Unification Problems Modulo a Theory of *Until*

(Extended Abstract)

Shreyaben Brahmakshatriya, Sushma Danturi, Kimberly A. Gero*and
Paliath Narendran*

University at Albany—SUNY, Albany, NY, USA
{brahmaks,sushmad,kgero001,dran}@cs.albany.edu

1 Introduction and Motivation

Our main aim in this paper is to investigate unifiability problems modulo an equational theory related to the idempotence property. The defining axioms of the theory are:

$$\begin{aligned} f(x,x) &= x \\ f(x,f(x,y)) &= f(x,y) \\ f(y,f(x,y)) &= f(x,y) \\ f(f(x,y),y) &= f(x,y) \\ f(f(x,y),x) &= f(y,x) \end{aligned}$$

We refer to this equational theory as W. We came across these axioms while studying identities or equivalences in Linear Temporal Logic (LTL) — a logic for reasoning about time with the "until" operator $\mathcal{U}$, but without the "next-time" operator $\bigcirc$[1]. It is not hard to see that the *until* operator $\mathcal{U}$ satisfies these identities. However, note that there are other models for these identities as well: the logical *and* ($\wedge$) and *or* ($\vee$) operations in boolean algebra are examples. Another is the "*max*" function over a domain such as $\mathbb{Z}$. In fact, any operator that has the properties of associativity, commutativity and idempotence (an **ACI**-operator) will suffice as a model. A more trivial (and hence less interesting) model is where f is interpreted as a projection function into the second argument.

In this paper we consider three computational problems for W. The first is the (usual) unifiability problem *with constants*, i.e., the terms to be unified will have free (uninterpreted) constants besides f. We show that the W-unifiability problem can be done in polynomial time if the input equations contain *at most* two constants. This was somewhat surprising, since the unifiability problem modulo idempotency (the first axiom) alone is NP-complete in the two constant case (see the reduction in [11]). The W-unifiability problem is NP-complete if the terms have 3 or more constants. The disunification problem modulo W is NP-complete *even in the two-constant case*. The asymmetric unifiability problem (a concept that was introduced only recently [7, 8]) modulo W is also NP-complete in the two-constant case.

We would like to note here that semantic unification problems modulo LTL have been investigated by Vladimir Rybakov (see [4, 14]). However, Rybakov's results cover a much more general theory—the entire LTL, in fact.

This is a preliminary report of our ongoing research. Some of the proofs are omitted due to lack of space. For more details and proofs please see our tech report [5].

*K. Gero and P. Narendran were supported in part by NSF grant CNS 09-05286.

[1] This subclass of LTL is called "stutter-free" LTL [9, 13]

2 Notation and Preliminaries

We assume the reader is familiar with the usual notions and concepts in term rewriting systems [1] and equational unification [3]. A rewrite rule $l \to r$ is *optimally reducing* if and only if for any substitution θ for which $\theta(r)$ is R-reducible, there is a proper subterm s of l such that $\theta(s)$ is R-reducible. A rewrite system R is optimally reducing if and only if every rule in R is optimally reducing [12]. It can be shown that the unifiability problem for convergent, optimally reducing term rewrite systems is NP-complete [12]. (This result was generalized by Comon-Lundh and Delaune in [6].)

3 The Convergent System

The equational theory W has the following convergent term rewriting system R:

$$f(x,x) \to x \quad (1) \qquad f(f(x,y),y) \to f(x,y) \quad (4)$$
$$f(x,f(x,y)) \to f(x,y) \quad (2) \qquad f(f(x,y),x) \to f(y,x) \quad (5)$$
$$f(y,f(x,y)) \to f(x,y) \quad (3) \qquad f(f(x,y),f(y,x)) \to f(y,x) \quad (6)$$

Lemma 1. *R is optimally reducing.*

4 The Two Constants Case is in Polynomial Time

Here we consider the unification problem modulo W in which there are only two constants, say a and b. In other words, we consider only terms from $T(\{f,a,b\},V)$ where V is a denumerable set of variables. We show that this problem can be done in polynomial time.

Some preliminary results are given below.

Proposition 1. *There are only 4 ground normal forms, namely, a, b, $f(a,b)$ and $f(b,a)$. Every ground term in $T(\{f,a,b\})$ reduces to one of these.*

Proof-sketch. By structural induction. □

Proposition 2. *Every term of the form $f(s,b)$ reduces to either b or $f(a,b)$. In particular, for all ground terms s, $f(s,b)$ reduces to $f(a,b)$ if and only if s contains a.*

Proof. Since the normal form of s has to be one of $\{a,\ b,\ f(a,b),\ f(b,a)\}$ the first part of the proposition follows easily. On the other hand, if s does not contain a, then $s \to^! b$ and $f(s,b) \to^! b$. In all other cases, namely $s \to^! a$, $s \to^! f(a,b)$ and $s \to^! f(b,a)$, the normal form of $f(s,b)$ is $f(a,b)$. □

Similarly,

Proposition 3. *Every term of the form $f(s,a)$ reduces to either a or $f(b,a)$. In particular, for all ground terms s, $f(s,a)$ reduces to $f(b,a)$ if and only if s contains b.*

Proposition 4. *(i) Every term of the form $f(s,\ f(a,b))$ reduces to $f(a,b)$.*
(ii) Every term of the form $f(s,\ f(b,a))$ reduces to $f(b,a)$.

