

〔公 開〕

TR-C-0110

S T R 手法と通信サービス競合
（非決定性）検証に関する考察

原田 良雄
Yoshio HARADA

1 9 9 5 3 . 2 0

A T R 通信システム研究所

STR手法と通信サービス競合（非決定性）検出
に関する考察

原田 良雄

目次

第1章	はじめに	1-1
第2章	STR手法について	2-1
2.1	通信サービス設計の範囲	2-1
2.2	STR手法の記述方式の概要	2-3
2.3	STR手法の適用方式の概要	2-8
2.4	サービスの記述性について	2-14
2.5	考察	2-19
第3章	通信サービス競合（非決定性）検出方式	3-1
3.1	通信サービス競合（非決定性）	3-1
3.2	非決定性の例	3-1
3.3	通信サービス競合（非決定性）検出方式の基本概念	3-4
3.4	考察	3-7
第4章	おわりに	4-1

1. はじめに

近年、電気通信分野での著しい技術革新の進展と通信サービスの普及に伴い、通信サービスに対する利用者の要求が多様化・高度化している。このような需要に対して、迅速な新規サービス開発と提供が必要となり、通信ソフトウェアの効率的かつ迅速な開発技術の確立が早急の課題となっている。通信サービス設計においては、新規サービス単独の動作仕様を規定するだけでなく、新規サービス仕様と既存サービス仕様との相互作用（サービスインタラクション）を解析し、サービス仕様間に生じる矛盾を解消し、サービス全体として矛盾のない仕様を作成する必要がある。ところが、一般に、相互作用の数は、既存サービス数が増えると組合せ的に増大し、その相互作用に隠れたサービス仕様矛盾の検証は、人手で行う事のできる範囲を越えた作業となっている。相互作用解消の問題は、1991年12月、CCITTによってインターネットネットワークのための技術的勧告として提示されたCS1 (IN Capability Set 1) においても、今後の重要な課題のひとつに挙げられている [1-2]。

サービス相互作用解析を機械的に支援するためには、まず、通信サービス仕様を形式的に記述する必要がある。通信サービス仕様の記述範囲は、システムをブラックボックスとみなし端末の動作に着目した動作を対象とし、サービス動作を形式的な規則形式により記述する手法 (STR: State Transition Rule) を提案している [9-11]。また、STR手法を用いて通信サービスは記述されていることを前提とし、新規サービスと既存サービスからサービス合成動作を作成し、そのサービス合成動作を基盤に相互作用に含まれる矛盾検出支援方式を提案している [14-16]。それらのおよび考察を含めてテクニカルレポート [21] に述べている。また、STR記述仕様につい

ては文献 [12] にまとめている。

本レポートでは、STR手法の記述性と通信サービス競合（非決定性）検出における状態列挙の違いに着目して補足説明する。

第2章では、STR記述手法について、記述範囲、記述性について議論する。

第3章では、通信サービス競合（非決定性）検出支方式の概要について述べ、2規則の競合判定に用いる状態の列挙の方法の違い（任意端末の状態列挙と複数端末状態列挙の方法）による利点と欠点について議論し、処理時間を考慮した融合案を示す。

第4章では、まとめについて述べる。

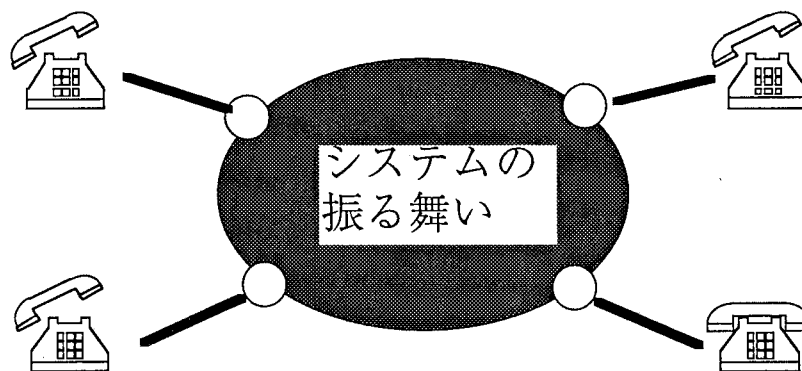
2. STR手法について

2. 1 通信サービス設計の範囲

本節では、通信サービスの設計対象とする範囲について述べる。利用者の視点から、外部仕様として、通信サービス動作を規定する。システム（ネットワーク）の振る舞いはブラックボックスとして扱う（図2-1）。図2-2および図2-3にサービス動作の例を示す。図2-2は、端末Aがアイドル状態のとき、受話器を挙げると、ダイヤル可能音が聞こえる状態（ダイヤル可能状態）に推移する動作を示す。図2-3は、端末Aがダイヤル可能状態、かつ、端末Aがアイドル状態のとき、端末Aが端末Bにダイヤルを行うと、端末AからBへの呼出中の状態（端末Aには呼戻音が出ており、端末Bには呼出音が出ている状態）に推移する動作を示す。

