条件部に無限集合をもつメンバシップ条件付きTRSの合流性について
Membership Conditional TRS with Infinite Sets in Conditions
(Extended Abstract)

山田順之介 (NTT基礎研究所)
Junnosuke Yamada
NTT Basic Research Laboratories
3-9-11, Midori-cho, Musashino, Tokyo 180 Japan

Abstract

A membership conditional term rewriting system (MCTRS) is a term rewriting system such that applications of its rules are restricted by membership conditions on the variables. This paper introduces a new technique of *split* for testing the confluence of MCTRS. The point of this technique is to divide membership conditions into simpler parts. By considering inductive structures of infinite *splits*, the confluence is shown for a MCTRS having infinite sets in its membership conditions. As an application of this *split* technique, a completion of McCarthy's 91-function is demonstrated.

1. Introduction

Term rewriting system, TRS in short, is an excellent machinery for automated theorem proving, algebraic data specification, and program verification and transformation. Conditional TRS arises naturally as its extension, and much research has appeared (e.g., [1]). Membership conditional TRS proposed by Toyama [16] is one of such extensions. Their rewriting rules are restricted by membership conditions on the variables in left hand sides of rules. Restrictions on types and values for variables in real programs can be expressed naturally using them. Moreover, such systems can describe an infinite number of rules in one rule. H. Kirchner [11] also proposed another notion called schematization to handle an infinite number of rules.

Discussions on the confluence of unconditional TRS can be classified into two categories: one assumes TRS to be left-linear and non-overlapping and the other assumes noetherian and overlapping. Research on conditional TRS also has two similar approaches. The results on membership conditional TRS in the former category are already known [16],[17] and the results in the latter will be shown here. The result is a critical pair lemma for membership conditional TRS. For that purpose the notions of critical pair and reductions are reformulated, so as to be able to handle membership conditions attached to rules and terms. This paper can treat a wider class of membership conditional TRS than previous work [18] by introducing contextual rewriting [19] which allows modification of context parts called *split*. The notion of *split* is necessary to overcome the difference of reductions in contextual TRS and its underlying membership conditional TRS. A completion algorithm for membership conditional TRS will be also proposed.

Sections 2 and 3 give preliminaries of TRS and membership conditional TRS respectively. Section 4 introduces contextual rewriting and related notions. Section 5 shows the critical pair lemma and a completion algorithm. Furthermore our algorithm demonstrates a property of the McCarthy's 91-function which needs some inductive method for its proof.

2. Term Rewriting Systems

This section briefly reviews TRS (cf., [2],[3],[5],[8]) and prepare necessary notions for the following sections.

A term set $T = T(F, V)$ is the set of first order terms composed of the elements in a set of function symbols F graded by arities and a denumerable set of variables V such that $F \cap V = \phi$. The set of all the variables in a term t is $Var(t)$. The identity of terms is denoted by $\equiv$.

For any term t its *occurrences* $\mathcal{O}(t)$, *subterm* t/u of t at occurrence $u \in \mathcal{O}(t)$, and replacement $t[u \leftarrow s]$ or simply $t[s]$ for $t, s \in T$ and $u \in \mathcal{O}(t)$ are defined completely same as in [5].

A *substitution* θ is a map from V to $T(F, V)$ such that $x\theta \equiv x$ almost everywhere. Substitutions are naturally extended as morphisms of T.

A *rewriting rule* on T is a pair of two terms (l, r) with $Var(l) \supset Var(r)$ and $l \notin V$. A set of rewriting rules is indicated by $\triangleright$, and $l \triangleright r$ iff $(l, r) \in \triangleright$. A term t *reduces to* a term t' at occurrence $u \in \mathcal{O}(t)$ by a rewriting rule $l \triangleright r$ iff $t \equiv s[u \leftarrow l\theta]$, $t' \equiv s[u \leftarrow r\theta]$ for some $s \in T$, substitution θ, and occurrence $u \in \mathcal{O}(t)$ such that $t/u \notin Var(t)$. The relation of the two terms is indicated by $t \rightarrow t'$ and the subterm t/u is called a *redex* of the rule in t. The transitive reflexive closure of $\rightarrow$ is denoted by $\rightarrow^*$.

Term rewriting system is defined as follows:

Definition 2.1. (Term Rewriting System)

A TRS is a structure $(T, \rightarrow)$ with an object set T and a binary relation $\rightarrow$ defined by a set of rewriting rules $\triangleright$ on T.

