移動体計算環境におけるアクティブデータベースの安全性解析手法

村瀬 亨[†,††] 塚本 昌彦[†††] 西尾 章治郎[†††]

無線通信による移動体計算環境におけるデータ統合を実現するうえで，移動や省電源切断動作などをイベントとしてとらえ，アクティブデータベースにおけるECAルールを適用することが有効であることをこれまでに筆者らは示してきた．この新しい環境では，搭載される複数システム間の通信機能も強化して計算機間の連携動作も可能にしている．作成されたECAルールには複数システム間で無限ループに陥らずに停止するかどうかの安全性判定が必要である．しかし，ECAルールを移動体環境に適用する場合は，ホスト計算機の移動によって実行シーケンスが変動するため，その実行シーケンスが停止するかどうかの判定が困難である．本論文では，このようなシステム動作の安全性について，特にルールの停止性について，移動による影響を含めて判定するための方法について考察する．その中で，イベント発生ホストとトリガーループの関係を σ マージグラフと呼ぶグラフとして表現することで効率良く安全性が判定できることを示す．

A Safety Analysis Method of Active Databases in Mobile Computing Environments

Toru Murase,[†,††] Masahiko Tsukamoto[†††] and Shojiro Nishio[†††]

For realizing the data integration in mobile computing environments, we showed that the notion of ECA rule in active databases is effective to describe events occurring in these environments such as moving of computers, disconnect operation for energy-saving, and so on. Though the ECA rule is useful, it has inevitable difficulty for guaranteeing that the rules can work without falling into infinite loop among hosts. Furthermore, if the system is applied for mobile computing environments, the complexity for predicting termination will be increased due to the facts that (1) trigger-chaining spreads over multiple computer hosts and (2) the network topology changes all the time because of the host's mobility. In this paper, we show analysis methods for the safety of our assuming active database. Especially, a method how to check the termination of ECA rules is proposed, where we introduce sigma-merge graph indicating the relationship among system events and trigger-chaining.

1. はじめに

無線通信機能を備えたパーソナルコンピュータ（PC）や携帯情報端末（PDS）による移動体計算環境が急速に普及しつつある[6),7)]．これらの端末は，モバイルeコマースとして実際のビジネスに活用されつつある．

筆者らはこれまでに移動体計算環境におけるデータ統合の有効性について議論し[13),14),16),17),19)]，特にアクティブデータベース[8),18)]の適用の可能性について考察を行ってきた[15)]．移動体計算環境では，すべての計算機は移動ホスト，固定ホスト，および移動ホストサーバとからなる．移動ホストは移動することでネットワークトポロジー上の位置を変更する計算機である．これに対して，固定ホストとはネットワークトポロジー上でその位置を変えない計算機をさし，そのうち無線通信機能を有して移動ホストと無線通信できる計算機を移動ホストサーバと呼ぶ．この移動体計算環境内で起こる移動ホストの通信セル間の移動や移動ホストの省電力設計のために実行される固定ホストとの間の通信切断動作（Disconnected Operation）[11)]などをすべて非同期なイベントとしてとらえ，それらに対してアクティブデータベースにおけるECAルールを適用して，データ統合を実現するAMDS（Active Mobile Database System）を提案した．さらに実際

† 住友電気工業株式会社
Sumitomo Electric Industries, Ltd.
†† 大阪大学大学院工学研究科情報システム工学専攻
Department of Information Systems Engineering, Graduate School of Engineering, Osaka University
††† 大阪大学大学院情報科学研究科マルチメディア工学専攻
Department of Multimedia Engineering, Graduate School of Information Science and Technology, Osaka University

のアプリケーションを ECA ルールを適用して実装してみることにより，AMDS は移動体計算環境での異機種分散システムの共通基盤として提供しうることを述べた．

ECA ルールはアクティブデータベースにおいて，イベント（Event），実行条件（Condition），操作（Action）をセットで記述し，イベントの発生により，SQL 形式のデータベース操作を実行することができる．AMDS では，この ECA ルールを移動体計算機環境に拡張し，移動ホストのセルへの出現を APPEAR，セルからの消失を DISAPPEAR として加えた．さらに，あるホストで発生したイベントを契機とするホスト間通信のためにアクションには SEND を，イベントして SEND を受ける RECEIVE をそれぞれ用意した．たとえば，研究室メンバの予定表アプリケーションでは，携帯端末（移動ホスト）を保持するメンバが研究室にきた場合は，研究室セクションごとの移動検出装置（固定ホスト）がそのメンバの来室をイベントとして予定表データベースを即時更新する．そして，現在位置を確認できない場合は予定表からの情報を得て，すべてのメンバの動静を統合ビューとして表示することができる[15]．

しかし，アプリケーションにとっては，開発された ECA ルール群がどのようなシーケンスをたどって実行されるのか，かつ連鎖的に実行されるシーケンスが停止するのかが重要である．特に，ユーザの特定のアプリケーション目的に従って作成，配置されたルールが予期しない移動ホストの出現によって，期待した動作をしないばかりか，実行シーケンス自体がホスト間で無限ループに陥ってしまい，AMDS として停止しない恐れもある．

