

複数のモバイル端末による協調ブラウジングのための 木構造型コンテンツ記述方式と分割方式

前川 卓也[†] 上向 俊晃^{††}
原 隆浩[†] 西尾 章治郎[†]

モバイルコンピューティング環境では、モバイル端末の機能的制限により、各モバイル端末に提供できるサービスも制限される。そこで本論文では、1つの端末だけでは閲覧することができないリッチなコンテンツを、身の回りにある複数の端末を用いてブラウジングを行う協調ブラウジングを想定する。協調ブラウジングでは、1つのコンテンツを複数のコンテンツに分割し、各端末に配布する。本論文では、協調ブラウジングを実現するためのコンテンツ記述方式およびコンテンツ分割方式を提案する。提案方式では、協調ブラウジングに用いるコンテンツは枝に重みがある木構造形式で記述し、その木を完全グラフに変換したうえでグラフ分割問題に帰着させる。このとき、端末の持つ機能や性能、端末に割り当て可能なコンテンツの量、ユーザの要望などを考慮することにより、柔軟な分割を行うことができる。

Tree Structured Content Description and Partitioning Methods for Collaborative Browsing by Multiple Mobile Users

TAKUYA MAEKAWA,[†] TOSHIAKI UEMUKAI,^{††} TAKAHIRO HARA[†]
and SHOJIRO NISHIO[†]

In a mobile computing environment, low functionality of mobile terminals restricts services provided for mobile users. In this paper, we assume that mobile users collaboratively browse rich contents, which one terminal cannot browse by itself, by using multiple terminals. In collaborative browsing, a rich content is divided into multiple contents and then each of them is distributed to each terminal. We propose a content description method and partitioning method for collaborative browsing. By using the proposed content description method, a content is described in the structure of an edge-weighted tree. Then, by using the proposed partitioning method, this tree is converted into a complete graph and a graph partitioning algorithm is applied to divide the content. In doing so, the functions of terminals, the size of contents which the terminals can receive, and the requests from users are taken into account for flexible partitioning.

1. はじめに

近年、計算機の小型化と無線通信技術の発展にとともに、モバイルコンピューティング環境に対する注目が高まっている⁷⁾。モバイルコンピューティング環境では、ユーザは携帯可能なモバイル端末を用いていつでもどこでもネットワークにアクセスし、情報を得ることができる。

しかし、依然として、モバイル端末はディスプレイサイズや計算能力などの機能的制限があるため、リッチなコンテンツを快適に閲覧することは困難である。そこで、貧弱な機能しか持たない単一のモバイル端末を利用するよりも、自身のモバイル端末と身の回りにある複数の端末とを組み合わせることで、よりリッチなコンテンツを閲覧することができる。本論文では、このように、複数の端末を用いて1つのコンテンツを閲覧することを協調ブラウジングと呼ぶ。協調ブラウジングでは、1つのコンテンツをさらに複数のコンテンツに分割することにより、複数の端末による協調作業を実現する。しかし、モバイルコンピューティング環境において、あるユーザが協調ブラウジングを行いたいときに、ユーザの身の回りある端末の数

[†] 大阪大学大学院情報科学研究科マルチメディア工学専攻
Department of Multimedia Engineering, Graduate
School of Information Science and Technology, Osaka
University

^{††} 大阪大学大学院工学研究科情報システム工学専攻
Department of Information Systems Engineering,
Graduate School of Engineering, Osaka University

や性能などが一定であるとは限らない。そのため、これらの条件に応じて柔軟にコンテンツを分割することは困難である。

一般的なコンテンツ記述言語として、WWW ページの作成に利用される HTML (Hyper Text Markup Language) があるが、HTML はコンテンツの表示形式しか表現することができない。一方、計算機に処理させるために開発された XML (eXtensible Markup Language) では、木構造を用いてコンテンツを階層的に表現できる。XML を用いてマルチメディアコンテンツのメタデータを記述する MPEG-7 では、コンテンツの階層構造を表現できるが、単に木構造として表現しているだけで、ノード (コンテンツ) どうしの関連性を表現できない。そのため、協調ブラウジングのための柔軟なコンテンツ分割を実現できない。

そこで本論文では、協調ブラウジングにおいて 1 つのコンテンツを任意の数の端末で協調して閲覧するためのコンテンツ記述方式および分割方式を提案する。提案方式では、MPEG-7 のようなメタデータの木構造に枝の情報を付加する。さらに、この重み付き木を完全グラフに変換することによりグラフ分割問題に帰着させ、端末の数や性能などに応じてコンテンツを分割する。

ここで、1 つの携帯端末には表示できないコンテンツを携帯端末上に表示することを目的とした従来研究には、文献 1) がある。文献 1) では、デスクトップパソコンでの閲覧向けに作成された WWW ページを、ユーザのプロファイル情報を用いて個人化することにより、携帯端末に表示可能な小さい WWW ページを作成する。具体的には、まず、ユーザの WWW ブラウジング履歴からユーザが閲覧したページに現れる単語を解析し、プロファイル情報を作成する。そして、WWW ページからユーザのプロファイル情報に合わない部分を省略した WWW ページを作成する。一方、本研究では、1 つのコンテンツを複数の携帯端末に分配して表示するため、コンテンツを簡素化せずに元のコンテンツと同じ内容で閲覧することができる。

さらに、身の回りにある機器を用いてリッチなサービスを提供する研究には、文献 4) がある。文献 4) では、ユーザは WWW ブラウザを操作することで、携帯端末だけでなく、ディスプレイやプリンタなどの機器ごとにあらかじめ設定したサービスを受けることができる。たとえば、プロジェクトを用いてプレゼンテーションを行うとき、ユーザは自身の持つ PDA 上の WWW ブラウザで「プロジェクトを用いたプレゼンテーション」というサービス、および、そのプレゼ

ンテーションファイルの PDA 上の所在を指定する。これにより、ユーザは、そのサービスに関連付けられている機器を用いたプレゼンテーションを行うことができる。しかし、このようなサービスを利用するには、そのサービスに対してあらかじめ機器を関連付けておく必要がある。一方、本研究では、コンテンツを閲覧するとき、ユーザの身の回りにある携帯端末の状況に応じてコンテンツを動的に分割して提供することができる。

以下では、2 章で本論文で想定する協調ブラウジングについて述べる。3 章では、その協調ブラウジングを実現するためのコンテンツ記述方式について述べ、4 章では、コンテンツ分割方式について述べる。5 章でコンテンツ分割方式の評価を行う。6 章で考察を行い、最後に 7 章で結論と今後の課題を述べる。

