.0003	-0.000573	-9.62E-05	-0.000246	-2.3E-05	-0.000123	-1.09E-05	-1.36E-05	-0.000107	
	1.5	-0.00031	-0.000452	-0.000264	-0.000212	-5.09E-05	-0.000131	-4.03E-06	-1.39E-05	-3.54E-06	
	1.8	-0.00041	-0.000502	-0.000263	-0.000225	-3.71E-05	-0.000125	-5.73E-06	0	-1.44E-05	
	2.1	-0.00042	-0.000344	-0.000376	-0.00021	-3.77E-05	-3.89E-05	-9.04E-05	-5.36E-06	-1.74E-05	
	2.4	-0.00103	-0.000327	-5.2E-05	-0.00037	-0.000193	-6.66E-05	-0.000104	-9.21E-07	-2.23E-05	
	2.7	-0.00112	-0.000161	-0.000173	-0.000228	-0.00036	-5.22E-05	-0.000101	-1.36E-05	-4.88E-05	
	3	-0.00114	-0.000139	-0.000178	-0.000247	-0.000243	-0.000144	-3.41E-05	-0.000108	-1.37E-05	



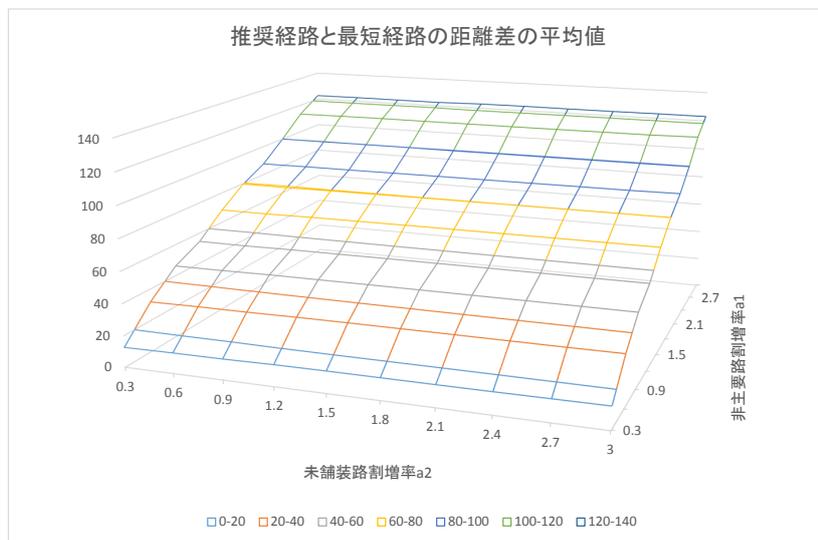
推奨経路と最短経路の距離比(割増率0.3刻み)

		未舗装路割増率a2									
		0.3	0.6	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3
非主要路割増率a1	0.3	1.010019	1.010533	1.010722	1.011441	1.01168	1.011714	1.011727	1.011828	1.012077	1.012317
	0.6	1.02423	1.024843	1.024937	1.025373	1.02568	1.02582	1.025899	1.025976	1.026009	1.026226
	0.9	1.037363	1.037589	1.03808	1.038186	1.038491	1.038594	1.038644	1.038736	1.038787	1.038779
	1.2	1.045982	1.045944	1.046349	1.046476	1.046779	1.046815	1.047026	1.047072	1.047089	1.047157
	1.5	1.059216	1.059136	1.059527	1.05965	1.059904	1.059973	1.060116	1.060141	1.060159	1.060137
	1.8	1.071956	1.071681	1.07202	1.07211	1.072346	1.072406	1.072607	1.072596	1.072596	1.072577
	2.1	1.080695	1.0809	1.080903	1.08099	1.081201	1.081183	1.081246	1.081378	1.081384	1.081581
	2.4	1.093914	1.094062	1.093843	1.09388	1.093603	1.093757	1.09368	1.093779	1.093743	1.093802
	2.7	1.106792	1.107089	1.107276	1.107152	1.107128	1.107092	1.107148	1.107265	1.107265	1.107827
	3	1.116789	1.116951	1.116781	1.116672	1.117369	1.117545	1.117364	1.117276	1.117504	1.117415



推奨経路と最短経路の距離差の平均値(割増率0.3刻み)

