

ホームネットワークシステムにおけるサービス要求を考慮した 環境競合検出法

池上 弘祐[†] 井垣 宏[†] 中村 匡秀[†]

[†] 神戸大学 〒657-8501 神戸市灘区六甲台町 1-1
E-mail: †{ikegami,igaki,masa-n}@cs.kobe-u.ac.jp

あらまし ホームネットワークシステム (HNS) を利用して複数の家電を協調動作させる, 家電連携サービスの研究が進んでいる. 単体では正常に動作する連携サービスを複数実行したとき, それらのサービスの持つ要求が互いに干渉・衝突し, ユーザの意図しない振る舞いが生じることがある. これをサービス競合と呼ぶ. 我々は先行研究において, HNS には「機器競合」と「環境競合」という2種類のサービス競合が存在することを述べている. このうち環境競合については, その定式化および検出法について十分な考察が行われていなかった. 本稿では, まずサービス要求と HNS 機器の環境に対する作用を踏まえて, HNS における環境競合を定式化するモデル化手法を提案する. さらに, 提案モデルを用いて, 環境競合を検出する手法を提案する. ケーススタディとして, 既存の連携サービスを提案モデルに基づいて記述し, 環境競合の検出を行う.

キーワード ホームネットワーク, 連携サービス, サービス競合, 環境競合

Detecting Environment Feature Interactions in Home Network System Based on Service Requirements

Kosuke IKEGAMI[†], Hiroshi IGAKI[†], and Masahide NAKAMURA[†]

[†] Kobe University rokkoudaityou 1-1, nada-ku, Kobe, Hyogo, 657-8501 Japan
E-mail: †{ikegami,igaki,masa-n}@cs.kobe-u.ac.jp

Abstract The integrated services of the home network system (HNS) orchestrate multiple networked home appliances to achieve value-added and comfortable services for home users. If multiple services are used at the same time, functional conflicts may occur among the services, which results in unexpected behaviors. This is known as feature interactions (FIs). We have previously shown that there are two kinds of FIs in the HNS, namely “appliance interactions” and “environment interactions”. Among the two kinds, we have not yet studied well the environment interactions. In this paper, we first propose a modeling method that formulates the environment interactions. To characterize the environment interactions, we especially count the service requirements to the environment and effects of appliance operations to the environment. Using the model, we also present a method that detects the environment interactions. As a case study, we describe actual HNS services with the proposed model, and detect environment interactions within the services.

Key words home network, integrated service, feature interaction, environment interaction

1. はじめに

近年, 家電機器のデジタル化やネットワーク化, センシング技術の発展に伴って, それらを利用したホームネットワークシステム (HNS) の研究・開発が盛んである. HNS では, テレビ, エアコン, 照明といった機器の持つ機能が, API(Application Program Interface) としてネットワークに公開される. これら

を利用して, 複数の家電やセンサを連携, 協調動作させ, 付加価値の高い連携サービスを実現することができる. 連携サービスの例として次のようなものが存在する.

DVD シアターサービス (DVD-T): テレビ, DVD レコーダ, カーテン, リビング照明を連動し, ワンタッチで映画館の雰囲気映画を視聴できるサービス. 起動すると, DVD とテレビの電源が入り, カーテンが開まり照明が消えて室内が暗くなり,

最後に映画が再生される。

お帰りサービス (CH): ユーザが帰宅すると玄関とリビングの照明が自動的に点灯し、部屋が明るくなる。

複数の連携サービスを同時に実行した場合、機器や環境に対する要求が干渉・衝突し、正常に動作しないことがある。これをサービス競合 (Feature Interactions, FI) という [1] [2]。例えば、ユーザ A が DVD シアターサービスを実行中にユーザ B が帰宅すると、以下のようなサービス競合が生じる。

サービス競合 (DVD-T & CH) -(a): DVD-Tによって消されていたリビング照明が CH によって点灯し、映画館の雰囲気損なわれる。

サービス競合 (DVD-T & CH) -(b): CH によって点灯した玄関照明が間接的にリビングを明るくし、映画館の雰囲気損なわれる。

我々は先行研究 [3] において、HNS には機器競合と環境競合という 2 種類のサービス競合が存在することを示している。ここで機器競合とは、異なる連携サービスが同一の機器に対して、両立しえない機器操作を要求した場合に発生するサービス競合である。上記 (a) のサービス競合に該当する。DVD-T の「リビング照明.off()」と CH の「リビング照明.on()」は、同じリビング照明で同時に実行できない操作である。