2. 協調ブラウジング

協調ブラウジングでは、ユーザは、他のユーザの携帯端末などの複数の端末を用いて、1 つの非力な端末では閲覧できないリッチなコンテンツを閲覧する。このとき、その 1 つのリッチなコンテンツは複数のコンテンツに分割される。より正確には、コンテンツを構成する複数のオブジェクトを、分割数に応じてオブジェクト集合に分ける。各端末には、割り当てられたオブジェクト集合が配布される。協調ブラウジングの応用例としては、下記があげられる。

● 協調 WWW ブラウジング

リンク数やコンテンツ量が多く、1 つの携帯端末では閲覧するのに困難な WWW ページを、友人などの複数の携帯端末に分配する。このとき、たとえば PDA などの端末には画像などを多く含む情報量の多いコンテンツが割り当てられ、携帯電話などの端末には情報量の少ないコンテンツが割り当てられる。そして、ユーザは各自の携帯端末に割り当てられた WWW ページを用いて情報を検索する。あるユーザがリンクを選択した際、そのリンク先のページをそれらの携帯端末に分配し、目的の情報を見つけるまで複数の端末を協調させることにより WWW ブラウジングを行う。このように、複数のユーザが分担して情報を検索するため、目的とする情報や、その情報へのリンクを容易に見つけることができる。

● 協調ゲーム

上述の協調 WWW ブラウジングが、携帯端末の制限上の問題を解決するための本来の目的に合致した例であるのに対し、協調ブラウジングを意図

的に利用した応用例として、協調ゲームがあげられる。協調ゲームでは、1つのゲームにおいて、画面やタスクを分割し、複数の携帯端末を用いて協調してゲームを行う。たとえば、迷路のアプリケーションにおいて、1つの携帯端末には表示しきれない迷路を分割する。ユーザは、各自の携帯端末に割り当てられた迷路を担当し、1つの駒をそれぞれのユーザがリレーすることにより、駒を全体の迷路のスタートからゴールまで導く。

ここで、想定している協調ブラウジングでは、ユーザの身の回りにある端末の数、性能は一定ではなく、また、貧弱な処理しかできない携帯電話などの端末も含まれるため、以下のような条件を考慮してコンテンツを分割する必要がある。

- 端末の数(コンテンツの分割数)
想定環境ではコンテンツの閲覧を行うときに利用できる端末の数が一定でないため、さまざまな分割数に対応する必要がある。
- 端末に割り当て可能なコンテンツの量
端末によっては利用可能なメモリやストレージの容量が制限される。たとえば、携帯電話には表示可能な Web ページの最大容量が制限されている。そのため、各端末に分配するコンテンツのサイズを考慮する必要がある。
- 端末の性能
端末によっては画面の大きさや処理能力などが制限される。そのため、コンテンツを複数の端末に分配するとき、各端末の性能比に応じてコンテンツの分配割合を決定する必要がある。
- 端末の機能
実行に特別な機能が必要なコンテンツ(オブジェクト)は、その機能を持つ端末に割り当てる必要がある。たとえば、白黒の画像しか表示できない携帯端末には、カラーの画像を割り当てない。
- 分割したコンテンツの関連性
分割されたコンテンツを構成するそれぞれのオブジェクト集合は、関連性のあるオブジェクトから構成されている方が利用しやすい。
- コンテンツの配布
オブジェクトの種類によっては、1つの端末だけでなくすべての端末に配布する必要のあるオブジェクトがある。たとえば、分割した全オブジェクトのインデックスを示すメニューなどがある。
- ユーザの要求
ユーザによっては、特定のコンテンツ(オブジェクト)を自身の端末に割り当てたい場合や割り当

てたくない場合がある。このような割当て要求や、非割当て要求を考慮して分割する必要がある。

以上に示した分割の条件は、協調ブラウジングを行う際に最低限必要と考えられ、アプリケーションによっては他の分割条件も必要となる。たとえば、協調 WWW ブラウジングでは、ユーザが興味のある情報を多く含むオブジェクトを割り当てる条件などが考えられる。具体的には、オブジェクトに個人化情報を付加し、あらかじめ設定したユーザのプロファイル情報と合致するようにコンテンツを分割する。また、協調ゲームの1つである迷路ゲームでは、全員をゲームに参加させるために、駒をスタートからゴールに移動させるにあたって、それぞれの携帯端末に割り当てられた部分迷路上を最低1度は駒が通過しなければゴールにたどりつけないように分割するための条件などが考えられる。具体的には、ゴールにたどりつくために必ず通らなければならない道を含むオブジェクトとそうでないオブジェクトを識別するための情報を付加し、必ず通らなければならないオブジェクトを各携帯端末に振り分けるように分割する。

3. コンテンツ記述方式

本章では、2章で述べた協調ブラウジングを実現するためのコンテンツの記述方式を提案する。

3.1 コンテンツの構造

提案するコンテンツ記述方式では、コンテンツのメタデータを XML 形式の木構造で記述する。図1にコンテンツ構造の概念図を示す。このコンテンツ構造において、葉ノード(以下、オブジェクトノード)は、コンテンツを構成する最小単位であるオブジェクトもしくはオブジェクト集合を表す。葉ノード以外のノード(以下、分類ノード)は、オブジェクトノードを階層的にグループ化するものであり、コンテンツの分割に利用する。オブジェクトノードは、たとえば、1つの画像ファイルである場合や、互いに関連性のある画像や文字列などのオブジェクト集合である場合が考えられる。図1では、各オブジェクトノードを3つの分類ノードによりグループ化し、階層的なコンテンツ構造を構成している。

3.2 オブジェクトノード間の関連性

提案するコンテンツ記述方式では、関連性のあるオブジェクトどうしのグループ化を実現するために、コンテンツ構造の各枝に重みを与え、2つのオブジェクトノードを結ぶパスの重みが小さいほど、その関連性が大きいと定義する。たとえば、図1のコンテンツ構造の場合、ノードDとEのパスの距離は2、ノ

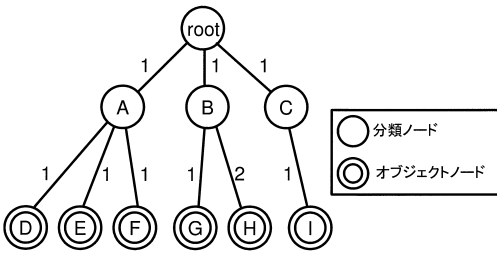


図 1 コンテンツ構造の概念図

Fig. 1 Basic concept of the content structure.

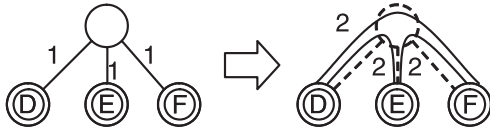


図 2 兄弟ノードの並びを考慮しない場合

Fig. 2 Correlation without considering the order of brother nodes.

