

コンテキスト依存型サービスにおけるスケーラブルな実行制御方式

内田 渉[†] 笠井 裕之[†] 倉掛 正治[†]

[†] 株式会社 エヌ・ティ・ティ・ドコモ ネットワーク研究所 〒239-8536 神奈川県横須賀市光の丘 3-5
E-mail: †{uchida,kasai,kurakake}@netlab.nttdocomo.co.jp

あらまし コンテキストが予め定められた実行条件を満たす場合に実行されるサービス（コンテキスト依存型サービス）では、適切なサービスを各時点で実行するために膨大な数のコンテキスト取得や収集などの実行制御動作が必要となる。本稿では、実行条件が満たされた適切なサービスを実行するとともに、実行制御動作の総数を大幅に削減するサービス実行制御方式を提案する。提案方式は各コンテキストの値の確率分布を時系列の形でモデル化し、それを用いてサービス実行に適した状態である確率を求める。そして、その確率が高い時刻にのみ適切なコンテキストの取得と収集を行う。また、提案方式のシミュレーション実験による評価結果を示し、適切なサービスを実行するとともに、単純な方式と比較して実行制御動作の実行数を大幅に削減することを示す。

キーワード コンテキスト依存型サービス, 実行制御, スケーラビリティ

Scalable Execution Control Method in Context-dependent Services

Wataru UCHIDA[†], Hiroyuki KASAI[†], and Shoji KURAKAKE[†]

[†] Network Laboratories, NTT DoCoMo Inc. Hikarinooka, Yokosuka-shi, Kanagawa, 239-8536, Japan
E-mail: †{uchida,kasai,kurakake}@netlab.nttdocomo.co.jp

Abstract A service which is triggered when context is satisfied with a pre-defined execution condition, context-dependent service, needs innumerable service execution control operations such as context acquisitions and collections to execute appropriate services at appropriate timings. We propose a new service execution control method which reduces the cost of execution control operations tremendously while it surely executes appropriate services when their execution conditions are satisfied. In our method, values of each context source is modeled by a probabilistic time series, and thus the probability that the execution condition is satisfied can be calculated. Appropriate context is acquired and collected when the probability is sufficiently large. Moreover, we show results of simulation experiments. The results demonstrate that our method can reduce the cost of context monitoring operations tremendously comparing with the naive method without a loss of the service quality.

Key words context-dependent services, execution control, scalability

1. はじめに

GPS (Global Positioning System) や非接触型 IC カードなどのコンテキストの取得機能を備えた端末の登場により、コンテキスト依存型サービス [3], [7] を提供可能な環境が実現されつつある。コンテキストとは人や場所、オブジェクトなどのエンティティの状態を特徴づけ、ユーザとアプリケーション間の相互作用に利用可能なあらゆる情報を指す [3]。コンテキスト依存型サービスはユーザやその周囲の状況に応じて振る舞いを変え、実世界におけるユーザの行動を支援する。

本稿で想定するコンテキスト依存型サービスの例を次に示す。

- ユーザの位置と空席状況に基づくプッシュ型飲食店内サービス

飲食店を探すユーザを支援するサービス。サービスは店舗とユーザ間の距離と、店舗の空席状況の二つのコンテキストを常に把握する。ユーザを受け入れるための空席がある店舗のうち、ユーザが接近し来店候補となった店舗を推奨するメッセージをユーザの端末に自動的に出力する。ユーザと店舗間の距離はユーザの端末に備えられた GPS などの測位装置の測位結

果を用いて計算される。空席状況は、店舗に配置された POS (Point Of Sales) システムとの連携などにより把握する。

- ユーザのプレゼンスに基づくメッセージ配信サービス

施錠時の個人認証等に非接触型 IC カードが使用されているオフィスにおいて提供されるサービス。このサービスでは各所の非接触型 IC カード利用履歴を用いて各従業員がどの部屋にいるかというコンテキスト (プレゼンス) を把握する。ユーザは、配信したいプレゼンスの条件と宛先を指定して、メッセージをサービスに登録する。プレゼンスが配信条件に合致する状態へ変化した場合、指定された宛先にメッセージを配信する。配信条件の例としては、「会議室に誰もいなくなったら」や「従業員 A が自席に戻ったら」が挙げられる。