上記の外部仕様レベルでの、通信サービス記述および、サービス設計・検証支援を研究対象としている。

この記述手法 [9-11] をSTR (State Transition Rule) 手法と呼ぶ。



(注) システムの振る舞いを
ブラックボックスとして扱う

図2-1 通信サービス検討範囲

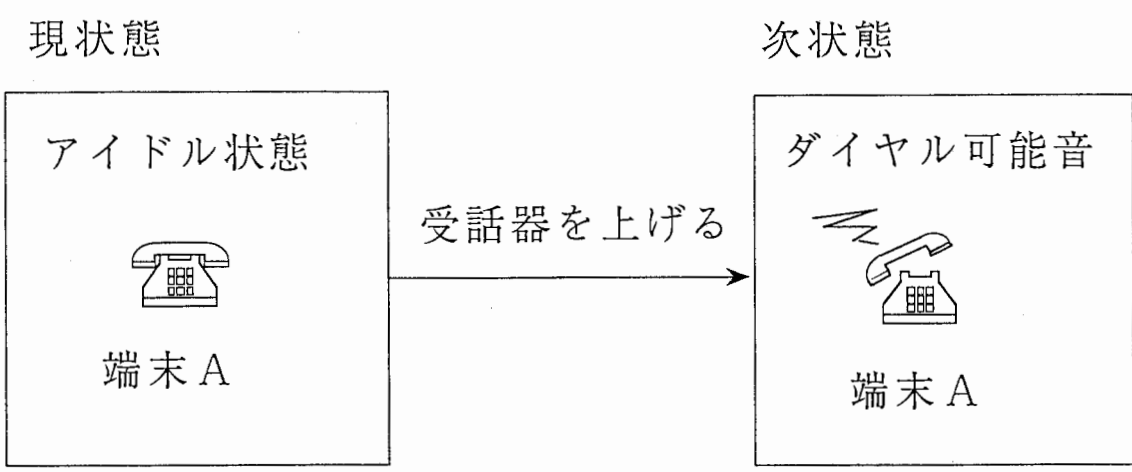


図 2-2 通信サービス動作例 1

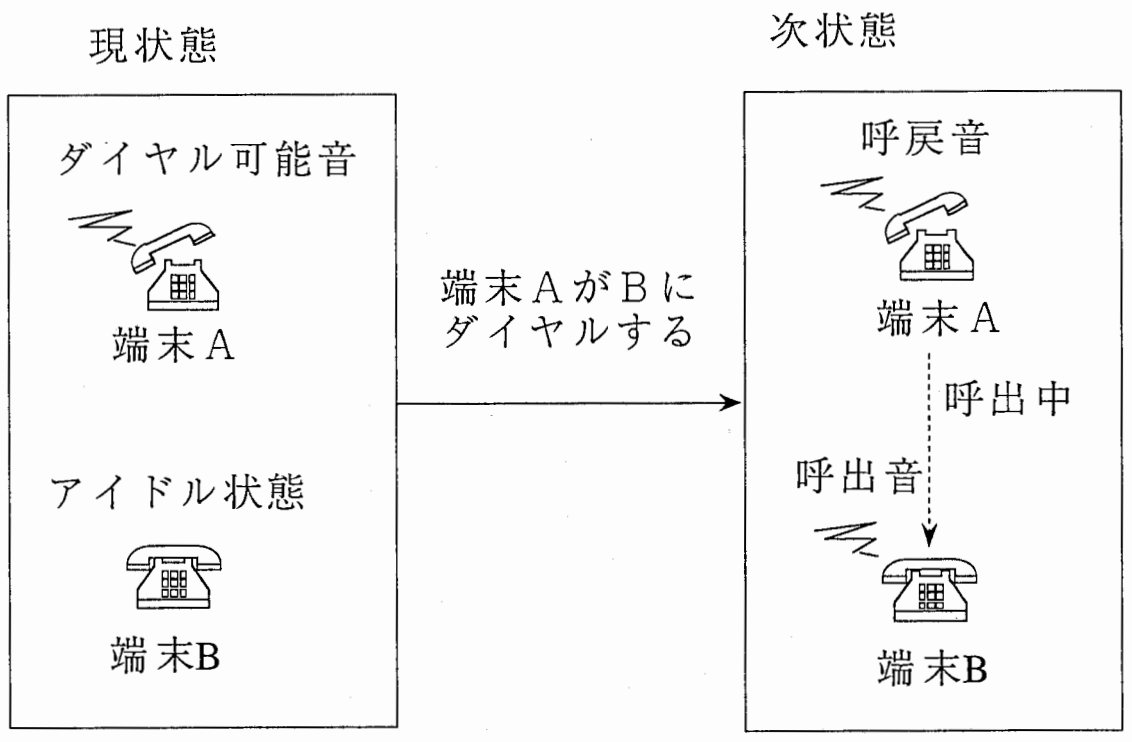


図 2-3 通信サービス動作例 2

2. 2 STR手法の記述方式の概要

状態推移を規定する方法として、基本的には「現状態 イベント： 次状態」という規則記述を用いる。現状態、次状態は状態記述要素の集合で表す。イベントは、状態推移を生起させるものである。状態記述要素、イベント、規則などの概要は以下の定義に示すとおりである。また、禁止状態集合記述、タイマー記述などの記述を用いるが、それぞれの規定は文献 [12] を参照されたい。本稿では、それらの概要の説明に留める。サービスの特徴的な動作は、それらの集合で表す。

[定義 2. 1] (状態記述要素)

状態記述要素 = 状態識別子 (対象物 [, 対象物])

状態記述要素とは、ある対象物の状態 (あるいは属性) を規定するための最小の規定要素をいう。状態記述要素は、状態識別名と左括弧に続く 1 つ以上 (2 つまで) の対象物識別子 (対象物を表す変数) から構成される。複数の変数を記述する際には、間にカンマを記述する。最後は右括弧で閉じる。括弧内の 1 番目の変数は、状態記述要素を保持する対象物を示し、2 番目の変数は 1 番目の対象物と関連する相手の対象物を示す。 □

[定義 2. 2] (対象物状態)

(1) ある対象物の状態は、状態記述要素の 1 番目の変数が同じ複数の状態記述要素の集合を用いて規定する。

(2) 複数の対象物の状態は、状態記述要素の 1 番目の変数が違う複数状態記述要素の集合を用いて規定する。 □

[定義 2. 3] (イベント)

状態推移を引き起こす契機 (トリガー) となる事柄をイベントと呼ぶ。STRの表記上、以下の3種類の表記方式を用いる。

(1) 利用者が行うイベント操作の表記: イベント名 (対象物 [, 対象物]) という形式で記述する。英数字のシンボルを用いて表現されたイベント名、左括弧、イベントが起動された変数 (必要ならばカンマの後にイベントの相手変数を記述する) を記述した後、右括弧で閉じる。第1引数はイベントを生起させる対象物であり、第2引数はイベントを関連させる対象物である。例えば、「ダイヤル (A, B)」は、端末AがBにダイヤルするイベントを表している。

(2) タイムオーバーによるイベントの表記: "timeover" の後に左括弧、タイマーの監視状態集合 (タイマー記述を参照) を記述し、右括弧で閉じる。

(3) 疑似イベント: ある状態になることにより起動がかかる場合、左鍵括弧 ([), 監視状態集合の後に右鍵括弧 (]) で閉じる。 □

[定義 2. 4] (規則記述)

Rule = 規則, I S = 規則の現状態, Event = 規則のイベント, N S = 規則の次状態, とする。規則 Rule は以下のように「現状態」「イベント」「次状態」の三つ組で表す。

I S Event : N S .

規則の現状態 = I S は、状態記述要素の集合で規定する。ただし、状態記述要素の否定条件 (状態記述要素の集合が存在しない) の表記方法は、not [状態記述要素の集合] と表記する。条件判定のみに用いて書き換えの対象としない状態記述要素に対しては cond: を状態記述要素の前に記述する。規則の次状態 = N S は、状態記述要素の集合で規定するが、規則上、現状態の全ての状態記述要素を取り去る場合のみ、明示

的に「empty」と記述する。

□

規則記述例：

規則を説明上区別するために、規則の識別子（例えば、pots-1）を便宜上附加する。

pots-1) idle(A) offhook(A): dial-tone(A).

pots-2) dial-tone(A), idle(B) dial(A, B): ringback(A, B), ringing(B, A).

pots-1)は、端末Aがアイドル「idle(A）」（空状態）のとき、オフフックする「offhook(A)」とダイヤル可能音表示「dial-tone(A)」になる動作を規定する。pots-2)は端末Aがダイヤル可能音表示「dial-tone(A)」かつ端末Bがアイドル「idle(B)」のとき、端末Aが端末Bにダイヤルをする「dial(A, B)」と端末Aが端末Bを呼び出している状態（端末Aには呼戻音「ringback(A, B)」，端末Bには呼出音「ringing(B, A)」が表示されている状態）に推移する動作を規定している。

[規則記述例終]

[定義2. 5]（禁止状態集合記述）

禁止状態集合記述は、規則が適用された結果の次状態が、あらかじめ与えておいた禁止状態集合を含む場合、規則適用を行わず、イベント生起端末の状態を、禁止状態集合記述の指定した状態記述要素に基づいて書き換える機能である。記述の方法を以下に示す。

inhibited primitive set:

{<Primitive-string1>} (<Primitive-string2>)

<Primitive-string1>：第1引数が同じ状態記述要素の集合を表す。

<Primitive-string2>：<Primitive-string1>で使用されている変数とイベント生起対象物を表す「\$」をもちいた状態記述要素の集合を表す。

{ } で括り指定した状態集合が、禁止状態集合を表す。この部分を禁止状態集合記述の条件部 (I F 部) である。後ろの () で指定した状態識別子が、書き換えに用いる状態記述要素である。この部分を禁止状態集合記述の動作部 (T H E N 部) である。 □

禁止状態集合記述例：

inhibitted primitive set:

{m-tcs(A, B), ringing(A, B)} (busy(\$))

これは、端末 A が端末 B からの着信を拒否するためにスクリーニングリストへ登録していた場合 (m-tcs(A, B)) , 端末 B からの呼びだし (ringing(A, B)) は拒否し、イベント生起端末をビジーにすることを表している。

[禁止状態集合記述例終]

[定義 2. 6] (タイマー記述)

状態記述要素または状態記述要素の集合に対して、指定した時間の監視を行い、指定した時間が経過したとき、タイマ処理のためのイベントを起動させる機能である。記述は、以下のように監視状態集合と監視時間を指定する。

timer: <Primitive-string> <Number>"sec."

<Primitive-string> : 状態記述要素の集合 (監視状態集合) を表す。

<Number> : 監視する時間 (秒) を表す。 □

タイマー記述例：

例 1 : 基本電話サービスより

timer: dial-tone(A) 5sec.

端末変数 A に対応する端末において、ダイヤル可能状態「dial-tone(A)」で 5 秒

間ダイヤルを行わなかった場合、ハウラー音に推移する。タイマー記述により、イベント「timer(dial-tone(A))」が生起される。推移規則を以下のように記述しておけば、ダイヤル可能音からハウラー音に推移する動作が表現できる。

```
dial-tone(A) timer(dial-tone(A)): howler(A).
```

例 2 : 転送サービス例より

不在転送サービスのオプションでは、呼出音を何秒かならす間に応答があれば通話になり応答がなければ転送登録先へ転送する機能がある。

記述例を以下に示す。

```
timer: ringing(B, A), m-fuzai(B) 5sec.
```

```
規則 1) ringback(A, B), ringing(B, A) offhook(B): path(A, B), path(B, A).
```

```
規則 2) ringback(A, B), ringing(B, A), cond: m-fuzai(B), cond:m-reg-fuzai(B, C),  
idle(C)
```

```
timer(ringing(B, A), m-fuzai(B) ):
```

```
ringback(A, C), idle(B), ringing(C, A).
```

これは、端末Aが端末Bから呼び出されているときに応答すれば、上記規則1により通話中になり、応答せず5秒間経過したときは、上記2の規則により端末Cへの転送呼び出しになることを表現している。この時点でさらにタイマの起動が必要ならばタイマーの宣言とタイマに関する規則を記述しておけばよい。

[タイマー記述例終]

