

情報家電機器におけるプロアクティブな利用支援機構

桐原 幸彦¹ 由良 淳一¹ 楠本 晶彦¹ 徳田 英幸^{1,2}

¹慶應義塾大学大学院 政策・メディア研究科 ²慶應義塾大学 環境情報学部

本論文では、様々な習熟度の情報家電機器ユーザにサービス機能を能動的に提供するプロアクティブ(能動的)な利用モデルを提案する。近年、家電機器の機能が高性能化、多機能化しているにも関わらず、その利用形態がリアクティブ(受動的)であり、ユーザインタフェースが複雑であるため、習熟度の低いユーザは機能を十分に使いこなせない場合が生じている。本稿で提案するプロアクティブな利用モデルは、機器利用に必要な情報収集や機器操作学習を行い、ユーザにサービス機能を容易に提供する。これにより、ユーザは要求を明示することなく操作簡便性を向上できる。本モデルを実現するために、情報家電機器の利用支援ミドルウェアである、Proactive Support system on Networked Appliances(PSNA)を開発した。PSNAを用いることにより、プログラマはプロアクティブな情報家電機器制御ソフトウェアを容易に構築できる。

PSNA:A Proactive Support System on Networked Appliances

Yukihiko Kirihara¹ Jun'ichi Yura¹ Akihiko Kusumoto¹ Hideyuki Tokuda^{1,2}

¹Graduate School of Media and Governance, Keio University

²Faculty of Environmental Information, Keio University

In this paper, we propose a proactive operation model of home appliances. Recently, users of home appliances have difficulty in operating appliances due to appliances' high functionality, complex use, interface, and users' lack of knowledge. While the current operational model of home appliances is a reactive model, we propose a proactive operational model which is realized by gathering operational information for appliances and learning operational pattern of appliances. With our model, users can access service of home appliances without expressing explicit request to the appliances. Based on the model, we have developed a middleware, called Proactive Support system on Networked Appliances (PSNA) with which a programmer can easily construct a software for our model.

1 はじめに

家電機器はハードウェア技術の発展により、家電機器本来の機能の性能向上だけでなく、多様なライフスタイルに対応するための開発が続けられている。これにより、自然環境に配慮した家電機器やネットワーク接続機能を備えた情報家電機器が登場した。また、情報通信技術の発達により、複数の情報家電機器をネットワークで接続し、制御する動きも顕著になっている。しかし、家電機器のユーザは、コンピュータユーザと異なりネットワーク環境に対する習熟度が様々である。従って、情報家電機器の利用を支援する必要がある。

本稿では、第2章で情報家電機器の利用形態の問題点を指摘する。次に第3章で新しい情報家電機器のプロアクティブな利用モデルを提案する。そして、第4章、第5章において、プロアクティブな利用モデルに基づいたProactive Support system on Networked Appliances(PSNA)の設計と実装について述べる。第6章では、PSNAの定性的及び定量的評価を行い、第7章で今後の課題について述べる。

2 情報家電機器環境

本節では、まず本稿の研究対象である情報家電機器環境を構成している情報家電機器ネットワークと情報家電機器ユーザについて述べる。また、現在の情報家

電機器の利用形態に着目し、その問題点を明確にする。

2.1 情報家電機器ネットワーク

近年の家電機器の機能拡張には、家電機器の情報化とそれに伴うネットワーク化がある。家電機器がデジタルデータに対応すると、圧縮や保存などの技術が利用できる。また、音声や映像などの情報をデジタルデータで扱えるため、ネットワークを介して情報交換できる。本稿では、このようなネットワーク接続機能を持った家電機器を情報家電機器と定義する。

家電機器メーカー各社は、情報家電ネットワーク構築のための技術の標準化に向けて提携などを行っている。例えば、日本と欧州の家電機器メーカー8社は、HAVi(Home Audio Video Interoperability)[1]というAV機器ネットワークのための基本仕様を定めた。また、情報家電機器の相互接続性を高める仕様として、Jini(Java Intelligent Network Infrastructure)[2]やUPnP(Universal Plug and Play)[3]が挙げられる。シャープはインターネット上にソフト、コンテンツ配信サービスを核にしたデジタル情報家電向けの情報サービス事業(Sharp Space Town)[4]を開始している。このように、ハードウェアとソフトウェアの両方から、情報家電機器ネットワークの構築を目指している。情報家電機器ネットワークを構築することによって、情報家電機器間で機能や情報の共有ができる。