これらのようなコンテキスト依存型サービスの目標を達成するためには、膨大な数の候補から適切なサービスを選択し、それらを適切な時刻に提供する必要がある。従って、できる限り多くのサービス候補に対してコンテキストを常に監視し、実行が適切か否かを判断する必要がある。例えば上記のプッシュ型飲食店内サービスではメッセージ配信に適した時刻を捉えるため、常に測位を実行し、POS システム等から現在の空席状

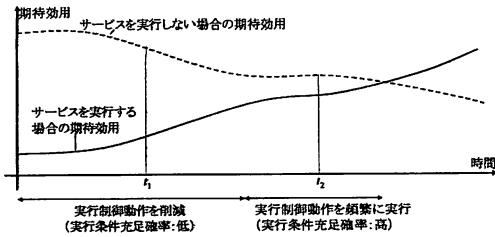


図2 提案方式における実行制御動作の削減

Fig.2 Reduction of execution control operations in our method.

減することにより、このトレードオフの関係を改善することが課題となる。

3. 提案方針

ここでは、提案サービス実行制御方式における実行制御動作削減の方針を述べる。

3.1 実行条件充足確率に基づく実行制御動作の削減

2.2に述べたように、低頻度で定期的に実行制御動作を行った場合、サービス実行の遅延、実行機会の喪失などの確率が生じる。しかし、これらの確率は時間により変化する。例えば、あるサービスが複数のコンテキストを用いてサービス実行時刻を判定しており、各コンテキストの値に対して与えられる条件(部分条件)が全て満たされた場合に実行されるものとする。部分条件のうち、ごく一部しか満たされていないことをサービス提供サーバが確認した直後の時刻は、条件に関する残りのコンテキストの取得装置から値を収集しない場合にもそれら全ての値が条件を満たす確率は低い。従って、その時刻にサービスの実行機会を失う確率も低い。

提案方式はこのような実行制御により実行に適した時刻であるか否かの判断を行わない間に、サービスを実行する場合の期待効用がサービスを実行しない場合の期待効用よりも高くなるような値を各コンテキストがとる確率(サービス実行条件の充足確率)を計算する。そして、その確率が十分低い場合のサービス実行制御動作を削減する。これにより、サービス実行の遅延やサービス実行機会の喪失を防ぎつつ、不要なサービス実行制御動作を削減する。

図2はサービス実行条件充足確率の変化と提案方式のサービス実行制御の頻度の変化を表している。実線および点線はそれぞれ、サービス提供サーバが全コンテキストの値の変化を把握していた場合のサービス実行およびサービスを実行しない場合の期待効用の変化を示している。時刻 t_1 の前後の時刻では、実行制御を行い、実行に適した時刻であるか否かの判断をコンテキストの値に基づいて計算しないと、サービス実行の期待効用がサービスを実行しない場合の期待効用よりも高い値をとる確率は低い。そのため、その間の実行制御動作は削減される。反対に時刻 t_2 の前後では、実行制御動作を行わない間にサービス実行の期待効用がサービスを実行しない場合の期待効用よりも高い値を取る確率が高いため、実行制御動作を頻繁に行う。

3.2 コンテキストとコンテキスト取得装置の特徴

実行条件の充足確率を計算するために、提案方式は次に述べるようなコンテキストとその取得装置における特徴を利用する。

● コンテキストの値の予測可能性の違い

コンテキストによって、将来の値を予測することが可能な場合と、予測が難しい場合がある。例えば歩行しているユーザの位置は、ある時刻における値を確認した場合、1分後の位置はその位置から数十メートル以内の近い確率が高いと予測できる場合がある。反対にある店舗への来客数などは、予測が難しく、様々な値をとる確率がある。

● 実行判断に対するコンテキスト間の影響度の違い

コンテキストの値の予測結果によっては、各コンテキストの取得結果がサービス実行の判断に与える影響が異なる場合があ

る。例えばプッシュ型店舗案内サービスにおいて、ユーザの位置が店舗の近くにあることは予測できるが、店舗内の空席数は予測が難しく、短い時間で満席となる可能性が高い場合を考える。ユーザの位置を収集するよりも、店舗の空席数を収集する方が、サービス実行条件が充足されている確率に対する影響が大きい。この場合、実行判断に対する各コンテキストの影響に差が存在する。

