

## 行動履歴と制約条件を考慮した情報家電制御機構

川原 圭博<sup>†</sup> 司 化<sup>††</sup> 猪鹿倉知広<sup>†††</sup> 登内 敏夫<sup>†††</sup> 森川 博之<sup>††</sup>  
青山 友紀<sup>†</sup>

<sup>†</sup> 東京大学 大学院情報理工学系研究科

<sup>††</sup> 東京大学 大学院新領域創成科学研究科

<sup>†††</sup> NEC インターネットシステム研究所

E-mail: <sup>†</sup>{kawahara,sihua,mori,aoyama}@mlab.t.u-tokyo.ac.jp, <sup>††</sup>t-igakura@bx.jp.nec.com,  
<sup>†††</sup>tonouchi@cw.jp.nec.com

あらまし 高齢化社会の到来に備え、ユーザの行動を先読みして煩雑な機器操作を簡略化したり、事故につながる操作を未然に防いで安心・安全なサービスを提供するコンテキストウェアサービスの実現に期待が寄せられている。こうしたシステムを実用的なものとするには、行動履歴をもとに個人に適応したサービス起動条件の学習と、様々な制約条件を考慮して安心・安全なサービス制御を行う汎用的な機構を実現しなければならない。本稿では、我々が開発しているコンテキストウェアサービス提供システム Synapse の設計指針とその実装について紹介する。

## An Information Appliance Control Mechanism based on User Action History and Constraint Conditions

Yoshihiro KAWAHARA<sup>†</sup>, Hua SI<sup>††</sup>, Tomohiro IGAKURA<sup>†††</sup>, Toshio TONOUCHI<sup>†††</sup>, Hiroyuki MORIKAWA<sup>††</sup>, and Tomonori AOYAMA<sup>†</sup>

<sup>†</sup> Graduate School of Information Science and Technology, The University of Tokyo

<sup>††</sup> Graduate School of Frontier Sciences, The University of Tokyo

<sup>†††</sup> Internet Systems Laboratories, NEC Corporation

E-mail: <sup>†</sup>{kawahara,sihua,mori,aoyama}@mlab.t.u-tokyo.ac.jp, <sup>††</sup>t-igakura@bx.jp.nec.com,  
<sup>†††</sup>tonouchi@cw.jp.nec.com

**Abstract** In view of the aging demography of several countries, context-aware services for networked home appliances which could simplify the operation of complex appliances and prevent operation failures are held as one of the next decade killer applications. In light of this, a general mechanism that provides services based on a user action history and constraint conditions is becoming indispensable in the realization of practical context aware systems. In this paper, we present the design and implementation of our context-aware service system named Synapse.

### 1. はじめに

ユビキタスネットワーク実現への第一歩として、様々なものがネットワーク接続される時代が来た。家庭においてもセンサやアクチュエータ、家電機器などをネットワーク接続し、いろいろな部屋で音楽や DVD を鑑賞したり、部屋のセキュリティ管理をしたりするホームネットワークの導入が始まっ

ている。こうしたホームネットワーク環境が普及するためには、機器同士の連携によって初めて可能になる魅力的なアプリケーションサービスが必要不可欠である。

「部屋に人がいないときには自動的に明かりを消灯する」「火の消し忘れを検知したらユーザに知らせる」といった機器の設定や連携の自動化はホームネットワークにおけるキラーアプリケーションの一つであると考えられる。とりわけ

これから高齢化社会を迎える先進諸国では、こうしたシステムが高齢者が自立した生活を送るための良き味方になると考えられる。知的な住宅環境を構築するという試みはこれまでに国内外で行われてきた。Tron 電脳住宅 [1], Aware Home [2], House\_n [3], Neural Network House [4] などでは同様のコンセプトが掲げられ実証的な研究が行われてきた。しかしながらこれらの研究においては、知的な住宅環境を構築することに主眼がおかれ、「陽気にあわせて自動開閉する窓」など様々なサービスを個別に組み込むというアプローチが中心であった。我々は、知的環境構築を実用的に構築、運用することを目的として、コンテキストに応じた機器連携のルールをユーザの手を煩わせることなく自動学習し、機器の変更や追加、嗜好の変化に対しても柔軟に振る舞える機器連携機構の検討を行っている。