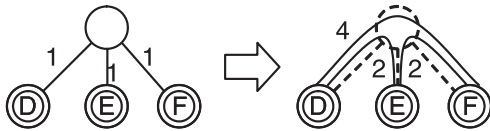


図 3 兄弟ノードの並びを考慮した場合

Fig. 3 Correlation considering the order of brother nodes.

ド D と G のパスの距離は 4 となるため、ノード D はノード G よりノード E との関連性が大きい。

しかし、この方法のみでは兄弟ノードをすべて対等に扱うため、兄弟ノードの並びの順番による関連性を定義することができない。たとえば図 2 において、オブジェクトノード間のパスの距離はすべて 2 となり、兄弟ノードの並びを考慮できない。そこで、図 3 に示すように、ノード D と E およびノード E と F の距離が 2 であるのに対し、ノード D と F の距離を、ノード D と E の距離およびノード E と F の距離の和である 4 と定義する。このように、兄弟ノード間の距離を隣接ノード間の距離の総和として求めることにより、兄弟ノードの並びを考慮した関連性を定義する。

3.3 オブジェクトノードの属性

提案するコンテンツ記述方式では、前章で述べた要求事項を満たすコンテンツ分割を行うために、オブジェクトノードに以下に示す属性を付加する。

- オブジェクトの量

実際のオブジェクト(集合)のデータサイズなど、オブジェクト(集合)の量の指標となるものを表す。この値はオブジェクトノード自体の重みと対応付ける。

- 実行に必要な端末の機能

オブジェクト(集合)を実行するために、端末に必要な機能を表す。

- オブジェクトの配布方法

オブジェクト(集合)を 1 つの端末のみに配布するか、すべての端末に配布するかを表す。

- オブジェクトの性質

オブジェクト(集合)の性質を特記する。この性質はアプリケーションごとに定義方法を変更できる。たとえば、2 章で示した協調 WWW ブラウジング特有の分割条件では、オブジェクトが持つ個人化情報を指定するために用いる。また、迷路ゲーム特有の分割条件では、駒が必ず通らなければならない道を含む部分迷路(オブジェクト)を指定するために用いる。

3.4 オブジェクトノード間の同期

3.3 節のオブジェクトノードの属性は、コンテンツ分割に必要なものであるが、協調ブラウジングのためのコンテンツ記述方式としては、オブジェクトノード間の提示の同期を指定する機能も必要となる。たとえば、協調ゲームの例である迷路のアプリケーションでは、位置的に隣接している部分迷路間で駒の往来があるため、その部分迷路が割り当てられている携帯端末どうしが、端末間での駒の移動に関する情報をやりとりし、同期をとる必要がある。

そこで、提案するコンテンツ記述方式では、別のユーザの端末に割り当てられたコンテンツとのインタラクションを実現させるために、オブジェクトどうしの同期方法を定義する。具体的には、オブジェクトノードの属性として、同期しているオブジェクトノード、および、そのオブジェクトノードとの同期関係について記述する。この同期関係には、そのオブジェクトノードのイベントが起因となって、同期しているオブジェクトノードに影響を及ぼす関係(能動)、同期しているオブジェクトノードのイベントが起因となって、そのオブジェクトノードに影響を及ぼす関係(受動)、または、それらのノードが相互に影響を及ぼす関係(相互)がある。この関係を設定しておくことにより、それらのオブジェクトノードが別々の携帯端末に分配された場合においても、それらの関係を保ったまま、インタラクションのあるコンテンツ提示を行うことができる。

3.5 オブジェクトノード間の実行順序

協調ブラウジングのために必要な機能として、コンテンツ同期のほかにも、実行順序(プレゼンテーション)の制御があげられる。

そこで、提案するコンテンツ記述方式では、各携帯

端末に割り当てたオブジェクト(集合)の実行順序を考慮するアプリケーションを実現するために、オブジェクトの実行時刻と実行期間を定義する。たとえば、スライドショーなどの時間軸に沿って実行されるコンテンツの閲覧を行うアプリケーションにおいて、オブジェクトを携帯端末上で実行する時刻とその期間をオブジェクトノードの属性として設定する。また、実行期間に重なりがあるオブジェクトどうしが1つの携帯端末に割り当てられた場合、重要なオブジェクトを優先して実行するために、実行の優先度をオブジェクトノードの属性として設定する。

3.6 メタデータの記述例

提案するコンテンツ記述方式では、コンテンツのメタデータをXML形式で記述する。図4は、図1に示す構造を持つコンテンツのメタデータの記述例である。以下に、記述に用いる各タグについて説明する。

[node タグ]

コンテンツ構造のノードを表し、入れ子構造にすることにより、階層的なコンテンツ構造を表現する。また、以下の属性を設定できる。

- type
ノードが、分類ノード (“grouping”) かオブジェクトノード (“object”) かを設定する。
- name
ノードの名前を設定する。
- weight
ノードがオブジェクトノード (type=“object”) の場合、そのオブジェクト量を設定する。設定がない場合は ‘1’ となる。
- edge_weight
ノードとその親ノードとの間を結ぶ枝の重みを設定する。設定がない場合は ‘1’ となる。
- child_location
ノードが分類ノード (type=“grouping”) の場合、ノードの子ノード間の位置関係を設定する。この値が “true” の場合、子ノードの位置関係は記述されている順に設定されるが、“false” の場合、子ノードの位置関係はすべて対等となる。設定がない場合は “false” となる。
- id
ノードの識別子を指定する。識別子は計算機がオブジェクトを識別する必要がある場合に用いる。
- priority
ノードがオブジェクトノード (type=“object”) の場合、そのオブジェクトの実行における優先度を設定する。

```
<?xml version="1.0"?>
<node type="grouping">
<node type="grouping" name="section1"
  child_location="true">
  <node type="object" name="subsection1"
    weight="3" id="1">
    <locator>./section1/subsection1.html</locator>
  </node>
  <node type="object" name="subsection2" id="2">
    <locator>./section1/subsection2.html</locator>
    <function>3</function>
  </node>
  <node type="object" name="subsection3" id="3">
    <locator>./section1/subsection3.html</locator>
  </node>
</node>
<node type="grouping" name="section2">
  <node type="object" name="subsection1" id="4">
    <locator>./section2/subsection1.html</locator>
  </node>
  <node type="object" name="subsection2"
    edge_weight="2" id="5">
    <locator>./section2/subsection2.html</locator>
  </node>
</node>
<node type="grouping" name="menu">
  <node type="object"
    name="table of contents" id="6">
    <broadcast/>
    <locator>./toc/toc.html</locator>
  </node>
</node>
</node>
```

図4 メタデータの記述例

Fig. 4 Example of metadata description.