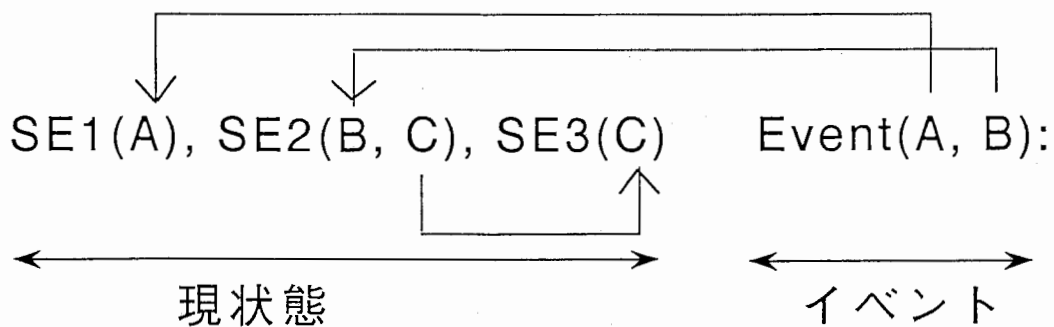


図 2-5 端末変数の通り方

2. 3 STR手法の適用方式の概要

2. 3. 1 ユニフィケーション

規則の条件部（現状態， イベント）が実際に満足するかどうかを判定するための変数のユニフィケーションの操作を以下に示す。また， 規則の現状態とイベントの記述から変数の通り方を図 2-5 に示す。

操作 1：規則のイベント名と実際に生じるイベント名が等しいとき， 変数に実対象物を代入し， イベントが等しいとみなす。

操作 2：実対象物名の状態を探索し， 規則の現状態にある同一端末（第 1 引数と同じ変数）の状態記述要素の有無を調べる。無い場合は， 規則の条件を満たさないので規則は適用されない。探索した状態記述要素の第 2 引数以降に実際の端末名を代入し， 全ての端末変数の関係が一致するまで， 操作 2 を繰り返す。

ここで， 以下のような状況を想定してみよう。

端末 P, Q, R の状態：

端末 P： SE1(P) 端末 Q： SE2(Q, R) 端末 R： SE3(R)

このとき， 端末 P から Q にイベント = 「Event(P, Q)」 が発生した。

上記の状況では， 図 2-5 のイベントと比較すると $A = P$ ， $B = Q$ と代入することに

より、イベントは等しいとみなすことができる（操作1）。端末Pの状態を探索し、「SE1(P)」の存在を確認する。一方、端末Qに対して同様に状態を調べる。「SE2(Q, R)」と「SE2(Q, C)」は端末変数C=Rと代入することで等しくなることが分かる（操作2）。さらに、端末Rに「SE3(R)」が存在していることを確認する（操作2）。この手順により、規則のIF部（現状態、イベント）が満足される。このような操作を行うことにより、規則の適用可能性を調べる。規則の記述では、イベントの端末変数から現状態に記述された端末変数が辿れるように記述することが前提である。詳細なSTR手法の適用規約は、次節に記述する。

2. 3. 2 STR手法の適用規約

規則適用は以下の2つの規約に従って行なう。

[定義2. 7] 規約1（規則の基本適用）

$$(G \supseteq IS) \wedge (GE = E) \rightarrow R$$

G： 任意の端末数から構成される状態，

GE： Gの状態で生起するイベント，

R： 規則，

IS：規則の現状態，

E： 規則のイベント，

NS：規則の次状態，とする。

（注）→の左側は規則適用の条件，→の右側は適用規則を示す。

Gの状態イベントGEが生起したとき、条件「 $(G \supseteq IS) \wedge (GE = E)$ 」を満足したとき、規則「R」は適用可能であり、現状態ISに該当するGの状態記述要素を次状態NSの状態記述要素に書き換える。□

[定義 2. 8] 規約 2 (規則の優先適用)

$$(I S_i \subset I S_j \rightarrow R_j) \vee (I S_i \supset I S_j \rightarrow R_i)$$

R_i, R_j : G において適用可能な規則, (G は定義 2. 7を参照)

$I S_i$: R_i の現状態,

$I S_j$: R_j の現状態, とする.

(注) \rightarrow の左側は規則適用の条件, \rightarrow の右側は適用規則を示す.

規約 1により適用可能な規則が複数あるとき, 規則の現状態の包含関係を見て, 完全に包含している方の規則を包含されている方の規則に優先して適用する.

□

事例を用いた規則適用の説明:

[定義 2. 9] (メモリ関連の状態記述要素)

状態記述要素として, 「m-」で始まるものを端末の状態として表面上には現われないメモリ関連の状態記述要素と呼ぶ. 例えば, サービスの契約を表す状態記述要素がある.

□

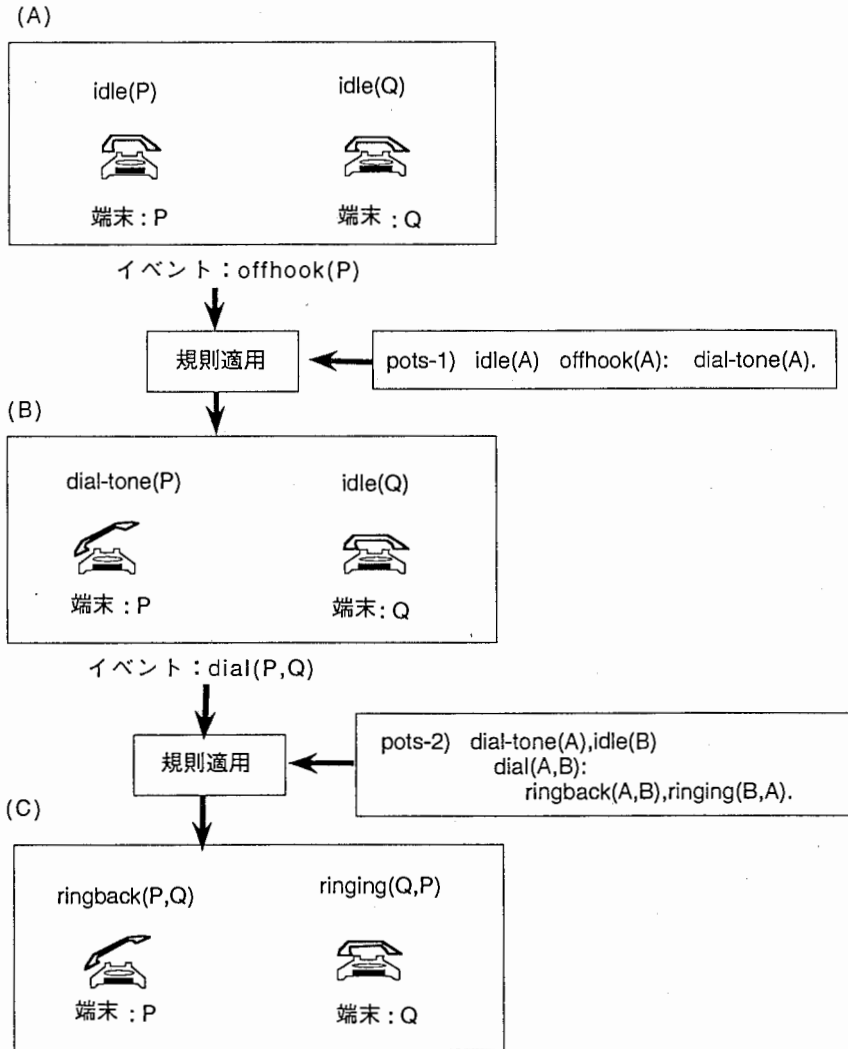


図 2-6 規約 1 (規則の基本適用)

規約 1 (規則の基本適用) :

規則適用規約 1 の例を図 2-6 に示す。図 2-6 の (A) の状態は端末 P, Q が共にアイドル状態である。この状態で端末 P が受話器をあげる操作 (イベント: offhook(P)) をしたとき, 規則「pots-1」の端末変数 $A = P$ とみなすことにより, 規則の現状態「idle(A)」は端末 P の状態「idle(P)」と等しく, 規則のイベント「offhook(A)」も端末 P が起こしたイベント「offhook(P)」と等しいことがわかり, 規則適用され, 端末 P の状態はダイヤル可能音「dial-tone(P)」に推移する (図 2-6 の (B))。同様に (B) の状態において, 端末 P が端末 Q にダイヤルをおこなう操作 (イベント: dial(P,Q)) をした場合, 規則「pots-2」の端末変数 $A = P, B = Q$ とみなすことにより, 規則適用され, 端末 P から Q への呼び出し状態に推移する (図 2-6 の (C))。

規約 2 (規則の優先適用) :

規則適用規約 2 の例を図 2-7 に示す。図 2-7 の (A) の状態は端末 P がダイヤル可能音「dial-tone(P)」, 端末 Q がアイドル状態「idle(P)」かつ端末 R に着信転送を登録している状態「m-cfv(Q,R)」, 端末 R が初期状態である。このとき端末 P が端末 Q に対してダイヤルをおこなう操作をすると, 規約 1 により, 2 つの規則「cfv-1」, 「pots-2」が適用可能であるが, 規約 2 によって現状態を完全に包含している規則「cfv-1」が優先されて, 次状態として, 端末 P から R を呼び出す状態となる (図 2-7 の (B))。 [事例を用いた説明終]