本稿では、将来のホームネットワーク用のコンテキストウェアシステムに求められる要件を明らかにし、この要件を満たすような学習機構と ECA ルールを利用したポリシ記述の両方に対応したコンテキストウェアサービスプラットフォーム Synapse について示す。

## 2. コンテキストウェアサービスとその提供方式

### 2.1 想定するアプリケーションサービス

Synapse では、センサネットワークが環境中に存在し、ドアの開け閉めや温度、明るさの変化といった環境情報の切り替わりや人々の行動（コンテキスト）をネットワークからセンシングすることが可能な環境を想定している。そして、環境中の照明やエアコン、テレビといった家電製品の制御や物理的なアクチュエータの駆動もネットワークから制御可能になるとする。Synapse が目指すコンテキストウェアサービスは、実空間の環境情報に応じて遍在する機器同士を自動的に操作するようなサービスである。このようなサービスはその性質により 2 つに分解できる。

(1) 「便利」を促進するサービス: 「ユーザの帰宅時間に合わせて、風呂を準備する時間を調整する」「ユーザのいつも見ている番組を覚えていて視聴を促す」「屋外がさわやかな天気の日にはエアコンではなく窓を開けて温度調節する」など、ユーザの典型的な行動を先読みし、これに合わせて環境が自動的に機器の制御等をおこなうサービス。

(2) 「安心」「安全」見守り型サービス: 「いつも施錠をしてから寝るのに、今日は施錠し忘れていた日にはユーザに警告する」「子供が一人で家にいるときにはガスレンジの利用を無効にする」など、異常あるいは危険な行動を察知して、適切に振る舞うサービス。

### 2.2 イベント情報とサービスの連携機構

前述したアプリケーションサービスを実現するための手法

としては

(1) システムデザイナーがコンテキストとサービスの対応を固定的に記述する

(2) ユーザが対応関係をプログラムする

(3) 自動的にコンテキストとサービスの対応を学習するという 3 つの実現方法が考えられる。

たとえば「部屋に誰もいないことが確認される場合、照明を OFF にする」といったシナリオでは、上記 (1) のようにシステムのデザイナーがサービスの起動条件を固定的にプログラムすることで対応可能である。こうした手法は公共空間でのサービスでは効果的である。しかし、サービスの個人へのカスタマイズを考えるとこの方法では柔軟性に欠ける。というのも、ユーザの好みや習慣は千差万別であり、あるセンサのデータの値が異なるユーザに対して異なるコンテキストとして解釈されることがしばしばあるため<sup>(注1)</sup>、サービスを均質に定義することが一般的に困難であるからである。あるサービスが適切ではないコンテキストに関連付けられ、望まない状況でサービスを自動実行してしまった場合、これはユーザにとって不快なサービスとなってしまふ。

これとは逆に、(2) に示したように、ユーザがそれぞれ実空間中のセンサ値に応じ、起動すべきサービスを指定するというアプローチが考えられる。こうした場合、サービスとコンテキストの対応関係を記述するための機構のデザインがシステムの使い勝手を大きく左右することになる。デザインを誤ると些細なサービスに対して一つ一つルールを設定することが非常にやっかいな作業となってしまふ<sup>(注2)</sup>。また、ユーザに自由にサービス起動条件を追加することを許す場合、ある一つのコンテキストについて、複数の矛盾するサービスの起動を許してしまうことも考えられる（例えばエアコンの温度設定や、テレビの番組チャンネル）

そこで、あらかじめ固定的に人手によりサービスの起動条件を設定するのではなく、ユーザのサービス利用履歴を記録することで、動的にコンテキストと提供サービスの関連を学習する方法が考えられる。学習ベースの手法は、より個人の嗜好にかなったきめ細かなサービスを期待できるが、学習するための訓練データが十分にたまるまでは正しく動作しないため、緊急時に自動的に実行されるサービス等事例が十分に蓄積されないようなサービスに対しては不利である。