		未舗装路割増率a2									
		0.3	0.6	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3
非主要路割増率a1	0.3	12.71097	13.03901	13.14421	13.61169	13.75655	13.76704	13.77488	13.82197	13.97032	14.11513
	0.6	31.79474	32.16154	32.19872	32.55357	32.68766	32.75266	32.80064	32.83414	32.84101	32.9755
	0.9	46.2758	46.39731	46.79896	46.86489	46.99475	47.03179	47.06212	47.11365	47.1284	47.12397
	1.2	54.43866	54.36983	54.70362	54.78574	54.92508	54.95203	55.0443	55.07013	55.06681	55.12213
	1.5	68.36449	68.30485	68.63313	68.7083	68.82551	68.86252	68.87857	68.90286	68.90281	68.88468
	1.8	79.60975	79.43672	79.72985	79.78133	79.90097	79.92973	80.02106	80.01191	80.01191	79.98537
	2.1	87.63547	87.81983	87.81991	87.88704	87.98576	87.97253	87.99868	88.07208	88.07943	88.37489
	2.4	99.95486	100.1292	99.95139	99.96569	99.78088	99.83164	99.77353	99.81567	99.79232	99.83959
	2.7	113.3833	113.5763	113.6794	113.5535	113.5525	113.484	113.5022	113.5619	113.5622	114.0591
	3	122.6556	122.7875	122.6518	122.5707	123.1641	123.2405	123.0946	123.061	123.1827	123.1291



推奨経路と最短経路の距離差の標準偏差(割増率0.3刻み)

		未舗装路割増率a2									
		0.3	0.6	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3
非主要路割増率a1	0.3	21.50778	21.76038	21.8797	22.53024	22.67235	22.68806	22.69216	22.75985	23.01367	23.32179
	0.6	55.04176	55.22673	55.26035	55.46065	55.43331	55.44223	55.44921	55.45923	55.46343	55.54768
	0.9	72.04749	72.04729	72.19303	72.2052	72.20855	72.21641	72.21462	72.21765	72.19861	72.18768
	1.2	81.95439	81.8836	81.93488	81.9606	82.0056	82.0464	82.06908	82.08186	82.05683	82.04446
	1.5	98.69779	98.73047	98.74038	98.73492	98.74696	98.73293	98.68434	98.68877	98.68104	98.63493
	1.8	111.1476	111.3468	111.4531	111.4108	111.3998	111.3935	111.4586	111.4344	111.4344	111.3795
	2.1	122.9972	123.1642	123.1431	123.218	123.2061	123.1835	123.1787	123.2389	123.2478	123.6233
	2.4	139.0836	139.3245	139.3735	139.3929	138.9775	138.84	138.7541	138.7167	138.677	138.7229
	2.7	156.0278	156.2458	156.2943	156.3028	156.3021	156.2064	156.2194	156.2576	156.2569	157.0799
	3	167.9495	168.2331	167.9834	168.0192	168.8042	168.7569	168.652	168.6351	168.7512	168.6601

推奨経路と最短経路の距離差の平均値+標準偏差(割増率0.3刻み)

		未舗装路割増率a2									
		0.3	0.6	0.9	1.2	1.5	1.8	2.1	2.4	2.7	3
非主要路割増率a1	0.3	34.21876	34.7994	35.02391	36.14193	36.4289	36.45511	36.46704	36.58182	36.98399	37.43692
	0.6	86.8365	87.38827	87.45907	88.01422	88.12097	88.19489	88.24985	88.29338	88.30444	88.52318
	0.9	118.3233	118.4446	118.992	119.0701	119.2033	119.2482	119.2767	119.3313	119.327	119.3116
	1.2	136.393	136.2534	136.6385	136.7463	136.9307	136.9984	137.1134	137.152	137.1236	137.1666
	1.5	167.0623	167.0353	167.3735	167.4432	167.5725	167.5955	167.5629	167.5916	167.5838	167.5196
	1.8	190.7574	190.7835	191.1829	191.1921	191.3008	191.3233	191.4797	191.4463	191.4463	191.3649
	2.1	210.6327	210.9841	210.963	211.105	211.1918	211.1561	211.1774	211.311	211.3272	211.9982
	2.4	239.0384	239.4536	239.3249	239.3586	238.7584	238.6717	238.5276	238.5324	238.4693	238.5625
	2.7	269.4112	269.8222	269.9737	269.8563	269.8546	269.6904	269.7216	269.8195	269.8192	271.139
	3	290.6051	291.0206	290.6352	290.5899	291.9684	291.9974	291.7466	291.6961	291.934	291.7891

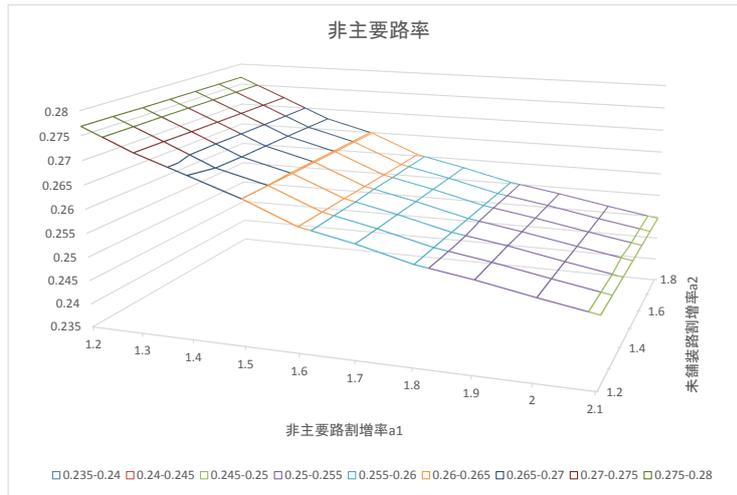
C.2 a_1, a_2 を 0.1 刻みで変化させた結果

非主要路率 (割増率0.1刻み)

