

確率モデルを用いた コンテキストウェアシステムの性能評価

柴山 裕輝 奥田 隆史 山崎 重行 井手口 哲夫 田 学軍

愛知県立大学大学院情報科学研究科

ITS 社会の実現に向け、ユーザに合わせた動作をするコンテキストウェア・アプリケーションシステムが注目されている。そのシステムでは、ユーザに最適な情報やサービスを提供する確率(コンテキストウェアネス確率)を高めることが重要である。そこで、本稿では、コンテキストウェア・アプリケーションシステムの導入・運用において、コンテキストウェアネス確率の変動する場合を想定し、確率モデルを用いて、コンテキストウェア・アプリケーションシステムの性能評価を行う。

Performance Evaluation of two Context-aware Systems by using Probability Models

Yuuki SHIBAYAMA Takashi OKUDA Shigeyuki YAMAZAKI
Tetsuo IDEGUCHI Xuejun TIAN

Graduate School of Information Science and Technology, Aichi Prefectural University

Context-aware service application systems are important to realize intelligent transport systems (ITS). In the applications, it is necessary to find out behavior and environments of user to provide useful information for users. In this paper, we present probability models for context-aware service application systems, evaluate the performance of the systems and show response time and context-awareness probability of the systems.

1. はじめに

ITS(Intelligent Transport Systems)は、自動車と道路交通環境を統合的に知能化、情報化することによって、安全で快適な交通システムを実現しようというものである[1]。ITS は単一システムではなく、様々なサブシステム、例えば、ナビゲーションシステムの高度化、自動料金収受、安全運転の支援、交通管理の最適化、道路管理の効率化、公共交通の支援、商用車の効率化、歩行者の支援、緊急車両の運行支援などを行うシステム集合体で

ある。

本稿では、これらのサブシステムの本質は、ユーザの状況に合わせて動作するコンテキストウェア・アプリケーションシステムとして考える。実際、コンテキストとは広義では「場所、時間、その他の様々な状況・環境・人の嗜好など」を意味する。コンテキストウェア・アプリケーションシステムとは、これらのコンテキストを認識し、それに適した形でプログラムの挙動を適応できる

アプリケーションのことである[2]。

このアプリケーションでは、コンテキストウェアアネス(Context Awareness, コンピュータやネットワークがコンテキストを認識すること)が重要である。すなわち ITS システムが社会に違和感なく受け入れられるためには、コンテキストウェアアネスの向上が必要となる。

本稿では、コンテキストウェア・アプリケーションシステムがユーザの意図した通りに働かないケースを想定し、それを踏まえた性能評価を行う。以下、2. では、コンテキストウェアの概要とコンテキストウェア・アプリケーションシステムの構成について述べ、コンテキストウェアネス確率について定義する。3. では、本稿で提案する性能評価モデルについて説明する。4. では、確率モデルを用いた性能評価をシミュレーションで行う。最後に、5. でまとめと今後の課題について述べる。

2. コンテキストウェア

コンテキストウェアとは、ユビキタス社会を実現するための方法の1つで、周辺状況などのコンテキストを理解し、それに応じて動作するものである[2]。

コンテキストウェアが実現すると、コンピュータがユーザの意図を読み取り、ユーザの行動に合わせて動作することが可能になると考えられている [3]。ITS におけるコンテキストウェアなアプリケーションシステムには、例えば、カーナビゲーションシステムがある。高度なカーナビゲーションシステムでは、運転者の要望に応じた目的先を自動的に検索し、提示することなどが考えられる。このようなコンテキストウェアなアプリケーションシステムをコンテキストウェア・アプリケーションシステムと定義する。

コンテキストウェア・アプリケーションシステムの情報通知タイミングの決定方法としては、あらかじめ時間や場所に対応する動作を決定しておく方法がある。しかし、様々な状況が存在するため、この方法だけで適切な通知を行うことは難しい [4]。したがって、ユーザ認証を行ってより適切な通知を行う必要がある。