2.2 情報家電機器ユーザ

情報家電機器のユーザである家電機器ユーザは、コンピュータユーザとは異なる。第1に家電機器の主な入力インタフェースはボタンであるが、PCやワークステーションの主な入力インタフェースはボタンよりも複雑なキーボードである。家電機器ユーザは通常利用している機器の操作インタフェースが違うため、操作可能なインタフェースに制限がある。第2に家電機器ユーザはコンピュータユーザに比べて専門的な知識が乏しい。トラブルシューティングの経験も少ないため、機器を利用するために多くのマニュアルを読むことに抵抗を感じる。すなわち、機器利用に必要な情報を収集する能力に違いがある。また、家電機器ユーザは、家庭内で家事に従事することが多いユーザだけでなく、普段は会社や学校に出掛けているユーザも存在する。このように、家電機器を利用する頻度に差があることから、家電機器ユーザの中でも家電機器に対する習熟度は様々であると言える。

2.3 リアクティブな利用形態

通常、家電機器を利用する場合、ユーザは何らかの方法を用いて家電機器に対して利用意志を伝える必要がある。例えば、テレビを見る場合、ユーザはリモコンの電源スイッチを押すことで、テレビに見る意志を伝える。また、ユーザが洗濯をする場合、洗濯機のスイッチを押すことで、洗濯機に利用する意志を伝える。このようにユーザからの要求を受けて機能を提供する特徴を持つ家電機器の利用形態は、ユーザにとってリアクティブであると言える。

図1にリアクティブな利用形態の概念図を示す。環境情報とは、温度情報、情報などの周辺の空間の属性に関する情報及び、利用する機器の状態、利用法などの機器の属性に関する情報を指す。リアクティブな利用形態では、ユーザが機器を利用するために必要な周辺情報、属性方法を収集し、機器に要求を明示する。機器は、ユーザからの要求提示を待ち、それに応答して機能を提供する。このように、ユーザから機器への単一方向の意志表示が行われ、機器は機能のユーザインタフェースを提示して、ユーザの要求を待つ。

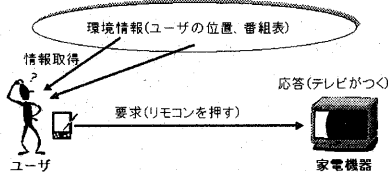


図1: リアクティブな利用形態

2.4 リアクティブな利用形態の問題点

情報家電機器ユーザは、利用可能な操作インタフェースに制限があり、機器利用のために情報を収集するこ

とに寛容でない。また、家電機器に対する習熟度が様々である。このような特徴を持つユーザに対して、情報家電機器をリアクティブな利用形態を持つ場合、以下のような問題が発生する。

情報取得負荷の増加

家電機器は情報化やネットワーク化が進み、より多くの情報を扱うようになった。それに伴い、ユーザが家電機器を利用して、対応すべき状況も増加した。しかし、家電機器ユーザは習熟度が様々であり、機器の機能を利用するために必要な情報取得能力に違いがある。そこで、習熟度の低いユーザの情報取得能力を補助し、情報取得にかかる負荷を軽減する必要がある。

機器操作負荷の増加

リアクティブな利用形態では機器の機能を直接操作するため、操作インタフェースを単純にする必要がある。しかし、機器の機能増加に伴い、機能を制御するインタフェースも複雑化しており、ユーザは多くのインタフェースの使用法について習熟しなければならない。そこで、機器操作を補助する必要がある。

3 プロアクティブな利用モデル

本研究では、情報家電機器のリアクティブな利用形態の問題点を解決するためにプロアクティブな利用モデルを提案する。本節では、提案したプロアクティブな利用モデルの概要、システム要件について述べ、リアクティブな利用形態との比較を行う。

3.1 概要

ユーザが機器の機能を利用する際に、要求を明示することなく、機能を提供できる家電機器はユーザにとってプロアクティブであると言える。例えば、自動ドアはユーザがドアの開閉機能に対して要求を明示するのではなく、赤外線センサがドアの前にいるユーザを感知し、ドアの開閉を行う。従って、ユーザにとってプロアクティブである。