A term t is said to be a *normal form* iff there is no t' such that $t \rightarrow t'$. A term t' is called a *normal form* of t iff $t \rightarrow^* t'$ and t' is a normal form and denoted by $t\downarrow$. Two terms t_1 and t_2 *converge* or are *convergent* iff there is a term s such that $t_1 \rightarrow^* s$ and $t_2 \rightarrow^* s$.

Two rules $l_i \triangleright r_i$ for $i = 1, 2$ with no common variables in a TRS are *overlapping* iff $l_i\theta_i/u \equiv l_j\theta_j$ for some $\theta_i, \theta_j, u \in \mathcal{O}(t)$ such that $l_i/u \notin V$. Hereafter any two rules are assumed to have no common variables, if not stated.

A critical pair of two overlapping rules is defined.

Definition 2.2. (Critical Pair)

A pair of terms $\langle P, Q\rangle$ is a *critical pair* of two rules $l_i \triangleright r_i$ for $i = 1, 2$ overlapping in $u \in \mathcal{O}(l_1)$ is:

$$P \equiv l_1\theta[u \leftarrow r_2\theta], \quad Q \equiv r_1\theta$$

where θ is the most general unifier of $l_1/u \notin V$ and l_2.

The following two notions characterize TRS.

Definition 2.3. (Noetherian)

A TRS $R = (T, \rightarrow)$ is *noetherian* iff every reduction in R terminates, i.e., there is no infinite reduction sequence as $t_1 \rightarrow t_2 \rightarrow t_3 \rightarrow \cdots$ where $t_i \in T$.

Definition 2.4. (Confluence and Local Confluence)

A TRS $R = (T, \rightarrow)$ is *confluent* iff

$$\forall t, t_1, t_2 \in T[t \rightarrow^* t_1, t \rightarrow^* t_2 \Rightarrow \exists t' \text{ such that } t_1 \rightarrow^* t', t_2 \rightarrow^* t']$$

and *locally confluent* iff

$$\forall t, t_1, t_2 \in T[t \rightarrow t_1, t \rightarrow t_2 \Rightarrow \exists t' \text{ such that } t_1 \rightarrow^* t', t_2 \rightarrow^* t'].$$

A TRS possessing both properties called *complete*. In such systems, every term necessarily has a unique normal form.

These two properties have been of our chief concern, because noetherian property guarantees the existence of normal forms, and confluence guarantees uniqueness of normal forms provided they exist. As they are generally undecidable, much research focuses to investigate some sufficient conditions of them. For noetherian unconditional TRS the following results on the confluence are well-known.

Lemma 2.5. (Critical Pair Lemma)
A noetherian TRS R is locally confluent iff every critical pair of R converges.

Note that the following lemma for general noetherian relations holds.

Lemma 2.6.
A noetherian relation is confluent iff it is locally confluent.

Combining these two lemmas, the next theorem on the confluence of unconditional noetherian TRS holds.

Theorem 2.7.
A noetherian TRS R is confluent iff every critical pair of R converges.

3. Membership Conditional Term Rewriting Systems

A kind of conditional TRS, membership conditional TRS was initially proposed in [16]. The rewriting rules of the system are restricted by membership conditions on the variables in the left hand sides of the rules. Membership conditional TRS are suitable to describe the restrictions on types, values of variables and treat an infinite number of rules in one. Membership conditional TRS treated in this paper are all assumed to be well-defined.

Definition 3.1. (c-Term, MC-Rule)
A *c-term* is a term with membership conditions on the variables in the term:

$$t : (x_1, \cdots, x_n) \in S_1 \times \cdots \times S_n$$

where $\{x_1, \cdots, x_n\} = Var(t)$ and $S_i \subset T$ for all i. A c-term is written simply as $t : c$ where $c = (x_1, \cdots, x_n) \in S_1 \times \cdots \times S_n$, and c is called the *context* of the c-term. An *MC-rule* $l \triangleright r : c$ is a rewriting rule $l \triangleright r$ with membership conditions c on the variables in l.

A term t reduces to a term t' by an MC-rule $l \triangleright r : (x_1, \cdots, x_n) \in S_1 \times \cdots \times S_n$ in a membership conditional TRS, when

$$t \equiv t[l\theta], \quad t' \equiv t[r\theta] \text{ for some substitution } \theta \text{ such that } x_1\theta \in S_1, \cdots, x_n\theta \in S_n.$$

Definition 3.2. (Membership Conditional TRS)
A *membership conditional TRS* is a term rewriting system defined by a set of MC-rules.