[locator タグ]

ノードがオブジェクトノード (type=“object”) の場合、その node タグ内に定義する。その node タグが表すオブジェクトのファイル名やファイル中のどの部分であるかなどを記述する。

[function タグ]

ノードがオブジェクトノード (type=“object”) の場合、その node タグ内に定義する。オブジェクトの実行に必要な機能を数字として記述する。設定がない場合は ‘0’ となる。これにより、その node タグが表すオブジェクトはその機能を持つ端末にしか配布されない。

[broadcast タグ]

ノードがオブジェクトノード (type=“object”) の場合、その node タグ内に定義する。その node タグが表すオブジェクトを、その機能を持つ機器すべてに配信する。

[class タグ]

ノードがオブジェクトノード (type=“object”) の場合、その node タグ内に定義する。そのノードの表すオブジェクトの性質を特記する。性質は、計算機がそのオブジェクトの簡単な分類を知る必要がある場合

に用い、アプリケーションごとに定義方法を変更できる。また、以下の属性を指定できる。

- name

ノードが持つ性質の名前を設定する。

[synchro タグ]

ノードがオブジェクトノード (type="object") の場合、その node タグ内に定義する。そのノードと同期しているノードとの関係を設定する。以下の属性を指定できる。

- relation

同期しているオブジェクトとの関係を、能動 ("active"), 受動 ("passive"), または相互 ("mutual") から選択する。

[schedule タグ]

ノードがオブジェクトノード (type="object") の場合、その node タグ内に定義する。その node タグが表すオブジェクトの実行スケジュールを設定する。また、以下の属性を指定できる。

- start

ノードが表すオブジェクトの実行開始時刻を設定する。

- length

ノードが表すオブジェクトの実行期間を設定する。

4. コンテンツ分割方式

本章では、前章のメタデータ記述方式に基づいてコンテンツの分割方式を提案する。この分割方式では、メタデータの木を完全グラフに変換し、そのグラフを分割することによりコンテンツ分割を実現する。ここで、グラフ分割の問題は、VLSI 設計やプログラムのセグメント化などの分野でよく用いられ、NP-完全問題として知られている。グラフ分割問題を解くための有名な発見的アルゴリズムとしては、Kernighan らによる KL アルゴリズム³⁾ や、KL アルゴリズムをもとにした Fiduccia らによるネットワーク分割問題を解くための FM アルゴリズム²⁾ がある。また、これらの発見的アルゴリズムは、グラフの 2 分割問題だけでなく、グラフをそれ以上のグループにクラスタリングするアルゴリズムにも簡単に拡張することができる。たとえば、Suaris らによる FM アルゴリズムに基づくネットワーク 4 分割アルゴリズム^{8),9)} や、Sanchis による FM アルゴリズムに基づく複数分割アルゴリズム^{5),6)} がある。本論文で提案するコンテンツの分割方式は、FM アルゴリズムを拡張したものである。

4.1 コンテンツ分割の定義

コンテンツの分割は、3 章で述べたメタデータの木

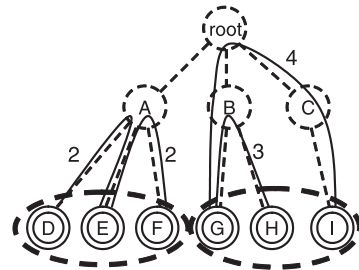


図 5 コンテンツ分割の例

Fig. 5 Example of content partitioning.

をもとに作成した完全グラフを対象に行う。完全グラフのノードはメタデータのオブジェクトノードのみから成り、ノード間の枝の重さは、メタデータの木構造におけるノード間のパスの重さに対応する。この変換により、完全グラフにおけるすべてのノード間の枝の重みは、それぞれのノードの関連性を表す。このような完全グラフを用いた簡単なコンテンツ分割結果の例を図 5 に示す。この例では、分割数が 2 で、分割後の 2 つのコンテンツの量が等しく、さらに、関連のあるオブジェクト (集合) どうしが同じ分割コンテンツになるようにグループ化されている。オブジェクトノードどうしを接続している実曲線は、完全グラフの枝を表しており、同じ分割コンテンツにグループ化されているオブジェクトノードどうしを連結している。このように、完全グラフの枝から、接続するオブジェクトノードどうしがグループを構成するようにカットする枝を選択し、コンテンツの分割を行う。

ここで、提案するコンテンツ分割方式では、2 章で述べた条件を考慮することから、分割の目的が一般のグラフ分割問題とは異なる。そのため、グラフ分割を評価する際の新たな指標が必要になる。そこで、分割結果の有効性を評価するための評価関数を導入する。評価関数 f は以下の式で表し、評価値が小さいほど条件を満たした分割であることを示す。ただし、 c_2, c_3, c_4 はそれぞれ f_2, f_3, f_4 の重みを決定するための係数であり、この値によってそれぞれの関数の重要度を調整できる。

$$f = f_1 \cdot (c_2 f_2 + c_3 f_3 + c_4 f_4)$$

f_1 は、配布するコンテンツを端末上で実行できない度合いを示す関数であり、以下の式で表す。

$$f_1 = \frac{W_o + W_f + 1}{W_a}$$

$$W_a = \sum_{i=0}^{n-1} v_i \quad (\text{コンテンツ全体の重みの総和})$$

n = オブジェクトノードの数

v_i = オブジェクトノード i の重み

W_o = 端末の容量を超えて割り当てられたオブジェクト (集合) の重みの総和

W_f = 割り当てられたオブジェクト (集合) のうち実行する機能がないものの重みの総和

f_2 は、分配したコンテンツ量が指定した割合にどれだけ遠いかを示す関数であり、以下の式で表す。

$$f_2 = \frac{1}{k} \sum_{i=0}^{k-1} \frac{(w_i - W_a \alpha_i)^2}{(W_a \alpha_i)^2}$$