それに対して環境競合は、異なる連携サービスが異なる機器に対して機器操作を要求し、それらの操作が HNS をとりまく環境を介して間接的に矛盾する場合に発生するサービス競合である。上記 (b) のサービス競合に該当する。DVD-T の「リビング照明.off()」と CH の「玄関照明.on()」は、操作する機器は異なるが、「部屋の明るさ」を介して間接的に衝突する。他にも、DVD を視聴するサービスと音楽を鳴らすサービスの間、あるいは、エアコン冷房を使うサービスとファンヒータを使うサービスの間にも環境競合が生じる。

2 種類の競合のうち機器競合については競合検出・解消システムを実装し、実際の HNS に適用し有効性を評価している [4] [5]。しかしながら [3] の定義ではいくつかの環境競合をうまく表現できないことがわかった。以前の定義では「複数の機器操作が同一の環境プロパティへ同時アクセスを行う」という粗い条件により、環境競合を直感的に定式化した。しかしこの条件にはサービスの要求が考慮されておらず、競合と考えられないものも競合と判断してしまう。例えば上記 (b) の競合は、DVD シアターの明るさに対する要求次第では無視できるかもしれない。

本稿では HNS における環境競合をより正確に検出するために、HNS を取り巻く環境、機器操作の環境に対する影響、環境競合をモデル化する新たな手法を提案する。また、提案した環境競合のモデル、及び検出法を利用して、連携サービス間の環境競合の検出実験を行い、有効性の評価を行う。

2. 先行研究 [3]

2.1 HNS のモデル化

我々は先行研究 [3] において、HNS の構成要素を「オブジェクト」としてモデル化する手法を提案している。この手法では、HNS における各機器をプロパティ (内部状態を表す属性) とメ

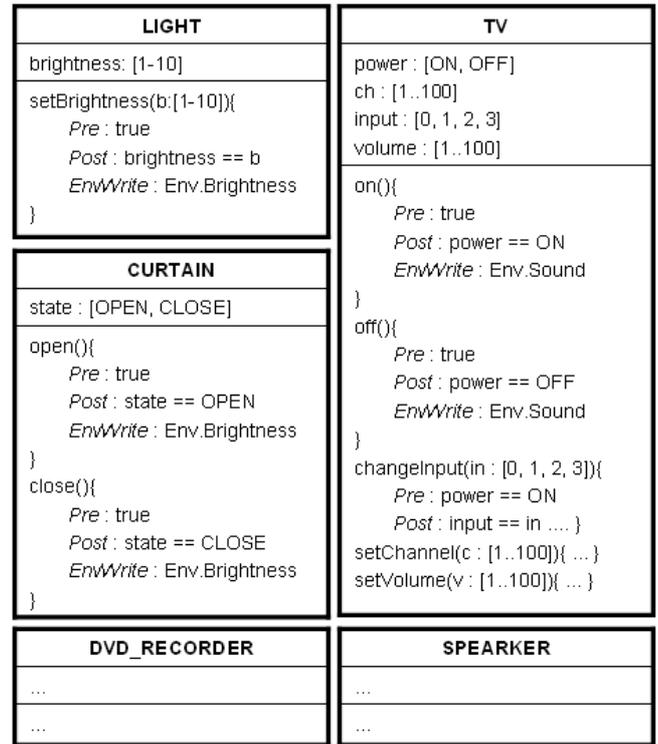


図 1 先行研究 機器モデル

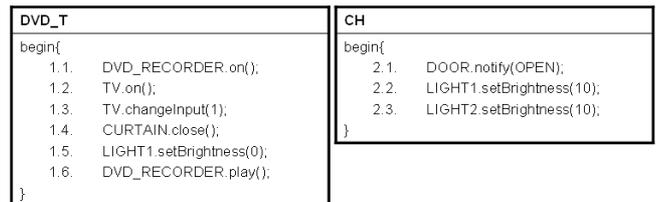


図 2 先行研究 連携サービスモデル

ソッド (機器操作) によりモデル化する。

図 1 に HNS の機器モデルの例を UML クラス図風で表したものを示す。例えば照明 (LIGHT) は明るさ (brightness) というプロパティを持ち、setBrightness() というメソッドによって操作されることを表している。各メソッドは、以下に示す事項で定義される。

