

移動時間を考慮した位置情報によるサービス検索

6 E-8

村上朝一¹ 楠本晶彦³ 由良淳一³ 徳田英幸^{2,3}¹ 慶應義塾大学 総合政策学部 ² 慶應義塾大学 環境情報学部 ³ 慶應義塾大学大学院 政策・メディア研究科

1 はじめに

プリンタやディスプレイなどのデバイスによるサービスを検索するには、物理的位置情報の利用が有効であり、様々な位置情報を用いたサービス検索システムが提案されている。しかし従来のシステムでは、ユーザの実際の移動距離、移動にかかる時間を考慮できなかった。そのため最も近接するサービスを検索した結果が、ユーザの体感距離にそぐわない場合がある。

本稿では、ユーザの移動時間から部屋や階段などの空間同士の距離関係を計算し、サービス検索を利用するシステムを提案する。本システムではユーザの過去の移動時間を蓄積し、動的に空間間の距離関係に反映する。これにより、従来に比べユーザの移動を考慮してサービスを検索できる。

2 研究の概要

本研究では部屋間の距離関係を計算するために、ユーザの過去の移動時間を用いる。移動時間を利用することにより部屋の配置、階段、エレベータなどの建物の構造に関係なく空間同士に対するユーザの体感距離が比較できる。

2.1 対象環境

本研究はオフィスやキャンパスのような複数の部屋、建物を想定する。その環境では、プリンタやディスプレイなどのサービスが散在している。また全てのユーザがRF-Code[1]のタグやJavaCard[2]をはじめとする、部屋単位でユーザの位置情報が取得できるデバイスを持ち歩き、リーダーが各部屋、建物の入り口などに設置されているものとする。

このような環境において、本システムを用いてユーザがプリンタを検索する例を説明する。利用可能なプリンタがユーザのいる 2 階と 1 階の部屋に複数存在する。ここで、2 階にあるプリンタは同じ階ではあるが、遠くの部屋にあり、1 階にあるプリンタは階段の近くにあるので、ユーザのいる部屋からは 2 階のプリンタと比べて近い。ユーザは相対的に近い 1 階のプリンタを利用できる。

2.2 既存システムの問題点

位置情報をもとに近接するサービスを検索するシステムとしては、SDS[3] や FLUSH[4] などがある。

SDS は空間の位置関係を階層構造で表現している。そのため、最も近接するサービスを検索し、同じ階層に複数のサービスが存在した場合にどちらがより近いかを判断できない。

FLUSH システムは空間の位置関係を階層構造で表現しているが、同じ階層にある空間同士で距離に重み付

Location based Service Discovery System considering Accessing Cost

¹ Faculty of Policy Management, Keio University
5322, Endo, Fujisawa, Kanagawa 252, Japan
E-Mail: tomo@ht.sfc.keio.ac.jp

² Graduate School of Media and Governance, Keio University

けをしている。そのため、最も近接するサービスをひとつに絞りこむことができる。しかし FLUSH システムでは、現実には違う階層に存在する空間の方が同じ階層に存在する空間よりも近い場合に正確な空間同士の距離関係の比較ができない。

また、これらの既存システムでは、建物の構造を物理的な位置関係を用いて表現している。そのため、ユーザの実際の移動という視点から見た場合に、階段やエレベーターなどの構造を評価できない。さらに、サービスまでの距離を計算するために、建物の物理的構造がシステム内に保持されている必要があり、管理者が建物の構造を入力する必要がある。

2.3 研究の目的

本研究では以上のような問題点を解決し、ユーザの体感距離を考慮したサービス検索システムを構築する。

ユーザの要求に該当し、ユーザに最も近接するサービスを検索するシステムには、以下の要件がある。

まず、ユーザの現在位置を把握する。次に、保持しているサービスの位置情報から、ユーザの要求にあつたサービスが存在する空間を検索する。さらに、ユーザの要求に該当するサービスが複数存在した場合に、最も近接するサービスを特定する。

ユーザに最も近接するサービスを特定するためには、ユーザのいる部屋からサービスの存在する部屋までの距離を物理的距離だけでなく、階段やエレベーターなどの構造をユーザの体感距離という視点から評価する必要がある。

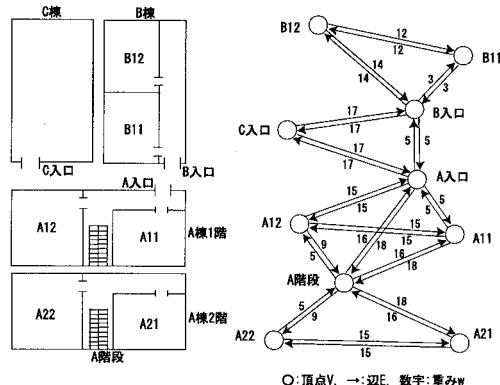


図 1: 建物の構造と重み付き有向グラフ G_T

2.4 体感距離の表現

本研究では、建物の構造を重み付き有向グラフ $G_T = (V, E, w)$ で抽象化する。ここで $u \in V$ は、部屋などの空間、建物の入り口、階段などの構造とする。また、辺 $(u, v) \in E$ の重み $w(u, v)$ は、ユーザの移動平均時間で計算し、体感距離を表す。ただし、 $w(u, v) > 0$ とする。この関係を図 1 に示す。