● 計算資源とコンテキスト発生位置の地理的分散

コンテキストは通常、サーバから地理的に遠い位置にある複数のコンテキスト取得装置から収集される。コンテキスト取得装置が携帯端末などの計算能力をもつものの場合、コンテキスト取得装置において予め取得した値に対して処理を行うことが可能である。これによりコンテキストの発生源に近い位置でサービス実行の判断を行うことでコンテキスト収集を行わずにサービス品質を向上する可能性がある。

3.3 コンテキストとその取得装置の特徴を利用した実行条件充足確率の計算

提案方式における上記の特徴の利用方法を次に示す。

● サービス実行判断結果の予測

コンテキストの真の値ではなく予測により得られた値を用いてサービス実行判断を行う。これによりコンテキストの取得や収集を行わずに各時点におけるサービス実行条件の充足確率を計算する。

● 影響の大きいコンテキストの収集

時間変化する各コンテキストの実行判断に対する影響度を計算する。各コンテキストの値はその影響度が高い場合のみ取得し、サービスに対して通知される。これにより、少数のコンテキスト取得と収集により、各時点におけるサービス実行条件の充足確率を計算する。

● コンテキスト取得装置における予備的な実行判断

あるコンテキストの値のみを用いてサービス実行条件の充足確率の高低を判断できる場合は、サービス提供サーバへの収集を行わずにコンテキストの取得装置がサービス実行の判断を行うこともできる。これを利用して、コンテキスト取得装置はコンテキスト収集よりも頻繁に取得のみを行う。取得値が、サービス実行条件が充足される確率が高い値の場合にはその値をサービス提供サーバへ通知する。これによりコンテキストの発生箇所における計算資源を利用し、サービス品質の向上を図る。

通常、サービスの実行はその判断に影響をもつ全てのコンテキストの値を考慮して判断される。そのため、各コンテキストにおけるサービス実行条件を満たす確率が高い値は、同じサービスの実行判断に用いられる他のコンテキストがとる値によって変化する。提案方式はこの変化を考慮して、通知すべき取得値を更新する。

4. サービス実行制御方式

4.1 概要

図3に提案方式の概要を示す。3章に示した方針に従い、提案方式はサーバ側制御器とコンテキスト取得装置の協調動作により実現される。サーバ側制御器は各コンテキストの取得装置からコンテキストを収集し、各サービスの代わりにサービス実行判断を行う。指示を行う時刻はサービス実行条件充足確率と各コンテキストの影響度の違いを考慮して決定される。

また、サーバ側制御器は警戒領域と呼ぶ値の範囲を各コンテキスト取得装置へ通知する。各コンテキスト取得装置はサーバ側制御器からの指示に従いコンテキストを取得し、その値を通知するとともに、警戒領域を用いてサービス実行条件の充足確率の高低を判断し、高い場合は自発的にサーバ側制御器へコンテキストを通知する。

以上のように、提案方式は二種類の制御器の連携により各時点で必要なコンテキスト取得装置から値を収集し、適切なサービスを適切な時刻に実行する。これによりサービス品質を保ちつつ、サービス実行制御動作の実行回数を削減する。

4.2 サーバ側制御器

サーバ側制御器の機能構成を図4に示す。サーバ側制御器

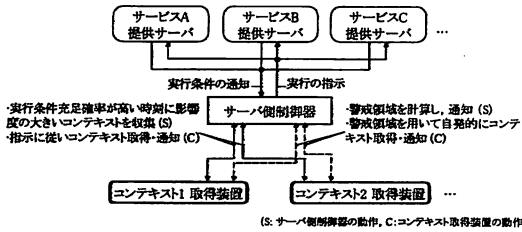


図3 提案方式の概要
Fig. 3 Outline of the proposed method.