		未舗装路割増率 a_2						
		1.2	1.3	1.4	1.5	1.6	1.7	1.8
非主要路割増率 a_1	1.2	0.276859	0.276832	0.276741	0.276736	0.276727	0.276717	0.276719
	1.3	0.272339	0.27234	0.272268	0.272275	0.272264	0.272264	0.272239
	1.4	0.268686	0.267701	0.26762	0.267678	0.267703	0.267692	0.267669
	1.5	0.264987	0.264985	0.264902	0.264902	0.264901	0.264898	0.264867
	1.6	0.260524	0.260509	0.260508	0.260431	0.260429	0.260389	0.260381
	1.7	0.258269	0.25816	0.258125	0.258057	0.258055	0.258012	0.258002
	1.8	0.255425	0.255452	0.255433	0.255371	0.25535	0.255348	0.255335
	1.9	0.253688	0.25369	0.253718	0.253546	0.253528	0.253526	0.253526
	2	0.251569	0.251604	0.251608	0.251513	0.251498	0.251492	0.251491
	2.1	0.249625	0.249632	0.249633	0.249593	0.249638	0.249633	0.249631

● 値の差 (黄色表示: 差の大きい部分 (0.003以上))

		未舗装路割増率 a_2						
		1.2	1.3	1.4	1.5	1.6	1.7	1.8
非主要路割増率 a_1	1.2⇒1.3	-0.00452	-0.00449	-0.00447	-0.00446	-0.00446	-0.00445	-0.00448
	1.3⇒1.4	-0.00365	-0.00464	-0.00465	-0.0046	-0.00456	-0.00457	-0.00457
	1.4⇒1.5	-0.0037	-0.00272	-0.00272	-0.00278	-0.0028	-0.00279	-0.0028
	1.5⇒1.6	-0.00446	-0.00448	-0.00439	-0.00447	-0.00447	-0.00451	-0.00449
	1.6⇒1.7	-0.00226	-0.00235	-0.00238	-0.00237	-0.00237	-0.00238	-0.00238
	1.7⇒1.8	-0.00284	-0.00271	-0.00269	-0.00269	-0.0027	-0.00266	-0.00267
	1.8⇒1.9	-0.00174	-0.00176	-0.00171	-0.00183	-0.00182	-0.00182	-0.00181
	1.9⇒2.0	-0.00212	-0.00209	-0.00211	-0.00203	-0.00203	-0.00203	-0.00203
	2.0⇒2.1	-0.00194	-0.00197	-0.00197	-0.00192	-0.00186	-0.00186	-0.00186



未舗装路率(割増率0.1刻み)

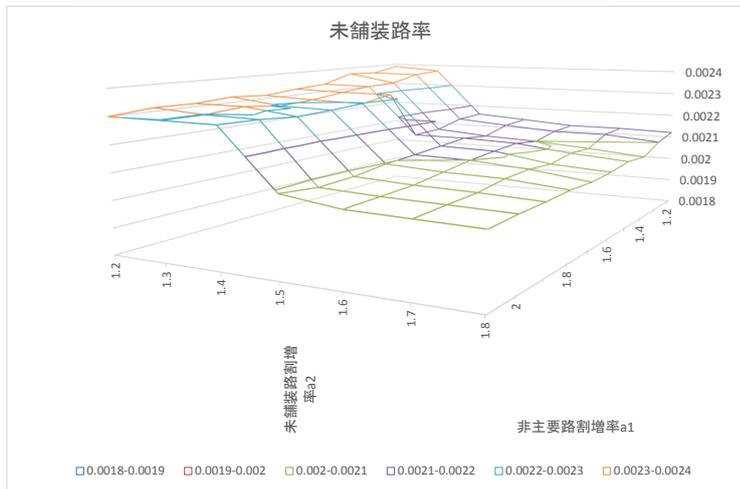
		未舗装路割増率a2						
		1.2	1.3	1.4	1.5	1.6	1.7	1.8
非主要路割増率a1	1.2	0.002389	0.002372	0.002159	0.002144	0.002127	0.002126	0.002121
	1.3	0.002366	0.002364	0.002154	0.002137	0.00212	0.00212	0.002101
	1.4	0.002372	0.002336	0.002127	0.002109	0.002099	0.002083	0.002063
	1.5	0.002335	0.002332	0.002123	0.002123	0.002115	0.002089	0.002072
	1.6	0.002326	0.00232	0.002319	0.002114	0.002109	0.002074	0.00206
	1.7	0.002313	0.002304	0.002298	0.002105	0.0021	0.002083	0.002054
	1.8	0.002322	0.002296	0.00229	0.002097	0.002085	0.00208	0.00206
	1.9	0.002312	0.002311	0.002285	0.002081	0.00207	0.002065	0.002056
	2	0.002314	0.002299	0.002294	0.002076	0.002066	0.002062	0.002057
	2.1	0.0023	0.002299	0.002297	0.00209	0.002062	0.002057	0.002052