本稿では、コンテキストウェア・アプリケーションシステムを認証システムとコンテキストウェアシステムから構成されたと考える。

2.1. 認証システム

認証システムは、ユーザ情報を認証する。その際認証方式が問題となる。本稿では、従来方式(図2)と EMAPP 方式(図3)について考える。なお、本稿においては、認証は毎回行い、ユーザは必ず認証に必要なユーザ情報を提供するとする。

従来方式

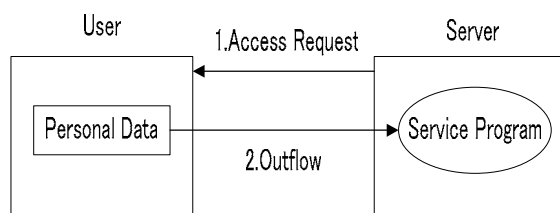


図2 Conventional method

従来使用されていたユーザ情報認証方式は、以下の手順に従うとする。

1. ユーザはサーバからユーザ情報を要求される。
2. ユーザは要求されたユーザ情報をサーバに送信する(送信時にユーザ情報は暗号化されるが、サーバ側で復号され、元のユーザ情報が利用可能となってしまう、ユーザ情報流出の危険性もある)。
3. サーバはユーザから受信したユーザ情報からユーザに最適なデータを選択し、ユーザに送信する。

エージェント流入型プライバシー保護方式(EMAPP)[5]

文献[5]で最近提案された方式として EMAPP がある。EMAPP(Encapsulated data and Mobile Agent based Privacy Protection)とは、モバイルエージェント(MA)流入型プライバシー保護方式である。EMAPP 方式の動作を図3に示す。

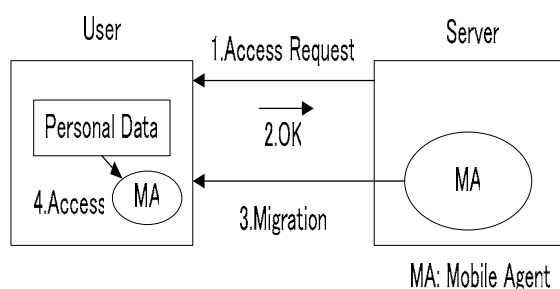


図3 EMAPP method

EMAPP 方式は、以下の 1~5 の動作手順に従う。

1. サーバは MA 送信許可を要求する。
2. ユーザは MA 送信を許可する。
3. サーバはユーザに対して MA を送信する。
4. MA は ES 内で動作し、ユーザの必要なユーザ情報のみを暗号化し、サーバに送信する。
5. サーバは MA から送られてきた暗号化された情報を符号表によって照合し、その情報を元にユーザに最適な情報を選択し、ユーザに送信する。

上記のように、MA はユーザ端末で動作し、必要なデータのみをハッシュ化してサーバに送る。サーバは MA によって送られてきたデータを自身の持つハッシュ表と照合して送信内容を知る。EMAPP 方式では送信される内容はあらかじめ決めておく必要がある。EMAPP 方式の採用によりユーザの安全な空間からユーザ情報がそのままの形式で流出することはなくなる。しかしながら、MA 使用によるユーザ端末の負荷の増加、サーバにおけるハッシュ値の照合のために処理負荷の増加が起こる。このため、EMAPP 方式ではユーザが増加するほど従来方式に比べ処理負荷が大きくなると予想される[5]。

2.2. コンテキストウェアシステム

本稿で扱うコンテキストウェアシステムについて述べる。本稿ではコンテキストウェアシステムは、ユーザ端末のユーザ情報を元にユーザに送信する情報を選択する。その際コンテキストウェアネス確率が高いことが重要となる。コンテキストウェアネス確率とは、コンピュータやネットワークが状況や人の嗜好などのコンテキストを認識し、適応して動作する確率とする。つまり、ユーザの求める情報を示す確率をコンテキストウェアネス確率と定義する。