は上述のようにサービスに代わってコンテキストを収集し、実行判断を行う。コンテキスト取得装置から値を収集する時刻は4.2.1で説明する方法により決定する。

また、サーバ側制御器は警戒領域を求め、コンテキスト取得装置へ伝える。警戒領域は、コンテキスト取得装置が取得したコンテキストの値がそれに含まれる場合、サービスの実行条件が充足される確率が高い値の範囲である。

4.2.1 コンテキスト収集時刻の決定

サーバ側制御器は各時点におけるサービス実行条件の充足確率を計算し、その値が高い時刻にサービスの実行判断に対して影響をもつコンテキストの取得装置から値を収集する。この確率やコンテキストの影響は各コンテキストの値の予測、およびサービス実行条件を用いて計算する。

まず、サービス実行条件の充足確率の求め方を示す。コンテキストの現在値から得られる現在の状態の代わりに、予測によって得られる将来時刻 t の状態の確率分布 $P^t(y_i)$ を用いることでその時刻における期待効用 EU^t_A を求めることができる。サービス実行制御を行うことでコンテキストの値を収集した結果、将来時刻 t における最大の期待効用をもつ動作 α^t がサービスを実行する動作となる確率が、実行条件の充足確率となる。

次に、コンテキスト間のサービス実行判断に対する影響の考慮方法を示す。サービス実行の判断に影響をもつ全コンテキストではなく、コンテキスト C の値のみを得た場合の条件付確率を用いて (1) 式を求めることで、その影響度を計算することができる。コンテキスト C がある値 c_i とした場合の決定動作 $\alpha_{c_i}^t$ を求め、現在の決定動作 $\alpha_{current}^t$ と比較する。それらが異なる確率、すなわちサービス実行条件が充足される値をとる確率 (Probability that the Decision Changes: PDC) が、そのコンテキストの影響の大きさとなる。

$$PDC_C^t = \sum_{\{i|\alpha_{c_i}^t \neq \alpha_{current}^t\}} P^t(C = c_i). \quad (3)$$

コンテキストの収集時刻の決定には、閾値 Th_{PDC} を用いる。サーバ側制御器は各時刻の PDC_C^t を各コンテキストに対して求める。そして、最初に一つ以上のコンテキストの PDC_C^t が Th_{PDC} 以上となる時刻に、それらに対応するコンテキスト取得装置から、コンテキストを収集する。

(3) 式を計算するために、コンテキスト C の値を得た場合の状態の確率分布を計算する必要がある。この計算を行うために、変数間の条件付確率の推論を含む意思決定に用いる表現形式として、ベイジアンネットワーク [6] (Bayesian Network: BN) でサービス実行条件を記述する。

図5は、プッシュ型飲食店内サービスのサービス実行条件をBNで表現したものである。楕円で示す各ノードは決定動作の判断に用いるコンテキストを表す。例えば Accepts user (G) と記述したノードは、店舗がそのユーザを受け入れ可能か否かを示すコンテキストを表し、可 (True) と不可 (False) の二値をとる。ノード間の矢印は、矢印の出る元の親ノードが、矢印が指す先の子ノードに直接影響を及ぼすことを示す。

ノード間の影響は、条件付確率表 (Conditional Probability Table: CPT) として表現される。CPT における行は親ノードの値に対する子ノードの値の条件付確率を示している。親をもたないノードは、事前確率が与えられる。例えば図5では、

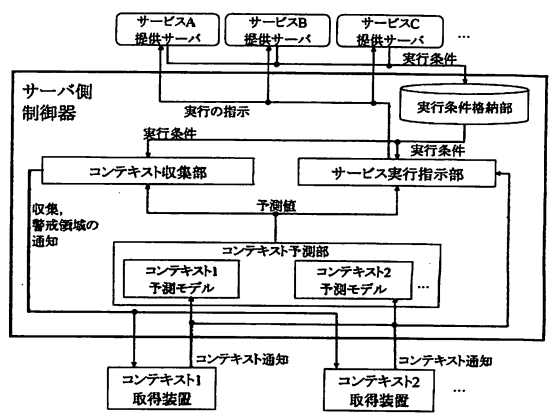


図4 サーバ側制御器の機能構成
Fig. 4 Function blocks of the server-side controller.