図2にプロアクティブな利用モデルを示す。プロアクティブな利用モデルは、情報取得、動作提案、ユーザ応答の3つの段階がある。まず、ユーザだけでなく、機器側でも機器利用に必要な情報を取得する。次に、取得した情報を基に機器がユーザに対して動作提案を行う。そして、ユーザはその動作提案に対して応答して機能を利用する。

3.2 システム要件

プロアクティブな利用モデルを実現するためには、情報取得、動作提案、ユーザ応答の各段階において、自律性、双方向性、操作簡便性を実現する必要がある。

自律性

プロアクティブな利用モデルでは、ユーザからの要求明示を受けずに、機器が動作する。従って、動

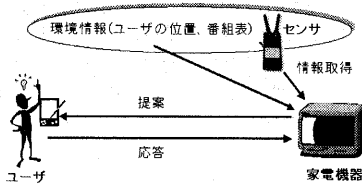


図 2: プロアクティブな利用モデル

作を決定するために必要な情報を自律的に収集する必要がある。

双方向性

リアクティブな利用形態では、ユーザと機器間の通信がユーザから機器への要求通知のみである。しかし、プロアクティブな利用モデルでは、機器からユーザへ動作提案を通知するため、通信の双方向性が必要である。

操作簡便性

プロアクティブな利用モデルの目的は、ユーザの機器利用を支援することである。従って、機器からの動作提案に対するユーザの応答は簡便である必要がある。

3.3 特徴

表 1 にリアクティブな利用形態とプロアクティブな利用モデルの比較を示す。

既存のリアクティブな利用形態では、ユーザが全ての機能を習熟する必要がある。家電機器ユーザは、コンピュータユーザに比べて、習熟度の浅いユーザが多いため、多くの機能は利用されない。習熟度の差が大きい家電機器ユーザにとって、ユーザの習熟度に依存したリアクティブな利用形態は効果的ではない。

一方、プロアクティブな利用モデルでは、ユーザの習熟度に依存することなく、ユーザに機能を提供できる。従って、情報家電機器上でプロアクティブな機器利用支援が行えれば、家電機器ユーザでも、コンピュータユーザと同様に情報家電機器の機能を使いこなせる。

表 1: 利用形態の比較

	リアクティブ	プロアクティブ
利用要求明示対象	機器	ユーザ
操作簡便性	習熟度依存	習熟度非依存
情報収集者	ユーザ	システム

3.4 関連研究

本節では、プロアクティブな利用モデルの関連研究として、GP(遺伝的プログラミング)マルチエージェントシステム [5] と Adaptive Control of Home Environments(ACHE)システム [6] を挙げる。

自律的に動作ルールを生成するシステムとして、名古屋大学で開発された GP マルチエージェントシステ

ムが挙げられる。GP マルチエージェントシステムは、GP による動作ルールの創発を Jini[2] を利用して、ネットワーク環境まで拡張したエージェントシステムである。エージェント・サービス利用者は、ネットワーク上のサービスを利用する時に、Jini の JavaSpace と呼ばれるオブジェクト交換機構から動作ルールをダウンロードする。エージェント・サービス利用者は、独自に動作ルールを保持しており、ダウンロードした動作ルールと自身の持つ動作ルールを遺伝的プログラミングを用いて、より効率的なルールを生成する。しかし、動作ルールの創発に遺伝的アルゴリズムを用いているため、生成された動作ルールにユーザの評価を組み込んでいない。従って、ユーザの意図しない動作ルールが創発される可能性があり、情報家電機器環境に適していない。

一方、情報家電機器ネットワークを利用して、各機器の自律的動作を目的として開発されたシステムとして、コロラド大学で開発された ACHE システムが挙げられる。ACHE システムは環境を監視し、その環境内での住人の動作パターンを推測するシステムである。家庭内の電気、ガス、水道などの情報を管理し、電灯、湯沸かし器、ガスコンロ、電気ヒーターなどの機器を制御する。制御する基準は、2つ存在する。1つは、住人の要求を満たすことである。もう1つは、電気、ガス、水道などの資源使用を最小限に抑えることである。ユーザの要求を満たすことと、資源使用の最適化が競合した場合、資源使用の最適化を優先させる。この場合、一時的にユーザに不快な思いをさせるが、最適化することにより、資源利用のためのコスト削減分をユーザに還元して、ユーザの満足度を向上させる。しかし、ユーザの要求はコスト削減だけでなく、多種多様な条件が動的に変化する。従って、ACHE システムは、コスト削減という限られた条件においてユーザの要求を満たしており、ユーザの満足度に対する柔軟性に欠ける。

