社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
SS2007-16 (2007-08)

左線形な定向条件付き項書換え系における到達可能な項集合の近似集合を認識する木オートマトン

村田 俊樹[†] 西田 直樹[††] 酒井 正彦[††] 坂部 俊樹[††] 草刈 圭一朗[††]

†,†† 名古屋大学大学院情報科学研究科
〒 464–8603 名古屋市千種区不老町
E-mail: †murata@trs.cm.is.nagoya-u.ac.jp, ††{nishida,sakai,sakabe,kusakari}@is.nagoya-u.ac.jp

あらまし 項書換え系（TRS）の到達可能性問題とは，与えられた 2 つの項の一方からもう一方に TRS での書換えにより到達できるか否かという問題であり，一般には決定不能である．そこで，到達可能な項集合（の近似集合）を認識する木オートマトンの生成法が広く研究されている．条件付き TRS（CTRS）に関しては，左線形な結合 CTRS について生成手続きが提案されているが，木オートマトンの自動生成の基準となる近似関数は CTRS について提案されていない．本稿では，結合 CTRS を含むクラスである定向 CTRS に着目し，左線形な定向 CTRS で到達可能な項集合の近似集合を認識する木オートマトンの生成手続きを提案する．さらに，結合 CTRS と定向 CTRS のための近似関数，結合 CTRS を定向 CTRS で表現する方法を提案し，結合 CTRS についての生成手続きと比較する．
キーワード 到達可能性，木オートマトン，簡約オートマトン，結合条件付き項書換え系

Recognizable Approximation of Descendant Sets for Left-Linear Oriented Conditional Term Rewriting Systems

Toshiki MURATA[†], Naoki NISHIDA[††], Masahiko SAKAI[††],
Toshiki SAKABE[††], and Keiichirou KUSAKARI[††]

†,††Graduate School of Information Science, Nagoya University
Furo–cho, Chikusa–ku, Nagoya, 464–8603, Japan
E-mail: †murata@trs.cm.is.nagoya-u.ac.jp, ††{nishida,sakai,sakabe,kusakari}@is.nagoya-u.ac.jp

Abstract The reachability problem for term rewriting systems (TRSs) is to decide whether one of two given terms is reachable to the other with respect to rewriting by a given TRS. Unfortunately, this problem is undecidable in general. Hence, several methods have been widely studied in order to construct a tree automaton that accepts all terms reachable from terms in a given recognizable set. In conditional rewriting, such methods have been proposed only for left-linear join conditional TRSs. However, the techniques of approximation functions used in automatically constructing tree automata have not been formalized. In this paper, we focus on oriented conditional TRSs since every join conditional TRS can be simulated completely by an oriented one. We first propose a method that, given a left-linear oriented conditional TRS and a recognizable set of terms, constructs a tree automaton recognizing an over-approximated set of reachable terms. Then, we formalize approximation functions for join conditional TRSs and oriented ones, respectively, and compare the constructions for join and oriented systems.
Key words reachability, tree automaton, reduction automaton, join CTRS

1. はじめに

項書換えの到達可能性問題とは，与えられた 2 つの項の一方（初期項）からもう一方（標的項）に項の書換えにより到達できるか否かという問題である．項書換え系（TRS）におけるこの問題の判定には，初期項から到達可能な項すべてからなる集合を認識する木オートマトンを生成し，標的項がこの木オートマトンで受理されるか否かで行う手法が広く研究されている [6] [7] [8] [9]．しかし，この問題は一般には決定不能であり上述した木オートマトンを生成できるとは限らない．そのた

め，初期項から到達可能なすべての項を含む近似集合を認識する木オートマトン（近似木オートマトン）を生成し，標的項がこの近似木オートマトンで受理されなければ，到達しないことを保証する手法も広く研究されている [2] [10] [11]．条件付き TRS（CTRS）に関しては，左線形な結合 CTRS について近似木オートマトンの生成手続き [1] が提案されているが，近似木オートマトンの生成の基準となる近似関数は CTRS について提案されておらず，人間が発見的に遷移規則を追加しなければならない．この近似関数は遷移規則の追加のための状態の割り当て方を定めるもので，これを近似木オートマトンの生成手続きに入力として与えることで遷移規則の追加を機械的に行うことが可能になる．

本稿では，結合 CTRS を表現できるクラスである定向 CTRS に着目し，[1] の手法を定向 CTRS について拡張，すなわち，左線形な定向 CTRS で到達可能なすべての項を含む近似集合を認識する木オートマトンの生成手続きを提案する．また，生成する木オートマトンが近似木オートマトンであることを証明することで，その正しさを示す．さらに，結合 CTRS と定向 CTRS のための近似関数，結合 CTRS を定向 CTRS で表現する方法を提案し，結合 CTRS についての生成手続きと比較する．具体的には，左線形な結合 CTRS と結合 CTRS のための近似関数が与えられたとき，結合 CTRS についての生成手続きで生成した近似木オートマトンの認識集合と，その結合 CTRS を定向 CTRS に変換し定向 CTRS についての生成手続きで生成する近似木オートマトンの認識集合が等しくなるような定向 CTRS のための近似関数を結合 CTRS のための近似関数から生成する方法を提案する．

本稿は次のように構成される．2. では項書換え系と木オートマトンに関する記法を説明する．3. では定向 CTRS についての近似木オートマトンの生成手続きを提案し，その正しさを示す．4. では CTRS のための近似関数を提案する．5. では定向 CTRS についての近似木オートマトンの生成手続きと結合 CTRS についての近似木オートマトンの生成手続きとの比較を行う．6. では今後の課題について議論する．

2. 準　　備

本稿では，項書換え系と木オートマトンの一般的な記法に従う [3] [4] [5]．

関数記号の集合 $\mathcal{F}$，変数の可算無限集合 $\mathcal{X}$ から構成されるすべての項の集合を $\mathcal{T}(\mathcal{F},\mathcal{X})$ と記す．基底項の集合 $\mathcal{T}(\mathcal{F},\emptyset)$ は $\mathcal{T}(\mathcal{F})$ と略記する．項 t 中のどの変数も 1 回しか現れないとき，t は線形であるという．項 $t_1,\cdots,t_n$ のいずれかに現れる変数の集合を $\mathcal{V}ar(t_1,\cdots,t_n)$ と記す．項 s と項 t が同一であるとき，$s \equiv t$ と記す．項 t の位置の集合を $\mathcal{P}os(t)$ と記す．位置 p にある t の部分項を $t|_p$ と記す．項 t の先頭（位置 ϵ）以外の関数記号の位置の集合を $\mathcal{P}os_{\mathcal{F}}(t)$ と記す．

ホール $\Box$ を 1 つ持つ文脈の集合を $\mathscr{C}(\mathcal{F},\mathcal{X})$ と記す．文脈 $C[\,]$ に出現する $\Box$ を項 t で置き換えることによって得られる項を $C[t]$ と記す．特に，文脈 $C[\,]$ における $\Box$ の出現位置 p を明記するときは $C[\,]_p$ と記す．