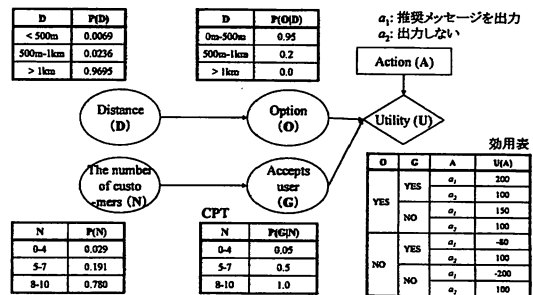


図5 ベイジアンネットワークの例
Fig. 5 Example of a Bayesian Network.

店舗内の人数が「0人から4人の間」という値 (n_1) にある確率 $P(N = n_1)$ は 0.029 と表現されている。長方形は意思決定に対する動作候補の集合を示す。例えば図5における長方形 (Action(A)) は二つの動作候補「推奨メッセージを出力」(α_1) と「推奨メッセージを出力しない」(α_2) を示す。菱形は各動作候補の効用を現している。効用は効用表の形式で表現される。効用表の各行はある状態 (効用に影響をもつ変数の値の組) において各動作を行った場合の効用値を現している。

BN で表現された各変数間の影響を利用し、一部のコンテキストの値のみが得られる場合、他コンテキストの値の確率分布を推論することができる。このような条件付の確率を計算するためのアルゴリズムがこれまでに多数提案されている [8]。

4.2.2 サーバ側制御器の動作手順

サーバ側制御器はコンテキストの値を収集した場合に、次に示す手順を実行する。

S1 収集したコンテキスト C を用いるサービスのうち、未選択のサービス S がある場合はそれを選択し、手順 S2 から S4 を実行する。

S2 サービス実行条件において C の値を収集した値 v 、その他のコンテキストの値を予測モデルにより得られた確率分布に設定して (2) 式を計算する。これにより、サービス S の動作 α を決定する。最も期待効用が高い動作 α が $\alpha_{current}^S$ と異なり、サービス実行を示す動作の場合はサービス実行条件が充足されたため、 S に対して実行を指示する。 $\alpha_{current}^S$ を α に更新する。 $\alpha_{current}^S$ はサービス S の現在の決定動作を示す。

S3 コンテキスト C の予測に用いるモデルのパラメータを v を用いて変更する。

S4 S4-1 から S4-2 の手順により、次回に収集するコンテキストと収集時刻を決定する。

S4-1 現在時刻を t とする。

S4-2 サービス S の実行条件におけるコンテキストの確率分

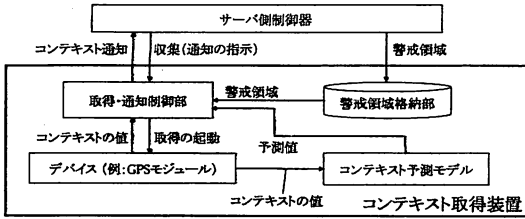


図6 コンテキスト取得装置の機能構成
Fig. 6 Function blocks of the context acquisition device.

布を、時刻 t における予測値に設定する。

S4-3 各コンテキストに対し、 PDC_C^t を計算する。

S4-4 PDC_C^t が閾値 Th_{PDC} 以上のコンテキストが存在した場合、それを時刻 t に収集することを決定する。存在しない場合は t を単位時間進め、S4-2へ戻る。

S4-5 各コンテキストの警戒領域を更新し、コンテキスト取得装置へ伝える。

4.3 コンテキスト取得装置

図6はコンテキスト取得装置の機能構成を示している。コンテキスト取得装置は指示に従いコンテキストをサーバ側制御器へ伝えるときに、サーバ側制御器が行うよりも高い頻度でコンテキストを取得する。取得した値がサーバ側制御器から伝えられた警戒領域に含まれる場合は、サーバ側制御器から指示される以前の時刻であってもその値をサーバ側制御器へ通知する。この場合、サーバ側制御器による予測では取りうる確率が低いと判断されたために収集が行われていないが、サービス実行条件が充足される確率が高い値を取っていることを意味する。

コンテキスト取得装置におけるコンテキスト取得の頻度も、サーバ側制御器がコンテキスト取得装置からコンテキストを収集する際に用いた方式と同様に決定される。コンテキスト取得装置は、警戒領域内の値を取る確率を各時刻に対して求め、その確率が閾値よりも高くなる時刻にコンテキストを取得する。