4 PSNA の設計

本節では、プロアクティブな利用モデルに基づいた PSNA の設計について述べる。まず、PSNA の概要について述べ、システムの構成を示す。そして、PSNA を構成する情報取得部、情報管理部、提案生成部の各サブシステムについて述べる。

4.1 システム概要

PSNA はプロアクティブな利用モデルを実現するために、ユーザのリアクティブな機器利用を学習し、その結果を用いてプロアクティブな動作を提案する。PSNA の各サブシステムの構成を図 3 に示す。情報取得部は機器の機能に関するデータやセンサが取得するデータを正規化し、情報管理部に送信する。提案生成部は、ユーザの機器利用法から機器利用の際の各センサの重みを学習し、重みデータを情報管理部に送信する。また、重みデータと情報取得部がセンサから取得したデータを利用して、ユーザへの動作提案を生成する。情報管理部

は、情報取得部から送信されるデータと提案生成部から送信されるデータをそれぞれデータベースに格納する。

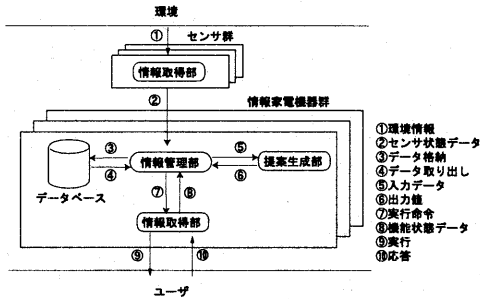


図 3: ソフトウェア構成

4.2 情報取得部

情報取得部はセンサ及び機器からデータを取得し正規化する。センサに搭載される場合は、データマネージャとセンサインタフェースから構成される。家電機器に搭載される場合は、データマネージャと機能インタフェースから構成される。両インタフェースはセンサ、機器の機能ごとの差異を隠蔽する。データマネージャは、各センサ、機器の機能から情報を取得し、情報管理部に送信する。

4.2.1 データマネージャ

データマネージャはセンサや機器ごとに存在し、センサや機器の機能から得られるデータを取得する。ユーザが機器を利用した際に、センサインタフェース及び機能インタフェースから取得したデータを情報取得部に送信する。

4.2.2 センサインタフェース

センサインタフェースは、多種多様なセンサの差異を隠蔽する。データマネージャがセンサインタフェースを通じて扱う情報は、センサ識別子、センサデータである。

センサ識別子

センサ識別子は、情報家電機器ネットワークに存在するセンサに一意につけられる。表2にセンサ識別子の構成を示す。システムはセンサ識別子の新規登録、変更、削除を行う。

表 2: センサ識別子の構成

ホスト名/センサの名前

センサ状態データ

センサから取得できるデータには、温度センサのように連続値で表すものと自動ドアの赤外線センサのように2値で表すものがある。センサ状態データはこのような様々なデータ形式を同様に扱うた

めに、0から1の間で正規化した値で表す。連続値の場合はセンサの上限値を1、下限値を0に設定し、2値の場合は、それぞれ1、0を設定する。表3に上限値170℃、下限値-30℃の温度センサを正規化した例を示す。

表 3: センサ状態データの例

温度センサ (°C)	-30	...	20	...	170
センサ状態データ	0	...	0.25	...	1

4.2.3 機能インタフェース

機能インタフェースは多種多様な機器の機能の差異を隠蔽する。データマネージャが機能インタフェースを通じて扱う情報は機能識別子と機能状態データである。

機能識別子

機能識別子は、情報家電機器ネットワークに存在する機器に一意につけられる。表4に機能識別子の構成を示す。システムは機能識別子の新規登録、変更、削除を行う。

表 4: 機能識別子の構成

ホスト名/機能の名前

機能状態データ

機能の状態はテレビのon/offのように2値で表せる場合とボリューム調整機能のように連続値で表す場合がある。機能状態データは機能のそれぞれの状態を0から1の間の値で正規化する。連続値の場合は上限値を1、下限値を0に設定し、2値の場合はそれぞれ1、0を設定する。表5に上限値-100デシベル、下限値0デシベルのボリューム調整機能を正規化した例を示す。