- *Pre*: 事前条件。メソッドの実行に必要なとなる機器の状態に関する条件
- *Post*: 事後条件。メソッドの実行後に成立する機器の状態に関する条件
- *EnvRead*: メソッドによって参照される環境プロパティの集合
- *EnvWrite*: メソッドによって更新される環境プロパティの集合

連携サービスは、複数の機器メソッドの系列によって定義される。図 2 は、1. で述べた DVD シアターサービスおよびお帰りサービスのシナリオを表している。機器 d におけるメソッド m を、 $d.m$ と表記し、図中の DVD_RECORDER.on(), TV.on() といった一行一行がメソッドを表している。

環境モデルは、HNS が存在する空間の環境属性（環境プロパティ）を持つオブジェクトとしてモデル化される。代表的な環境プロパティには、室温 (Env.Temperature), 明るさ (Env.Brightness), 音量 (Env.Sound), 湿度 (Env.Humidity), 消費電力 (Env.Electricity) 等が存在する。

2.2 サービス競合の定式化

上記のモデルを用いて、機器競合および環境競合は以下のように定義される。 s_1, s_2 を任意の連携サービス, m_1, m_2 をそれぞれ s_1, s_2 に含まれる任意の機器メソッドとする。このとき、「 s_1 と s_2 が機器競合を生じる」とは、同機器内のメソッド m_1, m_2 が存在して以下の条件が成り立つときである。

- 条件 A1 : $Post(m_1) \wedge Post(m_2) = \perp$ または,
- 条件 A2 : $Post(m_1) \wedge Pre(m_2) = \perp$

直感的には、二つのサービスが同一機器内で目的の矛盾するメソッドを実行する (条件 A1) か、一方のメソッド実行により他方のメソッドが実行できなくなる (条件 A2) かで、機器競合を性質づけている。一方、「 s_1 と s_2 が環境競合を生じる」とは、異なる機器のメソッド m_1, m_2 が存在して以下の条件が成り立つときである。

- 条件 E1 : $EnvWrite(m_1) \cap EnvWrite(m_2) \neq \phi$ または
- 条件 E2 : $EnvWrite(m_1) \cap EnvRead(m_2) \neq \phi$

直感的には、2つのサービスが共通の環境プロパティを更新する (条件 E1) か、一方が更新したものを他方が参照するとき (条件 E2) に、環境競合が生じ得るとしている。

上記定義と前節のモデルによって、1. で述べた2つのサービス競合 (a),(b) を (一応は) 説明可能である。

サービス競合 (a) は、DVD-T のメソッドである LIGHT1.setBrightness(0) と CH のメソッドである LIGHT1.setBrightness(10) の競合として表される。この2つのメソッドはどちらも機器 LIGHT1 のメソッドであり、LIGHT1 の状態である brightness をそれぞれ異なる値で書き換える Post を持つ。これが条件 A1 に反するので機器競合が生じる。

また、サービス競合 (b) は、DVD-T のメソッドである LIGHT1.setBrightness(0) と CH のメソッドである LIGHT2.setBrightness(10) の競合として表される。この2つのメソッドは、同じ環境プロパティ Env.Brightness を更新する EnvWrite を持つ。これは条件 E1 に反するので環境競合が生じる。

2.3 環境競合に関する課題

上記の定式化において、機器競合に関しては実際の HNS サービスに問題なく適用でき、機器競合検出・解消システムを構築することができている [4]。しかしながら、環境競合に関して以下のような問題がある。

問題 P1: 環境への影響の向き, 大きさが考慮されない。

例えば HNS において、照明機器が複数存在することは十分に考えられる。同じ HNS 上に機器 LIGHT1 と機器 LIGHT2 が存在するとき、LIGHT1.setBrightness(10) と LIGHT2.setBrightness(10) は、どちらも Env.Brightness を更新するので、条件 E2 により環境競合となってしまう。しかし、照明を明るくするというメソッド同士が競合を起こすのは不自

然である。また、影響の向きが同じであっても、その強さは全く考慮されていない。例えば、テレビは照明ほどではないにしても、部屋の明るさに多少の影響を与えるはずである。しかし、既存のモデルでは表現できない。