従来のアクティブデータベースの分野では，ルールの停止条件を含めた安全性について，その動作特性を次のように定義している[1]．

- 停止性（Termination）：データベースに対するあらゆる状態変更が完了した後，ルールの処理が停止することが保証されていること．
- 合流性（Confluence）：実行権に優先付けがなされていないルール間で，実行順序によってデータベースの最終状態に差異が生じうるかどうか．いい換えれば，ルールの処理の過程で複数のルールが同時に発火した場合，ルール実行停止時のデータベースの最終状態がルールの実行順序に依存しないこと．
- 可観測決定性（Observable Determination）：ルールのアクションがデータベースの外部から見えることを観測性と呼ぶ．可観測決定性とは実行権に優先付けがなされていないルール間で，観測性のあるアクションの順序や結果がルールの実行順序に依存しないこと．

これらの特性分析は，単一のアクティブデータベース内のルール相互の安全性について解析手法に適用されたものである[5),10),20),22)]．

移動体計算環境においては，ネットワーク内の各ホスト上にアクティブデータベースが搭載されており，アプリケーションによっては ECA ルールの実行が複数のホストにまたがることがある．さらに，ECA ルールを保持している移動ホストと固定ホストとの接続関係が移動ホストの移動によってトポロジー的に変化する．従来 ECA ルールの実行停止性判定にはトリガーグラフが用いられてきたが，移動体計算機環境に適用した場合，考慮すべきトリガーグラフの構成回数が膨大となり，その停止性の判定が困難である．

本論文では，従来のトリガーグラフ構成による停止判定法をもとに σ マージトリガーグラフを提案し，移動体計算環境におけるアクティブデータベースの安全性の中でも特に移動によるルール配置の変化を考慮したルールの停止性について述べる．

以下，2 章では，AMDS の ECA ルールの実例をあげ，配置により停止しないケースを述べる．3 章では，本論文での議論の中心となる移動ホストの位置関係の縮退とルール相互の安全性に関する解析を展開する．4 章では本論文で取り上げた解析手法の考察し，5 章でまとめを行う．

2. AMDS での ECA ルール

本章では，まず，移動体計算機環境におけるアクティブデータベースとして AMDS を取り上げ，アプリケーションがホスト間にまたがって，ECA ルールの連鎖により実行される仕組みについて述べる．次に，アプリケーションにおいて，ECA ルールが連鎖的に実行される場合の停止性について述べる．

2.1 移動体計算環境における ECA ルール

本論文で扱う移動体計算環境モデルを**図 1** に示す．このモデルにおいて構成する計算機は以下の 3 種類からなる．

- 移動ホスト（*mobile host*，MH）：移動ホストは，可搬あるいは移動可能なホスト計算機で無線通信機能を有し，移動ホストサーバと通信することができる．
- 固定ホスト（*stationary host*，SH）：固定ホストは，その位置を変えないホスト計算機で，他の固

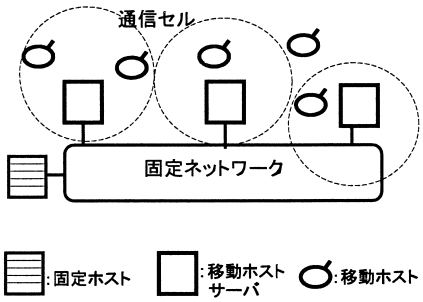

図 1 移動計算環境システムモデル
Fig. 1 A system model of mobile computing environments.

定ホストと通信することができる．

- 移動ホストサーバ（*mobile host server*，MHS）：移動ホストサーバは，固定ホストの一種であるが，無線通信機能を有して，その通信セル内にいる移動ホストと通信することができる．ここでは，簡単のため，移動ホストサーバは 1 つの通信セルを有し，通信セルは相互に重複なく設定されていると仮定する．

AMDS の動作は従来のアクティブデータベースと同様に，発生する事象（イベント），ルールの発火のための条件（コンディション），発生したさまざまな事象に応じて実行される操作（アクション）の 3 つ組で表される ECA ルールで記述する．AMDS のイベント，コンディション，アクションについてそれぞれ述べる．

- イベント：イベントには，データベース操作を起動するイベントとして従来のアクティブデータベースと同様に APPEND，DELETE，UPDATE，RETRIEVE がある．さらに，移動ホストが移動ホストサーバの管理するセル内に出現したことによって発生する APPEAR イベント，セルから退出することによって発生する DISAPPEAR イベントがある．そして，他の AMDS からのデータを受信することによって発生する RECEIVE イベントがある．AMDS で扱うことのできるイベントを**表 1** に示す．
- コンディション：コンディションには，特定のデータの指定や受信データの識別などが記述できる．
- アクション：アクションには，他の AMDS へデータ送信する SEND とデータベースへの問合せを行う QUERY が含まれている．アクションの種類を**表 2** に示す．

AMDS では，SEND と RECEIVE により，1 つのアクションが新たなイベントを発生させ，他のホストに伝えることが可能になっている．このため，あるホストで検出したイベント発生を契機として，複数の ECA ルールを駆動でき，その結果，複数のホストにまたがって連鎖的に ECA ルールを実行し，複雑な処理を実現できるようになっている．

表 1 AMDS のイベントと内部変数一覧
Table 1 Descriptors for events on AMDS.