表 5: 機能状態データの例

ボリューム (dB)	-100	...	-50	...	0
機能状態データ	0	...	0.5	...	1

4.3 提案生成部

提案生成部は学習モジュール、アクションマネージャ、提案生成インターフェースから構成され、ユーザの機器利用動作を学習し、動作提案を生成する。学習モジュールは、ユーザの機器利用時の各センサの重みデータを算出する。アクションマネージャは、学習モジュールの算出した重みデータを利用し、ユーザに対する動作提案を生成する。提案生成インターフェースは学習モジュールとアクションマネージャに対するインタフェースを定義する。

4.3.1 提案生成部インタフェース

提案生成部インタフェースはインタフェースとアルゴリズムを分離し、様々な学習アルゴリズムの組み替えを容易にした。これにより、PSNAの学習アルゴリズムに対する柔軟性が向上した。

4.3.2 学習モジュール

学習モジュールは、ユーザが通常の機器を利用した際のセンサ状態データから、機器利用に対するセンサの重みデータを算出する。算出された重みデータは、情報管理部に送信される。

4.3.3 アクションマネージャ

アクションマネージャはセンサ状態データが変化した際に、各センサのセンサ状態データと各センサの重みデータから機能状態データを算出する。図4にアクションマネージャの構造を示す。アクションマネージャは入力層、中間層、出力層から構成される。 I_k は入力値、 a_j は計算ユニット、 O_i は出力値、 W_{kj} は重みを表す。入力層は情報取得部が取得した各センサのセンサ状態データを受信し、中間層は出力値の計算を行い、出力層に計算結果を渡す。そして、出力層は計算された機能状態データを情報管理部に送信する。

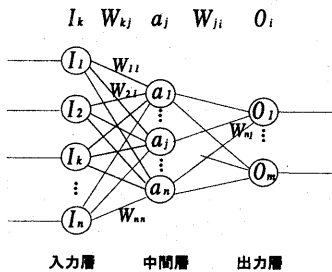


図4: アクションマネージャの構成

4.4 情報管理部

情報管理部は機器に搭載され、情報取得部から送信されたデータ及び提案生成部から送信されたデータを管理する。情報管理部はデータベースにデータを格納するデータベースマネージャと各種データベースから構成される。データベースは重みデータベースと利用状況データベースがある。

4.4.1 データベースマネージャ

データベースマネージャは情報取得部から送信される情報及び提案生成部から送信される情報を受信し、それぞれデータベースに格納する。情報取得部から送信されるデータは利用状況データベースに格納され、提案生成部から送信されるデータは重みデータベースに格納される。

4.4.2 重みデータベース

重みデータベースはユーザの動作学習に必要なセンサの重みデータを格納する。表6に重みデータベースの例を示す。重みデータベースはユーザ識別子、機器識別子、機能状態データ、重みデータの属性を持つ。

表6: 重みデータベースの例

ユーザ識別子	機器識別子	機能状態データ	センサA重み	センサB重み
yukihiko	dali/tv	0.9	0.3	0.4
yurayura	k2/air	0.4	0.4	-0.3
akihiko	k2/wash	0.3	-0.2	0.1

4.4.3 利用状況データベース

利用状況データベースはユーザが機器利用時の各センサの機能状態データを保持する。表7に利用状況データベースの例を示す。利用状況データベースはユーザ識別子、機器識別子、機能状態データ、センサ状態データの属性を持つ。

表7: 利用状況データベースの例

ユーザ識別子	機器識別子	機能状態データ	センサA状態	センサB状態
yukihiko	dali/tv	0.7	0.8	0.3
yukihiko	dali/tv	0.5	0.2	0.9
yukihiko	k2/air	0.2	0.9	0.3

5 PSNAの実装

本節では、まず第4節で述べた設計を基に実装したPSNAの実装概要について述べる。そして、本システムの応用アプリケーションとして実装したプロアクティブテレビについて述べる。

5.1 実装概要

実装はVineLinux2.1.5上でJava言語を用いて行った。開発ツールはJDKバージョン1.2を使用し、情報取得部、情報管理部、提案生成部をモジュール化した。また、学習アルゴリズムは誤差逆伝播法を用いた。

5.2 通信データ構成

PSNAの各サブシステム間の通信データは、データ性質とデータから構成される。データには、変更ノードデータ、状態データ、評価データ、入力データがある。表8に通信データの基本構成を示す。LABELは、データ性質を示し、それぞれのデータにdata, status, inputの4種類の識別子がある。以下に4種類の通信データについて説明する。

表8: 通信データの基本構成

PSNA://LABEL/DATA

変更ノードデータ (PSNA://data/...)

