社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2008－EMB－10 (12)
2008／11／28

並列動作する確率時間システムに対する拡張CEGAR

安井 雅俊†, 山根 智†, 山崎 真一†,

† 金沢大学大学院 自然科学研究科

近年, 無線センサネットワークがさまざまな分野において注目を集めているが, その検証法については幾つかの問題が存在し確立されていない. そこで, 本論文では, 無線センサネットワークを例に取った, 並列動作を行う確率時間システムの特性を検証するための手法として, 拡張 CEGAR を提案する.

Extended CEGAR for Parallel Probabilistic Timed Systems

Masatoshi YASUI † Satoshi YAMANE † Shinichi YAMAZAKI †

† Kanazawa University

Though wireless sensor networks attract attention in various fields in recent years,the verification method have not been established because there are some problems. Then, in this paper, we propose extended CEGAR to verify probabilistic real-time properties of parallel behaviers of wireless sensor networks.

1 まえがき

無線センサネットワーク (WSNs) の構成要素であるセンサノードは, 確率リアルタイムシステムとして表現することが可能であるため, その動作を確率時間オートマトン [1] を用いて記述し, 検証を行うことは可能である. しかし, そのノードが並列動作し, 広域をカバーするネットワークを構成する場合, ノードの数が莫大になり, ネットワーク全体のモデルに状態爆発現象が引き起こされるため, WSNs 全体としての検証を困難なものとしている. WSNs のモデル検査についての研究はほとんど例がなく, 大半が部分的なモデル検査にとどまっている. 関連研究としては, 確率リアクティブモジュールで WSNs をモデル化し, Approximate Probabilistic Model Checker というツールによって, 近似を行って状態空間の増大を抑え, かつネットワークの特性検証を行ったものがある [2]. しかし, 組込みシステムにおける重要な特性であるリアルタイム性の表現には至っていない.

そこで, 本研究では, WSNs の様な確率時間システムが複数並列に動作するシステムを対象とし, リアルタイム性と確率的な動作を扱うことができる確率時間オートマトンを用いモデル化を行い, 並列合成による状態爆発を削減することを目的とした検証手法として, 並列動作する確率時間システムに対する反例を用いた述語抽象化と精錬の枠組みである拡張 CEGAR を提案する.

2 確率時間オートマトン

本章では, 確率時間オートマトンのシンタックスとセマンティクスを定義し, 確率時間オートマトンの合成を定義する [1], [3].

2.1 前準備

定義 1 (離散確率分布)

可算状態集合 $\mathcal{S}$ 上の離散確率分布の集合を $Dist(\mathcal{S})$ で表す.$\mu \in Dist(\mathcal{S})$ は関数 $\mu : \mathcal{S} \to [0,1]$ である. ただし,$\sum_{s\in\mathcal{S}} p(s) = 1$ かつ集合 $\{s \mid s \in \mathcal{S}$ かつ $p(s) > 0\}$ は有限である.

次に, 時間経過を表すクロック変数とクロック変数の評価, クロックの制約を定義する.

定義 2 (クロック変数)

クロック変数は非負の実数値を取る変数であり, 全てのクロックが同じ速度で増加し, 遷移中に 0 にリセットすることが可能である.$\mathbb{R}_{\geq 0}$ 上のクロック変数の集合を $\mathcal{X}$ とする.

定義 3 (クロック評価)

クロック評価は関数 $\nu : \mathcal{X} \to \mathbb{R}_{\geq 0}$ である. $\mathcal{X}$ の全てのクロック評価の集合は $\mathbb{R}_{\geq 0}^{\mathcal{X}}$ と表す. $\mathbf{0}$ を $\mathcal{X}$ の全てのクロックに 0 を割り当てるクロック評価とする. $X \subseteq \mathcal{X}$ である集合 X に対して, X 内の任意のクロック変数 x を 0 にリセットし, $\mathcal{X}\backslash X$

内の任意のクロック変数 x は $\nu(x)$ である評価を $\nu[X := 0]$ と表記する．$t \in \mathbb{R}_{\leq 0}$ であるすべての t で $\nu + t$ は，すべての $x \in \mathcal{X}$ に対して，$\nu(x) + t$ の評価を与えるクロック評価とする．

定義 4 （ゾーン）
$\mathcal{X}$ 上のゾーン $\zeta \in Zones(\mathcal{X})$ はクロック評価の集合 $\mathbb{R}_{\geq 0}^{\mathcal{X}}$ の凸部分集合として以下のような構文で帰納的に定義される．

$$\zeta ::= x \leq c | x < c | x \geq d | x > c | x - y \leq d | x - y < d | \zeta \wedge \zeta$$

ここで，$x, y \in \mathcal{X}$，$c, d \in \mathbb{N}$ である．

クロック評価 ν が，ゾーン ζ を満足するとは，ゾーン中の各クロック変数 $x \in \mathcal{X}$ を ν によって対応するクロック値 $\nu(x)$ によって置き換えた後でクロック評価に関するゾーンの真偽値 $\zeta\nu \in \{true, false\}$ が $true$ であるとき，またその時に限る．