5. 性能評価

5.1 パラメータ設定と評価指標

本章では、提案方式と定期的に等しい頻度で実行制御を行う(Periodic Execution Control: PEC)方式のシミュレーション実験による評価結果について考察する。実験では、警戒領域を用いたコンテキスト取得や通知の制御は行わず、コンテキストの収集や取得をサーバ側制御器における時刻決定のみに従い実行した。

シミュレーション実験では1章に示したプッシュ型店舗案内サービスがユーザに提供されていることを想定した人工的なデータを用いた。サービスの実行時刻は、図5に示したサービス実行条件を用いて決定した。このサービスでは、ユーザの位置から得られる店舗からの距離と、現在の店舗内の空席数の二つのコンテキストを用いて実行を判断する。ユーザは10km四方の矩形内を移動するものとし、その位置を100mを単位とする二次元の離散的な座標 (x, y) ($0 \leq x < 100, 0 \leq y < 100$) で表現した。

単位時間は1分とし、ユーザは毎単位時間、前時刻の位置から距離 s 以内にある任意の座標へ等しい確率で移動するものとした。例えば $s = 1$ の場合、前時刻 t の位置が (x_1, y_1) とすると、時刻 $t+1$ の位置は $\{(x_1, y_1), (x_1 + 1, y_1), (x_1 - 1, y_1), (x_1, y_1 + 1), (x_1, y_1 - 1)\}$ から一つの座標を各々0.2の確率で選択することで決定される。その他のパラメータ設定を表1に示す。

提案方式では、上記のように与えた各コンテキストの値の推移を次のように予測した。ユーザの位置の確率分布は、前回の値を収集した時刻から t 単位時間後に、計測した位置を中心として平均0、分散 $s \cdot t/3$ の正規分布で与えた。また、前回の収集時に店舗内にいた客のうち、固定の滞在時間が経過していない未退出の客、および平均到着間隔2のポアソン分布で得られる新規到着客が店舗内にいるものとして、空席数の確率分布を

表1 評価におけるパラメータ設定
Table 1 Parameters for the evaluation.

パラメータ (単位)	値
シミュレーション時間 t_{total} (分)	100,000
店舗の位置	(50, 50)
ユーザの初期位置	(0, 0)
ユーザの最大移動速度 s	1, 10
店舗の席数	10
空席数の初期値	10
平均来客間隔 (分)	2
滞在時間 (分)	20
Th_{PDC}	0, 10^{-m} ($m = 2, 3, \dots, 25$)
PEC方式のコンテキスト収集間隔	2^n ($n = 0, 1, \dots, 6$)

予測した。

サービスは期待効用が最も高い動作が「推奨メッセージを出力しない」(a_2)から、「推奨メッセージを出力」(a_1)に変化した場合に推奨メッセージをユーザ端末へ出力するものと想定した。単位時間毎にコンテキストを収集する理想的な判断が可能な場合の判断動作が a_1 となる時間範囲(メッセージ出力が適している時間範囲)において、方式における判断動作が a_2 から a_1 へ変化した回数をサービス品質の評価指標とした。この指標を用いることで、評価対象の方式がメッセージ出力が適切な時間帯を捉える性能を評価することができる。

また、次式を用い、単位時間毎にコンテキストを収集する理想的な判断が可能な場合の判断動作の系列からの乖離も評価した。ここで、理想的な判断が可能な場合の時刻 t における判断動作を S_{ideal}^t 、評価対象の方式(提案方式もしくはPEC方式)の時刻 t における判断動作を S^t とする。

$$\sum_{\{t|t=0,1,\dots,t_{total}\}} d(t), \quad d(t) = \begin{cases} 1 (S^t \neq S_{ideal}^t) \\ 0 (S^t = S_{ideal}^t) \end{cases} \quad (4)$$

この値を求めることにより、サービス実行条件の充足の検出における適切さと検出の遅延を評価することができる。

実行制御方式のコストはコンテキストの取得数と収集数で測る。評価ではサーバ側制御器からの収集が行われる時刻のみコンテキストを取得するため、それらの数は等しい。従って、以下ではコンテキストの収集数のみを示す。