変更ノードデータは、機能状態データまたはセンサ状態データが変化した場合に、マルチキャストを用いて送信される。このデータは、LABELと変化した機能識別子またはセンサ識別子から構成される。

状態データ (PSNA://status/...)

状態データは、機能状態データまたはセンサ状態データを送信する際のデータ形式である。このデータは、変更された機能識別子やセンサ識別子及び機能状態データ及びセンサ状態データで構成される。

評価データ (PSNA://eval/...)

評価データは、提案生成部の決定した動作に対し、ユーザが入力するものである。動作した機能識別子、機能状態データ、ユーザの評価値で構成される。

入力データ (PSNA://input/...)

入力データは、情報管理部が提案生成部に送信するデータである。入力データは、センサ状態データ、動作した機能識別子、機能状態データで構成される。

提案生成部がシステム出力値を計算する際の入力値が含まれる。変更された機能状態データを正解出力値として計算する。

5.3 応用アプリケーション

本プロトタイプシステムを用いたアプリケーションとして、プロアクティブテレビを実装した。プロアクティブテレビは、部屋にどのユーザがいるか、という状況に応じてテレビの電源及びチャンネルの切り換え機能を能動的に提供する。

本アプリケーションは、ネットワークに接続したセンサや機器によって構築された SSLab 実験装置 [7] を用いて実装した。図 5 にプロアクティブテレビの構成を示す。テレビは、小型 PC である Tini [8] に接続したプラズマディスプレイを用いた。センサは、工業用小型 PC である Dounus [9] に接続した RF-ID [10] を使用した。以下に動作学習の手順とプロアクティブな動作提案の手順について述べる。

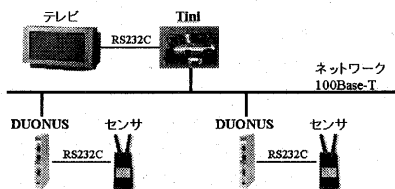


図 5: プロアクティブテレビの構成

5.3.1 機器利用の学習

図 6 に機器利用の学習の手順を示し、以下で説明する。

1. ユーザが機器に利用要求を明示し、機器を利用する。

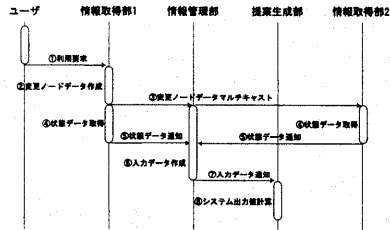


図 6: 動作学習の手順

2. 利用された機器上の情報取得部が、変更ノードデータを作成する。
3. 変更ノードデータをマルチキャストで送信する。
4. 変更ノードデータを受信した各情報取得部は、状態データを作成する。
5. 情報取得部は作成した状態データを情報管理部に送信する。
6. 状態データを受信した情報管理部は、入力データを作成する。
7. 入力データを提案生成部に送信する。
8. 提案生成部は、入力データからシステム出力値を計算して、正解出力値との誤差を修正する。

5.3.2 プロアクティブ動作

図 7 にプロアクティブ動作の手順を示し、以下で説明する。

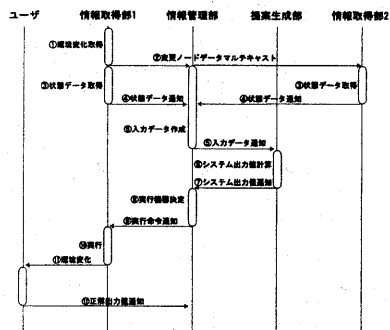


図 7: プロアクティブ動作の手順

1. 情報取得部が環境情報の変化を感知し、変更ノードデータを作成する。
2. 変更ノードデータを情報取得部にマルチキャストで送信する。
3. 変更ノードデータを受信した各情報取得部が、状態データを取得する。
4. 状態データを情報管理部に送信する。
5. 状態データを受信した情報管理部は、入力データを作成する。
6. 入力データを提案生成部に送信する。

7. 提案生成部は、入力データからシステム出力値を計算する。
8. システム出力値を情報管理部に送信する。
9. システム出力値を受信した情報管理部は、実行機器を決定する。
10. 情報管理部は、実行命令を情報取得部に送信する。
11. 実行命令を受けた情報取得部は機能を実行する。