以上の議論をふまえて、我々は学習ベースの推薦システムにおいてルールベースのポリシ処理エンジンを適材適所に組み合わせることにより、個人の嗜好を考慮したサービス自動生成をしつつも、まれな事象に対しては明示的にポリシベース

(注1): たとえば、快適だと思ふ室温は人によって異なる。

(注2): たとえば、現在 E-mail の送受信に使う多くの MUA には受信したメールをフィルタリングし、フォルダに整理するための機能が備わっている。しかしながら、コンピュータの扱いに慣れない人にとってはこれらのフィルタルールを作成することは思いのほかハードルの高い作業である。

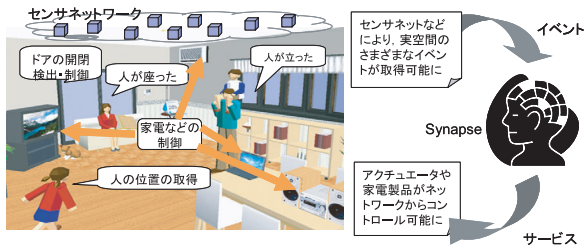


図 1 Synapse の概念

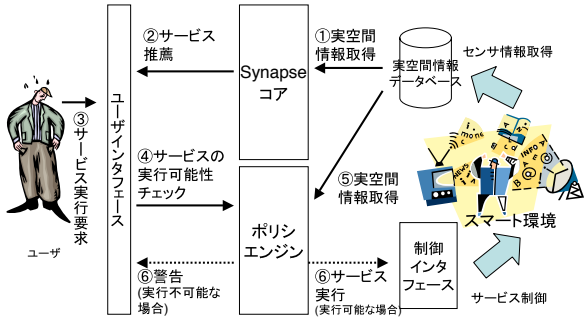


図 2 サービス推薦とポリシーによる実行制御

のサービス起動設定も可能なコンテキストウェアサービス基盤を実現する。

### 2.3 Synapse におけるアプローチ

Synapse は家庭やオフィス環境といった物理的に閉じた空間において発生したサービスの起動時刻とそれに先立つセンシングされたイベント情報をネットワーク上に蓄える (図 1)。その後、学習フェーズとして、これらイベントとサービスの関連性を統計的に学習する。実行フェーズでは、現在のセンサイベントを判断基準にして、次のステップに、もっとも起動されそうなサービスの上位いくつかを推薦リストとしてユーザーに提示し、実行するかどうか伺う。この学習 推薦の繰り返しが Synapse の基本動作となる [5]。

ところが、ユーザーの操作履歴が蓄積され、あるセンサイベントとあるサービスが関連していると判断されるに至るまでにはおよそ 10 から 20 回の事例が学習データとして必要になる<sup>(注3)</sup>。事例が十分に集まらないようなまれなサービスは Synapse での推薦対象となりにくい。また、確率モデルによって導きだされた推薦結果はサービスに関する一切のセマンティクスが考慮されていないため、たとえば、常識的に矛盾するサービスが同時に推薦されたり、物理的制約により現在実行することが適当ではないサービスが推薦されたりするという不具合が起こりうる。そこでまれな事象と「子供にアダルト番組へのアクセスを禁止する」「プレーカー容量を超える家電製品の起動を予防する」「音の出るサービスを同時に起動しない」などといった現実的な制約から導きだされる禁止事項に関しては、ポリシーとしてシステムに明示的に登録・実行することができるようにする [6]。