●値の差(黄色表示:差の大きい部分(0.0001以上))

		未舗装路割増率a2						
		1.2	1.3	1.4	1.5	1.6	1.7	1.8
非主要路割増率a1	1.2⇒1.3	-2.4E-05	-8.3E-06	-4.8E-06	-6.8E-06	-6E-06	-6E-06	-2E-05
	1.3⇒1.4	6.85E-06	-2.8E-05	-2.8E-05	-2.8E-05	-2.1E-05	-3.7E-05	-3.8E-05
	1.4⇒1.5	-3.7E-05	-4.1E-06	-3.5E-06	1.38E-05	1.6E-05	6.1E-06	9.32E-06
	1.5⇒1.6	-8.8E-06	-1.2E-05	0.000196	-9E-06	-6.5E-06	-1.5E-05	-1.2E-05
	1.6⇒1.7	-1.3E-05	-1.7E-05	-2.1E-05	-9.1E-06	-9.1E-06	9.58E-06	-5.3E-06
	1.7⇒1.8	9.4E-06	-7.5E-06	-8.1E-06	-7.6E-06	-1.5E-05	-3.5E-06	5.67E-06
	1.8⇒1.9	-1E-05	1.49E-05	-5.6E-06	-1.6E-05	-1.5E-05	-1.5E-05	-3.7E-06
	1.9⇒2.0	1.77E-06	-1.2E-05	9.65E-06	-5.3E-06	-4.1E-06	-3.6E-06	2.16E-07
	2.0⇒2.1	-1.4E-05	1.12E-07	2.92E-06	1.43E-05	-4E-06	-4.3E-06	-4.3E-06

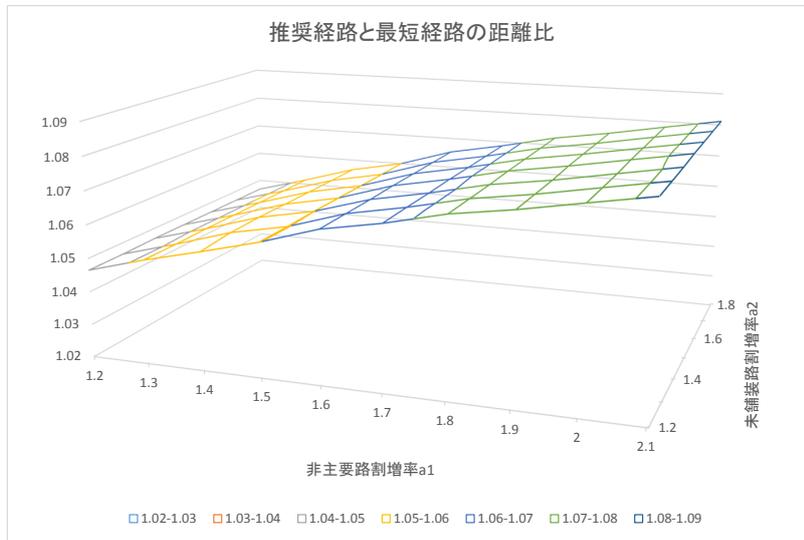
●値の差(黄色表示:差の大きい部分(0.0001以上))

		未舗装路割増率a2					
		1.2⇒1.3	1.3⇒1.4	1.4⇒1.5	1.5⇒1.6	1.6⇒1.7	1.7⇒1.8
非主要路割増率a1	1.2	-1.7E-05	-0.00021	-1.6E-05	-1.7E-05	-8.3E-08	-5.9E-06
	1.3	-1.7E-06	-0.00021	-1.8E-05	-1.6E-05	0	-2E-05
	1.4	-3.6E-05	-0.00021	-1.8E-05	-9.8E-06	-1.6E-05	-2E-05
	1.5	-3.1E-06	-0.00021	-3.4E-07	-7.6E-06	-2.6E-05	-1.7E-05
	1.6	-6.2E-06	-7.8E-07	-0.00021	-5.1E-06	-3.5E-05	-1.4E-05
	1.7	-9.4E-06	-5.3E-06	-0.00019	-5E-06	-1.7E-05	-2.9E-05
	1.8	-2.6E-05	-5.9E-06	-0.00019	-1.2E-05	-5E-06	-2E-05
	1.9	-1.2E-06	-2.6E-05	-0.0002	-1.1E-05	-5E-06	-8.8E-06
	2	-1.5E-05	-4.9E-06	-0.00022	-9.6E-06	-4.5E-06	-5E-06
	2.1	-9.7E-07	-2.1E-06	-0.00021	-2.8E-05	-4.8E-06	-5E-06