2.2 確率時間オートマトンのシンタックス

定義 5 （確率時間オートマトン）
確率時間オートマトン PTA は，組 $(L, \bar{l}, \mathcal{X}, Acts, inv, prob)$ で定義される．

- L は各ロケーションの有限集合．
- $\bar{l} \in L$ は初期ロケーション．
- $\mathcal{X}$ はクロックの有限集合．
- $a \in Acts$ は同期アクションの有限集合．
- 関数 $inv : L \to Zones(\mathcal{X})$ は各ロケーションに不変条件を割り当てる関数．
- $prob \subseteq L \times Zones(\mathcal{X}) \times Acts \times Dist(L \times 2^{\mathcal{X}})$ は有限の確率遷移関係．

確率時間オートマトン PTA の状態は対 $(l, \nu) \in \mathcal{S} \times \mathbb{R}_{\geq 0}^{\mathcal{X}}$ と表わされ，ν は $inv(l)$ を満足する．つまり，ν は $inv(l)$ が表すゾーンのクロック評価の集合の要素である．PTA は初期状態 $(\bar{l}, \nu_0)$ から動作を行う．ここで，ν_0 は全てのクロックの値を 0 とする初期クロック評価を表す．ある時点の状態 (l, ν) において，PTA は次のいずれかの動作を非決定的に選択する．

- 時間遷移
  ロケーション l に留まり，不変条件 $inv(l)$ に違反しないようにクロックの値を増加する．

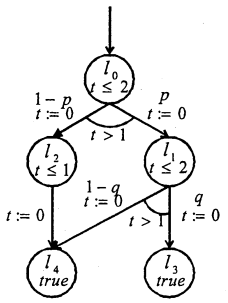

図 1: 確率時間オートマトンの例

- 確率遷移
  現在の状態を満足する確率遷移関係 $(l, g, a, \mu) \in prob$ が存在するとき，μ が指す遷移先のロケーション l' に遷移を行う．

確率遷移は遷移元ロケーション l の実行可能な確率遷移関係 $(l, g, a, \mu) \in prob$ によって行われる．ここで，ゾーン g は現在のクロック評価 ν を満足する．このとき，PTA のロケーションが l' に遷移し，集合 X 上の全てのクロックが 0 にリセットされる確率は $\mu(l', X)$ と表記する．

定義 6 （確率時間オートマトンの $edge$）
確率時間オートマトンの $edge$ は $(l, g, a, \mu) \in prob$ によって生成され，$\mu(l', X) > 0$ であるような組 (l, g, a, μ, X, l') の形をとる．$\mathrm{edgess}(l, g, a, \mu)$ は (l, g, a, μ) によって生成される $edge$ の集合とし，$\mathrm{edgess}(l, g, a, \mu) = \{(l, g, a, \mu, X, l') | (l, g, a, \mu) \in prob$ かつ $\mu(l', X) > 0\}$ であるとする．

確率時間オートマトンの例を図 1 に示す．まず，初期ロケーション l_0 でクロック $t = 0$ の状態から動作を開始する．次に $t > 1$ となるまで時間経過した後，離散遷移を行うか，$t = 2$ となるまで時間経過し，離散遷移する．確率 p でロケーション l_1 に確率遷移し，確率 $1 - p$ でロケーション l_2 に確率遷移する．ロケーション l_1 に遷移した場合，ロケーション l_0 と同様に時間経過し，離散遷移を行って，確率 q でロケーション l_3 に確率遷移するか，確率 $1 - q$ でロケーション l_4 に確率遷移する．

次に，確率時間オートマトンの並列合成について定義する．無線センサネットワークは複数のセンサノードが並列に動作し，同期することで構成されている．

定義 7 （確率時間オートマトンの並列合成）

二つの確率時間オートマトン $PTA_i = (L_i, \bar{l}_i, \mathcal{X}_i, Acts_i, inv_i, prob_i)$ $(i = 1, 2)$ について, $PTA_1 \otimes_f PTA_2 = (L, \bar{l}, \mathcal{X}, Acts, inv, prob)$ は PTA_1, PTA_2 の並列合成であり, 以下の要素からなる確率時間オートマトンである.

- $L = L_1 \times L_2$
- $\bar{l} = (\bar{l}_1, \bar{l}_2)$
- $\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2$
- $Acts = Acts_1 \cup Acts_2$
- $inv : L_1 \times L_2 \to Zones(\mathcal{X}_1) \wedge Zones(\mathcal{X}_2)$
- $prob \subseteq L \times (Zones(\mathcal{X}_1) \wedge Zones(\mathcal{X}_2)) \times f((Acts_1 \times \{0\}) \times (Acts_2 \times \{0\})) \times Dist(L \times 2^{\mathcal{X}})$

ここで, $p_{(1 \otimes_f 2)}(l, X) \in Dist(L \times 2^{\mathcal{X}}), l = (l_1, l_2), X = X_1 \cup X_2$ であるとき、 $p(l, X) = p_1(l_1, X_1) \cdot p_2(l_2, X_2)$. また, f は $(Acts \times \{0\}) \times (Acts \times \{0\})$ を $Acts$ に割り付けるような同期関数であり, f が自明であるとき $PTA_1 \otimes_f PTA_2$ を $PTA_1 \otimes PTA_2$ と書く.