禁止状態集合の適用方法, タイマーの適用法などについては, STR仕様書 (文献 [12]) を参照のこと。

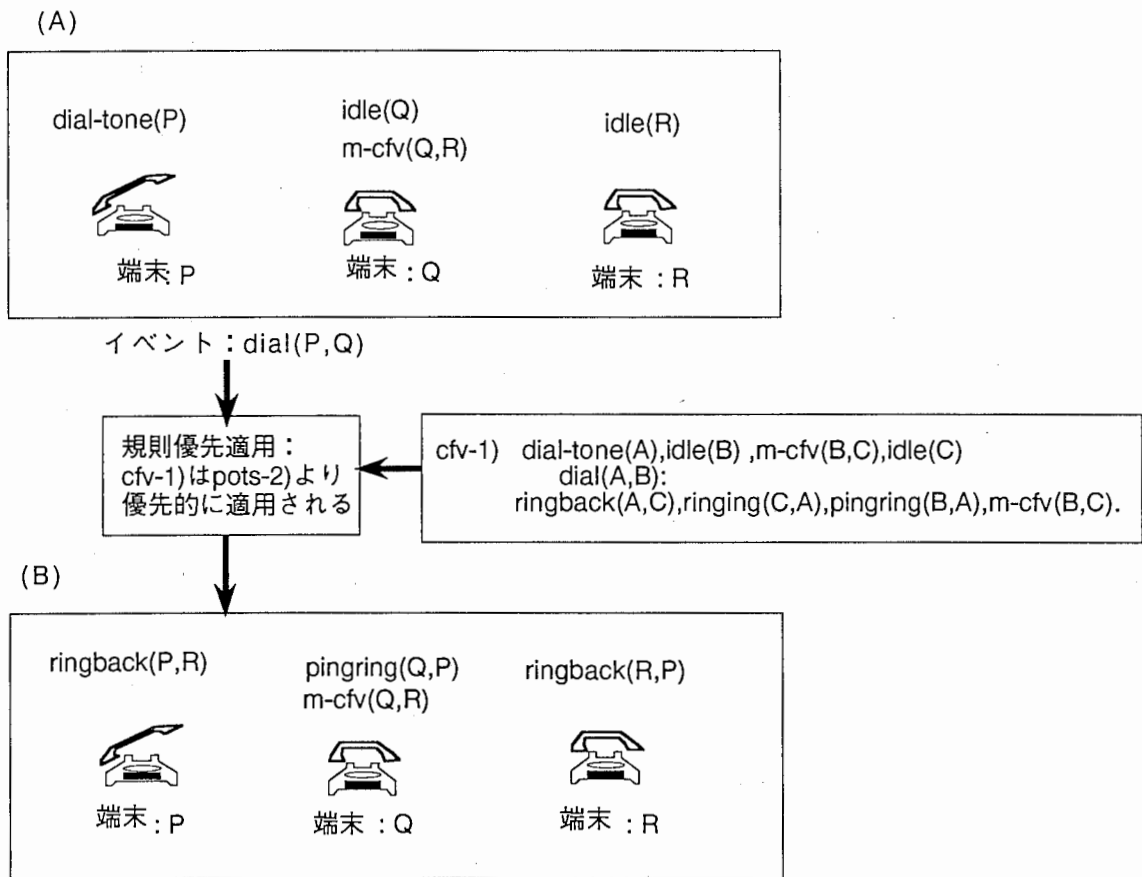


図 2-7 規約 2 (規則の優先適用)

2. 4 サービスの記述性について

2. 4. 1 対象物について

状態記述要素あるいはイベントにもちいる変数（対象物）のタイプの違いは必要に応じて対象物の宣言が可能である。

例えば、

対象物の宣言：

Terminal: A, B, C, D. ; 端末

User: U. ; UPTサービスのユーザ

Groupe: G. ; グループ識別子

などである。

2. 4. 2 状態記述要素について

状態記述要素は対象物、例えば端末上にあらわれる音とかランプなどのユーザから観察できる状態や、端末のサービス加入情報などの端末の属性など、推移の条件となりうるものを記述する。

例えば、音については、状態識別しとしてダイヤル可能音、呼戻音、呼出音などを記述する。ボタンについては、ボタンが点滅、点灯、緑色で点滅などを記述する。音がクラスでありダイヤル可能音などはインスタンスと捉えることが可能である。この状態識別子に変数を添えて規則を記述すると以下のような表現が可能になる。

ダイヤル可能音 (A) , アイドル (B) ダイヤル (A, B) :

呼戻音(A, B), 呼出音(B, A).

よって、状態記述要素＝状態識別子（対象物 [, 対象物] ）の状態識別子の意味としては、どのようなインスタンスを想定して記述してもよい。例えば、ある特定の日

を「x-day」として想定し、その日に限った推移を記述したい場合、

x-day(A), ダイヤル可能音 (A), アイドル (B) ダイヤル (A, B) :

通知(A, B), 通知(B, A).

といった記述も可能である.

状態記述要素の対象物の関係付けは、規則のイベントと現状態の関係付け (図 2-5 参照) を意識して設定する必要がある.

2. 4. 3 イベントについて

イベント記述を行う際、端末で具体的に操作する事象とイベント記述とは必ずしも一致する必要はない. 例えば、着信転送登録の解除として不在転送ボタンを2度押すという操作をして不在転送の解除を要求するイベントになる場合、「不在転送解除 (A)」というイベントを記述すればよい.

タイマーイベント、疑似イベントなどは、状態記述要素の集合であらわし、状態記述要素自体が自由な意味付けができるので、それらの集合も自由な意味付けとなる.

2. 4. 4 規則記述について

規則は、「現状態、イベント、次状態」の組で記述する. 規則の現状態、次状態は状態記述要素の集合であり、状態記述要素は任意の意味として記述できるのでその集合である現状態、次状態は任意の意味として表現できる. イベントも任意の意味として記述できる. よって、規則も任意の意味として記述可能である.

STR手法で扱える範囲は、ネットワークをブラックボックスと見て、対象物の関連する情報をもとに対象物の振る舞いを規定するものである. マルチメディアでの接続要求・接続・切断の手順は規定の対象となるが接続中は通話と同じ解釈であり通話中のエンドーエンド間で行うデータ処理などは通信サービスではなく対象外である.

ネットワーク内部の仕様に関しては規定の対象外である。但し、運用端末としてトラフィックの情報を見るという要求は、状態記述要素として「要求可能状態 (A)」、 「トラフィック表示 (A)」を想定し、イベントとして「トラフィック表示要求 (A)」を想定すれば、以下のように記述することが可能である。

要求可能状態 (A) トラフィック表示要求 (A) : トラフィック表示 (A) .

2. 4. 5 サービスの記述実験について

幾つかの違うタイプのサービスを題材にして、記述の網羅性の検討を試みた。BellcoreのLSSGR (サービス仕様) [20] をベースにした記述実験の報告 [13] では 28 サービスに対して記述でき単体サービスは記述できると報告されている。ここでは、PBXサービス (題材として「NTTビジネスホンE電話機の使い方」 [19]) に着目して、記述性について議論する。電話機の使い方の一覧を図2-9に示す。その数は43個である。まず、標準的な使い方に対する記述例を示す。

外へ電話をかけるとき (発信) の標準的な使い方の説明は表2-1に示すように記述されており、それを規則記述したものを図2-10に示す。説明書には通話になる状態などは明確になっていないので (表2-1) 補足して記述している。このように、記述すると殆どの電話の使い方は記述できる。外との通話中の便利な機能として「通話を録音する方法」が述べられている。これは、録音ジャックを内臓させ、これに録音装置を接続することにより、通話内容を録音することができるという説明である。電話機の操作はなく、ハードの機能のことをいっており、記述の対象外である。これ以外の42個の電話機の使い方は電話操作の使い方の説明であり記述可能である。

ネットワーク内部の仕様に関しては規定の対象外である。但し、運用端末としてトラフィックの情報を見るという要求は、状態記述要素として「要求可能状態 (A)」, 「トラフィック表示 (A)」を想定し、イベントとして「トラフィック表示要求 (A)」を想定すれば、以下のように記述することが可能である。

要求可能状態 (A) トラフィック表示要求 (A) : トラフィック表示 (A) .