問題 P2: 環境競合の条件にサービス要求が考慮されない。

環境競合をより正確に捉えるためには、サービスが環境に対してどのような状態を要求しているかを明示的に考慮すべきである。1. で述べたサービス競合-(b) は、DVD シアターが求めるリビングの部屋の明るさを考慮せず、どんなときでも環境競合と判断している。DVD シアターサービスが求めるリビングの明るさがどれくらいか、すなわちサービス要求を考慮することで、環境競合とすべきかの判断材料になる。

3. 環境を考慮した HNS 機器モデル化手法

3.1 HNS 機器の環境インパクトモデル

HNS の機器が環境に与える影響 (インパクト) をより詳細に捉えるため、状態遷移機械 (FSM) によるモデルを導入する。

[定義 1] E を環境プロパティの集合, d を HNS 機器とする。 d の環境インパクトモデルは状態遷移機械 $I_d = (S_d, M_d, T_d, \Delta, s_0)$ で定義される。ここで、

- S_d は d の状態の集合,
- M_d は d の持つ機器メソッドの集合,
- $T_d \subseteq S_d \times M_d \times S_d$ は状態遷移の集合,
- $\Delta : T_d \times E \rightarrow Z$ は環境インパクト関数,
- s_0 は初期状態。

環境インパクトモデルはメソッド実行による機器の状態遷移をモデル化したものである。各状態遷移が環境プロパティに対しどれだけインパクトを持つかを表す。

図 3 に本稿で利用する機器の環境インパクトモデルを示す。例えば TV は OFF(T_1), ON(T_2) の 2 状態を持ち、メソッド on() の実行によって T_1 から T_2 への遷移が起こる。遷移に伴って生じる環境インパクトは 3 つあり、

- $\Delta(\text{Env.Brightness}, (T_1, \text{on}(), T_2)) = +200$
- $\Delta(\text{Env.Sound}, (T_1, \text{on}(), T_2)) = +200$
- $\Delta(\text{Env.Electricity}, (T_1, \text{on}(), T_2)) = +300$

と定義されており、明るさ、音量、消費電力へそれぞれ 200,200,300 の正のインパクトを与えている。なお図では、 $\Delta(e, t) = +v$ (または $-v$) を、遷移 t のラベルとして $\Delta e = +v$ (または $-v$) と簡略表記にし、環境プロパティは頭文字で略記している。

環境インパクトは、機器メソッドの持つ環境に影響を与える能力の大きさを、加算が可能な次元に正規化したものである。環境インパクトの値 v は、環境への影響を相対的に表す数値であるので、必ずしも明度、温度等の実際の変動値と一致するものではない。実用上は v を環境プロパティの実際値への変換式が用意されているものと仮定する (本稿では紙面の制限のため詳細は言及しない)。