代入 σ の定義域，値域はそれぞれ $\mathcal{D}om(\sigma)$，$\mathcal{R}an(\sigma)$ とする．$\mathcal{D}om(\sigma)=\{x_1,\cdots,x_n\}$，$\sigma(x_i)\equiv t_i$ のとき，σ を $\{x_1\mapsto t_1,\cdots,x_n\mapsto t_n\}$ と記す．項 t に対して，$\sigma(t)$ を t の例と呼び，特に，$\sigma(t)\in\mathcal{T}(\mathcal{F})$ のとき，$\sigma(t)$ を t の基底例と呼ぶ．また，$\sigma(t)$ は $t\sigma$ と略記する．

$\mathcal{F}$ 上の結合条件付き書換え規則（join conditional rewrite rule）は次の条件を満たす 3 つ組 $(l,r,Cond)$ であり，$l\to r\Leftarrow Cond$ と記す：左辺 l は変数でない $\mathcal{T}(\mathcal{F},\mathcal{X})$ の項，右辺 r は $\mathcal{T}(\mathcal{F},\mathcal{X})$ の項であり $\mathcal{V}ar(l)\supseteq\mathcal{V}ar(r)$ を満たし，条件部 $Cond$ は $s_1\downarrow t_1,\cdots,s_k\downarrow t_k$ の形式をした結合条件 $s_i\downarrow t_i$ の列で，s_i,t_i は $\mathcal{T}(\mathcal{F},\mathcal{X})$ の項であり $\mathcal{V}ar(s_i,t_i)\subseteq\mathcal{V}ar(l)$ を満たす．$k=0$ のときは単に書換え規則といい，$l\to r$ と記す．規則には他の規則と区別するためのラベル ρ を用いて $\rho: l\to r\Leftarrow Cond$ と書くこともあり，略記するために ρ のみを記述してもよい．

$\mathcal{F}$ 上の結合条件付き書換え規則の有限集合 $\mathcal{R}_\downarrow$ の n レベルの書換え関係 $\underset{n}{\to}_{\mathcal{R}_\downarrow}$ は次のように再帰的に定義される．

(1) $\underset{0}{\to}_{\mathcal{R}_\downarrow}=\emptyset$

(2) $\underset{n+1}{\longrightarrow}_{\mathcal{R}_\downarrow}=\{(C[l\sigma]_p,C[r\sigma]_p)\mid \rho: l\to r\Leftarrow s_1\downarrow t_1,\cdots,s_k\downarrow t_k\in\mathcal{R}_\downarrow,\ C\in\mathscr{C}(\mathcal{F},\mathcal{X}),\ (\forall i.\ s_i\sigma \underset{n}{\overset{*}{\to}}_{\mathcal{R}_\downarrow}\underset{n}{\overset{*}{\leftarrow}}_{\mathcal{R}_\downarrow} t_i\sigma)\}$

$\mathcal{R}_\downarrow$ の書換え関係 $\to_{\mathcal{R}_\downarrow}$ は $\bigcup_{n\ge 0}\underset{n}{\to}_{\mathcal{R}_\downarrow}$ と定義する．結合条件付き項書換え系（結合 CTRS）は抽象書換え系 $(\mathcal{T}(\mathcal{F},\mathcal{X}),\to_{\mathcal{R}_\downarrow})$ であり，単に $\mathcal{R}_\downarrow$ を用いて表現する．$\mathcal{R}_\downarrow$ が条件部を持たない規則のみからなるとき，$\mathcal{R}_\downarrow$ は項書換え系（TRS）である．

$\mathcal{F}$ 上の定向条件付き書換え規則（oriented conditional rewrite rule）は，結合条件付き書換え規則 $l\to r\Leftarrow Cond$ の $Cond$ の形式が $s_1\to t_1,\cdots,s_k\to t_k$ に変更されたもので，$\mathcal{V}ar(l)\supseteq\mathcal{V}ar(r)$, $\mathcal{V}ar(s_i,t_i)\supseteq\mathcal{V}ar(l)$ の制約を満たす必要はない．$\mathcal{F}$ 上の定向条件付き項書換え系（定向 CTRS）は，結合条件付きの場合の $\underset{n+1}{\longrightarrow}_{\mathcal{R}_\downarrow}$ の定義の $s_i\sigma \underset{n}{\overset{*}{\to}}_{\mathcal{R}_\downarrow}\underset{n}{\overset{*}{\leftarrow}}_{\mathcal{R}_\downarrow} t_i\sigma$ を $s_i\sigma \underset{n}{\overset{*}{\to}}_{\mathcal{R}_\downarrow} t_i\sigma$ に置き換えて定まる抽象書換え系である．

条件付き書換え規則 $l\to r\Leftarrow Cond$ において，右辺もしくは条件部のみに現れる変数 $x\in\mathcal{V}ar(r,Cond)\setminus\mathcal{V}ar(l)$ を余剰変数と呼ぶ．CTRS $\mathcal{R}$ のすべての規則において，余剰変数が右辺には現れない，つまり $\mathcal{V}ar(r)\subseteq\mathcal{V}ar(l)$ を満たす $\mathcal{R}$ を 2-CTRS と呼ぶ．条件付き書換え規則 $\rho: l\to r\Leftarrow Cond$ 中に現れる変数の集合を $\mathcal{V}ar(\rho)$ で表す．

項の集合 E から $\overset{*}{\to}_{\mathcal{R}}$ で到達できるすべての項の集合を $\mathcal{R}^*(E)$（つまり，$\mathcal{R}^*(E)=\{t\mid t\in\mathcal{T}(\mathcal{F}),\ \exists s\in E.s\overset{*}{\to}_{\mathcal{R}}t\}$）で表す．

条件付き書換え規則の左辺（右辺）が線形であるとき，その規則は左線形（右線形）であるという．CTRS $\mathcal{R}$ のすべての条件付き書換え規則の左辺（右辺）が線形であるとき，$\mathcal{R}$ は左線形（右線形）であるという．

等式（不等式）制約とは，$\pi=\pi'(\pi\neq\pi')$ の形式をした $\mathcal{T}(\mathcal{F})$ 上の述語である（ただし，$\pi,\pi'\in\{1,\cdots,k\}^*$）．$\pi,\pi'\in\mathcal{P}os(t)$ かつ $t|_\pi=t|_{\pi'}(t|_\pi\neq t|_{\pi'})$ のとき，項 t が $\pi=\pi'(\pi\neq\pi')$ を満たすといい，$t\models\pi=\pi'(t\models\pi\neq\pi')$ と書く．

状態の有限集合を $\mathcal{Q}$ で表す．遷移規則とは，$t\in\mathcal{T}(\mathcal{F}\cup\mathcal{Q})$, $q\in\mathcal{Q}$, c は等式（不等式）制約のブール式であることを満たす 3

つ組 (t,q,c) であり，$t \xrightarrow{c} q$ と記す．遷移規則が制約を持たないとき，$t \to q$ と略記する．正規化された遷移規則とは，$f(q_1,\cdots,q_n) \xrightarrow{c} q$ の形式（$f \in \mathcal{F}$, $q_1,\cdots,q_n \in \mathcal{Q}$）をした遷移規則であり，単に遷移規則と呼ぶこともある．