3. 性能評価モデル

本稿では、ITS におけるコンテキストウェア・アプリケーションシステムとしての情報配信システムにおいて図 4 のような車載情報端末に情報配信するシステムを考える。このシステムは、図 4 の 1~4 のように動作するとし、図 5 のフローチャートで表現できる。以下に 1~4 の手順を述べる。

1. 情報端末がサービスセンターに情報要求する。
2. サービスセンターは情報端末に情報を送信す

る。

3. 情報端末に表示された情報をユーザが見る。ここでユーザが満足した場合(OK)は終了する。不満を感じた場合(NG)は 4 へ進む。
4. ユーザは情報端末で再送信要求をする。以下、1~4 の繰り返しとなる。

すなわち、図 4 は図 5 に示すフローチャートで表現することができる。なお、図 5 の斜線部分は認証部分に相当し、従来方式と EMAPP 方式で違いがある。

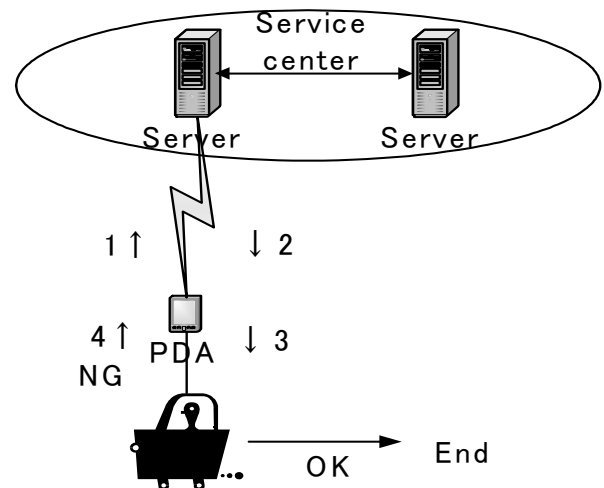


図 4 Evaluation model

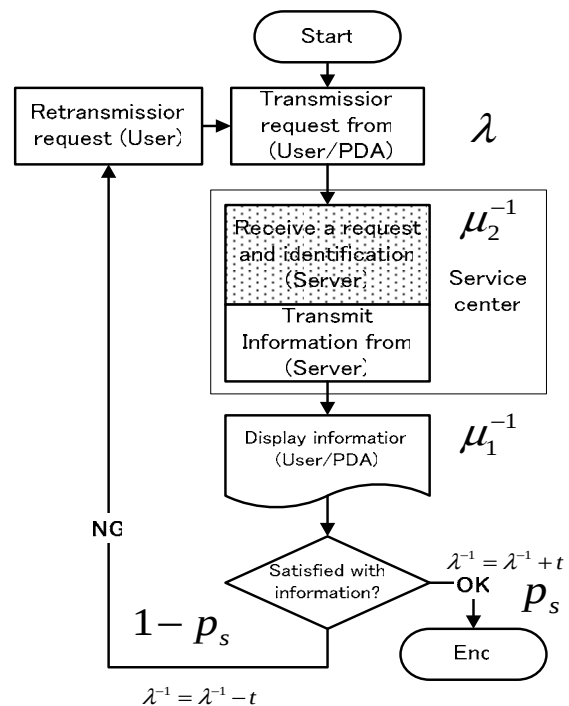


図 5 Conventional method

フローチャートにおいて、まず情報端末から呼が発生する。呼数の発生個数は到着率 λ 個/分のポアソン分布に従う。発生した呼はサービスセンターに到着する。サービスセンターで、サーバは n 個の情報の中から1つを三角分布(最小値 a ,最大値 b ,最頻値 c)に従い選択し、情報端末に送信する。本稿のシミュレーションではサーバが適切に情報を選ぶと仮定するので、ユーザが最も求める情報は c とする。