Proof. Straightforward. □

Proposition 5. *(i) Let $t \to^! a$ and t' be a term obtained from t by replacing an occurrence of a by $f(b,a)$. Then $t' \to^! f(b,a)$.*
(ii) Let $t \to^! b$ and t'' be a term obtained from t by replacing an occurrence of b by $f(a,b)$. Then $t'' \to^! f(a,b)$.

Proof. (i) By induction on $|t|$: if $|t| = 1$, then $t = a$ and the result trivially follows. Let $t = f(t_1, t_2)$ for some t_1 and t_2. Then it must be that $t_1 \to^! a$ and $t_2 \to^! a$. If the occurrence of a that is replaced by $f(b,a)$ is in t_1, then by the induction hypothesis $t_1' \to^! f(b,a)$ where t_1' is the term obtained from t_1 by replacing an occurrence of a by $f(b,a)$. Thus $t' = f(t_1', t_2) \to^* f(f(b,a), a) \to f(b,a)$.

The case where the occurrence of a that is replaced by $f(b,a)$ is in t_2 is similar, and we get $t' \to^* f(a, f(b,a)) \to f(b,a)$.

(ii) The proof is similar to that of the case above. □

Proposition 6. *Let $t \to^! s$ where $s \in \{f(a,b),\ f(b,a)\}$, and t' be a term obtained from t by either replacing an occurrence of a by $f(b,a)$, or b by $f(a,b)$. Then $t' \to^! s$.*

We omit the proof here and instead refer the reader to [5].

Proposition 7. *Let $f(s,a)$ be a term in normal form and let θ be a substitution whose range is $\{f(a,b),\ f(b,a)\}$. Then $\theta(f(s,a)) \to^! f(b,a)$.*

Proof. By Proposition 3, we have that every term of the form $f(s,a)$ reduces to either a or $f(b,a)$. Since the range of θ does not include a we have that $\theta(f(s,a)) \to^! f(b,a)$. □

Similarly,

Proposition 8. *Let $f(s,b)$ be a term in normal form and let θ be a substitution whose range is $\{f(a,b),\ f(b,a)\}$. Then $\theta(f(s,b)) \to^! f(a,b)$.*

We now consider equations which are in one of the following forms given below. It is assumed that the terms in an equation are in normal form modulo the rewrite system above. We will call the following equations "flexible equations".

(a) $s =^? t$ where both s and t are non-ground terms.
(b) $s =^? f(a,b)$ where s is a non-ground term.
(c) $s =^? f(b,a)$ where s is a non-ground term.

Proposition 9. *Let $s =^? t$ be a flexible equation that is unifiable. Then there is a unifier whose range is $\{f(a,b),\ f(b,a)\}$.*

Proof-sketch. The key idea here is that every replacement of the forms $x \mapsto a$ and $x \mapsto b$, can be changed to $x \mapsto f(b,a)$ and $x \mapsto f(a,b)$ respectively. □

Proposition 10. *An equation of the form $f(s,a) =^? f(t,b)$ is not unifiable.*

Proof. This follows from Propositions 2 and 3, since every term of the form $f(s,a)$ reduces to either a or $f(b,a)$ and every term of the form $f(t,b)$ reduces to either b or $f(a,b)$. □

Proposition 11. *Let $f(s_1, s_2) =^? t$ be a flexible equation where s_2 is not equal to either a or b. Then a substitution θ with range $\{f(a,b),\ f(b,a)\}$ is a unifier of $f(s_1, s_2) =^? t$ if and only if it is a unifier of $s_2 =^? t$.*

Proof. Since s_2 is not equal to either a or b, we need to consider three possibilities: (a) s_2 is a non-ground term, (b) $s_2 = f(a,b)$, or (c) $s_2 = f(b,a)$. If it is a non-ground term, then the variables are replaced by either $f(a,b)$ or $f(b,a)$, and hence the normal form of $\theta(s_2)$ must be either $f(a,b)$ or $f(b,a)$. By Proposition 4, any term of the form $f(s,\ f(a,b))$ or $f(s,\ f(b,a))$ reduces to $f(a,b)$ or $f(b,a)$ respectively and the result follows. Cases (b) and (c) can be dealt similarly. □

Proposition 12. *Let $f(s,\gamma)\ =^?\ f(t,\gamma)$ be a flexible equation where $\gamma \in \{a,b\}$. Then* any *substitution with range $\{f(a,b),\ f(b,a)\}$ is a unifier of the equation.*