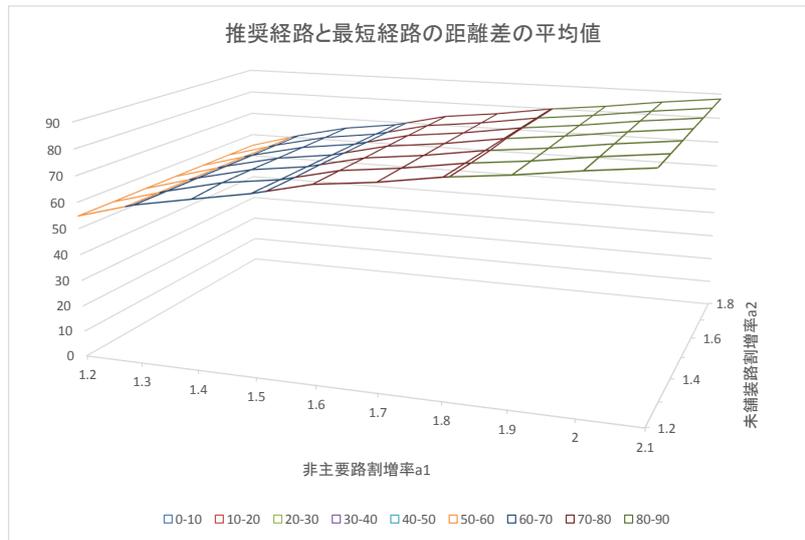
推奨経路と最短経路の距離比(割増率0.1刻み)

		未舗装路割増率a2						
		1.2	1.3	1.4	1.5	1.6	1.7	1.8
非主要路割増率a1	1.2	1.046476	1.046505	1.046763	1.046779	1.0468	1.046811	1.046815
	1.3	1.05121	1.051214	1.051445	1.051453	1.051474	1.051474	1.05151
	1.4	1.055301	1.056318	1.056563	1.056515	1.056495	1.056515	1.056552
	1.5	1.05965	1.059654	1.059904	1.059904	1.05991	1.059932	1.059973
	1.6	1.064938	1.064962	1.064964	1.065204	1.06521	1.065272	1.065294
	1.7	1.068071	1.068231	1.068291	1.068517	1.068522	1.068577	1.068608
	1.8	1.07211	1.072104	1.07213	1.072346	1.072375	1.07238	1.072406
	1.9	1.074719	1.074717	1.074711	1.0751	1.075124	1.075129	1.075131
	2	1.077901	1.077886	1.077888	1.078205	1.078225	1.078236	1.078239
	2.1	1.08099	1.080983	1.080984	1.081201	1.081168	1.081178	1.081183



推奨経路と最短経路の距離差の平均値(割増率0.1刻み)

		未舗装路割増率a2						
		1.2	1.3	1.4	1.5	1.6	1.7	1.8
非主要路割増率a1	1.2	54.78574	54.79432	54.91548	54.92508	54.93393	54.9503	54.95203
	1.3	60.8653	60.86715	60.96927	60.97069	60.97978	60.97978	61.00755
	1.4	64.84927	65.87234	65.98305	65.92424	65.90736	65.91644	65.9411
	1.5	68.7083	68.70963	68.82487	68.82551	68.8269	68.83558	68.86252
	1.6	73.81232	73.82575	73.83139	73.93972	73.94253	73.9822	73.99209
	1.7	76.28379	76.36544	76.39442	76.49709	76.49781	76.54134	76.55251
	1.8	79.78133	79.78131	79.80595	79.90097	79.9199	79.92062	79.92973
	1.9	82.19357	82.19325	82.19323	82.36755	82.38393	82.38464	82.37934
	2	85.28825	85.28789	85.29079	85.43298	85.44641	85.4495	85.44927
	2.1	87.88704	87.879	87.88184	87.98576	87.96922	87.97182	87.97253



推奨経路と最短経路の距離差の標準偏差(割増率0.1刻み)

		未舗装路割増率a2						
		1.2	1.3	1.4	1.5	1.6	1.7	1.8
非主要路割増率a1	1.2	81.9606	81.96027	81.99575	82.0056	82.0047	82.04725	82.0464
	1.3	88.98001	88.99053	88.96119	88.95829	88.95674	88.95674	88.94122
	1.4	93.60597	93.9575	93.95394	93.89203	93.83807	93.83613	93.82531
	1.5	98.73492	98.73449	98.74438	98.74696	98.74635	98.7404	98.73293
	1.6	103.5924	103.6075	103.617	103.6097	103.6116	103.6063	103.6188
	1.7	107.1317	107.2445	107.2822	107.28	107.2797	107.2755	107.2716
	1.8	111.4108	111.419	111.4203	111.3998	111.3965	111.3963	111.3935
	1.9	115.4082	115.4084	115.4163	115.4909	115.4884	115.4881	115.4725
	2	119.6168	119.6272	119.6375	119.6985	119.6959	119.6962	119.6966
	2.1	123.218	123.2079	123.213	123.2061	123.181	123.1838	123.1835