An example of membership conditional TRS and its reductions are shown below.

Example 3.3.
Let $F = \{f, d, +, s, 0\}$ and $F' = \{+, s, 0\}$. Next membership conditional TRS R defines the addition $+$, the double d, and f on the set of natural numbers $\mathbb{N} = T(\{s, 0\})$.

$$R: \begin{cases} x + 0 \rhd x : x \ \varepsilon \ T \\ x + s(y) \rhd s(x + y) : (x, y) \ \varepsilon \ T^2 \\ d(x) \rhd x + x : x \ \varepsilon \ T(F') \\ f(x, x) \rhd x : x \ \varepsilon \ T(F') \end{cases}$$

In this system, the following reduction sequence is obtained:

$$f(d(0), d(0)) \rightarrow f(0 + 0, d(0)) \rightarrow f(0 + 0, 0 + 0) \rightarrow 0 + 0 \rightarrow 0.$$

Note that $f(d(0), d(0))$ cannot directly reduce to $d(0)$ by the third rule in R since $d(0) \notin T(F')$.

Rules in membership conditional TRS might cause some interaction each other in a very different way from those in the other TRS. The following membership conditional TRS illustrates the situation:

$$R: \begin{cases} f(x) \rhd h(x) & : x \ \varepsilon \ g(T) \cup k(T) \\ g(x) \rhd x & : x \ \varepsilon \ T \end{cases}$$

where $g(T)$ and $k(T)$ denote sets of terms with outermost symbol g and k respectively, and $g(T) \cup k(T)$ a union of them. A single term $f(g(x))$ can be reduced by the both rules. A reduction sequence $f(g(x)) \rightarrow h(g(x)) \rightarrow h(x)$ is obtained by using the first rule and the second subsequently. However, if the second rule is applied preceding the first we have $f(g(x)) \rightarrow f(x)$, and R is not confluent. This problem is resulted from $f(x)$ is no longer satisfying the membership condition of the first rule. Then the following notions are introduced to prevent us from this problem.

Definition 3.4. (Closed, Quasi-Closed)
A set of terms S is *closed* iff $\forall s \in S \forall t \in T[s \rightarrow t \Rightarrow t \in S]$ A set of terms S is *quasi-closed* iff $s \downarrow \in S$ for all $s \in S$. A membership conditional TRS R is *closed* (resp. *quasi-closed*) iff every set $S (\neq T)$ in membership conditions of its rules is closed (resp. *quasi-closed*).

As for noetherian membership conditional TRS, it is not necessary to assume closedness and it is sufficient to assume quasi-closedness.

4. Contextual Rewriting

In membership conditional TRS, membership conditions attached to their rules must be carried by some mechanism. For that purpose the notions of context and contextual rewriting are introduced. The contextual rewriting introduced in this paper differs from the one in [19] whose rules and terms have conditions of boolean values.

We say $t(x_1, \cdots, x_n) \ \varepsilon \ S$ *holds under* c where $c = (x_1, \cdots, x_n) \ \varepsilon \ S_1 \times \cdots \times S_n$ iff $t(s_1, \cdots, s_n) \in S$ for all $(s_1, \cdots, s_n) \in S_1 \times \cdots \times S_n$.

Definition 4.1. (c-Substitution)
A map θ from $x : x \ \varepsilon \ S$ to a c-term $t : c$ is a *c-substitution* iff $x\theta \equiv x$ almost every where and $x\theta \ \varepsilon \ S$ holds under c. Naturally this map can be extended on c-terms.

Definition 4.2. (c-Match)
Let $t : c$ and $t' : c'$ be two c-terms. $t' : c'$ *c-matches* to $t : c$ iff there is a c-substitution θ from $t' : c'$ to $t : c$.

Now we define c-unifier of two c-terms which need some additional information on the domains of the variables substituted.

Definition 4.3. (c-Unification)
Let $t_1 : c_1$ and $t_2 : c_2$ be two c-terms with no common variables and $c_1 = (x_1, \cdots, x_m) \varepsilon S_1 \times \cdots \times S_m$, $c_2 = (x_{m+1}, \cdots, x_{m+n}) \varepsilon S_1 \times \cdots \times S_{m+n}$. A *c-unifier* of $t_1 : c_1$ and $t_2 : c_2$ is a pair of c-substitution θ and membership conditions $c = (x_1, \cdots, x_{m+n}) \varepsilon S'_1 \times \cdots \times S'_{m+n}$ where $S'_i \subset S_i$such that $x_i\theta \varepsilon S'_i$ holds under c. A c-unifier (θ, c) is a *most general unifier* of two c-terms $t_1 : c_1$ and $t_2 : c_2$ iff θ is a most general unifier of t_1 and t_2.