(注3): 4.2 において示す

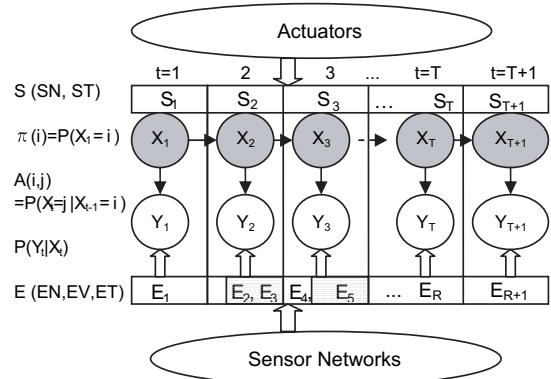


図 3 確率モデル

そして、サービスの推薦結果を基にユーザーがあるサービスを実行しようとする時、ポリシーエンジンがさまざまな制約を考慮してこのサービスの実行可能性を判断した上で実行またはユーザーに対する警告を行う。(図 2)

### 3. イベントとサービスの関連性学習機構

本節では、Synapse で用いられているイベントとサービスの関連性学習機構について述べる。

Synapse では隠れマルコフモデル (HMM [7]) を用いて実空間で発生したイベントと、そのときユーザーが欲したサービスの関係をモデル化する。図 3 に Synapse の確率モデルを示す。

まず、基本的なアイデアとして、Synapse では「人がいて部屋が暑い」「誰もいなくて暗い」などといったある室内環境の状況を HMM の (隠れた) 内部状態  $X_t$  として考える。そして、この内部状態は外部から複数のセンサ値を組み合わせさせたベクトル  $Y_t = (y_1, y_2, \dots, y_n)$  を観測値として現れる。そして、「部屋が暗ければ照明を点灯する」といったように、ユーザーはその内部状態に対応したサービスを一つ実行すると仮定し一つの内部状態から一つの対応するサービス  $S_t$  が記号として出力されるというモデル化を行う。

HMM の観測値となる  $Y_t$  は RFID やセンサネットワーク、あるいは天気予報や個人のスケジュールなど、あらゆる情報を取り込む必要がある。そこで、本実装ではファジーセットに倣い、すべての情報を 0 から 1 までの統一されたイベント値の組み合わせで表現することで統一的に情報を取り扱う。たとえば室温を表すイベントは、室温の有効範囲が 0 ~ 40 である場合、その値が線形に [0,1] の値にマッピングされる。あるいは人が入室したというイベントであれば、そのユーザーに対応するユーザー ID が [0,1] の間の離散値として与えられる。それぞれのイベントは  $E(EN, EV, ET)$  すなわち、イベント名とイベント値、そしてそれが発生したタイムスタンプと言う形で記録され、Synapse の管理対象空間で発生した過去すべてのイベントが  $\{E_1, E_2, \dots\}$  としてデータベースに記録される。

一方ユーザが起動したサービスは、Synapse 内部では  $S(SN, ST)$ 、すなわち、サービス名 (サービス ID)、サービス提供時刻の組として記録される。

ここで、 $SN \in \{S_1, S_2, \dots, S_n\}$  は、サービス名またはサービス ID である。サービス名とサービス ID は一対一で定義され、サービス ID は他との重複は許されない。そして、 $ST$  はサービスを提供した日付および時刻である。 $S(SN, ST)$  は Synapse を通じてユーザがサービスを呼び出すたびに記録される。

さて、HMM を取り扱う上で 3 つの重要なパラメータが存在する。すなわち初期状態分布  $\pi(i) = P(X_1 = i)$ 、内部状態が  $X_{t-1} = i$  から  $X_t = j$  に変化する確率を表す状態遷移確率  $A(i, j) = P(X_t = j | X_{t-1} = i)$ 、内部状態  $X_t$  とそのときの観測値  $Y_t$  の関係を表している  $P(T_t | X_t)$  である。このモデルを利用し、Synapse は学習フェーズにおいてモデルの各種パラメータを推定し、そしてその学習により求めたパラメータを用いて、次に呼び出しの起こりそうなサービスを列挙する。学習フェーズと実行フェーズに関して順を追って説明する。