5.2 結果と考察

各方式において適切な出力を行った回数とその場合のコンテキスト収集数の比較を図7に示す。ここで、ユーザの単位時間あたりの移動範囲 s は、歩行する程度を想定して1とした。PEC方式と比較して、提案方式は大幅に少ないコンテキスト収集回数で等しい回数の適切な出力を行っている。また、提案方式は収集の回数が2648を超えるると、116回の適切な出力機会を全て捉えている。すなわち、毎単位時間毎のコンテキストの値の変化を把握して実行制御を行う理想的な場合と比較してサービス品質を劣化させることなく、98%のコストを削減している。

(4)式を用いて得られる、理想的な動作系列からの乖離を比較した結果を図8に示す。両方式とも、コンテキストの収集回数が増えるほど理想的な動作からの乖離が少なく、サービス品質が向上する。提案方式はPEC方式と等しい乖離を、大幅に少ないコンテキスト収集回数で示している。またコンテキスト収集回数が2648回以上の場合、適切な出力回数と同様に理想的な動作系列と全く同じ動作を行っている。

ユーザの最大の移動速度を示す値 s を自動車での移動を想定して10に設定した場合の適切な実行回数を図9に示す。図7と比較して、提案方式がPEC方式と等しいサービス品質を保ちながら削減できるコンテキスト収集数の割合が減少している。これはサービス実行制御を行わない間のユーザの位置の予測が比較的難しく、正しく実行条件充足確率を把握するためには頻繁なコンテキスト収集が必要なためである。

以上の結果から、提案方式はPEC方式と比較して、等しいサービス品質で大部分のサービス実行制御コストを常に削減可能であるといえる。特に歩行時のように、コンテキストの予測

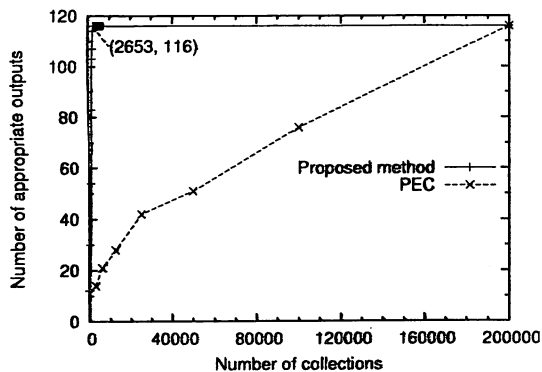


図 7 $s = 1$ の場合の適切な出力回数
Fig. 7 Number of appropriate outputs ($s = 1$).

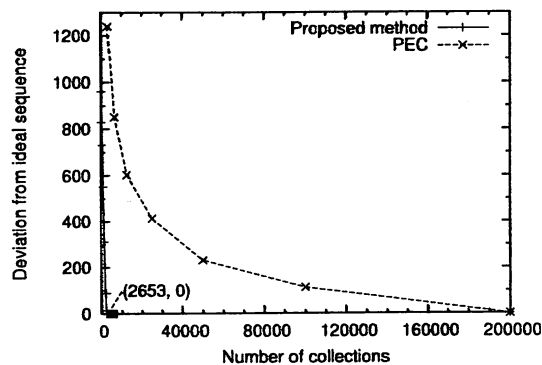


図 8 理想的な動作系列からの乖離
Fig. 8 Deviation from ideal sequence.

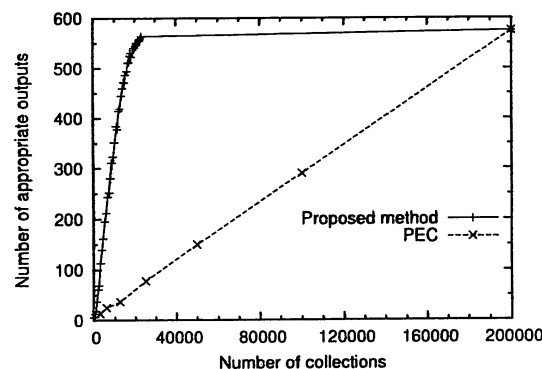


図 9 $s = 10$ の場合の適切な出力回数
Fig. 9 Number of appropriate outputs ($s = 10$).