イベント名	NEW	OLD
APPEAR:接続 MH	接続 MH 情報	—
DISAPPEAR:切断 MH	—	切断 MH 情報
RECEIVE:受信	受信データ内容	—
SELECT データ参照	参照タップル	—
APPEND:タップル追加	追加タップル	—
DELETE:タップル削除	—	削除タップル
UPDATE:タップル更新	更新後タップル	更新前タップル
TIMER:設定タイマ発火	タイマ識別子	—

表 2 AMDS のアクション
Table 2 Descriptors for actions on AMDS.

アクション名	アクション内容
QUERY([クエリー内容])	データベース操作
SEND([宛先],[送信内容])	データの送信
INSERT_ECA([ルール内容])	ECA ルール格納
DELETE_ECA([ルール識別子])	ECA ルール削除
ENABLE_ECA([ルール抽出条件])	ECA ルール有効化
DISABLE_ECA([ルール抽出条件])	ECA ルール無効化
SET_TIMER([タイマ条件])	新規タイマ設定
KILL_TIMER([タイマ識別子])	タイマ削除

2.2 ECA ルールの無限ループの例

ECA ルールの実行連鎖は，基本的にその停止性が確認されていなければならない．しかし，移動体計算機環境においては，移動ホストの移動によってその実行シーケンスが変化し，結果として無限ループに陥る恐れがでてくる．AMDS において，ECA ルールの実行でホスト間にまたがる無限ループが発生する例を遊園地入場者端末システムで示す．

入場者はそれぞれ入場者端末（移動ホスト）$m_x(x = 1, 2, 3)$ を携帯している．施設側では，入場者が利用する施設にそれぞれチェックイン装置（固定ホスト）v_1，v_2 を設置し，その通信セル内に入場者端末 m_x が入場したことを検知する機能を持っている．チェックイン装置 v_1 には 2 つのルールが配置されている．ルール 1 は，チェックイン装置 v_1 に特定の入場者端末 m_1 がチェックインしたことをイベント e_1 として，イベント e_1 をチェックイン装置 v_2 に伝えるルールである．ルール 2 はチェックイン装置 v_1 に不特定の入場者端末 m_x がチェックインしたことをイベント e_2 として，セル内にいるすべての入場者端末 m_x に伝えるルールである．

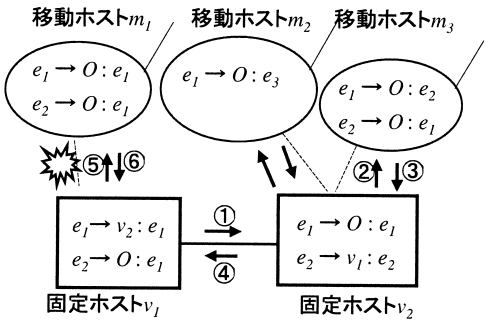

図 2 ホストをまたがって発生する無限ループの例
Fig. 2 An example of interminable loop of ECA rules through multiple hosts.

ルール 1：$e_1 \rightarrow v_2 : e_1$
Event: m_1 が v_1 と接続 (e_1).
Action: e_1 を v_2 に SEND.
ルール 2：$e_2 \rightarrow O :\ e_1$
Event: m_x が v_1 に接続 (e_2).
Action: e_2 を v_2 セル内にいるすべての m_x に SEND.

同様に，チェックイン装置 v_2 にも 2 つのルールが配置されている．
ルール 1：$e_1 \rightarrow O :\ e_1$
Event: e_1 を RECEIVE.
Action: e_1 を無線リンクでマルチキャスト SEND.
ルール 2：$e_2 \rightarrow O :\ e_1$
Event: e_x を RECEIVE.
Action: e_2 を v_1 に SEND.

図 2 にこの例でのホスト位置関係と，それぞれのホストの持つ ECA ルールのうち，イベント発生により SEND と RECEIVE によって実行無限ループが形成される可能性のあるもののみを表す．v_x は，入場者端末が直接接続されている移動ホストサーバ（固定ホスト）であり，図中四角で表す．また，移動ホストを m_x と表記し，図中円で表す．ルールのうち，イベント発生により誘発される ECA ルールの起動の流れを → で示す．ルール 1 の $e_1 \rightarrow v_2 : e_1$ は，e_1 が発生したときに，アクションとして v_2 に SEND メッセージを出し，v_2 では，それを受け取ると，e_1 イベントを発生するということを表す．また，O は，固定ホストの場合は接続されているすべての移動ホストに対するマルチキャストを表し，移動ホストの場合は接続されている固定ホストと同一セル内のすべての移動ホストに対するマルチキャストを表す．