Definition 4.4. (c-Reduction, c-Convergence)
A c-term $t : c$ is *c-reducible* by an MC-rule $l \triangleright r : (x_1, \cdots, x_n) \varepsilon S_1 \times \cdots \times S_n$ iff some subterm $t' \equiv t/u$ at occurrence u of t is $l\theta'$ for some substitution θ' and $x_i\theta' \varepsilon S_i$ for $i = 1, \cdots, n$ hold under c. Then $t : c$ *c-reduces to* $s : c \equiv t[u \leftarrow r\theta'] : c$ and we denote $t : c \twoheadrightarrow s : c$. In this definition, as each $x_i\theta'$ includes variables restricted by context c, and $x_i\theta' \in S_i$ have to be verified. The transitive reflexive closure of $\twoheadrightarrow$ is denoted by $\twoheadrightarrow^*$. A c-term $t : c$ is a *c-normal form* iff there is no $t' : c$ such that $t : c \twoheadrightarrow t' : c$, and $t' : c$ is a *c-normal form* of $t : c$ iff $t : c \twoheadrightarrow^* t' : c$, and $t' : c$ is a c-normal form. Two c-terms with a common context $t_1 : c$ and $t_2 : c$ *c-converge* iff there is a c-term $s : c$ such that $t_1 : c \twoheadrightarrow^* s : c$ and $t_2 : c \twoheadrightarrow^* s : c$.

The relation between terms and c-terms is formalized.

Definition 4.5. (Instance of c-Term, Associated c-Term)
A term $t\theta$ is called an *instance* of a c-term $t : c$ iff $x\theta$ satisfies the condition c for any variable $x \in Var(t)$. Conversely the c-term $t : c$ is called an *associated c-term* of $t\theta$. The set of all the associated c-terms of a term set T is denoted by T_c and called an *associated c-term set* of T.

Definition 4.6. (cTRS)
For a TRS $R = (T, \rightarrow)$ there are a set of associated c-terms T_c and a c-reduction relation $\twoheadrightarrow$, and a TRS $R_c = (T_c, \twoheadrightarrow)$ called the *associated cTRS* of R. Moreover the associated cTRS can necessarily be defined for any TRS.

Based on the notion of c-reduction, a critical pair of two MC-rules can be defined. Before that, the notion of overlapping of MC-rules have to be clarified.

Definition 4.7. (c-Overlapping)
Two MC-rules with no common variables

$$l_1 \triangleright r_1 : (x_1, \cdots, x_m) \varepsilon S_1 \times \cdots \times S_m \quad \text{and}$$
$$l_2 \triangleright r_2 : (x_{m+1}, \cdots, x_{m+n}) \varepsilon S_{m+1} \times \cdots \times S_{m+n}$$

are *c-overlapping* in $u \in \mathcal{O}(l_1)$ such that $l_1/u \notin V$ iff there is a c-unifier $(\theta, (x_1, \cdots, x_{m+n}) \varepsilon S'_1 \times \cdots \times S'_{m+n})$ of l_1/u and l_2 under the membership conditions in the rules.

Now c-critical pair of two c-overlapping rules can be introduced.

Definition 4.8. (c-Critical Pair)
Let

$$l_1 \triangleright r_1 : (x_1, \cdots, x_m) \ \varepsilon\ S_1 \times \cdots \times S_m \quad \text{and}$$
$$l_2 \triangleright r_2 : (x_{m+1}, \cdots, x_{m+n}) \ \varepsilon\ S_{m+1} \times \cdots \times S_{m+n}$$

be two MC-rules c-overlapping in occurrence $u \in \mathcal{O}(l_1)$. The *MC-critical pair* $\langle P, Q\rangle : c$ of the two MC-rules in $u \in \mathcal{O}(l_1)$ is

$$P \equiv l_1\theta[u \leftarrow r_2\theta], \quad Q \equiv r_1\theta, \quad \text{and} \quad c = (x_1, \cdots, x_{m+n}) \in S'_1 \times \cdots \times S'_{m+n}$$

where (θ, c) is a most general c-unifier of of two left hand sides c-terms of the two MC-rules with each membership conditions.