2.3 確率時間オートマトンのセマンティクス

確率時間オートマトンのセマンティクスを確率時間ストラクチャとして定義を行う. 確率時間ストラクチャはマルコフ決定過程 (MDP) の形をとり, 非決定的な遷移を行う.

定義 8 （確率時間ストラクチャ）

確率時間ストラクチャ $\mathcal{M}$ はマルコフ決定過程 $(Q, q_0, Steps)$ である.

- Q は状態の集合
- q_0 は初期状態
- $Steps \subseteq Q \times \mathbb{R}_{\geq 0} \times Dist(Q)$ は確率遷移関係であり, 全ての $q \in Q$ に対して, $\mu \in Dist(Q)$ かつ $t > 0$ であるような, $(q, t, \mu) \in Steps$ が存在する.

ここで, $(t, \mu) \in Step(q), t \in \mathbb{R}$ かつ $\mu \in Dist(Q)$ を非決定的に選択し, t 時間経過した後 $\mu(q') > 0$ である様な μ に従って, 目的の状態 q' への確率的な選択が行われる.

2.4 パス

確率時間ストラクチャのパスは非決定的選択と確率的選択の解決として表現される. 確率時間ストラクチャ $\mathcal{M} = (Q, q_0, Steps)$ のパス ω は以下の様な非空の有限あるいは無限列である.

$$\omega = q_0 \xrightarrow{t_0, \mu_0} q_1 \xrightarrow{t_1, \mu_1} q_2 \xrightarrow{t_2, \mu_2} \cdots$$

ここで, $0 \leq i \leq |\omega|$ において, $q_i \in Q, (q_i, t_i, p_i) \in Steps, p_i(q_i) > 0$ である. パス ω の i 番目の状態を $\omega(i)$, i 番目の遷移を $step(\omega, i)$ とし, ω が有限列であるならば, その長さを $|\omega|$, その最後の状態を $last(\omega)$ と表す. 状態 q から始まる全ての有限あるいは無限パスの集合を, それぞれ $Path_{fin}(q), Path_{ful}(q)$ と表す.

ここで, 非決定性のみ解決する表現として, 確率時間ストラクチャのアドバサリを導入する.

定義 9 （アドバサリ）

確率時間ストラクチャ $\mathcal{M} = (Q, q_0, Steps)$ のアドバサリ A は, $\mathcal{M}$ の全ての有限パス ω_{fin} を $(last(\omega_{fin}), \mu) \in Steps$ が存在する離散分布 μ に写像する関数である.

任意のアドバサリ A と状態 q に対して, $Path^A_{ful}(q), Path^A_{fin}(q)$ は, それぞれ A によって生じる $Path_{ful}(q), Path_{fin}(q)$ のサブセットを表すとし, $Adv_{\mathcal{M}}$ を確率時間ストラクチャ $\mathcal{M}$ のアドバサリの集合とする. また, パスの最後の状態が等しければパスに依らず全て同じ確率分布を返すアドバサリをシンプルなアドバサリ A_{simple} として区別する. 以降, 単にアドバサリと書くときはシンプルなアドバサリを指す.

アドバサリの元では, 確率時間ストラクチャの非決定的な選択は解決される. ここで, 確率時間ストラクチャ $\mathcal{M} = (Q, q_0, Steps)$ について, 与えられたシンプルなアドバサリ A_{simple} の下での振る舞いは離散時間マルコフ連鎖 $(DTMC)$ によって記述できる. 次に, アドバサリと関連付けたマルコフ連鎖と確率時間ストラクチャに現れるパスの発生確率を定義する.

定義 10 （離散時間マルコフ連鎖）

確率時間ストラクチャ $\mathcal{M} = (Q, q_0, Steps)$ について, 与えられたシンプルなアドバサリ A の下での $\mathcal{M}$ の振る舞いは離散時間マルコフ連鎖 $DTMC^A$ で記述でき, 組 $(Q^A, q_0^A, \mathbf{P}^A)$ と表す. ここで, 任意の状態 q, q' に対して,

$$\mathbf{P}^A(q, q') = \begin{cases} \mu(q')\ if\ \exists \omega. last(\omega) = q \wedge A(\omega) = \mu \\ 0 \qquad\qquad otherwise. \end{cases}$$

となる.