遷移規則の正規化とは，その規則の左辺項の状態でないすべての真部分項に状態を割り当て，正規化された遷移規則に分解することである．例えば，遷移規則 $f(g(q', f(a))) \to q$（$\mathcal{F} = \{f, g, a\}$, $\mathcal{Q} = \{q, q'\}$）を状態の列 q_1, q_2, q_3 で正規化するとき，$f(g(q', f(a)))$ の状態でない真部分項 $g(q', f(a))$, $f(a)$, a にそれぞれ q_1, q_2, q_3 を割り当て（それぞれの真部分項の位置が辞書式順序で小さい順に，列の前の状態から割り当てる），正規化された遷移規則に分解すると $\{f(q_1) \to q,\ g(q', q_2) \to q_1,\ f(q_3) \to q_2,\ a \to q_3\}$ となる．

等式不等式制約付き木オートマトン（AWEDC）とは，関数記号の集合 $\mathcal{F}$，状態の有限集合 $\mathcal{Q}$，最終状態の集合 $\mathcal{Q}_f$ ($\mathcal{Q}_f \subseteq \mathcal{Q}$)，正規化された遷移規則の集合 $\triangle$ からなる 4 つ組 $\langle \mathcal{F}, \mathcal{Q}, \mathcal{Q}_f, \triangle \rangle$ である．AWEDC のすべての遷移規則が等式（不等式）制約を持たない（遷移規則が $f(q_1,\cdots,q_n) \to q$ の形式をしている）とき，その AWEDC は木オートマトンである．簡約オートマトンとは，その状態が順序付けされており，すべての遷移規則 $f(q_1,\cdots,q_n) \xrightarrow{c} q$ において c が等式制約を持つならば q はどの q_i よりも真に小さいことを満たす AWEDC である．

$\mathcal{A}$ を木オートマトンとすると（$\mathcal{A} = \langle \mathcal{F}, \mathcal{Q}, \mathcal{Q}_f, \triangle \rangle$ とする），書換え関係 $\to_{\mathcal{A}}$ は $\triangle$ を書換え規則とみなした $\mathcal{T}(\mathcal{F} \cup \mathcal{Q})$ 上の書換え関係 $\to_{\triangle}$ で定義される．$\to_{\mathcal{A}}$ を $\to_{\triangle}$ と書くこともある．$\to_{\mathcal{A}}$ の推移的反射的閉包は $\to^*_{\mathcal{A}}$ と記す．

基底項 t ($t \in \mathcal{T}(\mathcal{F})$) が AWEDC $\mathcal{A}$ ($\mathcal{A} = \langle \mathcal{F}, \mathcal{Q}, \mathcal{Q}_f, \triangle \rangle$) によって $t \to^*_{\mathcal{A}} q$ となる $q \in \mathcal{Q}_f$ が存在するとき，$\mathcal{A}$ は t を受理するという．$\mathcal{A}$ が受理する項の集合を $\mathcal{L}(\mathcal{A})$ と記す．$L = \mathcal{L}(\mathcal{A})$ のとき，$\mathcal{A}$ は L を認識するという．$t \to^*_{\mathcal{A}} q \in \mathcal{Q}$ となる基底項 $t \in \mathcal{T}(\mathcal{F})$ の集合を $\mathcal{L}(\mathcal{A}, q)$ と記す．つまり，$\mathcal{L}(\mathcal{A}) = \bigcup_{q \in \mathcal{Q}_f} \mathcal{L}(\mathcal{A}, q)$ となる．

状態代入 $\sigma_{\mathcal{Q}}$ とは，その値域が $\mathcal{R}an(\sigma_{\mathcal{Q}}) \subseteq \mathcal{Q}$ となる代入である．状態代入の集合を $\Sigma(\mathcal{Q}, \mathcal{X})$ と記す．

木オートマトン $\mathcal{A}$ と基底項の集合 T に対して，$T \subseteq \mathcal{L}(\mathcal{A})$ のとき，$\mathcal{A}$ を T を認識する近似木オートマトン（単に近似木オートマトン）と呼ぶ．

以下の命題は，簡約オートマトンの性質を表すものである．

［命題 2.1］（[5]） (1) すべての項に対してその項の例を部分項に持つ項の集合を認識する簡約オートマトンが存在する．(2) 簡約オートマトンのクラスは積集合に閉じている．(3) 空問題は簡約オートマトンのクラスで決定可能である． □

項 t の例の集合を $G(t)$ と記す．$G(t)$ は簡約オートマトンで認識できる．

3. 定向 CTRS の近似木オートマトンの生成

本節では，定向 CTRS についての近似木オートマトンの生成手続き，すなわち，左線形な定向 CTRS で到達可能なすべての項を含む近似集合を認識する木オートマトンの生成手続きを提案し，その正当性を示す．なお，この手続きは結合 CTRS のための手続き [1] に変更を加えたものである．

まずは，左線形な定向 CTRS で到達可能なすべての項を含む近似集合を認識する木オートマトンの生成手続きを提案する．具体的には，初期項を含む集合 E を認識する木オートマトンを $\mathcal{A}_0$ とし，$\mathcal{A}_i$ から $\mathcal{A}_{i+1}$ を生成するアルゴリズムを以下に記す．ただし，今回考える左線形な定向 CTRS はその各規則 $l \to r \Leftarrow s_1 \to t_1, \cdots, s_n \to t_n$ が $\mathcal{V}ar(s_1,\cdots,s_n) \subseteq \mathcal{V}ar(l)$ を満たす定向 2-CTRS とする．このような定向 CTRS の規則の条件部 $s_1 \to t_1, \cdots, s_n \to t_n$ は 1 つにまとめることができる（例えば，新しい関数記号 tp_n を使って $\mathsf{tp}_n(s_1,\cdots,s_n) \to \mathsf{tp}_n(t_1,\cdots,t_n)$）ため，定向 CTRS の規則の条件部の条件は高々1 つであるとしても，一般性を失わない．これは，結合 CTRS についても同様である．以降では，条件は高々1 つとして議論する．

入力：左線形な定向 CTRS $\mathcal{R}_{\to}$
木オートマトン $\mathcal{A}_0 = \langle \mathcal{F}, \mathcal{Q}_0, \mathcal{Q}_f, \triangle_0 \rangle$
（ただし，$s\sigma_{\mathcal{Q}}$ を指標付けした状態 $q_{s\sigma_{\mathcal{Q}}}$ は $q_{s\sigma_{\mathcal{Q}}} \notin \mathcal{Q}_0$ を満たす(注1)）

出力：木オートマトン $\mathcal{A}_* = \langle \mathcal{F}, \mathcal{Q}_*, \mathcal{Q}_f, \triangle_* \rangle$