### 3.1 学習フェーズ

学習フェーズにおいて、Synapse はまず  $t = 1 : T$  において発生したイベントとサービスを記録する。そして、この記録を次の手順で利用する。あるサービス  $S(SN, ST)$  が呼び出されると、システムは時刻  $ST$  から過去のある一定期間の間に発生したイベントをチェックする。たとえば、図 3 において、 $S_2$  が実行されると、 $E_2(e_k, 0.4, ET_2)$  および  $E_3(e_g, 0.7, ET_3)$  という二つのイベントが記録される場合、システムでは  $Y_2 = (y_1, y_2, \dots, y_n)$  (ただし  $y_k = 0.4, y_g = 0.7$  で、その他は 0) という値が記録される。こうして学習アルゴリズムへの入力パラメータとしての  $Y_t$  を得る。

学習フェーズは、与えられた記号系列からモデルのパラメータを推定する問題である。つまりゴールは、 $Y_t(t = 1, 2, \dots, T)$  を用いて  $\pi(i), A(i, j)$  と  $P(Y_t | X_t)$  を計算することにある。しかしながら、隠れマルコフモデルでは、記号系列を生成した状態遷移系列は非観測であるため、直接、最尤推定などを用いてパラメータを学習することはできない。そこで典型的な手法である前向き・後ろ向きアルゴリズムと呼ばれる手法を適用する。まず、前向きアルゴリズムにおいて、時刻  $t$  で状態  $X_i$  に到達する前向き確率  $\alpha_t = P(X_t = i | Y_{1:t})$  を再帰的に計算する。そして、後ろ向き確率  $\beta_t = P(Y_{t+1:T} | X_t = i)$  を再帰的に計算する。つづいて  $\gamma_t(i) = P(X_t = i | Y_{1:T})$  を  $\alpha_t, \beta_t$  を用いて求める。同様に、 $Y_{1:T} = y_1 \dots y_T$  に対し、状態  $X_i$  から状態  $X_j$  への遷移が時刻  $t$  において生じた確率  $\xi_{t-1,t|T}(i, j) = P(X_{t-1} = i, X_t = j | Y_{1:T})$  も  $\alpha_t, \beta_t$  を用いて求める。最後に、EM アルゴリズムにより状態遷移行列  $A(i, j) = P(X_t = j | X_{t-1} = i)$  および、モデル  $P(Y_t | X_t)$  を  $\gamma_t(i), \xi_{t-1,t|T}(i, j)$  を用いて求めることができる。

### 3.2 実行フェーズ

Synapse では、ある環境状態ではそのユーザは再び同じサービスを実行したがるだろうという仮定をおいている。よって、実行フェーズの役割は、現在のセンサイベントを確認し、現在の観測値ベクトルすなわち出力記号系列から次に遷移しそうな状態を推定することである。

ある時間  $t$  において、まず filtering アルゴリズムを用いて現在の内部状態  $X_t$  を求める。

$$P(X_t | Y_t) \propto P(Y_t | X_t) * P(X_t | Y_{t-1}) \quad (1)$$

ここに、 $P(X_t | Y_{t-1})$  は時刻  $t-1$  における予測結果であり、 $P(Y_t | X_t)$  は学習結果で得られた既知のモデルパラメータである。この結果を基に予測アルゴリズムを用いて次の内部状態  $X_{t+1}$  をとる確率を求める。

$$P(X_{t+1} | Y_t) = P(X_{t+1} | X_t) * P(X_t | Y_t) \quad (2)$$

この  $P(X_{t+1} | X_t)$  は既知の状態遷移確率であり  $P(X_t | Y_t)$  は上式 1 で求めた結果である。

この計算により、次の状態の発生確率が計算できる。我々のモデルでは一つの内部状態が一つのサービスに対応しているので、次にどのサービスがユーザによって実行されそうであるかが上記のステップを踏むことによって計算可能になる。

以上のように、本手法はセンサイベントと起動されるサービスの関係は HMM において確率的に記述されるため、一切のセマンティクスを含まない。よって、センサやサービスの変更や追加、またユーザ自身の嗜好や行動パターンの変化にも徐々に適応することができるため、非常にスケーラブルなシステムである。