定義 11 （パスの確率）

確率時間ストラクチャ$\mathcal{M}$のアドバサリをAとする．このとき，パスの発生確率$Prob^A_{fin}$: $Path^A_{fin} \to [0,1]$を以下の様に定義する．

$$Prob^A_{fin}(\omega) = \begin{cases} \mathbf{P}^A(\omega(0),\omega(1))\cdots \\ \mathbf{P}^A(\omega(n-1),\omega(n)) & if\ |\omega| \neq 0 \\ 1 & otherwise. \end{cases}$$

以上より，確率時間オートマトンPTAのセマンティクスを確率時間ストラクチャと関連付けて定義を行う．

定義 12 （確率時間オートマトンのセマンティクス）

$PTA = (L, \bar{l}, \mathcal{X}, Acts, inv, prob)$を確率時間オートマトンとする．このとき，$PTA$のセマンティクスを確率時間ストラクチャ$\mathcal{M}_{PTA} = (Q_{PTA}, q_{0\ PTA}, Steps_{PTA})$として，以下のように定義する．

- $\mathcal{M}_{PTA}$の状態は$(l,\nu) \in Q_{PTA}$である．ここで$Q_{PTA} \subseteq L \times \mathbb{R}^{\mathcal{X}}_{\geq 0}$.
- 以下の何れかの状態を保つ時，またその時に限り$((l,\nu),t,\mu) \in Steps$である．
  時間遷移：$t \geq 0, \mu = \mu(l,\nu+t)$かつ全ての$0 \leq t' \leq t$について$\nu + t'$は$inv(l)$を満足する．
  離散遷移：$t = 0$かつνがgを満足するような$(l,g,a,\mu) \in prob$が存在するとき，全ての$\mu((l',X)) > 0$であるような(l',X)と任意の$(l',\nu') \in Q$に対して，$\nu[X := 0]$は$inv(l')$を満足し，以下の様になる．

$$\mu(l',\nu') = \sum_{X \subseteq \mathcal{X}\ \&\nu' = \nu[X:=0]} \mu(l',X)$$

2.5 確率到達可能性解析

本論文では，システムが目標とする状態へある確率以上で到達するか否かを検証する確率到達可能性解析を用いて，確率時間システム—無線センサネットワークの特性を検証することを目的とする．

定義 13 （確率到達可能性問題）

確率時間オートマトン$PTA = (L, \bar{l}, \mathcal{X}, Acts, inv, prob)$において，$l_{target}$を目標ロケーションとし，$>$と$\lambda \in [0,1]$によって，目標ロケーションへの到達確率を表すとする．このとき，PTAの確率到達可能性問題は，組$T = (l_{target}, >, \lambda)$で定義される．ここで，ある$\mathcal{M}$のアドバサリ$A \in Adv_{\mathcal{M}}$において，$PTA$の初期状態$\bar{l}$から始まり，$last(\omega) = l_{target}$となるパスが一つ以上存在し，そのパスの合計発生確率P_{max}が条件$P_{max} > \lambda$を満たすとき，かつその時に限り，PTAの確率到達可能性問題の答えは"*Yes,Reachable*"となり，そうでなければ"*No*"となる．

3 述語抽象化とその精錬

WSNs全体の動作はネットワーク全体を構成するノードの動作モデルを定義7の並列合成で示した通りに合成することでモデル化できる．一般に実用的なWSNs内のノード数は数十程度と言われており，合成したネットワーク全体の動作を記述しても，その状態数は非常に莫大なものとなり検証は非常に困難なものとなってしまう．そのため，計算機のメモリに乗せて現実的に検証可能な状態数になるよう抑え込む必要がある．そこで，状態空間を抽象化して表現する方法である述語抽象化を導入する [4].

3.1 述語抽象化

述語抽象化は無限状態遷移系の有限の近似を得るために用いられる．この手法は抽象化述語の集合に基づいて抽象化を行う．

定義 14 （抽象化述語）

ロケーションlにおける，クロック変数の集合$\mathcal{X}$に関する抽象化述語ψ^lは$\mathcal{X}$上の自由変数の集合を持つあらゆる論理式である．クロック変数の集合$\mathcal{X}$において，述語ψ^lは以下のように定義される．

$$\psi ::= x_1 \leq c | c \leq x_1 | x_1 < c | c < x_1 | x_1 - x_2 \leq d | x_1 - x_2 < d | true$$

ここで，$x_1, x_2 \in \mathcal{X}, c \in \mathbb{N}, d \in \mathbb{Z}$である．クロック評価$\nu$，抽象化述語$\psi$において，$\nu$に関する述語$\psi$の真偽値を$\psi\nu \in \{true, false\}$とすると，$\psi$に現れるクロック$x \in \mathcal{X}$に対応する値$\nu(x)$を代入した結果が真となるとき，かつその時に限り，$\nu$は述語$\psi$を満たし，$\nu \models \psi$と書く．また，全てのクロック評価$\nu \in \mathcal{V}_{\mathcal{X}}$において，$\nu \models true$とする．

ロケーションlにおける抽象化述語の集合$\Psi^l = \{\psi^l_1, \cdots, \psi^l_n\}$は，クロック評価$\nu$から長さ$n$のビットベクトル$b^l$へのマッピングである．ここで，全てのロケーションにおける抽象化述語の集合を