ここで，チェックイン装置 v_1 に入場者端末 m_1 が入ってきたとする．v_1 に移動ホスト m_1 の APPEAR イベントが発生する．チェックイン装置 v_2 にはすでに端末 m_2，m_3 が無線リンクを介して接続されている．この結果，チェックイン装置 v_1，チエツクイン装置 v_2，入場者端末 m_3，チェックイン装置 v_2，チエツクイン装置 v_1，入場者端末 m_x，入場者端末 v_1 の流れによって無限ループ（① → ② → ③ → ④ → ⑤ → ⑥ → ①...）が形成され，ルール実行が停止することはない．この場合，最初のイベント e_1 が v_1 で検出されたときに，m_3 が v_2 のセル内にいない場合は，この無限ループは構成されず，実行は停止する．

このように，それぞれのホストに配置された個別のルールは妥当であっても，移動体計算機環境ではホストの移動によって，ルール設計者の意図に反し無限ループを形成してしまう場合が存在する．移動体計算環境では，移動ホストと固定ホストとの位置関係の変化がルールの実行シーケンスに影響を与えるため，停止性を含めた安全性の解析手法が必要となる．

3. 安全性解析手法

本章では，まずホストとルールの配置関係についてのモデルを提示する．続いて，移動ホストと固定ホストが搭載する ECA ルールが実行上無限ループを発生するかどうかを判定する手法を導入する．さらにそれを拡張した σ マージトリガーグラフについて述べる．

3.1 要素の定義

ホストの集合を以下のように定義する．
V：固定ホストの集合
M：移動ホストの集合
ただし，$V \cap M = \phi$とする．また，
E：イベントの集合
とする．

固定ホスト $v \in V$ は，相互に有線通信リンクで接続されている．移動ホスト $m \in M$ は，移動によって固定ホスト v との接続関係が変化する．ここで，移動ホストは同時にはただ 1 つの固定ホストに接続されると仮定する．逆に，複数の移動ホストが 1 つの固定ホストに接続されることもありうるものとする．さらに固定ホストは移動ホストを不特定多数の対象と見なし，固定ホストから移動ホストを選別してイベントを送信することはできないと仮定する．実際上の選別送信は，受信した移動ホスト側で選択的に受信することでこれと等価な通信が行えるので，本モデルで取り扱うことができる．

固定ホストが，無線リンクによって移動ホストと接続されている関係を移動ホストの位置割当てと呼び，

次のように定義する．

$l : M \to V$：移動ホストの位置割当て

M，V上のすべての位置割当てからなる集合を Loc とおく．図2に示されるような位置関係にあるホストの位置割当ては，

$l\ (m_1) = v_1$

$l\ (m_2) = v_2$

$l\ (m_3) = v_2$

で表される．

次に，各ホストに配置するECAルールについての定義をする．

$r : E \to 2^{(V \cup \{I,O\}) \times E}$：固定ホストに対するECAルール

$r' : E \to 2^{\{I,O\} \times E}$：移動ホストに対するECAルール

ここで，$r\ (e) = \{(x_0, e_0), \ldots, (x_n, e_n)\}(e \in E, x_i \in V \cup \{I, O\}$（固定ホスト）あるいは，$x_i \in \{I, O\}$（移動ホスト），$e_i \in E(i = 0, \ldots, n))$ は直観的には，イベント e が起これば，ホスト x_i に対してイベント e_i を送信する $(i = 0, \ldots, n)$ というルールを表している．O は無線リンクを介して接続されるホストの全体を表し，I はルールにおけるアクションが移動ホスト内部のローカルな動作であることを表している．このような r，r' の全体をそれぞれ R_v，R_m と記す．

図2の例では，v_1 のECAルールは，

$r(e_1) = \{(v_2, e_1)\}$

$r(e_2) = \{(O, e_1)\}$

$r(e_3) = \phi$

と表される．

固定ホストおよび移動ホストへのECAルールの割当てを次のように定義する．

$a_v : V \to 2^{R_v}$

$a_m : M \to 2^{R_m}$

ここで，

A_v：a_v の全体集合

A_m：a_m の全体集合

とおく．

自発的イベント (x_i, e_i) とは，$V \cup M$ の元と E の元の組のことをいう．ホスト x_i において，イベント e_i が自発的に起こりうることを表す．具体的には，ホスト間の通信によって発生するイベント（表1におけるRECEIVE）以外のイベントのことを意図している．自発的イベント集合

$G = \{(x_0, e_0), \ldots, (x_m, e_m)\}$

$(x_0, \ldots, x_m \in V \cup M, e_0, \ldots, e_m \in E)$

とは，自発的イベントの有限集合である．たとえば，図2の例では，

$G = \{(v_1, e_1), (v_1, e_2), (v_2, e_1), (v_2, e_2)\}$

とすることができる．

3.2 実行モデル

本章では，ECAルールの実行モデルを定義する．ホストとイベントの組がECAルールの連続的実行を誘発する場合を取り上げる．

$l \in Loc$，$a_v \in A_v$，$a_m \in A_m$，とする．$x, x' \in V \in M$，$e, e' \in E$ に対して，次のいずれかの条件を満たす場合，(x, e) が (x', e') を誘発するといい，$(x, e) \to (x', e')$ と記す．