手順：$i = 0$ とし，$\mathcal{A}_i = \mathcal{A}_{i+1}$ になるまで以下を繰り返す．

1. $\mathcal{Q}_{i+1} := \mathcal{Q}_i$, $\triangle_{i+1} := \triangle_i$ とする．
2. $\mathcal{R}_{\to}$ の各規則 $\rho : l \to r \Leftarrow s \to t$ に対して(注2) $l\sigma_{\mathcal{Q}} \to^*_{\triangle_i} q \in \mathcal{Q}_i$ かつ $r\sigma_{\mathcal{Q}} \not\to^*_{\triangle_i} q$ を満たすすべての $(q, \sigma_{\mathcal{Q}})$ について (i),(ii) のどちらかを行う．ただし，$\{x_1,\cdots,x_k\} = \mathcal{V}ar(l)$, $\sigma_{\mathcal{Q}} = \{x_1 \mapsto q_{i_1}, \cdots, x_k \mapsto q_{i_k}\}$, $\mathcal{R}an(\sigma_{\mathcal{Q}}) \subseteq \mathcal{Q}_i$ とする．
   (i) $s\sigma_{\mathcal{Q}}$ を指標付けした状態 $q_{s\sigma_{\mathcal{Q}}}$ が $\mathcal{Q}_i$ に含まれている場合(注3)（もしくは ρ に条件部がない場合）．
   $\mathcal{L}(\mathcal{A}_i, q_{s\sigma_{\mathcal{Q}}}) \cap G(t) \stackrel{?}{=} \emptyset$ (注4) を判定(注5)し，これが偽なら，$r\sigma_{\mathcal{Q}} \to^*_{\triangle_{i+1}} q$ となるように $r\sigma_{\mathcal{Q}} \to q$ を正規化した遷移規則を $\triangle_{i+1}$ に追加し（ρ に条件部がない場合は判定することなく追加する），その際使用した状態を $\mathcal{Q}_{i+1}$ に追加する．
   (ii) それ以外の場合．
   状態 $q_{s\sigma_{\mathcal{Q}}}$ を $\mathcal{Q}_{i+1}$ に追加し，$s\sigma_{\mathcal{Q}} \to^*_{\triangle_{i+1}} q_{s\sigma_{\mathcal{Q}}}$ となるように $s\sigma_{\mathcal{Q}} \to q_{s\sigma_{\mathcal{Q}}}$ を正規化した遷移規則を $\triangle_{i+1}$ に追加し(注6)，その際使用した状態を $\mathcal{Q}_{i+1}$ に追加する．
3. $\mathcal{A}_{i+1} = \langle \mathcal{F}, \mathcal{Q}_{i+1}, \mathcal{Q}_f, \triangle_{i+1} \rangle$, $i := i + 1$ とする． □

(注1)：結合 CTRS の手続き [1] の場合，“$q_{c_l\sigma_{\mathcal{Q}}}, q_{c_r\sigma_{\mathcal{Q}}} \notin \mathcal{Q}_0$” である．
(注2)：[1] の場合，“$\rho : l \to r \Leftarrow c_l \downarrow c_r$” である．
(注3)：[1] の場合，“$c_l\sigma_{\mathcal{Q}}, c_r\sigma_{\mathcal{Q}}$ を指標付けした状態 $q_{c_l\sigma_{\mathcal{Q}}}, q_{c_r\sigma_{\mathcal{Q}}}$ が両方 $\mathcal{Q}_i$ に含まれている場合” である．
(注4)：[1] の場合，“$\mathcal{L}(\mathcal{A}_i, q_{c_l\sigma_{\mathcal{Q}}}) \cap \mathcal{L}(\mathcal{A}_i, q_{c_r\sigma_{\mathcal{Q}}}) \stackrel{?}{=} \emptyset$” である．
(注5)：この判定は命題 2.1 より決定可能である．
(注6)：[1] の場合，“$q_{c_l\sigma_{\mathcal{Q}}}, q_{c_r\sigma_{\mathcal{Q}}}$ のうち $\mathcal{Q}_i$ に含まれない状態を $\mathcal{Q}_{i+1}$ に追加し，$c_l\sigma_{\mathcal{Q}} \to^*_{\triangle_{i+1}} q_{c_l\sigma_{\mathcal{Q}}}$，$c_r\sigma_{\mathcal{Q}} \to^*_{\triangle_{i+1}} q_{c_r\sigma_{\mathcal{Q}}}$ となるように $c_l\sigma_{\mathcal{Q}} \to q_{c_l\sigma_{\mathcal{Q}}}$，$c_l\sigma_{\mathcal{Q}} \to q_{c_l\sigma_{\mathcal{Q}}}$ を正規化した遷移規則を $\triangle_{i+1}$ に追加する” である．

この生成手続き内には，正規化された遷移規則を追加する過程があるが，ここでは，遷移規則の正規化は人間が適当に状態列を割り当てて行うこととする（文献[1]も同様である）．

[例 3.1] $\mathcal{R}_1$ を条件付き書換え規則 $\rho: f(x,y) \to g(h(y)) \Leftarrow g(h(x)) \to g(x)$ のみからなる定向CTRS，$\mathcal{A}_0$ を $\mathcal{A}_0 = \langle \mathcal{F}, \mathcal{Q}_0, \mathcal{Q}_f, \triangle_0 \rangle$ （$\mathcal{F} = \{a,b,f,g,h\}$，$\mathcal{Q}_0 = \{q_f, q_0, q_1\}$，$\mathcal{Q}_f = \{q_f\}$，$\triangle_0 = \{a \to q_0, b \to q_1, f(q_0,q_1) \to q_f\}$）なる木オートマトンとする（また，$\mathcal{A}_i = \langle \mathcal{F}, \mathcal{Q}_i, \mathcal{Q}_f, \triangle_i \rangle$ で表す）．このとき，$\mathcal{A}_0$ から $\mathcal{A}_2$ を生成すると，以下のようになる．

（1ステップ目）

$f(q_0,q_1) \to^*_{\triangle_0} q_f$ かつ $g(h(q_1)) \not\to^*_{\triangle_0} q_f$ より，$(q_f, \{x \mapsto q_0, y \mapsto q_1\})$ を得る．この $(q_f, \{x \mapsto q_0, y \mapsto q_1\})$ に対して，ρ は条件部を持ち，$\mathcal{Q}_0$ に $q_{g(h(q_0))}$ は含まれないので，$q_{g(h(q_0))}$ を作り $g(h(q_0)) \to^*_{\triangle_1} q_{g(h(q_0))}$ なる遷移規則を生成する必要がある．$g(h(q_0)) \to q_{g(h(q_0))}$ を新しい状態 q_2 を使って正規化した結果，$\triangle_1$ に追加される遷移規則は $h(q_0) \to q_2$，$g(q_2) \to q_{g(h(q_0))}$ となる．

（2ステップ目）

$f(q_0,q_1) \to^*_{\triangle_1} q_f$ かつ $g(h(q_1)) \not\to^*_{\triangle_1} q_f$ より，$(q_f, \{x \mapsto q_0, y \mapsto q_1\})$ を得る．この $(q_f, \{x \mapsto q_0, y \mapsto q_1\})$ に対して，ρ は条件部を持ち，$\mathcal{Q}_1$ に $q_{g(h(q_0))}$ は含まれるので，$\mathcal{L}(\mathcal{A}_1, q_{g(h(q_0))}) \cap G(g(x)) \stackrel{?}{=} \emptyset$ の判定を行う．$g(h(a)) \in \mathcal{L}(\mathcal{A}_1, q_{g(h(q_0))}), g(h(a)) \in G(g(x))$ より，偽となるので，$g(h(q_1)) \to^*_{\triangle_2} q_f$ なる遷移規則を生成する必要がある．$g(h(q_1)) \to q_f$ を新しい状態 q_3 を使って正規化した結果，$\triangle_2$ に追加される遷移規則は $h(q_1) \to q_3$，$g(q_3) \to q_{g(h(q_1))}$ となる．