サーバが情報を選択し、選択した情報が情報端末に届くまでの時間を平均 μ_1^{-1} 分の指数分布とする。情報端末がサーバから受信した情報はユーザによって確率 p_s で OK となり、確率 $1-p_s$ で NG となる。以下 p_s を満足度と呼ぶ。

OK の場合は情報端末からの発生間隔が t 分長くなるとし、NG の場合は情報端末からの発生間隔が t 分短くなるとした。

また、EMAPP 方式のモデルでは、サーバからユーザ端末へのMAの送信とユーザ端末のMA処理にかかる合計時間を平均 μ_2^{-1} 分の指数分布とする。

4. 数値例

今回、シミュレーションには株式会社構造計画研究所の Visual SLAM[6]を用いた。なお、シミュレーション結果として、1000 単位時間の試行を 10 回行った結果の平均値を用いる。

評価方法としては、満足度(p_s)を 0.1~0.9 まで変動させた場合の OK, NG の数、レスポンスタイムを求める。その方法として、乱数を発生させ、あらかじめ決めた値になった時に OK, それ以外の時に NG として、OK, NG の数を集計し、その際のレスポンスタイムを計測する。サービスセンターはユーザに対する情報として、三角分布に従う数から 1 つの数を選択し、情報端末に返す。所要時間は平均 μ_1^{-1} 分の指数分布に従う。

EMAPP 方式は、基本的に従来方式と同じであるが、呼の発生後、MA が転送され、実行される時間を平均 μ_2^{-1} 分の指数分布と置いている。

EMAPP 方式では、MA 転送と MA による処理の時間を、処理時間が長い場合[2分(EMAPP(2))], 中間の場合[1分(EMAPP(1))], 処理時間が短い場合[30秒(EMAPP(0))]の3つの場合で考えた。

NG の時、再送信要求を行うため、 $\lambda^{-1} = 0$ 分を下限として $t = 0.1$ 分小さくなり、通信発生間隔が短くなる。また、OK の時はそれ以上再送信要

求を行わないので、 λ^{-1} が初期値を上限として $t = 0.1$ 分大きくなり、通信発生間隔が長くなる。

表 1 パラメータ設定

呼の発生間隔	λ^{-1}	5 (分)
サーバが持っている情報	n	10 (個)
三角分布の最小値	a	0
三角分布の最大値	b	10
三角分布の最頻値	c	5
サーバから情報端末への情報到達時間	μ_1^{-1}	1 (分)
MA による遅延	μ_2^{-1}	2, 1, 0.5 (分)

以下にシミュレーション結果を載せる。図 6 は従来方式での、各満足度に対する OK, NG の反応数を表している。満足度が上がるにつれ、OK の数は 80 付近であまり変化がないのに対し、NG の数は 60 から 0 付近へと大幅に下がっている

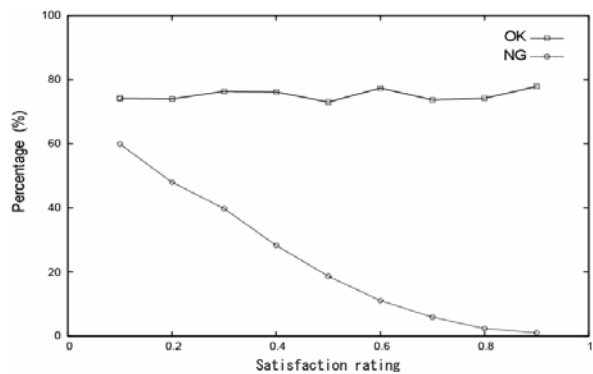


図 6 Number of OK and NG
(Conventional method)

図 7 は、従来方式での総集計数の内の OK, NG の割合とレスポンスタイムを表現している。図 7 の左縦軸は OK, NG の割合、右縦軸はレスポンスタイムとなっている。