6 評価

本節では PSNA のプロトタイプシステムの評価を定量的評価と定性的評価に分けて行う。

6.1 定量的評価

定量的評価は PSNA 上に構築したプロアクティブテレビを用いて行った。評価項目は学習から動作提案決定までの全体的な処理時間、計算回数に対する正解率、学習対象数に対する学習処理時間の3つを設定した。測定方法はそれぞれの測定を100回計測しその平均をとった。また、測定対象としたデータパターンは、RF-ID[10]を持つユーザ A, B が2人とも部屋にいる場合にテレビをつけるタイプ α とユーザ C, D のどちらかが部屋にいる場合にテレビを付けるタイプ β の2つのタイプである。測定環境を表9に示す。

表 9: 測定環境

CPU	PentiumIII 650MHz
主記憶	128MB
OS	Linux version 2.2.18-0vl4.2

測定 1: 全体的な処理時間

タイプ α のパターンを用いて、出力値計算からセンサの重みを更新するまでにかかるシステム内部のオーバーヘッドを測定した。表10に測定結果を示す。

表 10: システム内部のオーバーヘッド

処理段階	タイプ α (ms)
出力値計算	163
中間層重み更新	5
誤差逆伝播	8
入力層重み更新	2
合計	178

タイプ α において、システム全体の処理時間の約90%が出力値計算が占めている。これにより、システムのオーバーヘッドを減らすためには、出力値計算時間の短縮が必要であることが分かる。

測定 2: 計算回数に対する正解率

タイプ α において、正解率が95%を越えるまで学習させた。図8にタイプ α の測定結果を示し、それを対数近似によって数式化したものを式1, 2に示す。

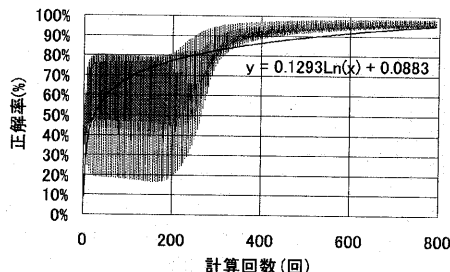


図 8: タイプ α の正解率の推移

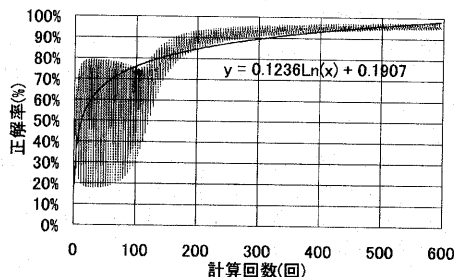


図 9: タイプ β の正解率の推移

$$C_{rate} = 0.1293 \log(N) + 0.0883; \quad (1)$$

$$C_{rate} = 0.1236 \log(N) + 0.1907; \quad (2)$$

式1, 2は、それぞれのパターンにおける正解率(C_{rate})を計算回数(N)で表したものである。両者とも正解率が100%に向けて収束している。従って、タイプ α 及びタイプ β のようなユーザの動作については、正解率が収束することが分かる。

測定 3: 学習対象数に対する学習処理時間

タイプ α のパターンを用いて、1つの情報家電機器において扱う学習対象が増加した時の学習にかかる処理時間を測定した。図10に測定結果を示し、それを線形近似によって数式化したものを式3に示す。

$$T(Learn) = 0.001x + 165.36 \quad (3)$$

式3は、出力値計算処理時間(T_{learn})を学習対象数(N)で表したものである。式3の傾きが0.001であることから、学習対象数に対するスケラビリティがあることが分かる。

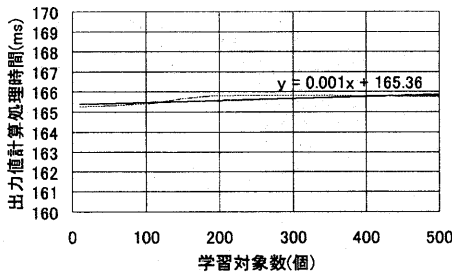


図 10: 出力値計算時間の推移

6.2 定性的評価

本節では、第 3.2 節で挙げたプロアクティブ利用モデルのシステム要件に基づいて、PSNA の性能評価を行う。

自律性

本システムの学習機能は、センサなどのから得られる外部情報をもとに、重みを更新している。したがって、仮定された状況以外でも行動でき、自律性を確保できる。

双方向性

本システムの学習機能によって更新された重みデータをもとに、システム出力値をユーザに提供している。また、システム出力値に対するユーザの評価を重みデータに反映する機能も備えている。従って、システムとユーザの双方向通信を実現した。