この結果，$\mathcal{A}_2$ は

$\mathcal{A}_2 = \langle \mathcal{F}, \{q_f, q_0, q_1, q_{g(h(q_0))}, q_2, q_3\}, \mathcal{Q}_f, \{a \to q_0, b \to q_1, f(q_0,q_1) \to q_f, h(q_0) \to q_2, g(q_2) \to q_{g(h(q_0))}, h(q_1) \to q_3, g(q_3) \to q_{g(h(q_1))}\}\rangle$

となる． □

次に，提案した生成手続きの正当性を示す．ここでの正当性とは，提案した生成手続きによって完成した近似木オートマトン $\mathcal{A}_*$ が初期項 E から定向CTRS $\mathcal{R}$ で到達可能なすべての項集合 $\mathcal{R}^*(E)$ の近似木オートマトンであること，つまり，$\mathcal{R}^*(E)$ のすべての項が $\mathcal{A}_*$ で受理されること（$\mathcal{R}^*(E) \subseteqq \mathcal{L}(\mathcal{A}_*)$）である．

以下の定理が生成手続きの正当性を表すものである．

[定理 3.2] E を項の集合，$\mathcal{R}$ をその各規則 $l \to r \Leftarrow s_1 \to t_1, \cdots, s_n \to t_n$ に関して，$\mathcal{V}ar(s_1, \cdots, s_n) \subseteqq \mathcal{V}ar(l)$ を満たす左線形な定向2-CTRS，$\mathcal{A}_0$ を $E \subseteqq \mathcal{L}(\mathcal{A}_0)$ を満たす木オートマトン，$\mathcal{A}_*$ を $\mathcal{R}$ と $\mathcal{A}_0$ を入力として提案した定向CTRSについての生成手続きによって完成した木オートマトン（$\mathcal{A}_* = \langle \mathcal{F}, \mathcal{Q}_*, \mathcal{Q}_f, \triangle_* \rangle$）とする．このとき，$\mathcal{R}^*(E) \subseteqq \mathcal{L}(\mathcal{A}_*)$ を満たす．

[略証] $\mathcal{R}^*(\mathcal{L}(\mathcal{A}_*)) \subseteqq \mathcal{L}(\mathcal{A}_*)$ は，$\forall s,t \in \mathcal{T}(\mathcal{F}), q \in \mathcal{Q}_*.\ s \to^*_{\triangle_*} q$ かつ $s \stackrel{*}{\to}_{\mathcal{R}} t$ ならば $t \to^*_{\triangle_*} q$ を示すことと同様のことである．これは，関係 $\xrightarrow[i]{k}_{\mathcal{R}}$ におけるレベル i とステップ数 k の辞書式順序に関する帰納法で証明できる．さらに，$E \subseteqq \mathcal{L}(\mathcal{A}_0) \subseteqq \mathcal{L}(\mathcal{A}_*)$ であるから，$\mathcal{R}^*(E) \subseteqq \mathcal{R}^*(\mathcal{L}(\mathcal{A}_0)) \subseteqq \mathcal{R}^*(\mathcal{L}(\mathcal{A}_*))$ を満たす．ゆえに，$\mathcal{R}^*(E) \subseteqq \mathcal{L}(\mathcal{A}_*)$ を満たす． □

4. CTRSのための近似関数

3.節で提案した生成手続き内の，$\triangle_{i+1}$ に追加する遷移規則を生成する過程，すなわち，遷移規則を正規化する過程で使われる新しい状態を選択するための近似関数は，CTRSについては提案されていない．そこで本節では，結合CTRSおよび定向CTRSのための近似関数をそれぞれ定義する．これらは文献[2]の近似関数を拡張したものである．

3.節の手続きでは，入力として与える木オートマトン（初期項を含む集合を認識する木オートマトン）$\mathcal{A} = \langle \mathcal{F}, \mathcal{Q}, \mathcal{Q}_f, \triangle \rangle$ から近似木オートマトンを生成するとき，新しく状態を追加する．近似関数はそのような状態をあらかじめ有限で用意して状態の割り当てを決めておく関数である．そこで，$\mathcal{Q}$ に現れない状態の集合を $\mathcal{Q}_{new}$ とする（つまり $\mathcal{Q}_{new} \cap \mathcal{Q} = \emptyset$）．定向CTRSについての近似木オートマトンの生成手続きでは，$s_j\sigma_{\mathcal{Q}}$ を指標付けした状態 $q_{s_j\sigma_{\mathcal{Q}}}$ は $\mathcal{Q}_{new}$ に含まれなければならない．また，集合 S に対して，S の基数を $\mathcal{C}ard(S)$ で表す．

4.1 結合CTRSのための近似関数

結合CTRSのための近似関数は，書換え規則の右辺，条件部の左辺，条件部の右辺のそれぞれに対して，関数記号が出現する先頭以外の位置の項それぞれにどの状態を割り当てるかを定めたものである．

[定義 4.1] $\mathcal{R}_\downarrow$ を結合CTRS，$\mathcal{X}$ を変数の集合，$\mathcal{Q}$ を状態の集合，$\mathcal{Q}_{new}$ を $\mathcal{Q}$ に現れない状態の集合，$\mathcal{Q}^*_{new}$ を $\mathcal{Q}_{new}$ の状態の列の集合，$\mathcal{Q}^{*\times 2}_{new}$ を $\mathcal{Q}_{new}$ の状態の列の対の集合とする．近似関数 γ は，$\gamma: \mathcal{R}_\downarrow \times (\mathcal{Q} \cup \mathcal{Q}_{new}) \times \Sigma(\mathcal{Q} \cup \mathcal{Q}_{new}, \mathcal{X}) \mapsto \mathcal{Q}^*_{new} \times \mathcal{Q}^{*\times 2}_{new}$ なる写像であり，$\gamma(l \to r \Leftarrow c_l \downarrow c_r, q, \sigma_{\mathcal{Q}}) = (q_{r1} \cdots q_{rn_0}, (q_{c_l1} \cdots q_{c_ln_1},\ q_{c_r1} \cdots q_{c_rn_2}))$ （ただし，$n_0 = \mathcal{C}ard(\mathcal{P}os_{\mathcal{F}}(r))$，$n_1 = \mathcal{C}ard(\mathcal{P}os_{\mathcal{F}}(c_l))$，$n_2 = \mathcal{C}ard(\mathcal{P}os_{\mathcal{F}}(c_r))$ である）という形式をしている． □

本稿では，$q_{r1} \cdots q_{rn_0}$，$q_{c_l1} \cdots q_{c_ln_1}$，$q_{c_r1} \cdots q_{c_rn_2}$ の列はそれぞれ r, c_l, c_r に対する，その状態を割り当てられる項の位置が辞書式順序に並ぶように並んでいるとする．