k = 端末数

w_i = 端末 i に割り当てる実行可能なオブジェクト (集合) の重み

α_i = ユーザが指定した端末 i に割り当てるオブジェクト (集合) の割合

f_3 は、ユーザのコンテンツ割当て要求にどれだけ応えていないかを示す関数であり、以下の式で表す。

$$f_3 = \frac{N_u}{N_h}$$

N_u = ユーザの要求どおりに割り当てられなかったオブジェクトノードの数

N_h = ユーザが要求した割当て要求および非割り当て要求に含まれるオブジェクトノードの数

f_4 は、同じ分割コンテンツにグループ化されたオブジェクトノード間の関連性の低さを示す関数であり、以下の式で表す。

$$f_4 = \frac{\sum_{i=0}^{k-1} E_i}{E_a}$$

E_i = 端末 i に分配したオブジェクトノード集合から構成した最小スパンニング木における、すべての枝の重みの総和

E_a = メタデータをもとに作成した完全グラフのすべての枝の重みの総和

ここで、評価関数 f_1 は、配布するコンテンツを端末上で実行できない度合いを示しているため、 f_1 が小さいほど、ユーザに閲覧可能なコンテンツの量が多くなる。分割の良さを表す評価関数 f は、ユーザが閲覧可能なコンテンツの量に比例するのが妥当であり、他の関数とは積の関係となっている。また、分配したコンテンツ量が指定した割合にどれだけ遠いかを示す評価関数 f_2 、ユーザのコンテンツ割当て要求にどれだけ応えていないかを示す評価関数 f_3 、および、同じ分割コンテンツにグループ化されたオブジェクトノード間の関連性の低さを示す評価関数 f_4 は、それぞれ

```

Convert Tree into Complete Graph
Coordinate Graph
Create Initial k-way Partitioning
Compute  $f'$ 
 $minf = f'$ 
/*k-way FM Algorithm*/
Do While  $minf$  updates
  Unmark all nodes
  Do While Unmarked nodes exist
    Compute  $f$  in Unmarked nodes
    Find  $node_n$  and  $group_m$ 
    that minimize  $f$ 
    Move  $node_n$  into  $group_m$  and mark
  end Do
  Find groups that minimize  $f$  in the above loop
  and this minimum  $f$  is set to  $tmpminf$ 
  If( $minf > tmpminf$ )
     $minf = tmpminf$ 
  end If
end Do
    
```

図 6 コンテンツ分割アルゴリズム
Fig. 6 Content partitioning algorithm.

に独立している評価関数であるため、それらの和を求める形式が妥当である。

4.2 分割方式の手順

提案する分割方式では、まず、メタデータ木を完全グラフに変換した後、さまざまな条件を満たすよう完全グラフの調整を行い、初期のグラフ分割を行う。その後、複数分割に拡張した FM アルゴリズムを用いて、分割したグループ間でのオブジェクトノードの移動を行う。図 6 にアルゴリズムを示し、以下にその主要な点を説明する。

[Coordinate Graph]

グラフの調整は以下の手順で行う。

- (1) あるオブジェクトノードを実行するための機能を持つ端末が存在しない場合は、そのオブジェクトノードをグラフから削除する。
- (2) broadcast タグを持つオブジェクトノードがある場合は、あらかじめそのノードをすべての端末に割り当て、グラフから削除する。
- (3) 1つの端末に2つ以上のオブジェクトノードの割当て要求があった場合は、それらのオブジェクトノード間の枝の重みを0に設定する。これにより、要求されたオブジェクトノードが同じグループに割り当てられやすくなる。

[Create Initial k-way Partitioning]

初期のグラフ分割 (分割数 k) は以下の手順で行う。

- (1) 調整後のグラフから最小スパンニング木を求める。
- (2) 最小スパンニング木から $k-1$ 本の枝を以下の方法でカットすることにより、ノードを k 個のグループに分割する。

まず、1つ以上の割当て要求と、1つ以上の非

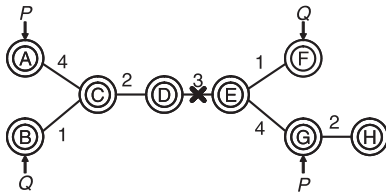


図 7 非結合要求を考慮したコンテンツ分割の例

Fig. 7 Example of content partitioning considering an uncombined request.

割当て要求があった端末において、割当て要求ノードと非割当て要求ノード間すべてに非結合要求を設定する。非結合要求とは、初期分割においてその要求が設定されたノードどうしを同じグループに割り当てないようにするものである。すべての最小スパニング木の枝の中から、より多くの非結合要求が設定されたノード間のパス上にあり、より重みの大きい枝を優先的に選択し、これをすべての非結合要求が満たされるまで、または、 $k-1$ 本の枝がカットされるまで繰り返す。カットする枝を選択する例を図 7 に示す。非結合要求 P がノード A と G 間に、非結合要求 Q がノード B と F 間に設定されている。この場合は、非結合要求をより多く満たし、重みの最も大きいノード D と E 間の枝がカットされる。

カットを $k-1$ 回行う前にすべての非結合条件が満たされた場合は、カットされていない枝のうちから重みの大きい枝からカットし、合計で $k-1$ 回カットするまで繰り返す。以上の手順で得られた k 個のグループを初期分割とする。次に、分割した k グループのコンテンツを端末に割り当てる際の可能なすべての組合せ中から、最小の評価値 f を持つ組合せを求め、これを初期分割における割当てとする。

- (3) 端末 $i (0 \leq i \leq k-1)$ に割り当てるオブジェクトノードの集合 $group_i$ を出力する。

[k -way FM Algorithm]

初期分割によって求めたグループ間でノードの移動を行い、さらに適切な分割を求める。

まず、ノード $n (node_n)$ が自身の属するグループ以外のグループ $group_m (node_n \notin group_m)$ に移動したすべての場合の評価値 f を、マークされていないすべてのノードについて求める。その中で最小の f を与えるノードの移動を選び、その移動を実行し、さらにそのノードをマークする。この手順をすべてのノードがマークされるまで繰り返す。この繰り返しの中で、

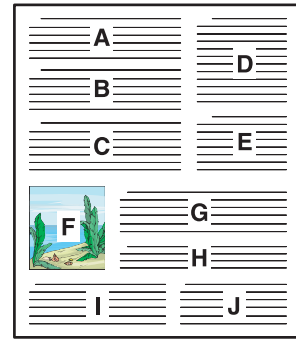


図 8 WWW ページの例

Fig. 8 Example of WWW page.

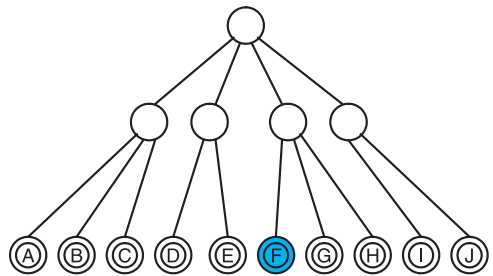


図 9 WWW ページのメタデータ

Fig. 9 Metadata of the WWW page.