2. 4. 5 サービスの記述実験について

幾つかの違うタイプのサービスを題材にして、記述の網羅性の検討を試みた。

BellcoreのLSSGR (サービス仕様) [20] をベースにした記述実験の報告 [13] では 28 サービスに対して記述でき単体サービスは記述できると報告されている。ここでは、PBXサービス (題材として「NTTビジネスホンE電話機の使い方」 [19]) に着目して、記述性について議論する。電話機の使い方の一覧を図2-9に示す。その数は43個である。まず、標準的な使い方に対する記述例を示す。

外へ電話をかけるとき (発信) の標準的な使い方の説明は表2-1に示すように記述されており、それを規則記述したものを図2-10に示す。説明書には通話になる状態などは明確になっていないので (表2-1) 補足して記述している。このように、記述すると殆どの電話の使い方は記述できる。外との通話中の便利な機能として「通話を録音する方法」が述べられている。これは、録音ジャックを内蔵させ、これに録音装置を接続することにより、通話内容を録音することができるという説明である。電話機の操作はなく、ハードの機能のことをいっており、記述の対象外である。これ以外の42個の電話機の使い方は、状態推移に分割され、個々の状態推移は規則として表現可能であった。

1. 外線へ電話をかけるとき（発信）
 - 1) 標準的な使い方
 - 2) 受話器をかけたまま発信するとき
（オンフックダイヤル）
 - 3) 最後にかけた相手にかけ直したいとき
（相手番号自動再送）
 - 4) いまダイヤルした番号を登録して再度かけたいとき（特定番号自動再送）
 - 5) 短縮ダイヤルのご利用方法
 - 6) ワンタッチボタンを利用した短縮ダイヤルの使い方
 - 7) 外線がすべてお話し中のとき（外線予約）
 - 8) 空いている外線を自動的に選ぶとき
（空外線自動選択）
 - 9) ボタンを押したときの確認音が必要なとき
（ボタン押下確認音）
2. 外から電話がかかってきたとき（着信）
 - 1) 標準的な受け方
 - 2) 受話器をあげただけで応答する方法
 - 3) 着信音を一時的に他の電話機に移す方法
（着信転送）
 - 4) 秘書の着信音を一時的に幹部の電話機に移す方法（幹部着信転送）
 - 5) 着信音を一時とめる方法
 - 6) 着信音を大きな音で聞く方法
3. 外との通話を保留するとき
 - 1) 標準的な保留のしかた
 - 2) 人にとられない保留の方法
（自己専用保留）
 - 3) 保留音の曲目を変える方法
4. 外との通話を他の人にまわしたいとき（転送）
 - 1) 標準的な転送のしかた
（転送先と話しをしてから転送するとき）
 - 2) 外線通話を使った転送方法
 - 3) 転送先へ直接とりつぐ方法
5. 外との通話中の便利な機能
 - 1) 他の人を通話に加える方法
（秘話解除）
 - 2) 通話を録音する方法
 - 3) 3分間の通話警告音
（3分間通話タイマー）
 - 4) 回転ダイヤル式回線でプッシュホンのダイヤル信号を出すとき
6. 外線と通話したいとき
 - 1) 標準的な通話のしかた
 - 2) 呼出音の音声・信号切替え
 - 3) 受話器をかけたまま応答する方法
（外線ハンズフリー応答）
 - 4) ワンタッチで呼出す方法
 - 5) お話し中でも呼び出す方法
（外線話中呼出）
 - 6) 外線代表呼出
 - 7) 外線ステップコール
 - 8) 外線キャンプオン
 - 9) 外線自動キャンプオン
 - 10) 外線不在転送
 - 11) 伝言（メッセージウェイティング）
 - 12) 外線自動応答
 - 13) 外線代理応答
 - 14) 外線アッドオン
 - 15) 二台の幹部電話を同時に呼び出すとき
 - 16) 外線通話の保留転送
7. 他の電話機の呼出し方法
 - 1) 他の電話機を一斉に呼び出す方法
 - 2) 構内放送をするとき

図 2-9 P B X サービス一覧

表 2-1 外へ電話をかけるときの標準的な使い方

手順	操作	端末状態
1	外線ランプがついていない外線ボタンを押します	
2	受話器をあげます	押された外線ランプが時々消える点滅となります
3	相手の電話番号をダイヤルします	ディスプレイに相手番号が表示されます (表示器付電話機の場合)
4	お話しが終わりましたら受話器をかけます	ディスプレイに相手番号が表示されます

手順 1 : idle(P) req-gaisen(P): idle(P), m-gaisen(P).

端末Pがアイドル (idle(P)) のとき外線使用要求したら (req-gaisen(P)) ,
外線へかけることが可能 (m-gaisen(P)) になる

手順 2 : idle(P), m-gaisen(P) offhook(P): lp-gaisen-on1(P), m-gaisen(P).

受話器をあげたら (offhook(P)) , 外線ランプがときどき消える点滅 (lp-gaisen-on1(P)) となる

手順 3 : lp-gaisen-on1(P), m-gaisen(P), idle(A) dial(P, A):

ringback(P, A), ringing(A, P), display(P, A), lp-gaisen-on1(P), m-gaisen(P)

相手の電話番号をダイヤルすると (dial(P, A)) , ディスプレイに相手番号が表示され (display(P, A))
(以下, 補足) 相手がアイドルであれば呼び出しとなる (ringback(P, A), ringing(A, P))

補足 : ringback(P, A), ringing(A, P), display(P, A), lp-gaisen-on1(P), m-gaisen(P)

offhook(A):

path(P, A), path(A, P), display(P, A), lp-gaisen-on1(P), m-gaisen(P) .

相手が応答したら (offhook(P)) , 通話 (path(P, A), path(A, P)) になる

手順 4 : path(P, A), path(A, P), display(P, A), lp-gaisen-on1(P), m-gaisen(P)

onhook(A): idle(P), busy(A).

話しがおわり受話器をかける (onhook(A)) と, 外線ランプは消え,

(以下, 補足) 外線利用の権利もなくなり, アイドルになり, 相手にはビジー音がでる

図 2-10 規則記述例

2. 5 考察

(1) 新規サービスの差分記述

通信サービスを効率よく設計するためには、既存仕様を再利用することが重要な要素である。サービスの個々の動作を規則形式で表現することにより、既存仕様の再利用は規則の再利用の問題として扱うことが可能となる。既存の基本電話の規則を再利用することで、本来記述すべき仕様の40%程度の記述量ですむ。つまり、サービスの特徴的な動作を記述し、新規サービス仕様と利用するサービス仕様とを組み合わせ、仕様を合成し、利用したサービス仕様の状態推移を取り除けば、目的のサービス仕様が合成でき、新規サービスの設計の効率化が実現できる。

(2) 将来の電話サービスとSTR手法との関係

将来のサービスに対する予測は難しく、かつ予測が正しいかどうかの判定すら困難であるが、過去の基本電話サービスから始まり今日の電話サービスの多様化への傾向から類推して議論することは可能であろう。まず、予測が難しいという例として9年前前に「新しい電話サービスのすべて」という文献[18]が発表されたが、今日ではすでに文献よりも多くの電話サービスが議論されている。この文献では、1)フリーダイヤル、2)トリオホン(3者通話)、3)でんわ会議、4)転送電話、5)音声蓄積、の5サービスを代表的という意味であろうが新しい電話サービスとして紹介している。フリーダイヤルサービスは着信側が料金を負担するサービスである。STR手法ではフリーダイヤルの端末操作は記述できる。料金に関する機能は課金仕様で表せばよい[17]。他の機能は現状のSTRの機能で記述可能である。

電話サービスは基本電話サービス(POTS)から始まり多様化する傾向にある。多様化の方向としては、1)接続端末を増やす方向(会議電話、三者通話など)、2)接

続条件に制約を付ける方向（着信／発信スクリーニングなど）， 3）着信先を変更する方向（着信転送など）， 4）情報表示の方向（発信番号表示など）， 5）イベント入力手順の簡易化・省略化の方向（短縮ダイヤルなど）， 6）端末以外の対象物を増やす方向（UPTサービスのユーザなど）などがあり，これらのサービスは現状のSTR手法で記述可能である．今後も上記の傾向でサービスの多様化が進むならば，場合によっては拡張の必要は生じるかもしれないが基本的にはSTR手法で対応可能と考える．理由としては，規則は「現状態，イベント，次状態」の組で記述する．規則の現状態および次状態は状態記述要素の集合であり（複数の端末の状態の記述が可能），状態記述要素は任意の意味として記述できるのでその集合である現状態，次状態は任意の意味として表現できる．複数の端末の接続状態は状態記述要素の第1引数の違う複数の状態記述要素の組み合わせで表現できるし，接続条件の組み合わせが複雑になっても状態記述要素の組み合わせで対応できる．イベントも任意の意味として記述できる．よって，規則も任意の意味として状態推移が記述可能である．また，端末以外の対象物が必要な場合は，随時，対象物を宣言して使うことができる．

（3）記述の省略化の検討

重箱のすみをつつくようなささいな問題はあるかもしれないが，現状では，一斉呼び出しにおける記述の省略化を可能にする程度の問題があるくらいと考えている．理由は，前述の項目（2）を参照して頂きたい．例えば，一斉呼び出しサービスではあらかじめ登録しておいた複数の呼び出し先を同時に呼び出すサービスであるが，リソースの問題等があり呼び出し先の登録個数は決められている．検討したPBXサービスの一斉呼び出しでは6個までであった．この程度ならば，現状のSTRの記述機能で十分記述可能である．呼びだしの端末数が増えると記述が面倒なので，記述の省略化の例を以下に示す．

ダイヤル可能 (A) , {cond:呼登録 (A, B1) , アイドル (B1) } , . . . , {cond:
呼登録 (A, Bn) , アイドル (Bn) } 一斉呼び (A) :

{呼戻音 (A, B1) , 呼出音 (B1, A) } , . . . , {呼戻音 (A, Bn) , 呼出音
(Bn, A) } .

|| で括った状態集合の組み合わせが, 1 から n まで繰り返されることを示してい
る. 省略を 「, . . . , 」 で表せば, 記述量が削減できる.