$\Psi^{all} = \{\Psi^{l_0} \cup \cdots \cup \Psi^{l_k}\}$ とすると，Ψ^{all} により抽象化関数 α が決定される．b^l の i 番目の要素はロケーション l において $\psi^{l_i}\nu$ が真となるとき，かつその時に限り真となる．ここで，l における長さ n のビットベクトルは集合 B_n^l の要素であるとし，B_n^l はドメイン $\{0,\cdots,n-1\}$ と変域 $\{0,1\}$ を持つ関数であると仮定する．また，全てのロケーションにおけるビットベクトルの集合を $\mathcal{B}$ とする．α の逆像は具体化関数 γ であり，これはビットベクトル b^l を，ビットベクトル b^l の i 番目の要素が真であるときは常に ψ_i^l を満たすような全てのクロック評価に写像する関数である．よって，具体状態 (l,ν) の集合は抽象化関数 α によって抽象状態 $\alpha(l,\nu)$ に写像され，抽象状態 (l,b^l) は具体化関数 γ により具体状態の集合 $\gamma(l,b^l)$ に写像される．以下に抽象化，具体化について定義を行う．

定義 15 （抽象化・具体化）

$\mathcal{X}$ はクロックの集合，$\mathcal{V}_\mathcal{X}$ はそれに対応するクロック評価の集合であるとする．述語の有限集合 $\Psi^{all} = \{\Psi^{l_0} \cup \cdots \cup \Psi^{l_k}\}$ が与えられたとき，抽象化関数 $\alpha : L \times \mathcal{V}_\mathcal{X} \to L \times \mathcal{B}$ は以下のように定義される．

$$\alpha(l,\nu)(i) = (l, \psi_i\nu)$$

また，具体化関数 $\gamma : L \times \mathcal{B} \to L \times \mathbb{R}_{\geq 0}^{\mathcal{X}}$ は以下のように定義される

$$\gamma(l,b^l) = \{(l,\nu) \in L \times \mathcal{V}_\mathcal{X} \mid inv(l) \wedge \bigwedge_{i=0}^{n-1} \psi_i^l\nu \equiv b^l(i)\}$$

α,γ に関して表記 $\alpha(Q) = \{\alpha(l,\nu) \mid (l,\nu) \in Q\}, \gamma(Q^\sharp) = \{\alpha(l,b^l) \mid (l,b^l) \in Q^\sharp\}$ を用いる．ここで抽象化・具体化関数の対 (α,γ) はガロア接続の形を成す．

定義 16 （抽象構造）

具体遷移系 $\mathcal{M} = \langle Q, q_0, Steps\rangle$ の抽象構造 $\mathcal{M}^\sharp = \langle Q^\sharp, q_0^\sharp, Steps^\sharp\rangle$ を構築する．抽象構造 $\mathcal{M}^\sharp$ は以下の要素からなる．

- $Q^\sharp = L \times \mathcal{B}$
- $q_0^\sharp = \alpha(q_0)$
- $Steps^\sharp \subseteq Q^\sharp \times Dist(Q^\sharp)$

$((l,b),\mu^\sharp) \in Steps^\sharp$ は $(l,\nu) \in \gamma((l,b))$ であるような $((l,\nu),\mu) \in Steps$ が具体構造上に存在するときに限り，抽象構造上で構築される．ここで，$\mu^\sharp$ は $\mu^\sharp((l',b')) = \mu((l',\nu'))$ である確率分布とする．

定義 16 より次の定理が導き出される．

定理 1 （抽象遷移と具体遷移の関係）

具体構造 $\mathcal{M}$ と遷移関係 $Steps$，および対応する抽象構造 $\mathcal{M}^\sharp$ と対応する遷移関係 $Steps^\sharp$ において，以下の式が成り立つ．

1. $(l,\nu) \subseteq \gamma(Steps^\sharp)$
2. $\alpha(Steps) \subseteq Steps^\sharp$

二つのリージョン等価関係の同値クラスを全て区別して表現できるような抽象化述語の集合を basis と呼ぶ．以下にその定義を示す．

定義 17 （*basis*）

PTA をクロック集合 $\mathcal{X}$ を持つ確率オートマトンであるとし，Ψ^{all} を抽象化述語の集合とする．Ψ^{all} が PTA における *basis* であるとは，全てのクロック評価 $\nu_1,\nu_2 \in \mathcal{V}_\mathcal{X}$ において

$$\forall\psi^l \in \Psi^{all}.\nu_1 \mid\approx \psi^l \Leftrightarrow \nu_2 \mid\approx \psi^l$$

であることと同値である．

3.2 並列同期システムに対する反例による抽象化の精錬

確率時間システムに対する反例による抽象化の精錬は，述語抽象化とその精錬による確率時間オートマトンの到達可能性解析手法 [4] によって示されている．ここで，[4] を拡張して，システム全体に対して確率到達可能性解析を行うことで得られる目標ロケーションへの反例を，ノードに対応する形で分割し，その上で反例解析を行うことでノード単位での反例解析を行い精錬時にシステムのローカルな部分に対する精錬を行うことで効率的な検証を実現する．