最小の評価値を与えたグループ分けを暫定的な最適のグループ分けとする。次に、すべてのノードのマークを外したうえで、そのグループ分けからさらにグループ間でのノードの移動による分割の最適化を繰り返す。この繰返しは最小評価値の更新が行われなくなった時点で終了する。

4.3 分割方式の適用例

前節のアルゴリズムを用いて、図 8 に示す WWW ページを 3 つの端末に対して分割する例を示す。また、図 8 に示す WWW ページは、図 9 に示す木構造メタデータを持ち、A から J のオブジェクトノードが WWW ページ上の A から J のコンテンツに対応している。A~C, D~E, F~H, I~J のそれぞれは、WWW ページのレイアウト上で近くに位置し、内容的にも関連しているため、図 9 では 4 つの分類ノードにより各オブジェクトノードを分類している。さらに、A から J のオブジェクトノードのうち、F は画像ファイルを示し、端末 1 のみがこれを端末の画面上に表示できるものとする。その他の端末および分割の条件を表 1 に示す。

$c_2 = c_3 = c_4 = 1$ のときおよび $c_2 = c_3 = 1, c_4 = 3$ のときの結果を表 2 に示す。結果 1 で端末 0 に割り当てられていたオブジェクト G が、結果 2 では端末 1 に割り当てられている。これは、 $c_4 = 3$ と

表 1 端末と分割の条件

Table 1 Conditions of terminals and partitioning.

	容量	割り当てる コンテンツの比	割当て 要求	非割当て 要求
端末 0	-	5	J	H
端末 1	-	3	A	-
端末 2	2	2	-	-

表 2 割当て結果

Table 2 Results of allocation.

	結果 1 $c_2 = c_3 = c_4 = 1$	結果 2 $c_2 = c_3 = 1, c_4 = 3$
端末 0	B, C, G, I, J	B, C, I, J
端末 1	A, F, H	A, F, G, H
端末 2	D, E	D, E

することにより、オブジェクトどうしの関連性に重点が置かれたためであり、G と関連性の強いオブジェクト F と H と同じ端末に割り当てられている。

5. 評価

本章では、提案したアルゴリズムを用いたコンテンツ分割の評価を行う。

5.1 評価方法

評価実験では、コンテンツと端末および分割の条件を 300 セット生成した。コンテンツは、オブジェクトノード数が 15、深さが 2 の木となるように決定した。また、オブジェクトノードの重み、コンテンツを実行するために必要な機能、および、枝の重みはランダムに決定した。一方、端末の数は 2 から 5 の範囲でランダムに決定した。端末の持つ機能、端末ごとに割り当てるコンテンツ量の比、および、ユーザによる割当て、非割当て要求もランダムに決定した。 c_2, c_3, c_4 はすべて 1 とした。

提案方式 (MST-FM) の比較対象として、次の 3 つの方式についても同様の実験を行った。

- **EMST-FM** : 提案方式の Create Initial k-way Partitioning において、カットを $k - 1$ 回行う前にすべての非結合条件が満たされた場合、残りのカットしていないすべての枝から特定の条件を満たす枝を選択し、それらの枝をカットできるすべての組合せに対して評価値を求める。その中で最小の評価値を示す枝の組合せの枝をカットする。このとき、選択する枝は、まだカットしていない枝のうち、枝の重みが大きいものから、残りのカット回数を超える数までの枝とする。たとえば、重みが 4 の枝が 1 本、重みが 3 の枝が 2 本、重みが 2 の枝が 5 本まだカットされずに残っており、

残りのカット数が 2 の場合、重みが 4 と 3 の枝の 3 本を選択する。

- **RST-FM** : 提案した方式の Create Initial k-way Partitioning において、最小スパニング木の代わりにランダムに求めたスパニング木を用いる。
- **ERST-FM** : RST-FM と同様にランダムに求めたスパニング木をもとに分割を行い、EMST-FM と同様にカットする枝を選択する。

以上の評価環境において、以下に示す 8 つの評価項目について、300 セットの平均を求めた。

[評価値]

それぞれの分割方式における最終的な分割の評価値 f と、初期分割の段階における評価値 f' 。

[ノードの移動回数]

それぞれの分割方式におけるノード移動時におけるノードの移動回数 n_m 。

[計算時間]

それぞれの分割方式における最終的な分割の計算時間 T_p と、初期分割を終えるまでの計算時間 T_i 。ただし、実験には Pentium 4 (3.0GHz) の CPU を搭載した Red Hat Linux 9 マシンを用いた。また、シミュレータの実装には Java 言語を用いた。

[非結合条件が満たされるまでのカット数]

Create Initial k-way Partitioning において、非結合条件が満たされるまでに行ったカット数 n_c 。

[残りのカットできるすべての枝の組合せ数]

Create Initial k-way Partitioning において、EMST-FM と ERST-FM での、非結合条件が満たされた後、残りのカットできるすべての枝の組合せ数 n_p 。

[最小評価値が更新されなくなるまでの繰返し回数]

k-way FM Algorithm において、最小評価値の更新が行われなくなるまでの繰返し回数 n_i 。

[初期分割における評価値計算回数]

Initial k-way Partitioning における、分割したコンテンツの割当てでの評価値計算回数 n_{ei} 。

[k-way FM Algorithm での評価値計算回数]

k-way FM Algorithm における、評価値計算回数 n_{ek} 。

5.2 評価結果

評価実験の結果を表 3, 4, 5, 6, 7, 8, 9, 10 に示す。表 3 では、初期分割での評価値 f' は、ERST-FM, EMST-FM, MST-FM, RST-FM の順で大きくなっているが、評価値 f は、EMST-FM, MST-FM, ERST-FM, RST-FM の順に大きくなっている。このように MST-FM や EMST-FM の評価値が RST-

表 3 評価値

Table 3 Evaluation value.

	MST-FM	EMST-FM	RST-FM	ERST-FM
f	0.00349	0.00345	0.00353	0.00352
f'	0.05687	0.02866	0.06745	0.02242

表 4 ノードの移動回数

Table 4 Number of nodes' movements.

	MST-FM	EMST-FM	RST-FM	ERST-FM
n_m	6.83	4.58	7.84	6.34

表 5 計算時間 (msec)

Table 5 Computation time (msec).

	MST-FM	EMST-FM	RST-FM	ERST-FM
T_p	268.94	725.54	237.76	752.30
T_i	126.46	604.03	90.70	615.54

表 6 非結合条件が満たされるまでのカット数

Table 6 Number of cuts until all unconnected conditions are met.