図 7 の結果より、満足度が低い時はレスポンスタイムが長く、満足度が高くなるにつれ、レスポンスタイムが短くなっていることが分かる。

これは満足度が低い時には NG が多いために再送信要求が多くなり、そのことによる負荷が高くなるためだと思われる。

このことより、ユーザが満足する条件を適切に決定した場合、満足度が高いほど通信にかかる負荷は低くなっているといえる。

図 8 は EMAPP 方式を用いた場合の各満足度に対する OK, NG の数を表している。

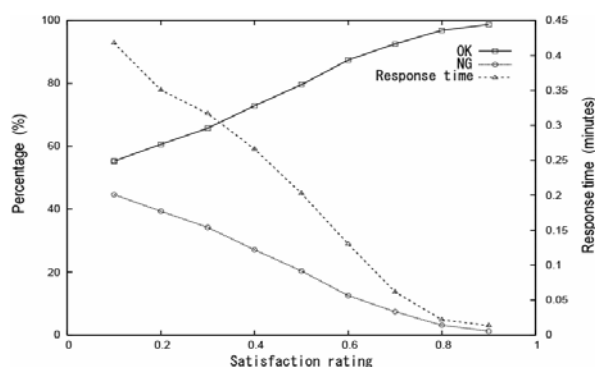


図 7 Response time and percentage of OK and NG(Conventional method)

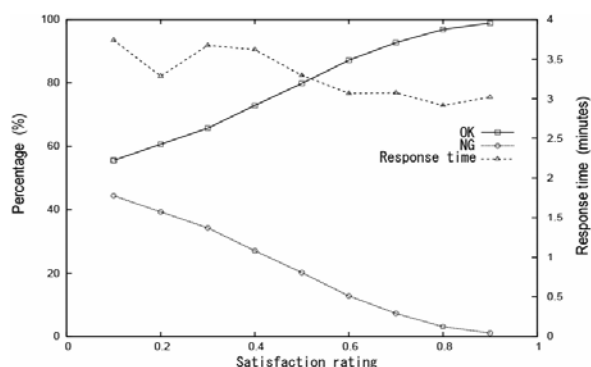


図 9 Response time and percentage of OK and NG(EMAPP(2))

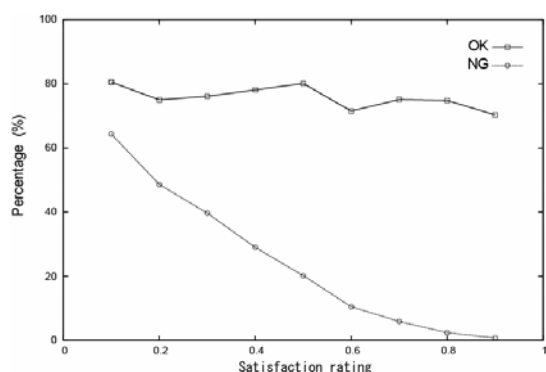


図 8 Number of OK and NG(EMAPP(2))

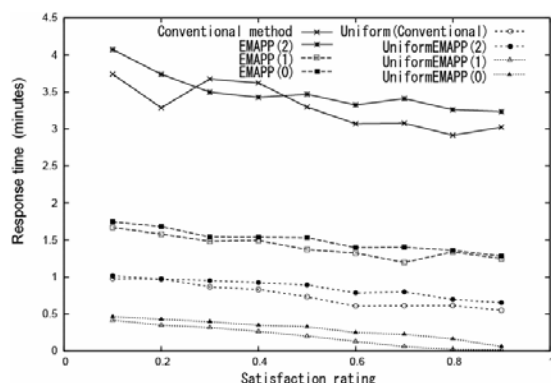


図 10 Response time of each methods

従来方式と同じく、EMAPP 方式でも満足度が上がるにつれ、OK の数に変化はあまりないのに対し、NG の数は下がっている。

図 9 は EMAPP 方式における総集計数の内の OK、NG の割合とレスポンスタイムを表現している。満足度によるレスポンスタイムの変化はほとんどなかった。これは、EMAPP 方式では通信時間よりも MA の流入、MA による作業に時間がかかることを示していると考えられる。