3.2.1 確率到達可能性解析

確率到達可能性解析は，システムが目標ロケーション l_{target} へ到達確率 λ より大きい確率で到達できなければ"Not Reachable"を出力し，到達できれば"Reachable"とそのロケーションへのパス (反例) を出力する．反例は抽象構造上での初期状態から目標ロケーションへのパスの集合 $\Omega_{reach}^\sharp$ として与えられる．ここで，$\Omega_{reach}^\sharp$ の要素であるパスを集めて，その合計到達確率が λ より大きくなる組み合わせのうち要素数が最小となる集合の中で確率最大のものを最小反例 $\Omega_{smallest}^\sharp \subseteq \Omega_{reach}^\sharp$ [5] と呼び，これを実際の反例として用いる．これは，次に続く反例解析の段階における計算量を削減するためである．

3.2.2 反例解析

定理1より, 抽象構造における遷移は具体構造における遷移のオーバー近似であるため, 具体構造に存在する反例は全て抽象構造における反例に含まれるが, その逆は必ずしも成り立たない. 言い換えると, 反例 $\Omega^{\sharp}_{smallest}$ に従った動作が, 具体構造上では実行不可能であることが起こりうる. そのため, 抽象構造に対する確率到達可能性解析によって得られた反例が具体構造上に存在するかどうかの判定を行う.

反例解析では, まず得られた反例 $\Omega^{\sharp}_{smallest}$ から反例の要素 $\omega^{\sharp}$ を一つ取り出し, それを合成前のモデル要素に対応する形で分解する. 次に分解した反例がそれぞれ対応する実際のシステム上で実行可能であるかを調べるパス反例解析を行い, それらの反例が並列同期システム上で同期しているかを調べる同期判定を行う. これらを $\Omega^{\sharp}_{smallest}$ が空になるまで繰り返し, その後それらの反例の要素が同一のアドバサリに従って実行することができるかを調べるアドバサリ反例解析を行う. 次に, これらの手順についての詳細な説明を加える.

(i) パス分割

反例のローカルな特性に対応するために, まず, 反例 $\Omega^{\sharp}_{smallest}$ から要素 $\omega^{\sharp}$ を一つ取り出し, 合成前のモデルに対応する形で分割する. 具体構造が $\mathcal{M} = \mathcal{M}_1 \otimes \cdots \otimes \mathcal{M}_n$ という具体構造の合成の形で表わされる時, 反例の要素であるパスは $\omega^{\sharp} = (q_0^1, \cdots, q_0^n) \rightarrow (q_1^1, \cdots, q_0^n) \rightarrow \cdots$ と表せられる. このとき, パス分割はパス $\omega^{\sharp}$ をモデルに対応するパス $\omega_i^{\sharp} = q_0^i \rightarrow q_1^i \rightarrow \cdots$, $1 \leq i \leq n$ に分解する操作を行う.

(ii) パス反例解析

分割された反例の要素が, 対応する合成前のモデル上で実際に実行可能であるかを調べる. ここでは前段階で得られた分割されたパス $\omega_i^{\sharp}$, $1 \leq i \leq n$ がそれぞれ対応する合成前の具体構造 $\mathcal{M}_i$, $1 \leq i \leq n$ 上で実行可能であるかを, 抽象パスの終端から対応する具体構造を逆に辿って解析する後方反例解析手法を用いて検証する.

(iii) 同期判定

それぞれの実行可能性を調べた反例の要素が, システム上で実際に同期動作するのかを判定する. パス反例解析により, 分割した反例の要素それぞれの単独での実行可能性は検証できたとして, それらがシステム全体で同期して実行できるのか, WSNの例で言い換えるとノード単独の動作が, 他のノードと同期して行えるのかを判定する.

(iv) アドバサリ反例解析

(i)〜(iii) を繰り返したのち, 解析された反例が同一のアドバサリ条件下で実行可能なのかを検証する. 具体構造上のある状態において, 時間遷移と離散遷移のどちらを選ぶかは非決定的である. このときアドバサリが与えられることによって非決定性が解決され, 結果として状態遷移列であるパスが与えられる. 一方で抽象構造においては時間が抽象化されているため, 到達可能性解析で得られた抽象反例に含まれる抽象パスが具体構造上では同時に実行できない可能性がある. そのため, このアドバサリ判定解析の段階において得られた反例についてアドバサリが同一であることを調べる.

3.2.3 精錬

反例解析で偽反例と判定された場合, その反例が存在しないように述語を追加して抽象状態を分割する精錬をおこなう. 精錬を行うために必要な情報は, 前段階の反例解析の結果から得る.

1) パス反例解析

パス反例解析において反例が偽反例となる場合は, 反例 $\Omega^{\sharp}$ の要素であるパス $\omega^{\sharp}$ を分割して得られるいずれかの分割抽象パス $\omega_i^{\sharp}$ が具体構造 $\mathcal{M}_i$ 上で実行不可能である時である. 言い換えると, 分割抽象パス $\omega_i^{\sharp}$ に対応するパスが具体構造上に存在しないことを意味する. このとき, 少なくとも $\omega_i^{\sharp}$ 内の一つの遷移 $q^{\sharp} \rightarrow q'^{\sharp}$ に対応する具体構造上の遷移 $q \rightarrow q'$ は, 遷移可能条件もしくは q' における不変条件を満たさないため遷移不可能であることが言える. しかし, 抽象構造上においては到達可能性解析から実行可能である. これは, ロケーションの遷移可能状態と不可能状態が抽象化によって同一視されているために起きる. 故に抽象状態を述語によって遷移可能な状態と不可能な状態に分割することで反例は実行不可能となる. この状態を分割する述語は, q でのゾーン, 及び遷移可能条件, あるいは q' の不変条件から選択する.