	MST-FM	EMST-FM	RST-FM	ERST-FM
n_c	0.24	0.24	0.27	0.27

FM や ERST-FM よりも小さくなるのは、MST-FM や EMST-FM での初期分割で求めた解の近くにある局所最適解の評価値が、RST-FM や ERST-FM のそれより小さい場合が多いためであると考えられる。また、EMST-FM での初期分割で求めた解の評価値は、MST-FM での初期分割で求めた解の評価値より局所最適解に近く、同様に、ERST-FM での初期分割で求めた解は、RST-FM での初期分割で求めた解より局所最適解に近いと考えられる。このことは、表 4 における EMST-FM と ERST-FM の n_m がそれぞれ MST-FM と RST-FM より小さくなっていることから分かる。一方、表 8、表 10 での ERST-FM の繰返し回数 n_i と評価値計算回数 n_{ek} が RST-FM より大きくなっていることと、 n_m が小さくなっていることから、1 度移動したノードが繰り返して移動している場合が多いことが分かる。このように、ERST-FM は、初期分割の段階で最も小さい評価値を求めているにもかかわらず、RST-FM より多くの計算回数を必要とするため、提案方式で利用したノードの移動戦略に適していないと判断できる。

表 6 では、初期分割において最小スパニング木を用いている MST-FM と EMST-FM の n_c が小さくなっている。これは、ランダムなスパニング木を用いるよりも最小スパニング木を用いる方が非結合条件を効率良く満たすことを示している。また、表 7 に示すように ERST-FM の n_p が EMST-FM より大きいこと

表 7 残りのカットできるすべての枝の組合せの数

Table 7 Number of combinations of edges that have not been cut yet.

	EMST-FM	ERST-FM
n_p	57.79	64.97

表 8 最小評価値の更新が行われなくなるまでの繰返し回数

Table 8 Number of iterations until the minimum f is not updated.

	MST-FM	EMST-FM	RST-FM	ERST-FM
n_i	2.21	2.08	2.28	2.34

表 9 初期分割における評価値計算回数

Table 9 Number of f 's calculations in initial partitioning.

	MST-FM	EMST-FM	RST-FM	ERST-FM
n_{ei}	25.01	4723.15	25.01	5217.68

表 10 k-way FM algorithm での評価値計算回数

Table 10 Number of f 's calculations in the k-way FM algorithm.

	MST-FM	EMST-FM	RST-FM	ERST-FM
n_{ek}	516.8	481.6	532.8	554.8

から、表 9 では ERST-FM の n_{ei} が EMST-FM より大きくなっており、ERST-FM の f' が EMST-FM より小さくなっている。しかし、ERST-FM はカットする枝の組合せを多く考慮しなければならないため、表 5 に示すように EMST-FM と比べて、初期分割に要する計算時間 T_i が多くかかっている。さらに、EMST-FM と ERST-FM では、初期分割の際に考慮すべき枝の数が多いため、グラフの大きさが大きくなると、組合せの爆発を起こす。実際に、ERST-FM の T_i は、MST-FM や RST-FM の 4 倍以上かかっている。また、初期分割での評価値計算回数 n_{ei} においても、EMST-FM と ERST-FM の計算量が圧倒的に多くなっている。このように、EMST-FM と ERST-FM は、カットできる枝の組合せを考えようとして、分割したコンテンツを端末に割り当てる組合せも考えているため、初期分割での計算量が大きくなる。また、MST-FM は RST-FM に比べて T_i が多くかかっている。これは、最小スパニング木を求める手間のためである。ERST-FM での T_i が EMST-FM より多くかかっているのは、最小スパニング木を求める手間より、カットの組合せを求める手間が上回っているためであると考えられる。

本論文で提案する協調ブラウジングにおいては、コンテンツ分割の要求と同時にその結果を返す必要があるため、ある程度大きなコンテンツに対しては、計算時間が短い MST-FM や RST-FM の方が適している。

また、RST-FM は MST-FM に比べノードの移動回数が多く、オブジェクトノード数が増加するのに従って、k-way FM Algorithm における f の計算量が多くなる。これは、別の評価実験により、オブジェクトノード数が 25 のとき、MST-FM の計算時間の方が短いことを確認している。さらに、RST-FM はランダムなスパニング木を用いているため、計算時間や評価値がそのスパニング木に大きく依存し、つねに安定した性能を示すことができない。したがって、協調ブラウジングに用いる分割方法として、提案方式 MST-FM が最も適していると考えられる。

6. 考 察

本章では、協調ブラウジング、および、提案方式について考察する。

6.1 協調ブラウジングについて

協調ブラウジングを行うことにより、身の回りにある端末を用いてサービスを受けることができるため、ユーザは小型で非力な携帯端末だけでは受けることができないサービスを受けることができる。また、それぞれの端末の条件に従って分割されたコンテンツの閲覧を行うことができる。

しかし、実際に身の回りにある端末を用いてブラウジングを行う場合、身の回りにある利用可能な端末を検索する機構や、ユーザごとの端末へのアクセス権などを管理する機構が必要となる。また、1つの端末に複数ユーザが同時に利用要求を行った場合などは、ユーザの身の回りにある他の端末と連携して、ユーザごとのサービスの品質を保つ機構を実現する必要がある。さらに、身の回りにある端末だけでは品質の高いサービスを利用できない場合は、サービスの質を必要最小限まで劣化させる機構が必要となる。

ここで、1つのアプリケーションを協調して利用するユーザを、友人同士に限定するのではなく、不特定多数のユーザと利用することにより、より柔軟な協調ブラウジングを実現できる。たとえば、現在参加しているユーザが所持する携帯端末の中に、あるオブジェクトの実行に必要な機能を持つものがない場合、その機能を備えた端末を所持する人を、周囲の中から募ることが考えられる。また、ユーザ間でお互いの携帯端末の画面を見せ合うなどのインタラクションがあるため、ユーザ間のコミュニケーションを図ることもできる。ただし、実環境において協調ブラウジングを運用するためには、セキュリティやプライバシーの問題を考慮する必要がある。たとえば、セキュリティの問題では、他のユーザから送信されたメッセージに不適切な

内容が含まれていることなどが考えられ、プライバシーの問題では、ユーザのプロファイルを用いてコンテンツを分割すると、他のユーザにそのプロファイルの内容を知られてしまうことなどが考えられる。したがって、協調ブラウジングの実現には、他ユーザの信頼性を保障するために、アプリケーションの実行前にユーザの認証を行うなどの機構が必要になるものとする。

6.2 コンテンツ記述方式と分割方式について

提案したコンテンツ分割方式では、分割した後のグループを端末に対して割り当てる際に、すべての組合せを考慮するため、端末数が増えると計算量が多くなる。今後は、この問題について検討する必要がある。