結合CTRSの規則 $l \to r \Leftarrow c_l \downarrow c_r$ に対して $\gamma(l \to r \Leftarrow c_l \downarrow c_r, q, \sigma_{\mathcal{Q}}) = (\vec{q_r}, (\vec{q_{c_l}}, \vec{q_{c_r}}))$ とする．$r\sigma_{\mathcal{Q}} \to q$，$c_l\sigma_{\mathcal{Q}} \to q_{c_l\sigma_{\mathcal{Q}}}$，$c_r\sigma_{\mathcal{Q}} \to q_{c_r\sigma_{\mathcal{Q}}}$ を γ により正規化した遷移規則はそれぞれ $r\sigma_{\mathcal{Q}} \to q$ を $\vec{q_r}$，$c_l\sigma_{\mathcal{Q}} \to q_{c_l\sigma_{\mathcal{Q}}}$ を $\vec{q_{c_l}}$，$c_r\sigma_{\mathcal{Q}} \to q_{c_r\sigma_{\mathcal{Q}}}$ を $\vec{q_{c_r}}$ で正規化した遷移規則である．

4.2 定向CTRSのための近似関数

定向CTRSのための近似関数は，書換え規則の右辺，条件部の左辺のそれぞれに対して，関数記号が出現する先頭以外の位置の項それぞれにどの状態を割り当てるかを定めたものである．

[定義 4.2] $\mathcal{R}_\to$ を定向CTRS，$\mathcal{X}$ を変数の集合，$\mathcal{Q}$ を状態の集合，$\mathcal{Q}_{new}$ を $\mathcal{Q}$ に現れない状態の集合，$\mathcal{Q}^*_{new}$ を $\mathcal{Q}_{new}$ の状態の列の集合，とする．近似関数 γ は，$\gamma: \mathcal{R}_\to \times (\mathcal{Q} \cup \mathcal{Q}_{new}) \times \Sigma(\mathcal{Q} \cup \mathcal{Q}_{new}, \mathcal{X}) \mapsto \mathcal{Q}^*_{new} \times \mathcal{Q}^*_{new}$ なる写像であり，$\gamma(l \to r \Leftarrow s \to t, q, \sigma_{\mathcal{Q}}) = (q_{r1} \cdots q_{rn_0},\ q_{s1} \cdots q_{sn_1})$ （ただ

し，$n_0 = Card(\mathcal{P}os_{\mathcal{F}}(r)), n_1 = Card(\mathcal{P}os_{\mathcal{F}}(s))$ である）という形式をしている． □

本稿では，$q_{r1} \cdots q_{rn_0}$，$q_{s1} \cdots q_{sn_1}$ の列はそれぞれ r, s に対する，その状態を割り当てられる項の位置が辞書式順序に並ぶように並んでいるとする．

定向 CTRS の規則 $l \to r \Leftarrow s \to t$ に対して $\gamma(l \to r \Leftarrow s \to t, q, \sigma_{\mathcal{Q}}) = (\vec{q_r}, \vec{q_s})$ とする．$r\sigma_{\mathcal{Q}} \to q$, $s\sigma_{\mathcal{Q}} \to q_{s\sigma_{\mathcal{Q}}}$ を γ により正規化した遷移規則はそれぞれ $r\sigma_{\mathcal{Q}} \to q$ を $\vec{q_r}$，$s\sigma_{\mathcal{Q}} \to q_{s\sigma_{\mathcal{Q}}}$ を $\vec{q_s}$ で正規化した遷移規則である．

[例 4.3]　$\mathcal{R}_1$ を例 3.1 の $\mathcal{R}_1$，$\mathcal{A}_0$ を例 3.1 の $\mathcal{A}_0$，γ を規則 ρ, 状態 $q \in \mathcal{Q}_2$, 状態代入 $\sigma_{\mathcal{Q}}$（$\mathcal{R}an(\sigma_{\mathcal{Q}}) \subseteq \mathcal{Q}_2$）から新しい状態の列への写像である，定向 CTRS のための近似関数（ただし，$\gamma(\rho, q_f, \{x \mapsto q_0, y \mapsto q_1\}) = (q_3, q_2)$）とする．このとき，$\gamma$ を用いて $\mathcal{A}_0$ から $\mathcal{A}_2$ を生成すると，正規化するときの新しい状態の選択は γ によって決まり，$\mathcal{A}_2$ として例 3.1 の $\mathcal{A}_2$ と同じものが生成される． □

5. 結合 CTRS についての生成手続きとの比較

本節では，3. 節で提案した定向 CTRS についての近似木オートマトンの生成手続きに関して，文献 [1] で提案されている結合 CTRS についての近似木オートマトンの生成手続きとの比較を行う．ここでは，結合 CTRS についての生成手続きで生成した近似木オートマトンと定向 CTRS についての生成手続きで生成した近似木オートマトンの近似精度を比較する．具体的には，左線形な結合 CTRS と結合 CTRS のための近似関数が与えられ結合 CTRS についての生成手続きで近似木オートマトンの生成に成功したとき，その近似木オートマトンと近似精度が等しい近似木オートマトンを定向 CTRS についての生成手続きで生成するための，結合 CTRS から定向 CTRS への変換，および結合 CTRS のための近似関数から定向 CTRS のための近似関数の生成方法を提案する．そして実際に，与えられた左線形な結合 CTRS がその各規則の条件部の左辺と右辺が別の規則の条件部に出現しないという条件下ならば，近似木オートマトンの精度が等しくなることを示す．

5.1 結合 CTRS から定向 CTRS への変換

ここでは，結合 CTRS から定向 CTRS への変換を定義する．

[定義 5.1]　$\mathcal{R}_\downarrow$ を $\mathcal{F}$ 上の結合 CTRS とする．$\mathcal{R}_\downarrow$ の各規則 $\rho : l \to r \Leftarrow c_l \downarrow c_r$ に対して，$\mathcal{F}$ にない記号 tp_2^ρ を用意する．このとき，$\mathbb{T}(\rho)$ を次のように定義する．$\mathbb{T}(\rho) = \{l \to r \Leftarrow \mathsf{tp}_2^\rho(c_l, c_r) \to \mathsf{tp}_2^\rho(x, x)\}$．ただし，$x \notin \mathcal{V}ar(\rho)$ を満たす．$\mathbb{T}$ は結合 CTRS 上に自然に拡張され，$\mathbb{T}(\mathcal{R}_\downarrow) = \bigcup_{\rho \in \mathcal{R}} \mathbb{T}(\rho)$ とする． □

$\mathcal{R}_\downarrow$ が左線形な結合 CTRS であるとき，$\mathbb{T}(\mathcal{R}_\downarrow)$ は左線形な定向 CTRS となる．また，変換 $\mathbb{T}$ により生成される定向 CTRS は，近似木オートマトンの生成手続きの適用対象となる定向 CTRS，すなわち，各規則 $l \to r \Leftarrow s \to t$ が $\mathcal{V}ar(s) \subseteq \mathcal{V}ar(l)$ を満たす 2-CTRS となる．

ここでは，変換 $\mathbb{T}$ の模倣完全性を表す以下の定理が成り立つ．

[定理 5.2]　$\mathcal{R}_\downarrow$ を $\mathcal{F}$ 上の結合 CTRS とする．任意の項 $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ に対して，$s \xrightarrow{*}_{\mathcal{R}_\downarrow} t$ ならば，かつそのときに限り，$s \xrightarrow{*}_{\mathbb{T}(\mathcal{R}_\downarrow)} t$ である． □