2) 同期判定

同期判定で反例が偽となる時は, 分割抽象パスには対応する具体パスが存在するが, 合成抽象パスには対応する具体パスが存在しないという場合である. つまり, 分割抽象パスの中にクロック同期がとれていないものが存在することを示す. このとき, ある具体状態 (q_i, q'_j) において少なくとも二つの分割具体パス ω, ω' の遷移対 $(q_i \rightarrow q_{i'}, q'_j \rightarrow q'_{j'}), i \leq$

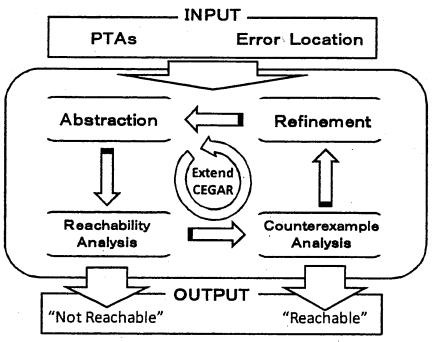

図 2: 拡張 CEGAR のモデル図

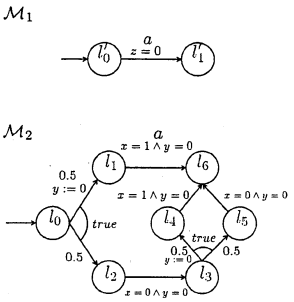

図 3: $PTA\ \mathcal{M}_1, \mathcal{M}_2$

$i', j \leq j'$ における出発時ゾーンの連言 $\zeta \wedge \zeta'$ は空となる．よってこの二つの状態を分割するように $\zeta \wedge \zeta'$ から述語を選択し精錬する．

3) アドバサリ反例解析

アドバサリ反例解析における判定で偽反例が見つかったとき，ある抽象状態 $q^\sharp$ において異なる遷移先が選択されていることになる．つまり，これを言い換えると，ある状態において時間遷移，離散遷移の異なる遷移が行われているということになる．よって，$q^\sharp$ を時間遷移に関して 2 つに分割すれば，時間遷移と離散遷移の競合不可能となる．したがって，この境界を示す時間条件を述語として追加する．

述語が追加された新たな抽象構造では，さきほどの偽反例は実行不能となる．よって，これを繰り返していくことにより，最終的に正しい確率を計算可能な抽象構造を構築することができる．

4 拡張 CEGAR

述語抽象化，反例による精錬を自動的に検証に適応していく手法が CEGAR の枠組み [6] である．ここでは並列同期システムの検証を目的とした形に CEGAR を拡張した拡張 CEGAR の動作について説明する．まず，拡張 CEGAR による，到達可能性解析のモデル図を図 2 に示す．

1. Abstraction: 初期述語集合 Ψ^{init} から初期合成抽象構造 $\mathcal{M}^\sharp_{\Psi^{init}}$ を構築する．

2. Reachability Analysis: $\mathcal{M}^\sharp_\Psi$ 上で目的となるロケーションへの最大到達可能性確率を計算する．

3. Counterexample Analysis: 2. で目的ロケーションへ到達した反例 $\Omega^\sharp_{smallest}$ の要素をそれぞれについて分割し，パス反例解析，同期判定，再結合してアドバサリ反例解析を行って具体構造 $\mathcal{M}$ 上で到達可能かどうかを解析する．

4. Refinement: 3. の反例解析の結果より，2. で得られた反例が存在しないように抽象状態を分割する述語集合 Ψ^{new} を得る．

5. Abstraction: 述語が追加された述語集合 $\Psi' = \Psi \cup \Psi^{new}$ から新たな合成抽象構造 $\mathcal{M}^\sharp_{\Psi'}$ を得る．

6. 2. に戻る．

このループを繰り返していくことにより，システムが目的ロケーションに"Reach"か，あるいは"Not Reach"かを判定する．"Reach"ならば，目的ロケーションへの具体パスが与えられ，この情報をもとにシステムの仕様を変更，改良することが可能となる．

4.1 検証例

本節では，例を通して拡張 CEGAR の検証手順を示す．入力は図 3 に示した具体構造 $\mathcal{M}_1, \mathcal{M}_2$ と目的ロケーション $l_6 \in L_2$，目標到達確率 $\lambda = 0.3$ である．

まず初めに，入力された具体構造を合成し合成具体構造図 4 を得た後，初期述語集合 Ψ^{init} に従って初期抽象化を行う．次に得られた合成抽象構造 $\mathcal{M}^\sharp$ に対して目標確率 λ での確率到達可能性解析を行う．この結果，反例の候補として $(A^\sharp, \Omega^\sharp)$ が得られる．ここから最小反例 $\Omega^\sharp_{smallest}$ を選択する．$\Omega^\sharp_{smallest}$ の要素は $\omega^\sharp = (l_0, l_0') \rightarrow (l_1, l_0') \rightarrow (l_6, l_0')$ のみからなる．この反例に対して反例解析を行う．ま