(1) $x \in V$ かつ $x' \in V$ かつ $\exists r \in a_v(x) : (x', e') \in r(e)$

(2) $x \in V$ かつ $x' \in M$ かつ $\exists r \in a_v(x) : (O, e') \in r(e)$ かつ $l(x') = x$

(3) $x = x' \in M$ かつ $\exists r \in a_m(x) : (I, e') \in r(e)$

(4) $x = x' \in V$ かつ $\exists r \in a_v(x) : (I, e') \in r(e)$

(5) $x \in M$ かつ $x' \in V$ かつ $\exists r \in a_m(x) : (O, e') \in r(e)$ かつ $l(x) = x'$

$(x, e) \to (x', e')$ は，直観的には，ホスト x においてイベント e が発生し，その結果ホスト x' において新たなイベント e' が発生することを意味する．

次に，誘発関係により形成されるホスト，イベント組の集合の列 $E(i)(i = 0, \ldots, n,$ あるいは，$i = 0, \ldots)$ が有効イベント列であるとは，各 i に対して次のいずれかが成立することをいう．

(1) ある $(x, e) \in E(i)$ が $W = \{(x'_1, e'_1), \ldots, (x'_m, e'_m)\}$ の各元を誘発し，$E(i+1) = E \cup W$ が成り立つ．

(2) $E(i+1) = E(i) \cup W$，ただし，$W \subseteq G$.

(1)は，第 i ステップから次のステップに移ったときに，その誘発されたイベントが増えることを示す．(2)により，次のイベントの集合は，自発的イベントの集合からなっていることを示す．非活性化列とは，任意の i に対し，(1)の条件のみが成立するような有効イベント列のことをいう．

ここで，ホスト，イベント組の集合 E_0 が安全であるとは，E_0 で始まる任意の非活性化列が有限列であることをいう．また，位置割当て l が安全であるとは，自発的イベント集合が安全であることをいう．そして，システムが安全であるとは，任意の位置割当て l が安全であるこという．

本モデルによって安全なシステムでは，以下に述べる誘発有限性という好ましい性質が成立する．ホスト，イベント組の列 $(x_i, e_i)(i = 0, \ldots, n,$ あるいは，$i = 0, \ldots)$ が誘発列であるとは，任意の i に対し，$(x_i,$

e_i) が (x_{i+1}, e_{i+1}) を誘発することをいう．誘発列の長さに関する帰納法を用いて，次の補題を導くことができる．

補題 3.1（誘発有限性） 位置割当て l に対し，自発的イベントから始まる任意の誘発列が有限であることと l が安全であることは同値である．

3.3 トリガーグラフ

ここで，ルールにおけるトリガーの関係を表したトリガーグラフを使って ECA ルールの安全性，すなわち移動ホストの移動によって，固定ホストとの位置関係が変化し，搭載する ECA ルールが実行中に無限ループを形成するかどうかの判定に用いる．

位置割当て l におけるトリガーグラフは，イベントの発生するホストとそこでイベントの組をノードとし，イベントの誘発関係（送り先）をアークとする有効グラフである．すなわち，

ノード：$\mathrm{N} \subset (V \cup M) \times E$

アーク：$\mathrm{A} \subset N \times N$

ただし，アーク (x, e)，(y, e') は，

(1) $x \in V$ かつ $\exists r \in a_v(x) : (y, e') \in r(e)$ あるいは，

(2) $x \in V$ かつ $\exists r \in a_v(x) : (O, e') \in r(e)$ かつ $l(y) = x$ あるいは，

(3) $x \in V$ かつ $\exists r \in a_v(x) : (I, e') \in r(e)$ かつ $x = y$ あるいは，

(4) $x \in M$ かつ $\exists r \in a_m(x) : (O, e') \in r(e)$ かつ $l(x) = y$ あるいは，

(5) $x \in M$ かつ $\exists r \in a_m(x) : (I, e') \in r(e)$ かつ $x = y$

のときのみ存在する．このトリガーグラフは，移動ホストの位置割当ての部分以外は，通常のアクティブデータベースで用いられるトリガーグラフと同一のものである．このトリガーグラフを用いて，移動ホストの固定ホストとの接続関係が変化することを考慮してルールの停止性を調べることができる．

図 3 にルールの配置とそのトリガーグラフを示す．固定ホスト v_1 では，イベント e_1 が発生すると，v_2 に対してイベント e_1 を送り，イベント e_2 が発生するとすべての接続ホストに対してイベント e_1 を送るルールを割り当てられている．同じように固定ホスト v_2 と移動ホスト m_1 にもそれぞれルールが割り当てられている．このルールの発火の誘発関係のトリガーグラフは図 3 の右図のようになる．

定理 3.1 任意の位置割当てに対してトリガーグラフ上で自発的イベントから可到達なループがないこととシステムが安全であることは同値である．

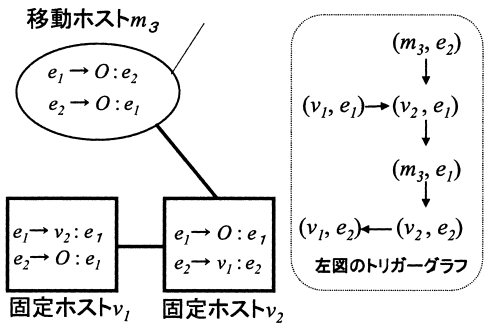

図 3 ルールの配置とトリガーグラフ
Fig. 3 An example of ECA rule allocation on hosts and its trigger graph.