環境インパクトモデルの導入により、機器操作が環境へ与える影響の向き大小を表現でき、問題 P1 の解決を図る。

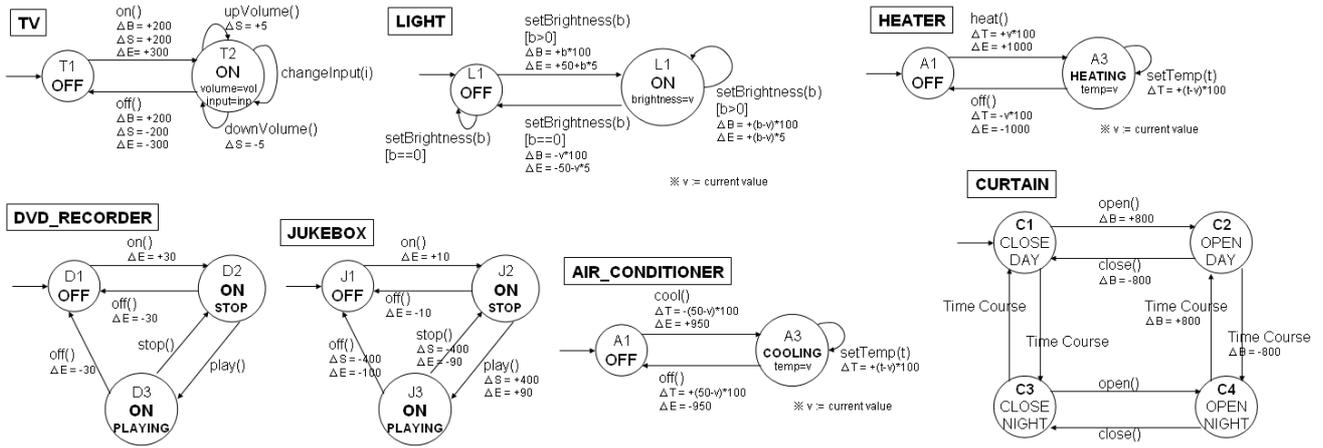


図 3 機器状態遷移モデル

3.2 連携サービス環境作用

連携サービスによって複数の機器メソッドが実行されいくと、環境に対して様々なインパクトが加えられていく。ここで、環境プロパティごとに環境インパクトの累計を計算し、連携サービスの環境に対する作用（環境作用、Environmental Effect）と定義する。

[定義 2] s を連携サービス, $[t_1, t_2, \dots, t_x]$ を s の実行によって起こった環境インパクトモデルの遷移系列とする。この時、ある環境プロパティ $e \in E$ に対する s の環境作用 $EE(s, e)$ を以下のように定義する： $EE(s, e) = \sum_{i=1}^x \Delta(e, t_i)$ 。

また、 $[t'_1, t'_2, \dots, t'_m]$ を現在実行中の全ての連携サービスの実行によって起こった遷移系列とする。この時、特定のサービス s によらない環境作用 $EE(e)$ を以下のように定義する： $EE(e) = \sum_{i=1}^m \Delta(e, t'_i)$ 。

最後に、ある連携サービス s を除くサービスによる環境作用を s の排他環境作用 $XEE(s, e)$ と呼び、以下のように定義する： $XEE(s, e) = EE(e) - EE(s, e)$ 。

$EE(s, e)$ は直感的に、連携サービス s の実行によって環境プロパティ e にどれだけのインパクトが蓄積されるかを定量化するものである。これを実行中のサービス全体に拡張したものが $EE(e)$ である。また $EE(e)$ のうち、 s が寄与しない部分が $XEE(s, e)$ で表され、 s 以外のサービスの実行によって蓄積されるインパクトを定量化する。

例として、図 2 の DVD シアターサービスの環境作用を図 3 の環境インパクトモデルを用いて説明する。DVD シアターサービスの機器メソッドのうち、環境インパクトに寄与するメソッドは、DVD_RECORDER.on(), TV.on() である。よって、 $EE(DVD_T, Env.Brightness)$ は TV.on() の +200、 $EE(DVD_T, Env.Sound)$ は、TV.on() の +200、 $EE(DVD_T, Env.Electricity)$ は、TV.on() の +300 と DVD_RECORDER の +30 を加えて +330 となる。

3.3 連携サービス環境要求

問題 P2 を解決するため、個々の連携サービスが環境に対して求める条件を環境作用を用いた論理式として定義する。この条件式をサービス環境要求と呼ぶ。

[定義 3] s を連携サービス, E を環境プロパティの集合とする。 s のサービス環境要求 $EReq(s)$ とは、環境作用 ($EE(s, e)$, $EE(e)$, $XEE(s, e)$ ($e \in E$)) を用いた条件の論理和 (\vee)、論理積 (\wedge) または否定 (\neg) で定義される。

例えば、DVD シアターサービスの環境要求として、「サービス実行中は、周りができるだけ暗くあってほしい」かつ「サービス実行中は、周りができるだけ静かであってほしい」という条件を考える。この要求は例えば以下のように記述できる。

$$EReq(DVD_T) = [XEE(DVD_T, Env.Brightness) < 200] \wedge [XEE(DVD_T, Env.Brightness) < 200]$$

「周りが」という条件を排他環境作用を用いて記述していることに注意されたい。

3.4 環境制約

サービスが環境に対して環境要求を示しても、それらは元来環境で定められる「使用ルール」や「容量」といった制約に従う必要がある。例えば、どの家にも電気容量があり、連携サービスは容量を超えた電力消費の要求はできない。このように、サービスに依存せずに環境が元来持つ制約条件を環境制約と呼び、環境作用の論理式で定義する。

[定義 4] E を環境プロパティの集合とする。HNS の環境制約 G とは、環境作用 ($EE(e)$ ($e \in E$)) を用いた条件の論理和 (\vee)、論理積 (\wedge) または否定 (\neg) で定義される。