For example we find a c-critical pair of the following two rules:

$$f(f(y)) \triangleright h(y) : y \ \varepsilon\ g(T) \cup h(T) \quad \text{and}$$
$$f(g(z)) \triangleright g(z) : z \ \varepsilon\ T.$$

There is a substitution $y/g(z)$ to make the two rules c-overlapping. Then some membership condition on z satisfying both $z \ \varepsilon\ T$ and $g(z) \ \varepsilon\ g(T) \cup h(T)$ have to be found. From the former condition, some subcondition of $z \ \varepsilon\ T$ is obtained from $g(z) \ \varepsilon\ g(T) \cup h(T)$. In this example, such a condition $z \ \varepsilon\ T$ and a c-critical pair $\langle f(g(z)), h(g(z))\rangle : z \ \varepsilon\ T$ can be found immediately.

Contextual rewriting as c-reduction and related notions are too restricted. Here the following membership conditional TRS which defines the addition *add* on the set of natural numbers $\mathbb{N} = T(\{s, 0\})$ is considered.

$$\begin{cases} add(x, 0) \triangleright x & : x \ \varepsilon\ \mathbb{N} & (1) \\ add(x, s(y)) \triangleright add(s(x), y) & : (x, y) \ \varepsilon\ \mathbb{N}^2 & (2) \\ add(x, s(y)) \triangleright s(add(x, y)) & : (x, y) \ \varepsilon\ \mathbb{N}^2 & (3) \end{cases}$$

Two rules (2) and (3) are c-overlapping in the associated cTRS and we have:

$$add(x, s(y)) : (x, y) \ \varepsilon\ \mathbb{N}^2$$
$$\swarrow \qquad\qquad \searrow$$
$$s(add(x, y)) : (x, y) \ \varepsilon\ \mathbb{N}^2 \qquad\qquad add(s(x), y) : (x, y) \ \varepsilon\ \mathbb{N}^2$$

These two c-terms are c-irreducible, even though the corresponding membership conditional TRS is locally confluent. To overcome the difference of the reduction power, c-reduction is extended using a new concept defined below. Then the extended c-reduction can handle this example properly.

Definition 4.9. (Split)
A context $c = (x_1, \cdots, x_k) \ \varepsilon\ S_1 \times \cdots \times S_k$ has a *split* into $c_1, \cdots, c_n$ iff S_i's are disjoint unions of several sets, i.e., $S_i = \sqcup_{j_i \in J_i} S_i^{(j_i)}$, and there is at least one i such that J_i has more than one elements. This is denoted by $c = c_1 \sqcup \cdots \sqcup c_n$ where $c_i = (x_1, \cdots, x_k) \ \varepsilon\ S_1^{(j_1)} \times \cdots \times S_k^{(j_k)}$.

Definition 4.10. (c-Reduction with Split)
A c-term $t : c$ c-reduces c-terms $t_i : c_i$'s *with split* iff there is a split $c = \sqcup c_i$ such that $t : c_i \twoheadrightarrow t_i : c_i$ for all i.

Definition 4.11. (c-Convergence with Split)
Two c-terms $t_1 : c$ and $t_2 : c$ with a common context c c-converge or are c-convergent *with split* iff there is a split $c = c_1 \sqcup \cdots \sqcup c_n$ such that $t_1 : c_i$ and $t_2 : c_i$ are c-convergent for all i.

By a split $\mathbb{N}^2 = \mathbb{N} \times \{0\} \sqcup \mathbb{N} \times \{n \geq 1\}$, the pair of terms in the preceding example $s(add(x, y)) : (x, y) \ \varepsilon\ \mathbb{N}^2$ and $add(s(x), y) : (x, y) \ \varepsilon\ \mathbb{N}^2$ c-converges.

5. Confluence of Membership Conditional Term Rewriting Systems

Next lemma is the critical pair lemma for membership conditional TRS.

Lemma 5.1. (Critical Piar lemm for MCTRS)
Let R be a noetherian quasi-closed membership conditional TRS and R_c be its associated cTRS. If every c-critical pair in R_c c-converges with split, then R is locally confluent.

Proof.
Let a term t reduces two distinct terms t' and t'' by two MC-rules (r1) $l_1 \triangleright r_1 : c_1$ and (r2) $l_2 \triangleright r_2 : c_2$ on redexes t/u and t/v respectively, where u and v are two occurrences of t.

There are two cases by the relative position of u and v.

Case1: u and v are disjoint.

Reductions at t'/v by (r2) and t''/u by (r1) result both in an identical term.

Case2: u and v are not disjoint.