推奨経路と最短経路の距離差の平均値+標準偏差(割増率0.1刻み)

		未舗装路割増率a2						
		1.2	1.3	1.4	1.5	1.6	1.7	1.8
非主要路割増率a1	1.2	136.7463	136.7546	136.9112	136.9307	136.9386	136.9976	136.9984
	1.3	149.8453	149.8577	149.9305	149.929	149.9365	149.9365	149.9488
	1.4	158.4552	159.8298	159.937	159.8163	159.7454	159.7526	159.7664
	1.5	167.4432	167.4441	167.5692	167.5725	167.5732	167.576	167.5955
	1.6	177.4047	177.4333	177.4484	177.5494	177.5541	177.5885	177.6109
	1.7	183.4155	183.6099	183.6766	183.7771	183.7775	183.8168	183.8241
	1.8	191.1921	191.2003	191.2263	191.3008	191.3164	191.3169	191.3233
	1.9	197.6018	197.6016	197.6095	197.8585	197.8723	197.8727	197.8519
	2	204.9051	204.9151	204.9282	205.1315	205.1423	205.1457	205.1459
	2.1	211.105	211.0869	211.0949	211.1918	211.1503	211.1557	211.1561

付録D 評価実験資料

D.1 実験計画書

実験計画書（経路案内）

更新日：2014/11/18

作成者：千田隼人

目的

1. 推奨経路の妥当性を確認すること.
2. 最短経路案内の有用性を確認すること.

観察者

1. 千田隼人

実験参加者

A. **** (実験参加者氏名)

B. ****

C. ****

D. ****

E. ****

F. ****

G. ****

H. ****

I. ****

J. ****

K. ****

L. ****

歩行する経路

1. 合宿所バス停⇒開学記念館
推奨経路：964m（約13分），最短経路：612m（約9分）
2. 桐葉橋下の交差点⇒本部棟
推奨経路：751m（約11分），最短経路：652m（約10分）
3. 本部棟⇒自然系学系 E 棟
推奨経路：667m（約9分），最短経路：623m（約10分）

実験参加者の割当て

	経路 1	経路 2	経路 3
推奨経路	A, B, C, D	E, F, G, H	I, J, K, L
最短経路	I, J, K, L	A, B, C, D	E, F, G, H
キャンバスマップ	E, F, G, H	I, J, K, L	A, B, C, D

※アプリを使用している歩行では、1回目でiOS、2回目でAndroidを使用する。

準備物（実験参加者 1 人分）

1. 本紙
2. XPERIA（CampusAR（スタート地点を固定できるもの）をインストールしておく）
3. iPhone（CampusAR（スタート地点を固定できるもの）をインストールしておく）
4. 実験参加者行動記録用紙×3 枚
5. アンケート（印刷物）
6. キャンパスマップ（中地区，南地区）×2 セット
7. バインダ
8. ボールペン（赤，黒）
9. ストップウォッチ

実験手順

1. 実験参加者にアンケート（印刷物）を見せ，内容を確認してもらう。
2. 実験参加者を経路 1 のスタート地点まで連れていく。その際，実験で歩行する道避ける。
3. 記録用紙，筆記用具，ストップウォッチを用意する。
4. [アプリ使用时]アプリにスタート地点とゴール地点と経路種別を入力する。
5. [アプリ使用时]端末を実験参加者に手渡し，経路 1 のゴール地点まで画面表示に従って歩行するよう指示する。地図の移動と拡大縮小以外の操作を行わないよう指示する。
[キャンパスマップ使用时]キャンパスマップを実験参加者に手渡し，経路 1 のゴール地点まで自分で経路を決めて歩行するよう指示する。
6. ストップウォッチを作動させる。
7. 実験参加者を追跡しながら実験参加者の行動を実験参加者行動記録用紙の行動記録欄に記入する。
[キャンパスマップ使用时]歩行経路をキャンパスマップに記入する。
8. 実験参加者が経路 1 のゴール地点に到着したらストップウォッチを停止させる。
9. 道を間違えた場合は理由を実験参加者に聞く。
10. [アプリ使用时]経路表示の分かりにくい箇所があったかどうかとその理由を実験参加者に聞く。
11. 実験参加者行動記録用紙の事後記録欄に必要事項を記入する。
12. 2~11 をあと 2 回繰り返す。
13. 2 日後までにアンケートに回答するよう実験参加者に指示する。
14. 実験内容を絶対に口外しないよう実験参加者に指示する。
15. 実験参加者を工学系学系 F 棟まで連れていく。
16. 実験参加者にメールでアンケート（電子ファイル）を送付する。

D.3 アンケート

アンケート(経路案内)