さらに分割方式では、ユーザの割当て要求や非割当て要求などの条件により、関連性のあるオブジェクトどうしが別の端末に割り当てられることがある。たとえば、協調 WWW ブラウジングの場合、ある画像とその解説文が別の端末に割り当てられることなどが考えられる。そこで、このような分割が行われた場合でも、ユーザが快適にブラウジングできるように、コンテンツの提示方法を工夫する必要がある。たとえば、協調 WWW ブラウジングの場合、自身の端末に表示されたコンテンツに関連し、他の端末に表示されたコンテンツを閲覧するために、その端末を所持するユーザ名を提示したり、そのコンテンツを自身の端末に表示するためのリンクなどを提示したりすることが必要になる。

また、本論文では、コンテンツを木構造で記述することを想定しているが、グラフ構造やネットワーク構造など、木構造以外のコンテンツを記述・分割できるように拡張する必要がある。

6.3 実行順序があるコンテンツについて

提案したコンテンツ記述方式では、オブジェクトどうして実行の期間に重なりがある場合、各オブジェクトに優先度を設定することにより、重要なオブジェクトを優先的に実行することができる。しかし、優先度の低いオブジェクトが優先度の高いオブジェクトと同じ端末に割り当てられた場合、優先度の低いオブジェクトを実行することができない場合がある。そこで、優先度の低いオブジェクトでも実行できるように、実行スケジュールに重なりのあるオブジェクトどうしをできるだけ別々の携帯端末に割り当てる必要がある。オブジェクトどうして実行時間に重なりがないような分割方法の検討は、今後の重要な課題の1つである。

7. おわりに

本論文では、1つの端末では閲覧できないようなリッ

チなコンテンツを、複数の端末を用いて閲覧する協調ブラウジングを実現するためのコンテンツ記述方式とコンテンツ分割方式を提案した。提案したコンテンツ記述方式を用いることにより、コンテンツの分割に必要な条件を記述することができる。また、提案したコンテンツ分割方式を用いることにより、指定した分割条件を考慮した分割を行うことができる。

本論文の提案方式は、コンテンツ分割を主目的としているため、オブジェクト実行優先度などの指定はできるものの、実行時の動作制御やレイアウトは実装依存としている。協調ブラウジングという観点からは、動作制御やレイアウトの指定も重要な機能と考える。これらの実現は今後の重要な課題の1つである。また、オブジェクトの実行時間にできるだけ重なりがないような分割を行う方式も、今後検討する必要がある。さらに、協調ブラウジングを用いたアプリケーションの実装を行い、実際の利用環境において分割に必要な条件などを再検討する予定である。

謝辞 本研究の一部は、平成16年度受託研究(独立行政法人情報通信研究機構)「モバイル端末による協調ブラウジングのためのアプリケーション構築に関する研究」、および文部科学省21世紀COEプログラム(研究拠点形成費補助金)の研究助成によるものである。ここに記して謝意を表す。

参 考 文 献

- 1) Anderson, C.R., Domingos, P. and Weld, D.S.: Personalizing web sites for mobile users, *Proc. WWW'01*, pp.565-575 (2001).
- 2) Fiduccia, C.M. and Mattheyes, R.M.: A linear time heuristic for improving network partitioning, *Proc. Design Automation Conference*, pp.175-181 (1982).
- 3) Kernighan, B.W. and Lin, S.: An efficient heuristic procedure for partitioning graphs, *Bell System Technical Journal*, Vol.49, No.2, pp.291-307 (1970).
- 4) Masuoka, R., Labrou, Y., Parsia, B. and Sirin, E.: Ontology-enabled pervasive computing applications, *IEEE Intelligent Systems*, Vol.18, No.5, pp.68-72 (2003).
- 5) Sanchis, L.A.: Multiple-way network partitioning, *IEEE Trans. Comput.*, Vol.38, No.1, pp.62-81 (1989).
- 6) Sanchis, L.A.: Multiple-way network partitioning with different cost functions, *IEEE Trans. Comput.*, Vol.42, No.12, pp.1500-1504 (1993).
- 7) Scholtz, J.: Ubiquitous computing goes mobile, *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol.5, Issue 3, pp.32-38 (2001).
- 8) Suaris, P.R. and Kedem, G.: Quadrisection: A new approach to standard cell layout, *Proc. International Conference on Computer-Aided Design*, pp.474-477 (1987).
- 9) Suaris, P.R. and Kedem, G.: Standard cell placement by quadrisection, *Proc. International Conference on Computer-Aided Design*, pp.612-615 (1987).

(平成15年9月25日受付)

(平成16年1月19日採録)

(担当編集委員 石川 博, 市川 哲彦, 原 隆浩,
佐藤 聡, 土田 正士)



前川 卓也

2003年大阪大学工学部情報システム工学科卒業。2004年同大学院情報科学研究科博士前期課程修了。現在、同大学院情報科学研究科博士後期課程在学中。モバイルコンピューティング、ユビキタスコンピューティングに興味を持つ。



上向 俊晃

2000年大阪大学工学部情報システム工学科卒業。2001年同大学院工学研究科博士前期課程修了。現在、同大学院工学研究科博士後期課程在学中。モバイルコンピューティング、ユビキタスコンピューティング、アドホックネットワークに興味を持つ。



原 隆浩(正会員)

1995年大阪大学工学部情報システム工学科卒業。1997年同大学院工学研究科博士前期課程修了。同年同大学院工学研究科博士後期課程中退後、同大学院工学研究科情報システム工学専攻助手、2002年より同大学院情報科学研究科マルチメディア工学専攻助手となり、現在に至る。工学博士。1996年本学会山下記念研究賞、2000年電気通信普及財団テレコムシステム技術賞、2004年本学会研究開発奨励賞を各受賞。データベースシステム、分散処理に興味を持つ。IEEE、電子情報通信学会、日本データベース学会の各会員。



西尾章治郎(フェロー)

1975年京都大学工学部数理工学科卒業。1980年同大学院工学研究科博士後期課程修了。工学博士。京都大学工学部助手、大阪大学基礎工学部および情報処理教育センター助教授、大阪大学大学院工学研究科情報システム工学専攻教授を経て、2002年より同大学院情報科学研究科マルチメディア工学専攻教授となり、現在に至る。2000年より大阪大学サイバーメディアセンター長、2003年より大阪大学大学院情報科学研究科長を併任。この間、カナダ・ウォータールー大学、ビクトリア大学客員。データベース、マルチメディアシステムの研究に従事。現在、ACM Trans. on Internet Technology, Data & Knowledge Engineering, Data Mining and Knowledge Discovery, The VLDB Journal等の論文誌編集委員。情報処理学会フェローを含め、ACM、IEEE等9学会の会員。