証明 3.1 システムが安全でない場合は，任意の有効イベント列において無限の誘発列が存在するため停止しない．そのときの位置割当てに対するトリガーグラフはそのループを構成する有効イベント列に同じホスト，イベント組が 2 回以上出現する．したがって，トリガーグラフに同じノードが 2 回以上出現していれば，その区間がグラフ上ループであることが示せる．以上のことにより，補題 3.1 を用いて定理が証明される．［証明終］

3.4 σ マージトリガーグラフ

トリガーグラフでは，ノードにホスト，イベント組をとり，アークとして ECA ルールの誘発関係をとって構成している．トリガーグラフの構成の回数を減らすために，別のグラフの構成方法を導入する．

V の部分集合の集合 p が，V の分割であるとは，

$\bigcup p = V$，$\forall x, y \in p : x \neq y$ ならば $x \cap y = \phi$

が成り立つような集合のことをいう．V の分割全体を P と記す．$p \in P$ に対して，$x \in v$ を含む p の元は唯一つきまり，これを $[x]$ と記す．

位置割当て l に対する拡張位置割当て，$\sigma : M \rightarrow P$ とは，$m \in M$ に対し，$\sigma(m) = [l(m)]$ で定まる関数のことをいう．分割 p がすべて単一元からなるとき，拡張位置割当ては前述の位置割当てと自明な対応関係を持つ．拡張位置割当ての全体を LOC と記す．任意の移動ホスト x に対して $\sigma(x)$ が単一元の集合となるようなものの全体が $Loc \subseteq LOC$ となる．位置割当て l，および分割 p に対する σ マージトリガーグラフは次のようなグラフのことをいう．

ノード：$\mathrm{N} \subset (V \cup M) \times E$

アーク：$\mathrm{A} \subset \mathrm{N} \times \mathrm{N}$

ただし，アーク (x, e)，(y, e') は，

(1) $x \in V$ かつ $\exists r \in a_v x) : (y, e') \in r(e)$ ある

いは，

(2) $x \in V$ かつ $\exists r \in a_v(x) : (O, e') \in r(e)$ かつ $y \in \sigma(x)$ あるいは，

(3) $x \in V$ かつ $\exists r \in a_v(x) : (I, e') \in r(e)$ かつ $x = y$ あるいは，

(4) $x \in M$ かつ $\exists r \in a_m(x) : (O, e') \in r(e)$ かつ $y \in \sigma(x)$ あるいは，

(5) $x \in M$ かつ $\exists r \in a_m(x) : (I, e') \in r(e)$ かつ $x = y$

のときのみ存在する．ここで σ は，l，p に対する拡張割当てである．(2) で特に，$y \neq l(x)$ となるリンクを拡張リンクという．拡張リンクを含まないループを有効ループと呼ぶ．

定理 3.2（σ マージトリガーグラフ） 分割 p に対して，システムが安全であることと，p に対する任意の拡張位置割当てに対して σ マージグラフに自発的イベントから到達可能な有効ループがないこととは同値である．

証明 3.2 定理 3.1 を用いて，有効ループがあれば安全でないことは容易に示せる．安全でなければ，有効ループがあることは，無限の非活性化列がないことから示される．[証明終]

固定ホスト v，v' が互いに素であるとは，トリガーグラフで v，v' を含むノードがそれぞれ互いに非連結となっていることをいう．固定ホスト v，v' のマージトリガーグラフとは，移動ホスト m に対して，固定ホスト v との間で形成される位置割当てのうち，分割 $\{\{x\} | x \neq v \text{ かつ } x \neq v'\} \cup \{\{v, v'\}\}$ を用いた σ マージトリガーグラフのことをいう．

定理 3.3（素な固定ホストの場合） 固定ホスト v，v' が互いに素なら，システムが安全であることと，v，v' の σ マージトリガーグラフに有効ループがないこととは同値である．

証明 3.3 安全ならばトリガーグラフにループはない．安全かつ v，v' が互いに素であるならば，v，v' のマージトリガーグラフに新たなループができることはない．[証明終]