●実験日 2014 年 12 月 日	
●歩行経路 1回目: ⇒ : 2回目: ⇒ : 3回目: ⇒ :	
●質問 <b style="color: red;">注意:初めて訪れる施設の内部を歩行する場合を想定して回答してください。	
1. 推奨経路の総距離は最短経路の何%程度までなら増えても構わないと思いますか？ (選択式)	
2. 進路を確認する際にどのようなものを目印にして歩きましたか？ (例:案内板、景色、道の形、建物の表札) (自由記述式)	
3. AR表示はあった方が分かりやすいと思いますか？ (選択式)	
4. AR表示の精度はいかがでしたか？ (選択式)	
5. 歩行中に経路の他に提供してほしい情報はありますか？ (自由記述式)	
6. 推奨経路、推奨経路(階段無し)、最短経路(階段無し)の他に提供してほしい経路の種類はありますか？ (自由記述式)	
7. 初めて訪れる大型施設で本システムによる案内サービスが提供されているとしたら、積極的に利用したいと思いますか？ (選択式)	
8. Android版とiOS版のアプリを使用して、両者のUIや操作性などに相違点を感じた場合はその内容を記入してください。(自由記述式)	
8. その他ご意見やシステムの不具合がありましたら記入してください。(自由記述式)	

ご協力ありがとうございました。

付録E マニュアル

CampusARユーザーマニュアル

平成26年11月11日



キャンパスARの基盤の構築

ART

文責：畑中裕太、萬成亮太 版：4.0

目次

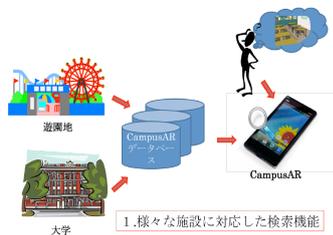
1	アプリ概要	3
1.1	何ができるのか？	3
1.2	動作環境	3
2	アプリで使うインターフェース	4
2.1	標準画面	4
2.2	メニュー	5
2.3	施設選択画面	6
2.4	お気に入り一覧画面	7
2.5	履歴一覧画面	8
2.6	検索画面	9
2.7	検索結果画面	10
2.8	検索結果のカテゴリ切り替え	11
2.9	ナビゲーション画面（地図）	12
2.10	経路選択ウィンドウ	13
2.11	お気に入りに登録・削除	14
2.12	ナビゲーション画面（AR）	15
2.12.1	地図とARの切り替え	16
3	注意事項	17

1 アプリ概要

1.1 何ができるのか？

CampusAR が実現する機能は主に以下の4つです。

1. どこに目的地があるかわからない時に、目的地の検索をすることができます。
2. 2次元地図とARによって行きたい建物にガイドしてくれます。
3. お気に入り機能と履歴機能により再度検索する手間を省くことができます。



2. 地図とARシステムを用いたナビゲーション機能



1.2 動作環境

Android4.1以降、またはiOS7以降で、背面カメラ、モーションセンサとGPSを内蔵しており、モバイルデータ通信、またはWi-Fiでの常時通信機能が使える携帯端末とタブレットで動作可能です。動作確認している機種を以下に示します。

- iPhone 6 Plus
- Galaxy note
- XPERIA Z1

2 アプリで使うインターフェース

2.1 標準画面



Android 版



iOS 版

① 検索ボックス

検索ボックスをタップすると、検索ボックスに文字を入力できるようになります。

② メニューボタン

メニューボタンをタップすると、施設選択、お気に入り、履歴にアクセスできるメニューが表示されます。

③ 状態遷移ボタン

状態遷移ボタンは以下の状態遷移をします。

- 現在位置に移動するボタン：現在位置が画面の中心になるように画面移動します。
- 画面回転ボタン：端末の向きに合わせて画面が回転するようになります。地図が45 仰角の俯瞰で見えているように表示が変わります。
- 画面回転停止ボタン：画面の回転を止め、地図の向きを北向きに変えることができます。

④ 現在位置の表示

GPS で取得した現在位置を地図上に表示します。端末の向いている方向に従い矢印の向きが変化します。

⑤ コンパス

N と書かれている方向が北を指しています。

2.2 メニュー



Android 版



iOS 版

- ① 施設選択
施設選択ボタンをタップすると、施設選択画面に移動します。
- ② お気に入り
お気に入りボタンをタップすると、お気に入り一覧画面に移動します。
- ③ 履歴
履歴ボタンをタップすると、履歴一覧画面に移動します。
- ④ ヘルプ
ヘルプボタンをタップすると、アプリマニュアルを表示します。
- ⑤ メニューを閉じる
半透明のスクリーンをタップすると、メニューを閉じます。