(4) 今後の課題

STR手法はひとつの動作(状態推移)を規則で表し, その集合でサービスを規定す
るものであるが, サービスの規定という観点では, もっと抽象的で概念的な捉え方が
できるのかもしれない. サービス要求レベルでは, あんなことがしたい, こんなこと
ができれば, といったかなり抽象的な要求がキーになる. このような, 抽象的な要求
を形式的に表現できるかどうか, また個々の規則との関係をどう扱うのかといったこ
ともサービス設計の効率化を考える上で検討すべき要因であろう.

3. 通信サービス競合（非決定性）検出手法

3. 1 通信サービス競合（非決定性）

状態推移モデル上の問題として、同一イベントに対して複数の推移先が生じる非決定性の問題は、規則適用の観点からみると、複数の動作規定が適用可能になり一意に決めることのできない規則競合の問題として扱うことができる。本章では、規則競合検出方式の概要と考察について述べる。

提案する競合検出方法は、2規則の始状態から2規則が同時適用される可能性のある状況（競合候補）を作成し、任意端末状態の集合を用いて、その状況（競合候補）が実際に在りえる状況かどうかを判定する。在りえない場合は、競合候補から除外する。また、規則適用規約2（定義2.8参照）を考慮して、2規則が優先関係にある場合は競合しないと判定する。競合候補に適用可能な他規則があり、他規則が2規則のいずれか、あるいは、両方に優先適用される場合は競合候補から除外する。つまり、本手法の特徴は、2規則が明らかに競合しない条件を判定基準に用い、その判定基準に該当する場合、競合候補から除外する方法である。この判定を通過した競合候補が在る場合、その競合候補の2規則は競合する可能性があるとして判定する。検出された2規則の競合する状況を画面に表示し、設計者の検証を促す。

3. 2 非決定性の例

STR手法では、一つの推移は一つの規則で記述される。規則の構文は、「現状態 イベント： 次状態」で表される。イベントが生起したとき、現状態を満足する状況があるとき、規則は適用される。つまり、競合検出は適用可能な規則が複数有り、実行させる規則がひとつに決めることのできない規則競合検出の問題として扱うこと

話中着信 (CW:Call Waiting)

```
dial-tone(A),m-cw(B),path(B,C)
dial(A,B):
ringback(A,B),cw-ringing(B,A),m-cw(B),path(B,C).
```

着信転送 (CFV :Call Forwarding Variable) :

```
dial-tone(A),path(B,C),m-cfv(B,D),idle(D)
dial(A,B):
ringback(A,D),ringing(D,A),path(B,C),m-cfv(B,D).
```

図 3 - 1 規則記述例

ができる。

S T R 手法で記述された、話中着信動作と着信転送動作の記述例を図 3 - 1 に示す。図 3 - 1 の 2 つの規則は、以下の動作を既定している。

話中着信動作 (C W) :

端末Aがダイヤル可能状態 (dial-tone(A)) , かつ端末Bが話中着信に加入している (m-cw(B)) , 話中 (path(B, C)) のとき、端末Aが端末Bにダイヤル (dial(A, B)) を行くと、端末Bへの話中着信 (cw-ringing(B, A), ringback(A, B)) となる。

着信転送動作 (C F V) :

端末Aがダイヤル可能状態 (dial-tone(A)) , かつ端末Bが着信転送を登録 (m-cfv(B, D)) しており通話中 (path(B,C)) 状態、かつ着信転送登録先端末Dがアイドル (idle(D)) のとき、端末Aが端末Bにダイヤル (dial(A, B)) を行くと、着信転送登録先へ着信 (ringing(D, A), ringback(A, D)) となる。

図 3 - 1 の 2 規則が同時に適用できる状況として図 3 - 2 の (A) に示す状況が想定される。つまり、端末Pはダイヤル可能状態 (dial-tone(P)) , かつ端末Qが端末Rと話中 (path(Q,R)) , 話中着信に加入し (m-cw(Q)) , 着信転送を端末Sに対して登録

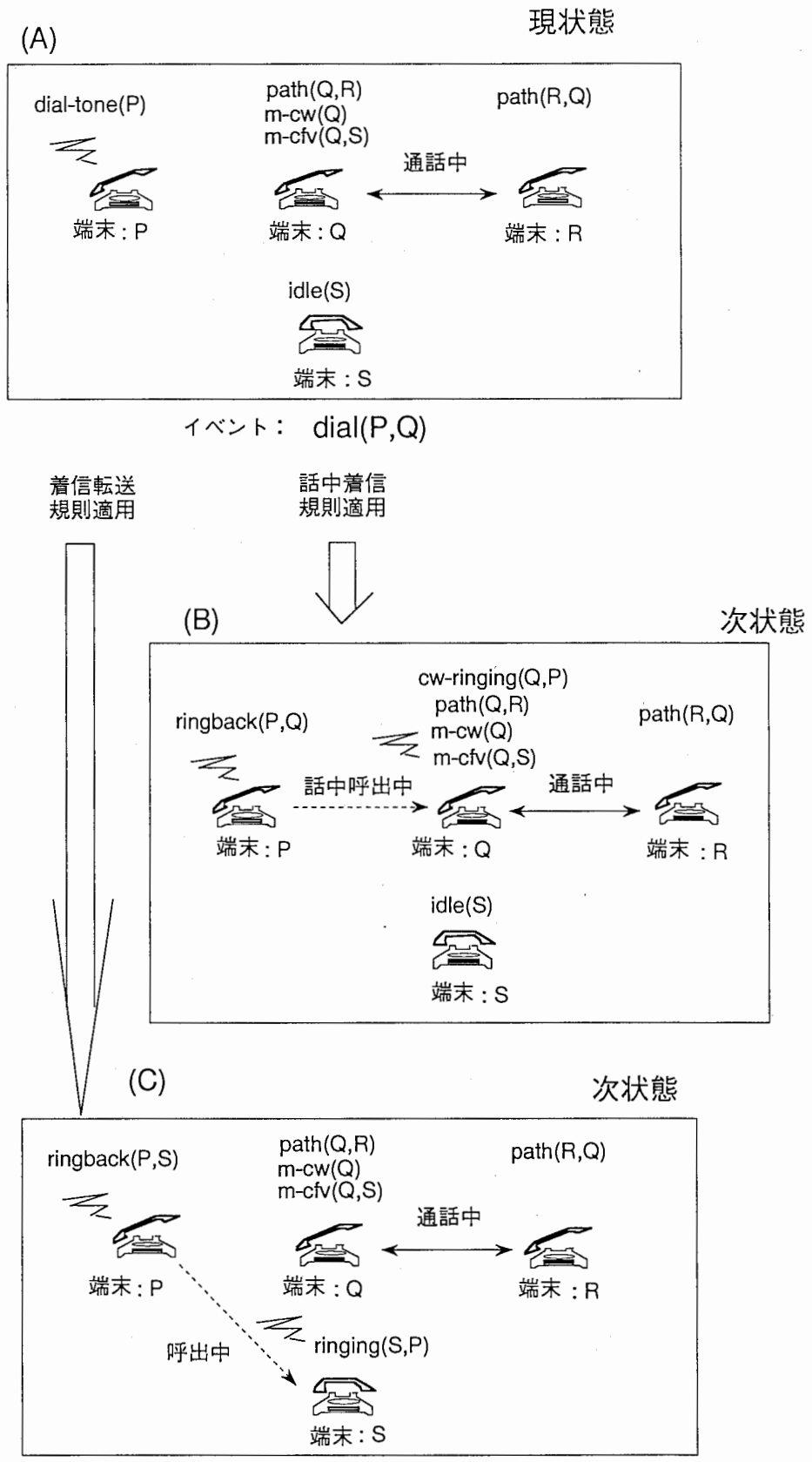


図 3-2 規則競合例

(m-cfv(Q,S)) しており，かつ端末 S が初期状態 (idle(S)) である．端末 P はダイヤル可能状態 (dial-tone(P)) である．

この状況で，端末 P から端末 Q へダイヤルしたとき (イベント: dial(P,Q))，図 3-1 の 2 つの規則の条件 (始状態) は両方とも満足され，図 3-2 に示す 2 つの異なる状態推移が起動されることになり，動作規定を一意に決定できない動作矛盾が発生する．

3. 3 通信サービス競合 (非決定性) 検出の基本概念

2 つの規則が同時に適用可能検査の基本的な考え方を以下の 2 規則を用いて説明する．

規則 1) $p1(A, B)$ $ev1(A, B):$ $p2(A, B)$.