例えばある家では、「同時使用電力が 2000kW 未満でなければならない」という使用ルールが決まっていたとする。この時の環境制約は、

$$G = EE(Env.Electricity) < 2000$$

と記述される。連携サービス s の環境要求 $EReq(s)$ は通常 G を満たす範囲で与えなくてはならない。

3.5 モデル構築例

提案した環境インパクトモデル (図 3) およびサービス環境要求を考慮した連携サービスの記述例を図 4 に示す。

DVD_Theater サービス (DVD_T): TV で快適に DVD を視聴するサービス。「周りが静かであってほしい」と「周りが暗くあってほしい」というサービス環境要求を持つ。

Jukebox サービス: JUKEBOX で音楽を流すサービス。

<pre> DVD_Theater (DVD_T) begin{ 1.1.1. DVD_RECORDER.on(); 1.1.2. TV.on(); 1.1.3. TV.changelnput(1); 1.1.4. CURTAIN.close(); 1.1.5. LIGHT.setBeighthness(0); 1.1.6. DVD_RECORDER.play(); } EReq{ 1.2.1. XEE(DVD_T, Env.Sound) < 200 1.2.2. XEE(DVD_T, Env.Brightness) < 200 } </pre>	<pre> Coming_Home (C_H) begin{ 3.1.1. AIR_CONDITIONER1.cool(); 3.1.2. AIR_CONDITIONER.setTemp(26); 3.1.3. LIGHT.setBrightness(10); } EReq{ } </pre>
<pre> Jukebox begin{ 2.1.1. JUKEBOX.on(); 2.1.2. JUKEBOX.play(); } EReq{ } </pre>	<pre> AirCondition (AC) begin{ 4.1.1. AIR_CONDITIONER2.cool(); 4.1.2. AIR_CONDITIONER2.setTemp(26); } EReq{ 4.2.1. XEE(AC, Env.Temperature) < 100 } </pre>
	<pre> Heating begin{ 5.1.1. HEATER.heat(); 5.1.2. HEATER.setTemp(22); } EReq{ 5.2.1. XEE(Heating, Env.Temperature) > -100 } </pre>

図 4 提案 連携サービスモデル

Coming_Home サービス (C_H) : ユーザーの帰宅時に、照明を付け、空調を行うサービス。

AirCondition サービス (AC) : 冷房を付け部屋を涼しくするサービス。「他の機器に部屋を暖めてほしくない」というサービス環境要求を持つ。

Heating サービス : ヒーターを付け部屋を暖めるサービス。「他の機器に部屋を冷やしてほしくない」というサービス環境要求を持つ。

連携サービス DVD.Teater, AirCondition, Heating サービスが持っているサービス環境要求は、それぞれサービスの排他環境作用が一定値以下もしくは以上であることを要求するものである。

サービス競合とは直感的に「単体では正常に動作するサービスを同時に複数実行した際に生じる不具合」を指す。このことを環境競合の面から考える。

まず、単体の連携サービスが(環境の面において)正常に動作するという概念を定義する。

[定義 5] s を連携サービスとする。 s が正常動作するとは、環境インパクトモデルにしたがって実行された s が、環境要求 $Req(s)$ と環境制約 G を共に満たすことと定義する。またこのことを、 $s \vdash Req(s) \wedge G$ と書く。

次に、連携サービス s_1, s_2 を考える。 s_1 と s_2 の環境競合は「それぞれは正常動作するが、両方実行すると正常動作しない」という条件になるから、以下のように定義できる。

[定義 6] (2 者間環境競合) s_1, s_2 を連携サービス、 $[s_1; s_2]$ を両サービスの逐次実行 (s_1 の後 s_2 の順) とする。この時、 s_1 と s_2 が環境競合を起こすとは、以下の条件が全て満たされることである。

条件 B1 : $s_i \vdash EReq(s_i) \wedge G$ ($i = 1, 2$)

条件 B2 : $[s_1; s_2] \not\vdash EReq(s_1) \wedge EReq(s_2) \wedge G$

s_1 と s_2 が環境競合を起こすとき、 $EnvFI(s_1, s_2)$ と書く。

条件 B1 は s_1 と s_2 がそれぞれ単独で正常動作することを表す。条件 B2 は両者を実行すると、 s_1 または s_2 の環境要求が

満たされなくなるか、環境制約 G を満たせなくなることを表している。

環境競合は複数のサービスからの影響を表す環境作用と、環境制約、サービス環境要求との間の競合である。よって、環境競合は 2 つのサービス間に留まらず、 n 個のサービス間で発生しうる。 n 者間に拡張した環境競合の定義を以下に示す。