## 4. 実装と評価

提案手法の有効性を確認するためにシステムの実装を行った。システムには図 2 に示すコンポーネントからなる。

**Synapse コア** Synapse コアは Matlab を用いて実装されており、蓄積されたセンサイベントからイベントとサービスの関連の学習や、現在のセンサイベントを判断基準にして、もっとも起動されそうなサービスの上位いくつかを推定し、推薦サービスリストとして生成する。

**ポリシー処理系** 学習では対応しきれない、まれな事象や現実的な制約から導きだされる事象に関して、ポリシーとしてシステムに明示的に登録・実行することができる。推薦されたサービスが実行可能かどうかポリシーによりチェックし、実行が不適当である場合にはユーザにその旨を知らせる。

**ユーザインタフェース** Synapse コアによって生成された推薦サービスリストは、Matlab Web サーバ機能を通じて XML 形式でユーザの UI に提供される。ユーザ端末では Macromedia Flash で作成された GUI スクリーンから起動したいサービスを選択するだけで任意のサービスを起動できる。Web イ

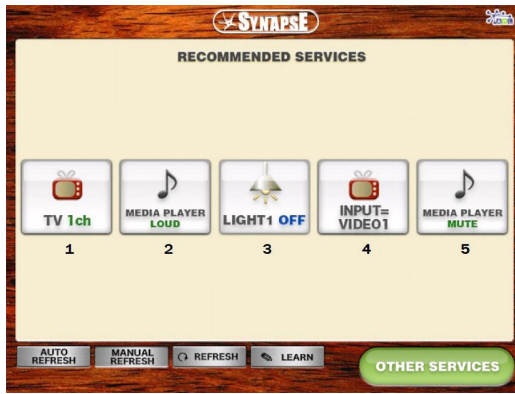


図 4 ユーザインタフェース

ンタフェースを利用しているため、ユーザはこのシステムを PC や PDA、携帯電話からも利用できる（図 4）

実空間情報データベース さまざまなセンサから実世界に関する情報を取得し Synapse コアにとって有用な形に情報を変換し、データベースにこれを保存する。たとえば部屋の明るさや温度は生データに対して単位の変換を施した後そのままの形でセンサイバントデータベースに保存される。その一方で「日付時刻」データなどは、この値から季節や曜日、一日のうちの時間帯という異なる複数のセンサイバントを生成し保存される。

サービス起動制御インタフェース 実世界に存在する家電製品やアクチュエータの制御を行い、サービスを起動する。情報家電などの複数の規格の差異はここで吸収する。

本実装では、RFID や無線センサネット U<sup>3</sup> [8] を含む 4 種類のセンサ（温度、明るさ、モーション）で 11 のイベントを生成している。そして、サービスとして、テレビや扇風機、メディアプレイヤーなど 5 種類のアプライアンスで 25 種類のサービスを提供している。これらのセンサやサービスを利用して自動的に構築されることを想定したサービスとしては、たとえば以下の 3 つがあげられる。

（1）照明点灯シナリオ:

部屋に活動中の人がいて、明るさセンサの値が暗い場合“照明 ON”のサービスが関連づけられる

（2）好みの番組選択シナリオ:

子供はリビングに一人ではいるときはテレビの 8ch を好んで視聴しているが、父親が部屋に入ってくるとチャンネルを 1ch に変更することが多く、部屋の中に誰がいるかという組み合わせ情報とテレビのチャンネルが関連づけられる

（3）音量調整シナリオ:

あるユーザがテレビを見たり音楽を聴いているとき、電話がかかってくると一時的に音の出る機器を消音設定にすることが多いため、電話の使用と機器の消音設定が関連づけられる