4. 考　　察

ECA ルールの実行モデルにおいて，移動ホストの位置割当てからイベントとホスト組の集合に着目し，トリガーグラフを構成して可到達なループの検出をすることにより，システムの安全性が確かめられる見通しが立った．しかし，対象となるすべての移動ホストと移動ホストサーバとの関係，すなわち移動ホストのすべての位置割当てについて調べるため，トリガーグラフの構成回数は $|V|^{|M|}$ 回になる．たとえば，固定ホスト 2 台，移動ホスト 2 台の場合の位置割当ては**図 4** のようなパターンをとりうる．システムが停止性において安全であるには，少なくとも移動ホストからみて，接続先の固定ホストに関係なく無限ループが存在してはならない．その場合，先にトリガーグラフにおいて調べ上げた場合のうち，単に移動ホストが接続される固定ホストが区別されたために発生した割当てを統合するほうが有利である．そこで，この数を減らして簡便に調べる方法が必要となる．次に導入した拡張位置割当て σ では，直観的には無限ループの形成が最低 2 本のアークが存在するはずとの前提に着目した．σ マージトリガーグラフの導入により，1 つでも有効ループを形成するトリガーグラフが検出されれば，その ECA ルールの位置割当ては安全でないことがいえるためである．これにより，ルールの実行停止性の検証が効率的に実行できる見通しがたった．特に，σ として $V \times V$ を選ぶと，マージトリガーグラフの有効ループを調べることで無限ループの危険性を確かめられることになる．算出されたループについてその妥当性をチェックすることで，すべての位置割当てについてトリガーグラフを構成する場合と同等の計算結果が得られるものと考えられる．すべての位置割当てについてトリガーグラフを構成する場合には，$|M| \times |V|$ とおりの位置割当てがあるため，固定局のリンク数を n とすると，計算量は $O((n \times |M|) \times |V|^{|M|})$ となるが，$V \times V$ マージトリガーグラフの場合には，妥当性のチェックを除く計算量は $O(n + |M|\ |V|)$ となる．

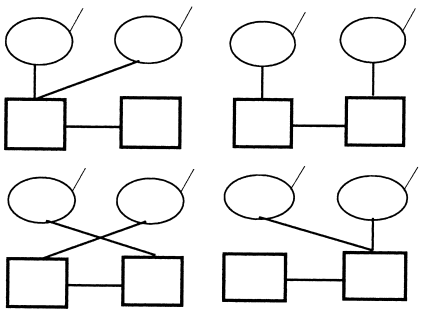

図 4 固定ホスト 2 台 × 移動ホスト 2 台の位置割当て
Fig. 4 Node-link assignments of two stationary hosts and two mobile hosts.

一方，本手法ではルールの位置割当てについては，移動という動作後の場合で論じた．正確には，移動というイベントが反映されない場合も出てくると考えられる．本手法は，ECA ルールが制作された時点で，実際のホストへの配備に先立って実施されるものと想

定しているため，判定のための実行時間は問題にはならない．しかし，新しい移動ホストに対して，出現と同時に判定を実施する場合は，本手法の実行時間が問題となる．このための対応としては，移動ホストが非常に短時間内にセルへの出入りを繰り返した場合は，イベントとして判定するためのフィルタを用意し，イベント判定の閾値を高めて安定した動作を図ることが必要となろう．また，本論文ではECAルールのコンディション（条件）の部分を考慮していないが，安全システムの誘発有限列というシステム解析上好ましい性質が得られた．安全性判定に条件部を考慮した効率良い解析手法を検討する必要がある．さらに，移動体計算環境では非同期に複数のルールが同時に発火した場合の合流性や特定ルールの実行権に優先度付けがなされる場合の可観測決定性についても検討する必要がある．

アクティブデータベースの分野では，ECAルールのループに対して有向グラフを使った停止性が論じられている[2)~4),9)]．さらに，アクティブデータベースのECAルールの条件部を考慮した安全性としては，Leeらの研究[12),21)]もみられる．しかし，いずれも単一システムにおける安全性に関しての工夫であって，本論文で取り上げた複数のアクティブデータベース間で，しかも移動計算環境におけるトリガーグラフの適用を試みた研究はみられない．

5. ま と め

本論文では移動体計算環境におけるデータ統合のためのAMDSにおけるECAルールの安全性について議論した．まず，移動体計算環境においては，ホストの移動により，移動ホストと移動ホストサーバの位置関係が変化するため，従来のトリガーグラフでは，その停止性の判定が困難であることを述べた．これに対して，あらたに σ マージトリガーグラフの導入することによって，複数のホスト間にまたがってループが形成され，しかも動的に多くの組合せが形成されるAMDSでのECAルールの停止性について効率良く判定できることを示すことができた．