Without loss of generality t/v is a subterm of t/u and we have only to consider a subterm t/u of t. This case has two subcases.

a. $v = u \cdot w$ where $w \in \mathcal{O}(l_1)$ and $l_1/w \notin V$.

Only in this subcase, t'/u and t''/u are instances of two c-terms $P : c$ and $Q : c$ where $\langle P, Q\rangle : c$ is a c-critical pair in R_c. Then there exists a substitution θ such that $t'/u \equiv P\theta$, $t''/u \equiv Q\theta$ and $x_i\theta \in S_i$ for every variable and its condition $x_i \in S_i$ appearing in c. As they c-converge also their instances converge. Even if either $P : c$ or $Q : c$ does not irreducible, we can choose appropriate c_i from some split $c = \sqcup c_i$ such that $\langle P, Q\rangle : c_i$ c-converges.

b. $v = u \cdot w_0 \cdot w_1$ where $w_0 \cdot w_1 \notin \mathcal{O}(l_1)$ and $l_1/w_0 \in V$.

We assume that (r1) dose not erase the variable l_1/w_0. First $t'/v \cdot w_0$ reduces to its normal form s which exists from noetherian property. Subsequently all the subterms of t' which correspond to the variable reduce to s, and a term $\tilde{t}'$ is obtained. Similarly t'' reduces to $\tilde{t}''$ using $t/v \cdot w_0 \rightarrow^* s$. By quasi-closedness (r1) is applicable to $\tilde{t}'$ and $\tilde{t}'$ reduces to $\tilde{t}''$. When (r1) erases the variable l_1/w_0, the proof is similar to this proof.
□

Thus we obtain a sufficient condition on the confluence:

Theorem 5.2. (Main Theorem)
Let R be a noetherian quasi-closed membership conditional TRS, and R_c be its associated cTRS. If every c-critical pair in R_c c-converges with split, then R is confluent.

Proof.
It is clear from lemma 5.1 and noetherian property using lemma 2.6. □

Based on theorem 5.2, a completion algorithm can be designed as in the unconditional case ([6], [12]) and the other conditional cases (e.g., [10]).

Let a set of quasi-closed membership conditional equalities E and some reduction ordering $\succ$ be given as inputs of the algorithm. In this algorithm, selection of equality $m = n : c$ from E assumed to satisfy the *fairness hypothesis* in [6]. The hypothesis ensures every equality E will be selected within a finite number of steps.

Completion Algorithm

E : a set of quasi-closed MC-equalities (given)

R : a set of MC-rules (initially $= \phi$)
While $E \neq \phi$ do
begin
$f := m = n : c$;;; A candidate of a new rule, chosen from E.
$R^* := \{l_i \triangleright r_i : c_i\}$
;;; New rules, l_i, r_i are c-normal forms of $m : c_i$, $n : c_i$
;;; by the current rule set R and $l_i \succ r_i$. If c-normal forms
;;; of $m : c_i$ and $n : c_i$ are IN-comparable by $\succ$, then it stops with failure.
$R' := \{l' \triangleright r' : c' \in R \mid l' : c'$ or $r' : c'$ c-reducible by some $l_i \triangleright r_i : c_i \in R^*\}$
$R'_{eq} := \{l' = r' : c' \mid l' \triangleright r' : c' \in R'\}$
$R := R + R^* - R'$
$E := E - \{f\} + R'_{eq} + CP\langle R, R^*\rangle$
;;; $CP\langle R, R^*\rangle$ is all the c-critical pairs between the rules in new R and R^*.
end
return(R) ;;; $E = \phi$ and it stops with success, R is complete.

Now a completeness theorem on the completeness of this algorithm and its proof is shown.

Theorem 5.3.
For a given set of quasi-closed MC-equalities E and a reduction ordering $\succ$, if this algorithm stops and a membership conditional TRS R is returned, then R is complete, that is, noetherian and confluent. Moreover $=_E \equiv \sim_R$. Here $=_E$ and $\sim_R$ are equivalence relations generated by $=$ of E and $\rightarrow$ of R respectively.
Proof.
$E, R, \cdots$ in i-th loop are indicated by $E_i, R_i, \cdots$ with suffix i. We prove by the induction on i. Base case is clear and $i+1$ case is shown assuming i case. As $=_{E_{i+1}} \cup \sim_{R_{i+1}} \supset =_{E_i} \cup \sim_{R_i}$ clearly and its converse is also true by $=_{CP\langle R_i, R_i^*\rangle} \subset \sim_{R_i} \cup =_{f_i}$, we have $=_E \equiv \sim_R$. When $E = \phi$, R is locally confluent because there is no critical pair, and noetherian by the ordering used in the algorithm. □

This completion algorithm can be used as a proof method similar to inductionless induction in the unconditional case [7]. Under axioms described by a complete membership conditional TRS R, it can be proved whether or not $m = n : c$ is a theorem of R by trying to complete $R \cup \{m = n : c\}$. If the completion stops with success the equality is a theorem and if it stops with failure, it is not a theorem.