以下では、上記のシナリオに沿って、Synapse の性能を 3 つの側面から評価する。

## 4.1 アルゴリズムの計算量に関する議論

まず、Synapse の学習フェーズと実行フェーズに必要な計算量に関してそれぞれ議論する。アルゴリズムに必要な計算量は、モデルがとりうる状態の数  $M$  と学習データの長さ  $T$  が深く関係する。学習フェーズでは、もし  $M$  個の状態数があれば、それぞれのタイムスライスにおいて、 $M \times M$  のサイズの行列と  $M \times 1$  の行列の積を取る必要があるため、 $O(M^2)$  がそれぞれのタイムスライスにおいて必要な計算オーダーとなる。そして、この操作を  $t = 1, 2, \dots, T$  について繰り返さなければならないため、 $O(M^2T)$  が学習フェーズに必要な計算オーダーとなる。また、学習フェーズを完了するためには、 $t = 1, 2, \dots, T$  に対して、 $\alpha_t = P(X_t = i | Y_{1:t})$  ( $i = 1, 2, \dots, M$ ) を保持しなければならないためメモリ空間としては  $O(MT)$  必要となる。

実行フェーズでは、それぞれのタイムスライスにおいて学習フェーズとほぼ同じ操作が必要であるため、計算オーダーは  $O(M^2)$  になり、メモリコストは  $O(M)$  となる。

PentiumM 1.8GHz、メモリ 1G バイトの Windows XP を搭載した PC において計算を行った結果 50 の状態（サービス）と 2600 の学習データ（操作履歴）を処理するのにかかった時間は 80 秒であった。ユーザがサービスを利用することで得られた結果はいち早くシステムに反映された方がよいが、これはたとえば一日に一度ユーザがシステムを利用していない時に行えばよい操作であり、80 秒という計算所要時間は十分実用的である。

## 4.2 学習に必要な事例数に関する議論

ロバスタな学習モデルを構築するために何回同じような事例をシステムに与えればよいかを評価することも重要になる。そこで、20 種類の異なるトレーニング用のサンプルを用いてモデルを評価した。

図 5 にその結果を示す。

Y 軸に示す対数尤度は、モデルが訓練された特徴を認識した確率を示し、対数尤度が高ければ高いほど、モデルが安定していることを示す。結果より、Synapse のコアアルゴリズムは、約 10 回の操作により安定し、19 反復目に収束していることがわかる。

従って、一日に 5、6 回行う典型的な行動パターンに関しては、2~3 日後には自動的に推薦されることになり、これは十分実用的であると言える。

### 4.2.1 シナリオ別の妥当性

本節では実際に、特定のシナリオを想定した妥当性の評価をする。前節より、対数尤度が 10 回程度で落ち着くという結果が得られているため、

前述の 3 つのシナリオに関して、それぞれ 15 回ずつのサービス呼び出し（ただし“消音シナリオ”は、15 回の消音サービス操作と 15 回の元の音量に戻すサービス操作を伴う）に

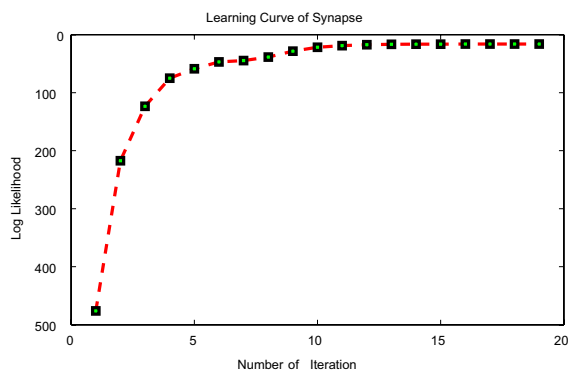


図 5 Synapse の学習曲線

表 1 シナリオ別の推薦サービス正解率

機器	サービス	正解率
照明	暗いと感じるときに照明 ON を推薦	96.7%
TV	人に応じて好みの番組 ch. を推薦	90.0%
CD プレーヤー	電話使用時に消音	93.3%
	消音復帰	96.7%