[略証]　まず，$s \xrightarrow{*}_{\mathcal{R}_\downarrow} t$ ならば $s \xrightarrow{*}_{\mathbb{T}(\mathcal{R}_\downarrow)} t$ は，関係 $\xrightarrow[i]{k}_{\mathcal{R}_\downarrow}$ におけるレベル i とステップ数 k の辞書式順序に関する帰納法で証明できる．また，$s \xrightarrow{*}_{\mathbb{T}(\mathcal{R}_\downarrow)} t$ ならば $s \xrightarrow{*}_{\mathcal{R}_\downarrow} t$ は，関係 $\xrightarrow[i]{k}_{\mathbb{T}(\mathcal{R}_\downarrow)}$ におけるレベル i とステップ数 k の辞書式順序に関する帰納法で証明できる． □

5.2 定向 CTRS のための近似関数の生成方法

ここでは，結合 CTRS のための近似関数から定向 CTRS のための近似関数への変換 $\mathbb{T}_\Gamma$ を提案する．この $\mathbb{T}_\Gamma$ は，与えられた左線形な結合 CTRS $\mathcal{R}_\downarrow$ とそのための近似関数 γ から結合 CTRS の生成手続き [1] で生成した近似木オートマトンと，$\mathbb{T}(\mathcal{R}_\downarrow)$ と $\mathbb{T}_\Gamma(\gamma)$ から定向 CTRS の生成手続きで生成した近似木オートマトンとの近似精度が等しくなるような変換として提案している．

[定義 5.3]　$\mathcal{R}_\downarrow$ を左線形な結合 CTRS，$\mathcal{R}_\to$ を $\mathcal{R}_\downarrow$ を変換してできた左線形な定向 CTRS（$\mathcal{R}_\to = \mathbb{T}(\mathcal{R}_\downarrow)$），$\gamma_\downarrow$ を結合 CTRS のための近似関数（$\gamma_\downarrow : \mathcal{R}_\downarrow \times (\mathcal{Q} \cup \mathcal{Q}_{new}) \times \Sigma(\mathcal{Q} \cup \mathcal{Q}_{new}, \mathcal{X}) \mapsto \mathcal{Q}^*_{new} \times \mathcal{Q}^{*\times 2}_{new}$）とする．$\mathcal{Q} \cup \mathcal{Q}_{new}$ に含まれない状態 $q_\perp$ を用意する．このとき，$\gamma_\downarrow$ から変換 $\mathbb{T}_\Gamma$ により生成した定向 CTRS のための近似関数 $\gamma_\to$（$\gamma_\to = \mathbb{T}_\Gamma(\gamma_\downarrow)$）を図 1 のように定義する（$\gamma_\to : \mathcal{R}_\to \times (\mathcal{Q} \cup \mathcal{Q}'_{new}) \times \Sigma(\mathcal{Q} \cup \mathcal{Q}'_{new}, \mathcal{X}) \mapsto \mathcal{Q}'^*_{new} \times \mathcal{Q}'^*_{new}$）．ただし，$\mathcal{R}_\to$ の各規則は，$l \to r \Leftarrow \mathsf{tp}_2^\rho(c_1, c_2) \to \mathsf{tp}_2^\rho(x, x)$（ただし，$\rho$ は $\rho : l \to r \Leftarrow c_1 \downarrow c_2$）であり，$x$ は $\mathcal{R}_\downarrow$ には出現しない唯一の新しい変数，すなわち $\mathcal{V}ar(\mathcal{R}_\to) = \mathcal{V}ar(\mathcal{R}_\downarrow) \uplus \{x\}$ を満たすものとする． □

変換して得られた $\mathcal{R}_\to$ には $\mathcal{F}$ にない関数記号が存在する．そのような記号を不当に含む項を特別に扱うために $q_\perp$ を導入している．

次に，左線形な結合 CTRS $\mathcal{R}_\downarrow$ がその各規則の条件部の左辺と右辺が別の規則の条件部に出現しないという条件下ならば，$\mathcal{R}_\downarrow$ と結合 CTRS のための近似関数 $\gamma_\downarrow$ を用いて結合 CTRS についての生成手続きで生成した近似木オートマトンと，$\mathbb{T}(\mathcal{R}_\downarrow)$ と $\mathbb{T}_\Gamma(\gamma_\downarrow)$ を用いて定向 CTRS についての生成手続きで生成した近似木オートマトンの，それぞれの認識集合が等しくなることを表す以下の定理が成り立つ．

[定理 5.4]　$\mathcal{R}_\downarrow$ をその各規則の条件部の左辺と右辺が別の規則の条件部に出現しない（ただし，真部分項としての出現は許す）左線形な結合 CTRS，$\mathcal{R}_\to$ を $\mathcal{R}_\downarrow$ を変換してできた左線形な定向 CTRS（$\mathcal{R}_\to = \mathbb{T}(\mathcal{R}_\downarrow)$），$\gamma_\downarrow$ を結合 CTRS のための近似関数，$\gamma_\to$ を $\gamma_\downarrow$ から生成した定向 CTRS のための近似関数（$\gamma_\to = \mathbb{T}_\Gamma(\gamma_\downarrow)$），$\mathcal{A}_k$ を $\mathcal{R}_\downarrow, \gamma_\downarrow$ を用いて結合 CTRS についての生成手続きで木オートマトンを生成するときの k ステップ後の木オートマトン（$\mathcal{A}_k = \langle \mathcal{F}, \mathcal{Q}_k, \mathcal{Q}_f, \Delta_k \rangle$），$\mathcal{A}'_k$ を $\mathcal{R}_\to, \gamma_\to$ を用いて定向 CTRS についての生成手続きで木オートマトンを生成するときの k ステップ後の木オートマトン（$\mathcal{A}'_k = \langle \mathcal{F}', \mathcal{Q}'_k, \mathcal{Q}_f, \Delta'_k \rangle$）とする（ただし，$\mathcal{F} \subseteq \mathcal{F}', \mathcal{Q}_0 = \mathcal{Q}'_0, \Delta_0 = \Delta'_0$）．このとき，任意の i に対して，$\mathcal{L}(\mathcal{A}_i) = \mathcal{L}(\mathcal{A}'_i)$ を満たす． □