Example 5.4. (91-function)
McCarthy's 91-function f in [13] is recursively defined by

$$f(x) = \text{ if } x \leq 100 \text{ then } f(f(x+11)) \text{ else } x - 10$$

and has a property

$$f(x) = 91 \text{ for all } x \leq 100.$$

The definition of f can be reformulated by the following membership conditional equalities (1) and (2), the property by (3). Membership conditions are indicated by equalities or inequalities which define them. e.g., $x \leq 100$ means $x \ \varepsilon \ \{n \in \mathbb{N} \mid n \leq 100\}$.

$$E: \begin{cases} f(x) = x - 10 \quad : 101 \leq x & (1) \\ f(x) = f(f(x+11)) \quad : x \leq 100 & (2) \\ f(x) = 91 \quad : x \leq 100 & (3) \end{cases}$$

Now the property (3) is proved using our completion algorithm for membership conditional TRS. That is, our completion generates a complete set of MC-rules from the MC-equality system E. Before completing above E, we notice that our algorithm also succeeds in completing $\{(1),(2)\}$. From now on, the current MC-rule set is denoted by R and assume the order $\succ$ defined from $f \gg s \gg 0$ where s is the successor function and $\mathbb{N} = T(\{s,0\})$.

The following membership conditional TRS R is obtained by choosing (1) and (3) from E or reverse order.

$$R: \begin{cases} f(x) \triangleright x - 10 & : 101 \leq x \quad (1') \\ f(x) \triangleright 91 & : x \leq 100 \quad (3') \end{cases}$$

The last equality (2) in E is chosen as a new rule, and the equality is reduced as below by $(1')$ and $(3')$.

$$\begin{array}{ccccccc} f(x) : x \leq 100 & = & f(f(x+11)) : x \leq 100 & \twoheadrightarrow & f(x+1) : 90 \leq x \leq 100 & \twoheadrightarrow & x - 9 : x = 100 \\ \downarrow & & \downarrow & & \downarrow & & \downarrow \\ 91 : x \leq 100 & & f(91) : x \leq 89 & & 91 : 90 \leq x \leq 99 & & 91 : x = 100 \\ & & \downarrow & & & & \\ & & 91 : x \leq 89 & & & & \end{array}$$

This shows that the both sides of equality (2) reduce to the same, there remains no equality in E and our completion succeeds.

Thus the intended property of 91-function is proved, applying our completion algorithm to the membership conditional TRS which defines the function and states the property. If membership conditional TRS is not utilized, then f must be defined by the following unconditional TRS which includes many rules and takes much effort for completion.

$$\begin{cases} f(s^{11}(0)) \triangleright f(0) \\ f(s^{11}(s(0))) \triangleright f(s(0)) \\ f(s^{11}(s(s(0)))) \triangleright f(s(s(0))) \\ \quad \vdots \\ f(s^{11}(s^{91}(0))) \triangleright f(s^{91}(0)) \\ f(s^{101}(x)) \triangleright s^{91}(x) \end{cases}$$

Compared with this method, our method is simpler and more efficient as it has much fewer equalities and rules.

Remark 5.5.

This completion algorithm computes normal forms of c-terms in contextual critical pairs with split to know their convergence. However there appears a nonterminating computation of normal forms even in the following very simple membership conditional TRS R with two rules $f(s(x)) \triangleright f(x) : x \varepsilon \mathbb{N}$ and $f(x) \triangleright f(0) : x \varepsilon \mathbb{N}$.

$$\begin{array}{ccccccc} f(x) : x \,\varepsilon\, \mathbb{N} & \twoheadrightarrow & f(x') : x' \,\varepsilon\, \mathbb{N} & \twoheadrightarrow & f(x'') : x'' \,\varepsilon\, \mathbb{N} & \twoheadrightarrow & \cdots \\ \downarrow & & \downarrow & & \downarrow & & \\ f(0) : x = 0 & & f(0) : x = 1 & & f(0) : x = 2 & & \end{array}$$

where $x = s(x') = s(s(x'')) = \cdots$. This example shows that even if a membership conditional TRS R is noetherian, the cTRS which is associated with R is not noetherian for c-reduction with split. This phenomena is caused by extractions of infinite patterns in a membership condition, and is stated as a divergence in breadth in [11]. For treating infinite splits cases as above, the completion algorithm should have some mechanism which find inductive structures of infinite splits and execute inductive proofs on them. We believe that the techniques proposed in [14],[15], can be effectively applied to this purpose. This direction is our further work.