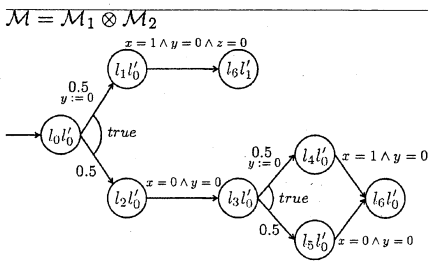

図 4: $PTA\ \mathcal{M}$

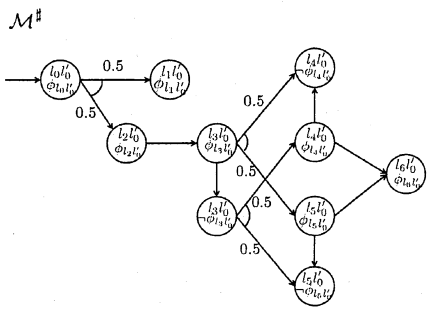

図 5: 精錬後の合成抽象構造

ず, 反例の分解を行い分割反例 $\omega_1^\sharp = l_0 \to l_1 \to l_6, \omega_2^\sharp = l_0' \to l_0' \to l_1'$ を得る. これらはパス反例解析において単独で実行可能であるが, 同期判定の段階で遷移 $l_1 \to l_6$ と遷移 $l_0' \to l_1'$ 間で同期をとることができないと判定される. よって述語 $\psi_{l_1 l_0'} = z \leq 0$ を用いて抽象状態 (l_1, l_0') を分割することで抽象構造 $\mathcal{M}^\sharp$ を精錬し, 再度抽象構造を構築する. その結果として, 抽象状態 (l_6, l_1') へ到達不可能となるためこの反例は反例の候補から除かれる.

次に, 到達可能性解析を行うと前段階での反例が除かれた新しい反例の候補 $(A'^\sharp, \Omega'^\sharp)$ が得られる. 最小反例 $\Omega^\sharp_{smallest}$ は $\Omega'^\sharp$ に等しく, 反例の要素となるパスは, $\omega_1^\sharp = (l_0, l_0') \to (l_2, l_0') \to (l_3, l_0') \to (l_4, l_0') \to (l_6, l_0')$ と $\omega_2^\sharp = (l_0, l_0') \to (l_2, l_0') \to (l_3, l_0') \to (l_5, l_0') \to (l_6, l_0')$ である. これらはパス反例解析, 同期判定においては偽反例と判定されないが, アドバサリ反例解析において同時実行できないことが示され, 偽反例であると判定される. これはロケーション (l_3, l_0') において 1 単位時間経過すると $\omega_1^\sharp$ に対応する具体パス ω_1 が実行可能であるが, このとき $\omega_2^\sharp$ に対応する具体パス ω_2 は実行不可能であり逆もまた同様である. よって, これらのパスは同一のアドバサリによって実行できないため反例の候補 $(A'^\sharp, \Omega'^\sharp)$ は偽反例となる. この時, この反例が到達可能性解析に再び現れないように, 述語によって抽象ロケーション (l_3, l_0') を時間経過した状態としない状態に分割する.

以上の様に, 拡張 CEGAR に従って検証手順を踏んでいくことにより, 最終的に目的ロケーション l_6 に目標到達確率 $\lambda = 0.3$ で到達するパスは合成抽象構造図 5 に存在しないこととなる, 言い換えると"Not Reachable"と言う検証結果が得られる.

5 まとめ

本論文は, 無線センサネットワークを例とする並列動作する確率時間システムの特性検証について, 述語抽象化とその精錬の枠組みを拡張し導入することで, 効率的な検証手法を考案した.

今後の課題としては, WSNs における, センサノードの時間的な生成消滅や通信状況の変化による通信路変化等の動的な特性まで含めた検証法の確立が考えられる.

参考文献

[1] M. Kwiatkowska, G. Norman, R. Segala, and J. Sproston. Automatic verification of real-time systems with discrete probability distributions. *TCS*, Vol. 282, pp. 101–150, 2002.

[2] A. Demaille, T. Herault, and S. Peyronnet. Probabilistic verification of sensor networks. *Research, Innovation and Vision for the Future*, pp. 45– 54, 2006.

[3] M Kwiatkowska, G Norman, J Sproston, and F Wang. Symbolic model checking for probabilistic timed automata. *Information and Computation*, Vol. 205, pp. 1027–1077, 2007.

[4] 駒形, 森下, 山根. 述語抽象化とその洗練による確率時間オートマトンの到達可能性 解析手法. 信学技報, CST2008-5, pp. 1–6, 2008.

[5] T. Han and J. P. Katoen. Counterexamples in probabilistic model checking. *LNCS*, Vol. 4424, pp. 72–86, 2007.

[6] E. M. Clarke. Counterexample-guided abstraction refinement. *LNCS*, Vol. 1855, pp. 154–169, 2000.