[定義 7] (n 者間環境競合) $[s_1; s_2; \dots; s_{n-1}]$ を現在実行中の連携サービスの系列とし、 s_n を新たに実行する連携サービスとする。このとき、実行中のサービス $\{s_1, s_2, \dots, s_{n-1}\}$ と s_n が環境競合を生じる ($EnvFI(\{s_1, \dots, s_{n-1}\}, s_n)$ と書く) とは、以下の条件が全て満たされることである。

条件 C1 : $\neg EnvFI(\{s_1, \dots, s_{i-1}\}, s_i)$ ($i = 2, 3, \dots, n-1$)

条件 C2 : $[s_1; s_2; \dots; s_n] \not\vdash EReq(s_1) \wedge \dots \wedge EReq(s_n) \wedge G$

条件 C1 は 2 者間環境競合 (定義 6) を基底とする再帰的な条件となっている。 $n-1$ 個のサービス間では環境競合が起こらなかったことを表している。この条件の下、条件 C2 で n 個目のサービスを実行すると環境要求または環境制約が満たせなくなってしまうということを表している。

ここで例として、図 4 に示す DVD.Theater サービスと Coming_Home サービスの環境競合を説明する。まず、DVD.Theater サービスを実行する。サービスは単独で正常に動作する前提により問題なく実行され、DVD.Theater サービスの関わる環境作用は、 $EE(DVD.T, Env.Brightness) == 200, EE(DVD.T, Env.Sound) == 200, EE(DVD.T, Env.Electricity) == 330$ となる。次に、Coming_Home サービスを実行する。機器競合は発生せず、C.H サービスの関わる環境作用が $EE(C.H, Env.Brightness) == 1000, EE(C.H, Env.Temperature) == -2400, EE(C.H, Env.Electricity) == 1100$ となるが、これにより排他環境作用 $XEE(DVD.T, Env.Brightness) \in EReq(DVD.T)$ を満たさなくなるため、環境競合が生じる。

3.6 環境競合の検出

定義 7 に基づき、環境競合を検出する手法を考える。一般にサービス競合を検出する方法として以下の二つの方法が考えられる [1]。

オフライン検出 : 事前に競合する連携サービスの全ての組み合わせについて競合検出を行う。

オンライン検出 : サービス実行時に、実際に実行中のサービスの組み合わせにおいて競合検出を行う。

n 者間環境競合の検出にあたっては、連携サービスの可能な組み合わせ数が膨大になることから、本稿ではオンライン検出を行う手法を検討する。以下に環境競合のオンライン検出アルゴリズムを示す。

オンライン環境競合検出法

前提 : サービス s_1, \dots, s_n が実行中で、新たにサービス s_{new} を実行する。

出力 : $EnvFI(\{s_1, \dots, s_n\}, s_{new})$ が成立するかどうか。

手順 :

(1) 各環境プロパティ $e \in E$ について、実行中のサービス s_1, \dots, s_n に対して環境作用 $EE_{prev}(e)$ を求める。

表 1 検出された環境競合

	DVD_Theater	JukeBox	Coming_Home	AirCondition	Heating
DVD_Theater		環境競合 XEE(DVD.T, Sound) < 200	環境競合 XEE(DVD.T, Bright) < 200	なし	なし
JukeBox	環境競合 XEE(DVD.T, Sound) < 200			なし	なし
Coming_Home	環境競合 Xeff(DVD.T, Bright) < 200	なし		なし	環境競合 XEE(Heating, Temperature) > -100
AirCondition	なし	なし	なし		環境競合 XEE(AirCondition, Temperature) < 100 XEE(Heating, Temperature) > -100
Heating	なし	なし	環境競合 XEE(Heating, Temperature) > -100	環境競合 XEE(Heating, Temperature) > -100 XEE(AirCondition, Temperature) < 100	

(2) 環境インパクトモデルを参照し, s_{new} で実行されるモジュールで実行される遷移系列 $T_{new} = [t_1, t_2, \dots, t_m]$ を求める.

(3) T_{new} に含まれる遷移に基づき, 全ての環境プロパティ $e \in E$ について, $EE(s_{new}, e) = \sum \Delta(e, t_i)$ を算出する.