2.3 施設選択画面



Android 版



iOS 版

① 施設一覧

施設名をタップすると、選択した施設で検索できるようになります。

② バックボタン

バックボタンをタップすると、前の画面に戻ります。

2.4 お気に入り一覧画面



Android 版



iOS 版

- ① お気に入り一覧
場所の名前をタップすると、その場所への案内を開始します。
- ② バックボタン
バックボタンをタップすると、前の画面に戻ります。

2.5 履歴一覧画面



Android 版



iOS 版

① 履歴一覧

場所の名前をタップすると、その場所への案内を開始します。履歴一覧には検索結果から案内を開始した場所が自動的に登録されます。

② バックボタン

バックボタンをタップすると、前の画面に戻ります。

2.6 検索画面



Android 版



iOS 版

① 検索ボックス

検索ボックスをタップすると、キーボードが表示されます。キーボードから検索したいワードを入力し、検索を行うことができます。

② 検索ボタン

キーボードの検索ボタンをタップすると、入力したキーワードで検索されます。

③ キーボードを閉じる

地図をタップすると、キーボードが閉じます。

2.7 検索結果画面



Android 版



iOS 版

- ① 検索結果
場所の名前をタップすると、その場所への案内を開始します。
- ② カテゴリ
一覧に表示されているカテゴリを示します。
- ③ バックボタン
バックボタンをタップすると、前の画面に戻ります。

2.8 検索結果のカテゴリ切り替え



Android 版



iOS 版

① 検索結果

検索一覧を横方向にスワイプすることで、検索結果のカテゴリを切り替えることができます。

2.9 ナビゲーション画面（地図）



Android 版



iOS 版

- ① 目的地名
目的地の名前を表示します。目的地名をタップすると、案内を終了し、新たに検索することが出来ます。
- ② 詳細情報
目的地の詳細情報を表示します。
- ③ 総距離
案内を開始した地点から目的地までの距離を表示します。
- ④ 経路選択ボタン
案内経路の種類を切り替えるウィンドウを表示します。
- ⑤ お気に入りボタン
案内中の場所をお気に入りに登録、または解除します。
- ⑥ 案内終了ボタン
案内を終了します。
- ⑦ 目的地
案内中の目的地の場所を表示します。
- ⑧ 経路
案内を開始した地点から目的地までの道のりを表示します。

2.10 経路選択ウィンドウ



① 経路の種類

経路の種類をタップすると、現在地から目的地までの経路を再計算し、画面に新しい経路を描画します。

② ダイアログを閉じる

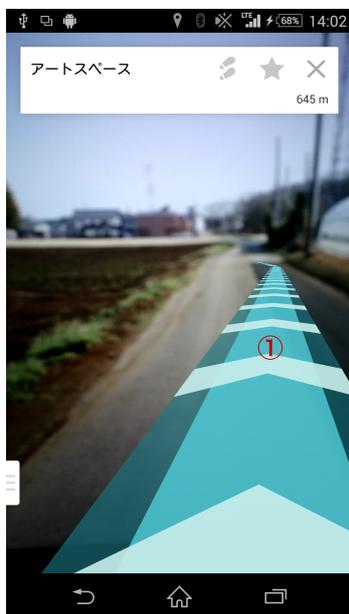
地図やキャンセルボタンをタップすると、経路選択をキャンセルし、ウィンドウを閉じます。

2.11 お気に入りに登録・削除



お気に入りボタンをタップすると、案内中の場所をお気に入り登録、または削除することができます。
アイコンが黄色く点灯している場所は、お気に入りに登録されていることを示します。

2.12 ナビゲーション画面 (AR)



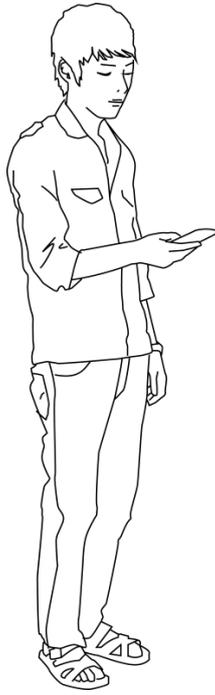
Android 版



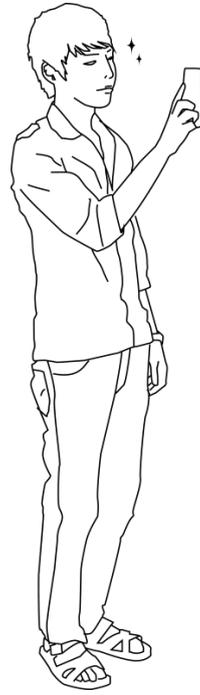
iOS 版

① 経路
経路をARで表示します。

2.12.1 地図とARの切り替え



地図



AR

端末を下に向けると地図、立ち上げるとARに自動的に切り替わります。

3 注意事項

アプリを使用する上での注意事項

- 端末の GPS 機能を ON にして使用してください。
- インターネット通信ができる状態でのみ使用することができます。
- このアプリケーションが案内するのは建物までです。教室までは館内の見取り図などから自力で到達してください。
- 自転車などの乗り物に乗りながら使用しないでください。
- 機種によってはカメラがなかなか起動しないものもあります。