精度が高い場合には、常にコンテキストの値を把握した場合の理想的な実行制御と等しいサービス品質を保ちながらほとんどの実行制御コストを削減可能であることを確認した。

6. 関連研究

コンテキスト依存型のサービスとコンテキストの発生源との間の通信量を削減することによりスケーラビリティを向上するための研究がこれまでに行われている [2], [3], [5]。これらでは、ネットワーク上に分散した計算資源を利用するイベント型の方式が用いられている。ネットワーク上の各エンティティはある固定の条件が満たされた場合に、コンテキストを転送する。例えば Context Toolkit [3] では、widget と呼ぶエンティティを定義している。widget はコンテキスト取得装置に配置され、ユーザの接近などの定められたイベントが発生する度にその事実を

サーバへ通知する。

ネットワークを通じて収集される、車両などの位置を管理する移動オブジェクトデータベースを想定し、本研究と関連した研究が行われている [1], [4]。データベースに対して発行された移動体の位置に関する連続的なクエリに回答するために必要な、移動体の位置の登録に伴う移動体とデータベース間の通信回数を削減する方式が提案されている。これらでは、移動体の近隣の位置へ発行されたクエリなどの情報を各移動体に伝える。各移動体はそれを用いて位置更新の要否を判断し、必要と判断した場合のみデータベースへ位置を登録する。

これらの方式はいずれもサービスの実行やクエリ結果の変更を、一つのコンテキストの値のみで判断できることを想定している。各サービスが多くのコンテキストを用いて実行の判断を行う場合、一つのコンテキストの値のみでサービス実行は判断できない。この場合、提案方式のようにサービスの実行に影響をもつ全コンテキストの値を考慮して収集や取得の回数を削減する必要がある。

7. まとめ

コンテキスト依存型サービスにおいて、サービス品質を低下させずにサービスの実行制御に関する動作の実行数を削減する実行制御方式を提案した。提案方式はサーバがコンテキスト収集などの実行制御動作を行わない間にサービス実行条件が充足される確率を計算し、その確率が十分低い時刻におけるコンテキストの取得とサーバへの収集の実行数を削減する。また、シミュレーション実験の結果を示し、提案方式がコンテキストを等しい頻度で定期的に収集する方法と比較して、サービスの品質を保ちつつサービス実行制御の動作の実行数を大幅に削減することを確認した。

今後は様々なサービスを想定し、提案方式の有効性を評価する。また、警戒領域の設定方法を具体化し、それを用いた実行制御を評価する予定である。

文 献

- [1] Y. Cai, K. A. Hua, and G. Cao, "Processing Range-Monitoring Queries on Heterogeneous Mobile Objects," Proc. 2004 IEEE Int'l Conf. on Mobile Data Management (MDM '04), pp. 27-38, 2004.
- [2] G. Chen and D. Kotz, "Context Aggregation and Dissemination in Ubiquitous Computing Systems," Proc. IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '02), pp. 105-114, 2002.
- [3] A. K. Dey, D. Salber, and G. D. Abowd, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-aware Applications," Human-Computer Interaction (HCI) Journal, vol.16 (2-4), pp. 97-166, 2001.
- [4] H. Hu, J. Xu, and D. L. Lee, "A Generic Framework for Monitoring Continuous Spatial Queries over Moving Objects," Proc. 2005 ACM SIGMOD Int'l Conf. on Management of Data, pp. 479-490, 2005.
- [5] T. Nadeem, A. Youssef, S. Banerjee, M. Youssef, S. Agarwal, K. Kamel, A. Kochut, C. Kommareddy, P. Thakkar, B. Trinh, A. U. Shankar, and A. Agrawala, "Implementation of a Scalable Context-aware Computing System," Proc. Personal Wireless Communications (PWC '03), pp. 364-374, 2003.
- [6] S. J. Russell and P. Norvig, Artificial Intelligence: Modern approach, Prentice Hall, 1995.
- [7] W. N. Schilit, "System Architecture for Context-aware Mobile Computing," PhD thesis, Columbia University, 1995.
- [8] G. Shafer, Probabilistic Expert Systems, CBMS-NFS Regional Conference Series in Applied Mathematics (SIAM '96), 1996.