参 考 文 献

1) Aiken, A., Hellerstein, J. and Widom, J.: Static Analysis Techniques for Predicting the behavior of Active Database Rules, *ACM Trans. Database Syst.*, Vol.20, No.1, pp.3–41 (1995).
2) Baralis, E., Ceri, S. and Paraboschi, S.: Runtime Detection on Non-Terminating Active Rule Systems, *DOOD*, pp.59–68 (1992).
3) Baralis, E. and Widom, J.: An Algebraic Approach to Rule Analysis in Expert Database Systems, *20th VLDB Conf.*, pp.475–486 (1994).
4) Ceri, S. and Widom, J.: Deriving Production Rules for Constraint Maintenance, *Proc. 16th International Conference on VLDB*, pp.566–577 (Aug. 1990).
5) Hanson, E.: An lnitial Report on the Design of Ariel: A DBMS with an Integrated Production Rule System, *SIGMOD RECORD*, Vol.18, No.3 (1989).
6) Imielinski, T. and Bardrinath, B.R.: Mobile Wireless Computing: Solutions and Challenges in Data Management, Technical Report DCS-296, Dept. of Computer Science, Rutgers University (1993).
7) Imielinski, T. and Badrinath, B.R.: Mobile Wireless Computing, *Comm. ACM*, Vol.37, No.10, pp.18–28 (1994).
8) 石川 博：データベース，情報処理，Vol.35, No.2, pp.120–129 (1994).
9) Karadimce, A.P. and Urban, S.D.: Refined Trigger Graphs: A Logic-Based Approach to Termination Analysis in an Active Object-Oriented Database, *Proc. ICDE'96*, pp.384–391 (1996).
10) Kifer, M., Ramajrishnan, R. and Siberschhanz, A.: An Axiomatic Approach to Deciding Query Safety in Deductive Database, *Proc. PODS '88*, pp.52–60 (1988).
11) Kistler, J.J. and Satanarayanan, M.: Disconnected Operation in the Coda File System, *Proc. 13th Symposium on Operating System Principles*, pp.13–16 (1991).
12) Lee, S.Y. and Lin, T.W.: A Path Removing Technique for Detecting Trigger Termination, *Proc. 6th International Conference on EDBT'98*, pp.341–355 (March 1998).
13) 劉 渤江，塚本昌彦，西尾章治郎：移動体計算環境におけるデータベースビュー，*Proc. Advanced Database System Symposium'94*, pp.9–18 (1994).
14) 劉 渤江，仲秋 朗，塚本昌彦，西尾章治郎：移動体計算環境におけるデータベースビュー定義言語，電子情報通信学会技術研究報告，Vol.95, No.81 (DE95-1〜8), pp.25–32 (1995).
15) 村瀬 亨，塚本昌彦，西尾章治郎：データベースシステムによる移動体計算境におけるデータ統合，電子情報通信学会データ工学研究会技術研究報告，Vol.95, No.287, pp.41–48 (1995).
16) 仲秋 朗，劉 渤江，塚本昌彦，西尾章治郎：移動体データベースのビュー雑持手法，情報処

理学会研究報告，Vol.95, No.31 (95-DBS-102), pp.33–40 (1995).
17) 白井博章，仲秋 朗，劉 淘江，塚本昌彦，西尾章治郎：移動型データベースのビュー機構の設計および実装，電子情報通信学会技術研究報告，Vol.95, No.148 (DE95-27〜37), pp.1–8 (1995).
18) Stonebraker, M.: The lntegration of Rule Systems and Database, *IEEE Trans. Knowledge and Data Engineering*, Vol.4, No.5, pp.415–423 (1992).
19) Tsukamto, M., Liu, B. and Nishio, S.: MobiView: A Database View for Mobile Computing Environments, ISE-TR-95-O13, Dept.lnformation Systems Engineering, Faculty of Engineering Osaka University (June 1995).
20) Vaduva, A., Gatziu, S. and Dittrich, K.R.: Investigating Termination in Active Database Systems with Expressive Rule Languages, ifi-97.03, Institut fur Informatik, Universitat Zurich (March 1997).
21) van der Voort, L. and Siebes, A.: Termination and confluence of rule execution, *2nd International Conf. on Information and Knowledge Management* (1993).
22) Vardi, M.: Decidability and Undecidability Results for Boundedness of Linear Recursive Queries, *Proc. PODS'88*, pp.341–351 (1988).

(平成 14 年 1 月 21 日受付)
(平成 14 年 6 月 4 日採録)

村瀬 亨（正会員）

1976 年京都大学工学部精密工学科卒業．同年住友電気工業（株）入社．1987 年〜1989 年スタンフォード大学客員研究員．同社 IT 技術研究所を経て，現在，経営開発部に勤務．大阪大学大学院工学研究科在籍．分散システム，インターネットプロトコル，移動体システムに興味を持つ．電子情報通信学会，ACM 各会員．

塚本 昌彦（正会員）

1987 年京都大学工学部数理工学科卒業．1989 年同大学院工学研究科修士課程修了．同年，シャープ（株）入社．1995 年大阪大学大学院工学研究科情報処理システム工学専攻講師，1996 年同専攻助教授，2002 年より同大学院情報科学研究科マルチメディア工学専攻助教授，現在に至る．工学博士．時空間データベースおよびモーバイルコンピューティングに興味を持つ．ACM，IEEE 等 7 学会の会員．

西尾章治郎（正会員）

1975 年京都大学工学部数理工学科卒業．1980 年同大学院工学研究科博士課程修了．工学博士．京都大学助手，大阪大学基礎工学部および情報処理教育センター助教授を経て，1992 年大阪大学大学院工学研究科情報システム工学専攻教授，2002 年より同大学院情報科学研究科マルチメディア工学専攻教授，現在に至る．2000 年より大阪大学サイバーメディアセンター長併任．この間，カナダ・ウォータールー大学，ビクトリア大学客員．データベース，知識ベース，分散システムの研究に従事．現在，ACM Trans. on Internet Technology, Data & Knowledge Engineering，Data Mining and Knowledge Discovery，The VBDL Journal 等の論文誌編集委員．本学会フェロー．ACM，IEEE 等 8 学会の会員．