本システムではこの G_T をサービス検索に利用する。 G_T を生成する際には、部屋だけでなく建物の入口、階段なども V として E の数を減らし、効率良く距離の比較ができる。このように G_T を生成することで階段の上り下りや、エレベータなどの建物の構造を考慮に入れた体感距離の計算ができる。

また、本システムは $w(u, v)$ をユーザの移動時間を取得する度に更新するため、ユーザの空間から空間への移動で自動的に G_T が生成される。そのため、管理者はリーダを設置し、部屋にあるサービスを登録するだけで、建物内外の構造、地図をあらかじめシステムに入力する必要がない。

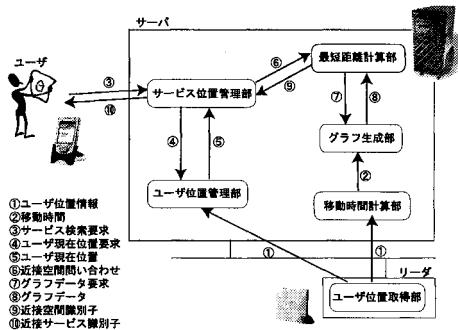


図 2: システム構成図

3 システムの設計と実装

本システムは以下の機能とモジュールから構成される。またこの関係を図2に示す。

- ユーザの現在位置取得機能
 - a ユーザ位置取得部
 - b ユーザ位置管理部
- サービス検索機能
 - c サービス位置管理部
- 近接サービス特定機能
 - d 移動時間計算部
 - e グラフ生成部
 - f 最短距離計算部

ユーザ位置管理部、サービス位置管理部、移動時間計算部、グラフ生成部、最短距離計算部は想定環境であるキャンパスやオフィスに存在するひとつのサーバ上で動作する。またユーザ位置取得部は、ユーザの位置情報を取得するデバイス上で動作する。

以下に各モジュールについて述べる。

- ユーザ位置取得部 (a)
- ユーザの空間への入退出イベントをタイムスタンプとともに、ユーザ位置管理部と移動時間計算部に送る。
- 移動時間計算部 (d)
- ユーザが空間 u から出た時間ととなりの空間 v に入った時間の差分から $(u, v) \in E$ の移動時間を計算する。計算した移動時間をグラフ生成部に渡す。
- グラフ生成部 (e)
- 移動時間計算部から渡される (u, v) の移動時間の算術平均を計算し、その値を w とする。ユーザの移動

時間が移動時間計算部から渡される度に w を再計算し、 G_T を更新する。全てのユーザの移動時間を利用することにより、サービスを検索するユーザが通ったことのない経路でも、距離の比較ができる。

● 最短距離計算部 (f)

サービス位置管理部からユーザの現在いる空間 $s \in V$ と、サービスの存在する複数の空間 $v_0 \dots v_n \in V$ を受け取る。空間 s から v_k までの最短距離 $\delta(s, v_k)$ をDijkstraアルゴリズムを用いて計算する。結果として $\delta(s, v_k)$ が最小となる v_k をサービス位置管理部に返す。

● ユーザ位置管理部 (b)

ユーザの現在位置を空間単位で管理する。サービス位置管理部からのユーザの現在位置要求に対してユーザの現在いる空間識別子を返す。

● サービス位置管理部 (c)

サービスの位置を空間単位で管理する。サービスを新たに追加する場合、システムの管理者がそのサービスの空間を登録する。

ユーザからユーザ識別子とユーザの利用要求サービス名を受け取り、サービス識別子を返す。ユーザの要求に該当するサービスが複数存在する場合は、ユーザの現在位置をユーザ位置管理部に問い合わせ、ユーザの存在する空間識別子とサービスの存在する空間識別子を最短距離計算部に渡し、最も近接する空間を問い合わせユーザに結果のサービス識別子を返す。

本システムはJava言語で実装した。ユーザの位置情報を取得するデバイスとしてはRF-Codeを用いた。

4 まとめと今後の課題

本稿では、ユーザの移動時間を考慮したサービス検索を実現するシステムの設計、実装について述べた。本システムを用いることにより、ユーザの体感距離を考慮して、ユーザに近接するサービスを検索できる。また、管理者が建物の構造を入力することなく、空間間の距離関係を比較できる。

今後は、より個々のユーザの体感距離に合ったサービスの検索を可能にするため、全てのユーザでひとつのグラフを共有するのではなく、ユーザごとのグラフも生成する。ユーザの移動時間からより信頼性の高い重み w を計算するために、他の計算方法も評価する。また、本システムの対象環境を、ひとつのキャンパス、オフィスなどから複数のキャンパス、オフィスなどに広げるため、スケーラビリティを考慮する。さらに本システムのサービス検索以外の利用についても考察する。

参考文献

- [1] RF Code, Inc., <http://www.rfcode.com/>
- [2] Sun Microsystems, Inc., <http://java.sun.com/products/javacard/>
- [3] Czerwinski, S. E., Zhao, B. Y., Hodges, T. D., Joseph, A. D. and Katz, R. H.: An Architecture for a Secure Service Discovery Service: The Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp.4-33 (1999) The Ninja Team: The Ninja Project. <http://ninja.cs.berkeley.edu>
- [4] 楠本晶彦, 岩井将行, 中澤仁, 大越匡, 徳田英幸: 物理的位置情報に基づくサービスの自動検出を実現するミドルウェアの構築: マルチメディア、分散、協調、とモバイル(DICOMO)シンポジウム (2000)