規則 2) $p3(A, B), p4(B, A)$ $ev1(A, B):$ $p5(A, B), p6(B, A)$.

規則 1) の始状態は「 $p1(A, B)$ 」である．これは端末 A が端末 B に対して「 $p1$ 」の状態であることを示す．このとき，端末 A が端末 B に対して $ev1$ のイベントを行うと，次状態は「 $p2(A, B)$ 」になる．規則 2) の始状態は「 $p3(A, B), p4(B, A)$ 」である．これは，端末 A が端末 B に対して「 $p3$ 」の状態であり，端末 B が端末 A に対して「 $p4$ 」の状態であることを示す．このとき，端末 A が端末 B に対して $ev1$ のイベントを行うと，次状態は「 $p5(A, B), p6(B, A)$ 」になる．

端末 A, B ごとに，この 2 つの規則の現状態の状態記述要素の和集合を作成すると以下のようなになる．

端末 A = { $p1(A, B), p3(A, B)$ }

端末 B = { $p4(B, A)$ }

この状況で、端末Aが端末Bに対してev1のイベントを行うと、2つの規則は同時適用可能になり競合が発生する。つまり、2つの規則が同時適用可能かどうかの判定は、2つの規則の始状態から端末毎の状態記述要素の和集合をとった状況が存在可能かどうかの判定問題として扱うことができる。

競合しない例を、以下に示す2つの規則を用いて示す。

規則1) dial-tone(A), idle(B) dial(A, B):

ringback(A, B), ringing(B, A).

規則2) dial-tone(A), not[idle(B)] dial(A, B):

busy-dial(A,B).

端末A, Bごとに、この2つの規則の始状態の状態記述要素の和集合を作成すると以下のようになる。

端末A = {dial-tone(A)}

端末B = {idle(B), not[idle(B)] }

端末Bの状態はアイドル状態 (idle(B)) とアイドルを含まない状態 (not[idle(B)]) の組み合わせになっており、この組み合わせはありえない状態であることがわかる。よって、この2規則は競合しないと判定される。

競合検出の手順の概要を以下に示す。

手順1：実状態集合の作成

任意の端末の取りえる状態を数え上げる。

[定義 3. 1] 実状態集合

任意の端末の取りえる状態を実状態集合と呼ぶ。

□

手順 2：競合候補検出

(1) 2 規則抽出：

同一イベントの 2 規則の組み合わせを規則集合から抽出する。

(2) 2 規則間の優先適用検査：

規則適用規約 2（第 2 章；定義 2. 8 参照）に該当しているかどうかの検査を行う。該当している場合は、競合しないと判定する。

(3) 競合候補作成：

2 規則の始状態から端末変数の可能な組み合わせを作成し、それぞれの端末変数の組み合わせごとに各端末毎の状態記述要素の和集合を作成する。これを、競合候補と呼ぶ。

2 規則から作成された競合候補は複数存在する場合がある。以下に示す検査（4）～（5）は競合しない条件に該当した競合候補を除外するものであり、その結果、競合候補がひとつでも残った場合は、2 規則は競合する可能性があるとして判定する。

(4) 競合候補の存在可能性検査：

競合候補の全端末状態が実状態集合に存在しているかどうか調べる。競合候補の全端末状態が実状態集合に存在していない場合は競合候補は競合しないと判定し競合候補から除外する。これは、1) 規則が適用可能なためには、規則の現状態が満足される状態が存在する必要がある、2) もし端末ごとの状態が存在しなければ規則は適用できない、という規則適用の特性を用いて判定するものである。

(5) 他規則優先適用検査：

競合候補を満足する状況に対して、2規則のいずれかに優先して適用される他の規則（イベントは2規則と等しく、2規則のいずれかの始状態を完全に包含している規則）がある場合、あるいは、2規則の両方に優先して適用される他規則がある場合は、その状況では2規則は競合しないと判定し、競合候補から除外する。

提案した非決定性検出方式の詳細な説明は、文献 [14, 15] を参照して頂きたい。

3. 4 考察

提案した手法は、イベントの同じ2規則の現状態から競合候補を作成し、あらかじめ列挙した任意端末の状態と比較して競合かどうかを判定する手法である。状態の獲得の方法として、任意端末の状態を列挙する方法と関連端末の状態を列挙する方法を検討した。まず、それぞれの利点と欠点について述べ、使い易さを考えた融合案を示す。

(A) 任意端末の列挙+2規則競合検出 [14, 15]

利点1) 5つのサービスの組み合わせによる実験結果（表3-1）では、機械的に検討対象を絞り込む効果が大きく設計支援に有効であることが確認できた。例えば、799の規則の組み合わせにおいて検討対象の規則の組み合わせは5組である。

利点2) 2規則から作成した競合候補は、変数のすべての可能性のある組み合わせを網羅しているため、検出した競合可能性候補は検討範囲を漏れなく網羅している。設計者は、絞り込まれた範囲に対して競合の可能性を検討すればよい。

利点3) 端末の状態は、任意端末ひとつの状態を対象にしているため、端末数を設定したりする必要もなければ、処理時間も比較的早い。

欠点1) 任意端末の状態だけに着目し他端末の状態を仮定しているために、任意端末と他端末の間に実際には存在しない状態を規則の現状態の条件として成立するとみ

表 3-1 非決定性検出実験結果

サービスの 組み合わせ	2 規則の組み合 わせ比較	競合する 2 規則の 組み合わせ数	抽出した 2 規則の 組み合わせ数
POTS+CW		0	146
POTS+CW+CFV		2	259
POTS+CW+CFV +3WC		2	726
POTS+CW+CFV +3WC+CCBS		5	799

なすことがあり、規則適用の結果、実際には存在しえない状態を列挙することがある。そのため、検出した2規則の競合可能性のある組み合わせの中に、調べてみると実際には生じ得ない組み合わせが含まれることがある。上記、5つのサービスの組み合わせにおいて、5組の2規則の競合候補があったが、その中の1組が競合しない組み合わせであった。つまり、規則適用時において他端末状態の同時存在性の判定を行いつつ状態の列挙を行えば結果として、競合判定の精度は向上する。

(B) 複数端末状態の列挙+2規則競合検出

検出精度を向上させるため、任意端末の状態と他端末状態との同時存在可能性判定を行う方法を検討した。詳細は、文献 [16] を参照のこと。規則適用して獲得する状態を1) 単独端末状態、2) 複数端末状態に分ける。前者は、状態記述要素の引数がひとつで同じ端末の状態である。例えば、「dialtone(A)」や「dialtone(A), m-cw(A)」である。後者は、状態記述要素の第2引数が指し示している端末間の状態で

ある。例えば、「path(A, B), path(B, A)」などである。これらの、規則の適用時に規則の現状態の同時存在性可能性の判定を行い、状態獲得を行う。同時存在性判定の特徴は、一般に、状態は構成端末が違えば別々な端末間で独立に存在しえるという性質を利用していることである。

簡単な例を以下に示す。

規則として

ダイヤル可能音 (A) , 通話 (B, C) , 話中着信可能 (B)

ダイヤル (A, B) :

呼戻音 (A, B) , 話中着信音 (B, A) , 通話 (B, C) , 話中着信可能 (B) .