6. Conclusion

A completion algorithm for membership conditional TRS has been presented. A new framework for contextual rewriting of membership conditional TRS has been introduced based on [19]. A new technique of *split* has been utilized to show the confluence of membership conditional TRS, which cannot be proven without *split*. Finally the completion algorithm has been demonstrated by an inductive proof for a property of McCarthy's 91-function.

To extend *split* technique for the case of infinite splits is our further work. Since our membership conditional TRS has a close connection to the metarewriting system proposed by H. Kirchner in [11], we believe that split technique will be also effective to prove confluence of the metarewriting system.

Acknowledgement

The author is very grateful to Yoshihito Toyama for helpful discussions and valuable suggestions. He also thanks colleagues in Koyama Research Group of NTT Research Laboratories for their careful reading of the preliminary form of this paper.

References

[1] Bergstra, J. and Klop, J. : *Conditional Rewrite Rules*, J. of Comp.and Sys.Sci. 32 (1986), pp.323-362.

[2] Buchberger, B. : *History and Basic Features of Critical-pair/completion Procedure*, J. of Symbolic Computation 3 (1987), pp.3-38.

[3] Dershowitz, N. : *Termination of Rewriting*, J. of Symbolic Computation 3 (1987), pp.122-157.

[4] Dershowitz, N., Okada, M., and Shivakumar, G. : *Confluence of Conditional Rewrite Rules*, Springer LNCS 308 (1988), pp.31-44.

[5] Huet, G. : *Confluent Reductions : Abstract Properties and Applications to Term Rewriting Systems*, J. of ACM 27 (1980), pp.797-821.

[6] Huet, G. : *A complete Proof of Correctness of the Knuth-Bendix Completion Algorithm*, J. of Comp. and Sys. Sci. 31 (1981), pp.11-21.

[7] Huet, G. and Hullot, J.-M. : *Proofs by Induction in Equational Theories with Constructors*, J. of Comp. and Sys. Sci. 25 (1982), pp.239-266.

[8] Huet, G. and Oppen, C. : *Equations and Rewrite Rules : A Survey*, in R. Book (ed.) : Formal Language : Perspectives and Open Problems, Academic Press (1980), pp.349-405.

[9] Kaplan, S. : *Conditional Rewrite Rules*, Th. Comp. Sci. 33 (1984), pp.175-193.

[10] Kaplan, S. : *Simplifying Conditional Term Rewriting Systems : Unification, Termination and Confluence*, J. of Symbolic Computation 4 (1987), pp.295-334.

[11] Kirchner, H. : *Schematization of Infinite Sets of Rewrite Rules Generated by Divergent Completion Processes*, Th. Comp. Sci. 67 (1989), pp.303-332.

[12] Knuth, E. and Bendix, B. : *Simple Word Problems in Universal Algebras*, in J. Leech (ed.) : Computational Problems in Abstract Algebra, Pergamon Press (1970), pp.263-297.

[13] Manna, Z. et al. : *Inductive Methods for Proving Properties of Programs*, Proc. ACM Conf. on Proving Assertions on Programs (1972).

[14] Sakai, M., Sakabe, T. and Inagaki, Y. : *A Method for Inductive Properties of Algebraic Specification*, IEICE Japan Tech. Rep. COMP88-86 (1989).

[15] Thiel, J.-J. : *Stop Losing Sleep over Incomplete Data Type Specification*, Proc. of 11th POPL (1984), pp.76-82.

[16] Toyama, Y. : *Term Rewriting Systems with Membership Conditions*, Springer LNCS 308 (1988), pp.228-244.

[17] Toyama, Y. : *Membership Conditional Term Rewriting Systems*, Trans. IEICE Japan, E72 (1989), pp.1224-1229.

[18] Yamada, J. and Toyama, Y. : *Confluence of Membership Conditional TRS*, IEICE Japan Tech. Rep. COMP89-8 and IPSJ Tech. Rep. SF-89-28-8 (1989).

[19] Zhang, H. and Remy, J.-L. : *Contextual Rewriting*, Springer LNCS 202 (1985), pp.46-62.