操作簡便性

本システムの学習機能はセンサ情報が変化した場合にシステム出力値を計算し、適切な機能が合った場合にその機能を実行できる。従って、ユーザは環境が変化した場合、要求を明示することなく、機能を利用できるため、操作簡便性がある。

7 今後の課題

PSNA ではユーザの個人情報機器に、登録して保持する手法をとっている。しかし、ユーザの個人情報は、ユーザ本人だけが保持していることが望ましい。現在、ユーザが常に着用する形で運用できるウェアラブルコンピュータ (WC)[11] の研究や WC を用いたアプリケーションフレームワーク [12] の研究が進められている。このようなユーザ周辺の WC 環境を利用して個人情報を管理し、PSNA のユーザ認証機構実現する必要がある。また、機器が動作提案をユーザに伝えるユーザインタフェースは、機器インタフェースの実装に依存しているが、今後、WC を利用し、動的に応答インタフェースを切り換え可能にする必要がある。

8 おわりに

本稿では情報家電機器のリアクティブな利用形態による問題点を解決するために、プロアクティブな利用

モデルを提案し、これを実現するための PSNA の設計、実装、評価を行った。本システムを利用することで、ユーザは機器利用に必要な情報収集や機器操作を補助できるため、ユーザの操作簡便性が向上する。プログラマは PSNA を用いることにより、プロアクティブアプリケーションを容易に構築できる。また、PSNA はアプリケーションごとに学習アルゴリズムを変更して、多様なプロアクティブアプリケーションの構築を支援する。

謝辞

本研究を進めるにあたり、慶應義塾大学徳田研究室の皆様にも多大な御助言、御協力を頂きました。特に同研究室中澤仁氏には数多くの議論をして頂き、深く感謝致します。

参考文献

- [1] Sony and Matsushita and Philips and Thomson and Hitachi and Toshiba and Sharp and Grundig: Specification of the Home Audio/Video Interoperability (HAVi) Architecture (2000). <http://www.havi.org/techinfo/docs/HAVi10.pdf>.
- [2] Sun Microsystems Inc.: Jini Architecture Specification Version1.1 (2000). <http://www.sun.com/jini/specs/jini1.1.pdf>.
- [3] Universal Plug and Play Forum: Universal Plug and Play (1999). <http://www.upnp.org>.
- [4] Sharp: Sharp Space Town (1999). <http://www.spacetown.ne.jp>.
- [5] 上捨石弘, 横井茂樹, 安田孝美: Jini を利用した分散環境における GP マルチエージェント学習, コンピュータシステム・シンポジウム, 情報処理学会 (1999).
- [6] C.Mozer, M.: The neural network house: An environment that adapts to its inhabitants, *The American Association for Artificial Intelligence Spring Symposium* (1996).
- [7] Okoshi, T., Wakayama, S., Sugita, Y., Aoki, S., Iwamoto, T., Nakazawa, J., Nagata, T., Furusaka, D., Iwai, M., Kusumoto, A., Harashima, N., Yura, J., Nishio, N., Tobe, Y., Ikeda, Y. and Tokuda, H.: Smart Space Laboratory Project: Toward the Next Generation Computing Environment, *Proceedings of IEEE Third Workshop on Networked Appliances (I-WNA) 2001* (2001).
- [8] Corp., D. S.: Introducing TINi: Tiny InterNet Interface (2000). <http://www.ibutton.com/TINI/index.html>.
- [9] 横河電機株式会社: Compact Field Server DUONUS (2000). <http://www.yokogawa.co.jp/DUONUS/Product/Product.htm>.
- [10] InData Systems: RF Code Spider RFID System. <http://www.indatasys.com/html/products/>.
- [11] MIT: Wearable Computing Web Page. <http://wearables.www.media.mit.edu/projects/wearables/>.
- [12] 村瀬正名, 若山史郎, 権藤俊一, 永田智大, 西尾信彦, 徳田英幸: 計算機環境に応じてサービス提供方式を適応させる通信用ツールキット, システムソフトウェアとオペレーションシステム研究会, 情報処理学会 (1999).