があり、獲得状態として、

単独端末状態：ダイヤル可能音 (A)

複数端末間状態：通話 (B, C) , 話中着信可能 (B) , 通話 (C, B)

があるでしょう。

このとき、端末Aの状態と端末B, C間の状態は構成端末が違うので、独立に成立する。規則は適用可能と判定し、規則適用の結果、以下の複数端末間状態を獲得する。

複数端末間状態：呼戻音 (A, B) , 話中着信音 (B, A) , 通話 (B, C) ,

話中着信可能 (B) , 通話 (C, B)

利点1) この方法により、ひとつの端末に着目した状態獲得の方法では、存在し得ない状態が含まれていたが、それらの状態は取り除くことが可能となり、その結果、競合検出の精度が向上した。実際に表3-1の5組の2規則の競合候補の中の1組が競合しない組み合わせであったが、この手法でその組み合わせは取り除かれてた。つまり、規則適用時において他端末状態の同時存在性の判定を行いつつ状態の列挙を行

えば結果として、競合判定の精度は向上する。

欠点1) 複数端末間で取り得る状態を列挙するため、処理時間が多くかかる。

(C) 両者の長所の融合案について

2規則の検査ステップに要する処理時間はわずかであるが、端末状態の獲得ステップが問題である。任意端末に着目し他端末状態を成立するものと仮定して状態を列挙し2規則競合を判定する方法を処理1とし、任意端末の状態と他端末状態との同時存在可能性判定を行い状態を列挙し2規則競合を判定する方法を処理2とする。処理1で検出された競合状況をもとに分かる範囲で競合を解消し、疑わしい競合状況に対しては、処理2を用いて確認するという運用を行えば使い勝手は向上すると考える。そのときに、処理1において、各端末の加入しているサービスの組み合わせがわかるので、処理2の環境設定を拡張し各端末毎にサービスの組み合わせを設定できるようにしておけば処理時間は大幅に削減されると予測できる。なぜならば、現在は、複数端末状態作成時に端末は全て等しくサービスを利用可能と想定しているため、サービス数と端末数の組み合わせ分の端末状態が生成される。検査すべき端末とサービスの組み合わせが固定できれば、処理時間は大幅に削減される。例えば、図3-2のような競合は、端末Qが話中着信サービスと着信転送サービスに加入しているときに起り、他の端末はPOTSサービスでよい。このような状況を確認したいとき、端末とサービスの組み合わせを設定して処理2を起動すればよい。また、獲得した既存サービスの状態を格納しておき、新規サービスとの組み合わせを行えば、既存サービスの状態を新たに作成しなくてもよい分、処理時間は幾分かは短縮される。このような、運用面から処理の組み合わせを行えば、処理時間の短縮化が可能と考える。

イベントの同じ2規則は競合する可能性があるが、規則の現状態はその規則が適用されるとき条件を記述しているだけで必ずしも規則が適用されるとき全ての状態

を記述しているわけではない（定義2.7参照）。イベントの同じ2規則の現状態の組み合わせで競合する可能性のある状況を推測できる場合もあるが、個々の端末の実際の状況がどういうものを把握することは難しい。つまり、どの端末にどういうサービスを設定したらよいかは、任意端末状態の列挙を行った上での判定状況を参考にした方がよい。

4. おわりに

4. 1 STR手法について

ネットワーク内部をブラックボックスとして捉え、外部から観測可能な対象物（主に端末）の状態推移を「現状態 イベント： 次状態」の規則形式で記述し、それらの規則の集合としてサービス仕様を規定する手法を検討してきた。本稿では、記述および適用方法の概要を紹介し、記述性について議論した。現状態および次状態は状態記述要素の集合として規定され、各状態記述要素は任意の意味として記述できるのでその組み合わせである現状態、次状態は任意の意味として記述できる。イベントも任意の意味として記述でき、現状態、イベント、および次状態の組み合わせである規則も自由な意味付けが可能である。電話サービスは基本電話サービス（POTS）から始まり多様化する傾向にある。多様化の方向としては、1）接続端末を増やす方向（会議電話、三者通話など）、2）接続条件に制約を付ける方向（着信／発信スクリーニングなど）、3）着信先を変更する方向（着信転送など）、4）情報表示の方向（発信番号表示など）、5）端末以外の対象物を増やす方向（UPTサービスのユーザなど）、6）イベント入力手順の簡易化・省略化の方向（短縮ダイヤルなど）などがあり、これらのサービスは現状のSTR手法で記述可能である。今後も上記の傾向でサービスの多様化が進むならば、STR手法で対応可能と考える。

4. 2 通信サービス競合（非決定性）検出手法について

通信サービス相互作用に含まれる矛盾検出の困難さの主な要因は、設計者がサービスを組み合わせた時に生じるであろう状態および推移を把握・予測することが困難なことに起因する。この点を解決する枠組みの確立が、サービス相互作用設計支援を行う上での鍵となる。

形式的仕様記述方式としては、SDL [3] , Petri Net [4, 5] , LOTOS [6, 7] , 等があるが、SDL [3] , Petri Net [4, 5] は、状態推移を明示的に表現するけれども、状態を合成する概念がなく、サービス相互作用の矛盾を検出する機能には限界がある。LOTOS [6, 7] は、サービス動作をイベント系列の集合で規定する手法であるが、状態を明示的に表現できないため、サービス合成における矛盾する状態の解析を行うには自ずと限界がある [8] .

STR手法を用いて通信サービスを規則の集合として捉え、各サービスの規則を適用することによりサービス合成における状態を列挙することが可能になり、矛盾検出のアルゴリズムと組み合わせることにより検証支援が可能となる。サービス競合（非決定性）検出では、イベントの同じ2規則の競合の可能性を機械的に絞り込み設計者に提示する手法である。提案手法は絞り込み効果が大きく検証支援に有効であった。基本的なアルゴリズムの検討を中心に行ってきたが、3章の考察で述べたような処理時間も考慮した運用を考えれば、実用面からも有用であると考ええる。

STR手法を前提としたサービス競合（非決定性）検出手法の限界としては、ネットワーク内部の処理はブラックボックスと想定した枠組みを前提としているので、ネットワーク内部でのシステムの機能分割に起因する処理の競合や矛盾、あるいはプロトコル仕様に依存した信号の送受信のタイミングなどに起因する競合や矛盾は扱うことができない。それらは次工程（詳細化レベル）での検証の問題と考えている。

参考文献

- [1] Duran J. M and Visser J. :International Standards for Intelligent Networks, IEEE Communication Magazine, Vol.30, No. 2, pp. 34-42 (1992-02)
- [2] Collet P. and Kung R. : The Intelligent Networ in France, IEEE Communication Magazine, Vol. 30, No. 2, pp. 82-89 (1992-02)
- [3] CCITT Recomendations Z.100 to Z.104 1985
- [4] J. L. Peterson: Petri Net Theory and Modeling of Systems, Prentice-Hall (1981)
- [5] J. L. ピータースン著, 市川惇信/小林重信訳: ペトリネット入門;
情報システムのモデル化, 共立出版 (1984)
- [6] ISO8807: Information processing systems - open systems interconnection -
LOTOS - a formal description technique based on the temporal ordering of
observational behavior, ISO publication (1988)
- [7]高橋薫, 神長裕明, 白鳥則郎: LOTOS 言語の特質と処理系の現状と動向,
情報処理, Vol. 31, No. 1, pp. 35-46 (1990-01)
- [8]長谷川晴朗: ペトリネットを用いた通信サービス仕様設計支援, 信学論文誌
(B-I), Vol. J74-B-I, No. 6, pp. 445 - 455 (1991)
- [9]原田良雄, 平川豊, 竹中豊文, 門田充弘: サービス仕様の自動生成に関する
考察, 情処 39 回全国大会, 5S-5 (1989)
- [10] Hirakawa Y., Harada Y., Takenaka T. :Behavior Description for A System
which Consist of An Infinite Number of Processes, 1990 BILKENT International
Conference on New Trends In Communication, Control, and Signal Processing,
pp.59 - 68, (1990-07)

- [11] Hirakawa Y., Takenaka T. : Telecommunication Service Description Using State Transition Rules, in Proc. Sixth Int. Workshop on Software Specification and Design, pp. 140 - 147 (1991-10)
- [12] STR (State Transition Rule) 記述仕様書, 平成 6 年 3 月, STR-1.1.1
- [13] 高見一正, 太田理 : 状態遷移規則による通信サービス記述性の評価, 1993年信学春全国大会, B-573 (1993-03)
- [14] 原田良雄, 平川豊, 竹中豊文 : 通信サービス競合検出システムの試作と評価, 信学技法, Vol. 92, No. 75, SSE92-8, pp. 13-18 (1992-05)
- [15] Harada Y., Hirakawa Y., Takenaka T., Terashima N. : A Conflict Detection Support Method for Telecommunication Service Descriptions, 信学論文誌, Vol. E75-B, No. 10, pp. 986-997 (1992-10)
- [16] Harada Y., Ohta T., Terashima N. : A Method of Automatic State Acquisition for Conflict Detection of Telecommunication Service Descriptions, Int. Journal on EAAI (Engineering Application of Artificial Intelligence) (to be published)
- [17] 原田良雄, 高見一正, 太田理, 寺島信義 : 通信サービス課金処理競合検出方式, 情処論文誌, Vol. 34, No. 5, pp. 1064-1073 (1994-05)
- [18] 大島一征, 小森稔彦 : 新しい電話サービスのすべて, ビジネスコミュニケーション, Vol. 23, No. 3, pp. 39 - 53 (1986-03)
- [19] NTTビジネスホンE操作マニュアル
- [20] Bell Communication Research: Technical Reference, Issue 2 (1987-06)
- [21] 原田良雄 : 通信サービス設計および検証における形式的支援に関する研究, ATR Technical Report, TR-C-0106 (1995-03)