[略証]　$\mathcal{R}_\downarrow$ の仮定と "$l\sigma_{\mathcal{Q}} \to^*_{\Delta'_i} q \wedge r\sigma_{\mathcal{Q}} \not\to^*_{\Delta'_i} q \wedge q \in \mathcal{Q}'_i \setminus \mathcal{Q}_i$

$$\gamma_{\to}(l \to r \Leftarrow \mathsf{tp}_2^{\rho}(c_l, c_r) \to \mathsf{tp}_2^{\rho}(x,x), q, \sigma_{\mathcal{Q}})$$

$$= \begin{cases} (q_{r1}\cdots q_{rn_0},\ q_{c_l\sigma_{\mathcal{Q}}} q_{c_l1}\cdots q_{c_ln_1} q_{c_r\sigma_{\mathcal{Q}}} q_{c_r1}\cdots q_{c_rn_2}) & q \in \mathcal{Q}\cup\mathcal{Q}_{new} \text{ かつ } \mathcal{R}an(\sigma_{\mathcal{Q}}) \subseteqq \mathcal{Q}\cup\mathcal{Q}_{new} \text{のとき.} \\ & \text{ただし,} \gamma_{\downarrow}(l \to r \Leftarrow c_1 \downarrow c_2, q, \sigma_{\mathcal{Q}}) = (q_{r1}\cdots q_{rn_0},\ (q_{c_l1}\cdots q_{c_ln_1},\ q_{c_r1}\cdots q_{c_rn_2})) \text{ である.} \\ (q_{\perp}\cdots q_{\perp},\ q_{\perp}\cdots q_{\perp}) & \text{それ以外のとき.} \end{cases}$$

図 1　$\mathbb{T}_{\Gamma}(\gamma_{\downarrow})$ で定める近似関数 $\gamma_{\to}$ の定義

$\wedge\ \mathcal{R}an(\sigma_{\mathcal{Q}}) \subseteqq \mathcal{Q}'_i \setminus \mathcal{Q}_i$ なる $(q, \sigma_{\mathcal{Q}})$ は存在しない" ことと "$\mathcal{L}(\mathcal{A}_i, q_{c_l\sigma_{\mathcal{Q}}}) \cap \mathcal{L}(\mathcal{A}_i, q_{c_r\sigma_{\mathcal{Q}}}) \neq \emptyset$ ならばかつそのときに限り $\mathcal{L}(\mathcal{A}'_i, q_{\mathsf{tp}_2^{\rho}(c_l,c_r)\sigma_{\mathcal{Q}}}) \cap G(\mathsf{tp}_2^{\rho}(x,x)) \neq \emptyset$ を満たす" ことから，任意の i に対して，以下の 2 つを満たすことが証明できる．

- $\mathcal{Q}_i \subseteqq \mathcal{Q}'_i$ かつ $\triangle_i \subseteqq \triangle'_i$,
- $\triangle_i \cap \triangle'_i$ のすべての遷移規則は $f(q_1,\cdots,q_n) \to q$ ($f \in \mathcal{F}$, $q_1,\cdots,q_n, q \in \mathcal{Q}\cup\mathcal{Q}_{new}$)，$\triangle'_i \setminus \triangle_i$ のすべての遷移規則は $\mathsf{tp}_2^{\rho}(q_{c_l\sigma_{\mathcal{Q}}}, q_{c_r\sigma_{\mathcal{Q}}}) \to q_{\mathsf{tp}_2^{\rho}(c_l,c_r)\sigma_{\mathcal{Q}}}$ (ρ は $\mathcal{R}_{\downarrow}$ の規則であり，$\rho : l \to r \Leftarrow c_l \downarrow c_r$) という形式をしている．

これらから，$\mathcal{L}(\mathcal{A}_i) \subseteqq \mathcal{L}(\mathcal{A}'_i)$ と $\mathcal{L}(\mathcal{A}_i) \supseteqq \mathcal{L}(\mathcal{A}'_i)$ が証明できる．　□

6. おわりに

本稿では，左線形な定向 CTRS で到達可能な項集合の近似集合を認識する木オートマトンの生成手続きを提案し，その手続きで生成される木オートマトンは与えられた左線形な定向 CTRS で到達可能な項をすべて受理することを示した．また，結合 CTRS と定向 CTRS のための近似関数，結合 CTRS を定向 CTRS に変換する方法を提案し，文献 [1] で提案されている結合 CTRS についての生成手続きと我々が提案した定向 CTRS についての生成手続きとの比較を行った．この比較の結果，与えられた左線形な結合 CTRS と結合 CTRS のための近似関数に対して，その結合 CTRS の各規則の条件部の左辺と右辺が別の規則の条件部に出現しないならば，結合 CTRS についての生成手続きにより生成した木オートマトンの認識集合と等しい集合を認識する木オートマトンを定向 CTRS についての生成手続きでも生成できることを示すことができた．

今後の課題として，まずは，各規則の条件部の左辺と右辺が別の規則の条件部に出現しない左線形な結合 CTRS $\mathcal{R}_{\downarrow}$ と結合 CTRS のための近似関数 $\gamma_{\downarrow}$ を用いて結合 CTRS についての生成手続きで生成した近似木オートマトンと，$\mathbb{T}(\mathcal{R}_{\downarrow})$ と $\mathbb{T}_{\Gamma}(\gamma_{\downarrow})$ を用いて定向 CTRS についての生成手続きで生成した近似木オートマトンの，それぞれの認識集合が等しくなることが，一般の左線形な結合 CTRS でも成り立つか否かを証明することが挙げられる．

また，我々が提案した定向 CTRS についての生成手続きは，与えられる定向 CTRS に様々な制限を持たせているが，その制限をなくした一般の定向 CTRS についての生成手続きについて研究することも今後の課題である．

謝辞　本研究は一部，科研費#16650005，#17700009，#18500011 の補助を受けている．

文献

[1] G. Feuillade and T. Genet. Reachability in Conditional Term Rewriting Systems. Proc. of FTP'03, Workshop on First Order Theorem Proving, ENTCS86, No 1, May, 2003.

[2] T. Genet. Decidable Approximations of Sets of Descendants and Sets of Normal Forms. Proc. of RTA '98, LNCS 1379, pp.151–165, Springer-Verlag, 1998.

[3] F. Baader and T. Nipkow. Term Rewriting and All That. Cambridge University Press, 1998.

[4] E. Ohlebusch. Advanced Topics in Term Rewriting. Springer, 2002.

[5] H. Comon, M. Dauchet, R. Gilleron, F. Jacquemard, D. Lugiez, S. Tison, and M. Tommasi. Tree Automata Techniques and Applications. http://www.grappa.univ-lille3.fr/tata/

[6] F. Jacquemard. Decidable Approximations of Term Rewriting Systems. Proc. of RTA '96, LNCS 1103, pp.362–376, 1996.

[7] T. Nagaya and Y. Toyama. Decidability for Left-Linear Growing Term Rewriting Systems. Proc. of RTA '99, LNCS 1631, pp.256–270, 1999.

[8] T. Takai, Y. Kaji, and H. Seki. Right-Linear Finite Path Overlapping Term Rewriting Systems Effectively Preserve Recognizability. Proc. of RTA '00, LNCS 1833, pp.246–260, 2000.

[9] S. Vágvölgyi. Descendants of a Recognizable Tree Language for Sets of Linear Monadic Term Rewrite Rules. Information Processing Letters, vol. 99, No. 3, pp.111–118, 2006.

[10] T. Genet and F. Klay. Rewriting for Cryptographic Protocol Verification. Proc. of CADE-17, LNAI 1831, Springer-Verlag, pp.271–290, 2000.

[11] T. Genet, Y.-M. Tang-Talpin, and V. Viet Triem Tong. Verification of Copy-Protection Cryptographic Protocol using Approximations of Term Rewriting Systems. Proc. of Workshop on Issues in the Theory of Security (WITS'03), 2003.