対する学習評価について述べる。

評価には、3つのシナリオに利用されているランダムなセンサーデータの組み合わせを利用する。SynapseのUIに提示されるいくつかの推薦サービスのなかで、現在想定したシナリオに対応するサービスが第一位として推薦されたときのみ正解であるとする。実用的な観点からは、推薦するサービスが第一位に選択されていなくとも、上位のいくつかの候補に入っていればユーザはその中から簡単に選択可能である。しかしここでは学習アルゴリズムの評価をするという観点から、第一位としてリストアップされるという条件で評価する。

表1に、3つのシナリオにおける4つのサービスの評価結果を示す。本手法を用いることで100%ではないながらも、高い確率で適切なサービスをもっともユーザが欲しそうなサービスとしてユーザに提示することが可能である。

定性的には、環境に大きな変化がないようなサービスはその変化を予測しきれず、推薦精度が低下する。たとえば、テレビの番組チャンネルの推薦は部屋に存在する人と曜日や時間帯が大きく影響しているため、定常的に人が部屋に存在する場合には適切な推薦が行われないことがある。これに対して、電話中にいったん消音にしたCDプレーヤーを電話終了後に消音解除するなど「一連の手順」が比較的典型的に定まっている場合は正しく推薦できる確率が高い。

## 5. おわりに

本稿ではユーザの行動を先読みして煩雑な機器操作を簡略化したり、事故につながる操作を未然に防いで安心・安全なサービスを提供するコンテキストウェアサービスプラットフォーム Synapse の設計と実装について示した。

Synapse は隠れマルコフモデルを用いた統計的手法によ

り、あるセンサイベントとサービスの関連を学習する。本学習手法は、サービスとセンサの関係に関するセマンティクスを一切含まないため、サービスやセンサの変更、追加そして、ユーザ自身の嗜好や行動パターンの変化にも柔軟に対応することが可能である。また、学習ベースのアプローチでは対応しきれない緊急時における「安心」「安全」見守り型サービスに関しては、ポリシーによる明示的な実行制御を行うことができる。

使用シナリオを想定した初期的な評価では、90%以上の確率で期待されるサービスが推薦されるという結果が出ており、本システムの有用性が期待できる。現在は、本機構をテストベッドの中に組み込み、より現実の生活に近い環境での評価を検討している。

謝辞

本研究は、総務省からの委託研究の成果である。

## 文献

- [1] Tron 電腦住宅: <http://www.sakamura-lab.org/TRON/proj95/>
- [2] Dey, Anind K., Daniel Salber and Gregory D. Abowd.: "A Context-based Infrastructure for Smart Environments," In Proceedings of the 1st International Workshop on Managing Interactions in Smart Environments (MANSE '99), Dublin, Ireland, 1999.
- [3] Intille, S. S.: "Designing a home of the future," IEEE Pervasive Computing, vol. April-June, pp. 80-86, 2002.
- [4] Mozer, M. C.: "The neural network house: An environment that adapts to its inhabitants," In Proceedings of the American Association for Artificial Intelligence Spring Symposium on Intelligent Environments, pp. 110-114., Menlo, Park, 1998.
- [5] 川原 圭博, 司 化, 森川 博之, 青山 友紀: "コンテキストウェアサービス実現に関する一検討," 電子情報通信学会ソサイエティ大会, B-20-7, September 2004.
- [6] 川原 圭博, 司 化, 猪鹿倉 知広, 登内敏夫, 森川博之, 青山友紀: "行動履歴と制約条件を考慮したサービス推薦を行うコンテキストウェアシステムの設計," 電子情報通信学会ソサイエティ大会, B-19-24, September 2005.
- [7] Rabiner, L.R.: "A tutorial on Hidden Markov Models and selected applications in speech recognition," In Proceedings of the IEEE 1989, 77(2):257-286, 1989.
- [8] Y. Kawahara, M. Minami, H. Morikawa, and T. Aoyama: "Design and Implementation of a Sensor Network Node for Ubiquitous Computing Environment," In Proceedings of IEEE Semiannual Vehicular Technology Conference(VTC2003-Fall), Orland, USA, October 2003.