図 10 は各方式のレスポンスタイムを示している。各レスポンスタイムを比較すると、従来方式が最も短かった。また、図 10 において各方式のレスポンスタイムの最大値をそれぞれ 100%とした時の各満足度における違いを見た時、従来方式の下がり幅が最も大きく、EMAPP 方式の下がり幅は MA の作業時間に関わらずほぼ同じであった。

図 10 には比較のため、0 以上 10 以下の値をとる一様分布による乱数発生を行ったものを掲載している。一様分布を用いると各情報が等確率で選択されるため、コンテキストウェアな方式に比べ NG の回数が多くなり、元の方式と比べ、レスポンスタイムが長くなる結果となった。

5. まとめ

本稿では、ITS を認証システムとコンテキストウェアシステムからなるコンテキストウェア・アプリケーションシステムとして捉え、性能評価した。

レスポンスタイムの短さを重視すると、従来方式が最も優れている。しかし、サービス利用時のユーザ情報の安全性を重視すれば、EMAPP 方式のほうが優れているといえる。また、MA の流入、作業にかかる時間が短ければより短いレスポンスタイムでサービスを利用することが可能であると分かった。

今後の課題としては、従来方式、EMAPP 方式においてランダムな結果を返した場合のシミュレーションを行い、結果の比較を行う必要がある。さらに、ユーザの満足度とレスポンスタイムからどれくらいの満足度ならシステムとして成立するかを求める必要がある。

謝辞

本研究は平成 17 年度中部電力基礎研究所研究助成金の一部を用いて行われた。本研究を進める

にあたり，EMAPP に関して貴重なご意見を下さった国立情報学研究所の山田茂樹氏，田中聡氏に感謝いたします．

参考文献

- [1] 松下温，屋代智之，**ITS と情報通信技術**，NetCom ライブラリ，裳華房，2003．
- [2] 中島秀之，車谷浩一，伊藤日出男，“ユビキタス情報処理による社会支援”，**情報処理**，Vol. 45，No.9，pp.907-911，2004．
- [3] 山口明，大橋正良，小塚秀宣，嶋崎佳史，森川大輔，本庄勝，入内嶋洋一，“電子タグと携帯端末を用いるプレゼンスサービスの検討”，**情報処理学会MBL研究会**，Vol.2004 No.44，pp.1-5，2005．
- [4] 田島敬士，内藤栄一，小澤順，“通知に対する反応履歴を用いた通知タイミング学習方法”，**人工知能学会第 19 回全国大会**，1C1-01，pp.1-2，2005．
- [5] 田中聡，山田茂樹，上岡英史，その他，“ユーザ指向タイムクリティカルネットワーク実現に向けたエージェント流入型プライバシー保護方式”，**情報学ワークショップ 2005 論文集**，pp.249-253，September 2005．
- [6] 森戸晋，相澤りえ子，貝原俊也，**Visual SLAM によるシステムシミュレーション**，(株)構造計画研究所，March 2001．
- [7] 柴山裕輝，前田郁夫，奥田隆史，井手口哲夫，田学軍，“ネットワーク・ロボット連携型ホームサーバシステムの性能評価”，**情報学ワークショップ 2005 論文集**，pp. 153-156，September 2005．
- [8] 児玉正憲，**確率モデルの数理分析 - 生産・在庫モデルと信頼性・待ち行列モデル -**，(財)九州大学出版会，April 1996．
- [9] Sheldon M. Ross, *Probability Models for Computer Science*, Harcourt/Academic Press, 2002．
- [10] Vidyadhar G. Kulkarni, *Modeling, Analysis, Design, and Control of Stochastic Systems*, Springer-Verlag, 1999．