(4) S_{new} の実行によって累計される新たな環境作用 $EE_{new}(e) = EE_{prev}(e) + EE(s_{new}, e)$ を計算する.

(5) $EE_{new}(e)$ について, 環境制約 G およびサービス環境要求 $EReq(s_1), \dots, EReq(s_n), EReq(s_{new})$ が全て満たされるか検査する.

(6) いずれかが満たされていない場合, 環境競合 $EnvFI(\{s_1, \dots, s_n\}, s_{new})$ を検出して終了する.

上記のアルゴリズムにおいては, s_{new} を実際に実行する前に環境競合のチェックを行うことに注意されたい. 競合が検出されなかった場合には, s_{new} を実際に実行するが, 検出された場合には何らかの競合解消法がとられるべきである. 環境競合のための競合解消法については今後の課題としたい.

4. ケーススタディ

図 4 で示した 5 種類の連携サービスに対し, 2 者間環境競合のオンライン検出を行った. 今回は, 一方の連携サービスが実行された状態で, 他方の連携サービスを後から実行した場合を想定して, 想定した環境競合が全て検出できるかを検査した. その結果, 全ての連携サービスの組み合わせにおいて, 期待された全ての環境競合を検出できたことを確認した.

表 1 に, 環境競合の検出結果を示す. 各行に先に実行する連携サービスを示し, 各列に後から新たに実行する連携サービスを表している. また, 表中で環境競合が発生した組み合わせの下段には, 成り立たなかったサービス環境要求を示した.

以降, 環境競合が検出された特徴的な検出結果を詳説する.

環境競合 (DVD シアター&JukeBox)

DVD_Theater サービスを実行中に, JukeBox サービスを実行する例である. JukeBox サービスの実行により, Env.Sound への環境インパクトが追加される. その結果, DVD_Theater サービスのサービス要求「周りが静かであってほしい」が満たされなくなる. 結果として, 静かに映画を見たいというサービス要求が満たされなくなってしまう.

環境競合 (AC&Heating)

AirCondition サービスを実行中に, Heating サービスを実行する例である. AirCondition サービスにより, Env.Temperature を大きく下げる環境インパクトが追加されている. 新たに Heating サービスを実行することで,

Env.Temperature を大きく上げる環境インパクトが追加され, AirCondition の「部屋を暖めてほしくない」というサービス要求が満たされなくなるほか, Heating の「部屋を冷やしてほしくない」というサービス要求は最初から満たされない.

5. おわりに

本稿では, 環境を介するサービス競合である環境競合の定式化を行い, 環境競合の検出法を提案した. 先行研究における環境競合の問題点が解決され, より幅広く連携サービスの要求を満たし, ユーザーの快適性を向上させるものとなった. 今後は, 環境競合の解消法の提案を行い, 機器競合を含めたサービス競合検出, 解消システムを実現することを考えている.

謝 辞

この研究の一部は, 科学技術研究費 (若手研究 B 20700027, 21700077), および, パナソニック電工株式会社の助成を受けて行われている.

文 献

- [1] M. Calder, E. Magill, M. Kolberg, and S. Reiff-Marganiec, "Feature Interaction: A Critical Review and Considered Forecast", *Computer Networks*, Volume 41/1, pp.115-141, North-Holland, January 2003.
- [2] M. Kolberg, E. H. Magill, and M. Wilson, "Compatibility Issues Between Services Supporting Networked Appliances", *IEEE Communications Magazine*, vol. 41, no. 11, Nov 2003 pp. 136-147
- [3] M. Nakamura, H. Igaki, K. Matsumoto. "Feature Interactions in Integrated Services of Networked Home Appliances -An Object-Oriented Approach-," In Proc. of Int'l. Conf. on Feature Interactions in Telecommunication Networks and Distributed Systems (ICFI'05), pp.236-251, July 2005.
- [4] M. Nakamura, H. Igaki, Y. Yoshimura, and K. Ikegami. Considering Online Feature Interaction Detection and Resolution for Integrated Services in Home Network System. *10th International Conference on Feature Interactions in Telecommunications and Software Systems (ICFI2009)*, 191-206, June 2009.
- [5] 池上弘祐, 吉村悠平, 井垣宏, 中村匡秀, "サービス期間を考慮したホームネットワークサービス競合検出・解消システムの実装," 電子情報通信学会 OIS 研究会, IEICE-OIS-462, pp.7-12, March 2009.