Proof. Follows from Propositions 2 and 3. □

Thus the key ideas behind our algorithm are as follows (for more details see Appendix A):

1. Eliminate equations of the form $s =^? a$ and $s =^? b$ first by replacing every variable in them with a or b respectively.
2. For flexible equations, we proceed as follows:
   (a) For equations $x =^? y$ we replace x with y (if x occurs elsewhere).
   (b) Let $f(s_1,s_2)$ be any non-ground term that appears either as a left-hand side or as a right-hand side in an equation.
      i. If $s_2 = a$, then we replace $f(s_1,s_2)$ by $f(b,a)$
      ii. If $s_2 = b$, then we replace $f(s_1,s_2)$ by $f(a,b)$
      iii. Otherwise we replace $f(s_1,s_2)$ by s_2.

If none of these steps is applicable, then if the set of equations is in solved form, then the algorithm terminates with success; otherwise it terminates with failure.

Theorem 2. *Two Constants Case algorithm is sound and complete for unifiability.*

Proof. Only the 3 steps in 2(b) need explanation. The first case follows from Proposition 3. The second follows from Proposition 2. The third follows from Proposition 11. □

Theorem 3. *Two Constants Case algorithm terminates.*

Proof. Again, only the steps in 2(b) need be considered. Since we are dealing with a non-ground term, either the number of variable occurrences will go down (Cases i and ii) or the term size will (Case iii). □

5 NP-Completeness With At Least 3 Constants

Lemma 4. *Let a, b, c be constants. Then the only (normalized) unifiers for the following set of equations*

$$f(f(c,X), f(f(c,a), f(c,b))) =^?_W\ f(f(c,a), f(c,b))$$
$$f(X, f(a,b)) =^?_W\ f(a,b)$$

are $\{X \mapsto a\}$ and $\{X \mapsto b\}$.

The above lemma lets us force values assigned to variables to be from $\{a,b\}$. Now consider the equations

$$f(f(X,Y), f(Z,b)) =^?_W\ f(a,b)$$

$$f(f(X,Y),f(Z,a)) =^?_W f(b,a)$$

where the range of the substitutions allowed is restricted to $\{a,b\}$. Then the following are the only normalized solutions[2]:

$$\sigma_1 = \{X \mapsto a, Y \mapsto a, Z \mapsto b\} \qquad \sigma_2 = \{X \mapsto a, Y \mapsto b, Z \mapsto a\}$$
$$\sigma_3 = \{X \mapsto a, Y \mapsto b, Z \mapsto b\} \qquad \sigma_4 = \{X \mapsto b, Y \mapsto a, Z \mapsto a\}$$
$$\sigma_5 = \{X \mapsto b, Y \mapsto b, Z \mapsto a\} \qquad \sigma_6 = \{X \mapsto b, Y \mapsto a, Z \mapsto b\}$$

Lemma 5. *Unifiability modulo W is NP-complete.*

Proof-sketch: NP-hardness can be shown by reduction from Not-All-Equal-3SAT [10] which is known to be NP-complete. Let U be a an instance of Not-All-Equal-3SAT. For each propositional variable p we form a respective term variable V_p and equations

$$f(f(c,V_p),\ f(f(c,a),\ f(c,b))) =^?_W f(f(c,a),\ f(c,b))$$
$$f(V_p, f(a,b)) =^?_W f(a,b)$$

and for each clause $C_i = p_i \vee q_i \vee r_i$ we form the following equations where X_i, Y_i, and Z_i are the term variables that correspond respectively to the propositional variables p_i, q_i and r_i:

$$f(f(X_i,Y_i),f(Z_i,b)) =^?_W f(a,b) \qquad (\alpha_i)$$
$$f(f(X_i,Y_i),f(Z_i,a)) =^?_W f(b,a) \qquad (\beta_i)$$

Membership in NP follows from the fact that R is optimally reducing. □

6 Disunification

Disunification is the problem of deciding satisfiablity of a system of equations and disequations with respect to a given equational theory. We will be using the notation for disunification as given in [2]. Let $Sig(E)$ denote the signature (function symbols) of an equational theory E. We will consider the (E,Σ)-disunification problem with constants, where Σ is a finite superset of $Sig(E)$ and $\Sigma \setminus Sig(E)$ is a finite set of constants.

Lemma 6. *(W,Σ)-disunifiability is NP-hard even when the input equations and disequations have only two uninterpreted constants, i.e. $|\Sigma \setminus sig(E)| = 2$.*

Proof. This proof is also by a reduction from Not-All-Equal-3SAT. The reduction uses equations (α_i) and (β_i) for each clause (as in the proof of Lemma 5) along with the disequations $V \neq f(a,b)$, $V \neq f(b,a)$ for each variable V. This alleviates the need for an extra constant c, hence we do not need Lemma 4. □

7 A Brief Foray into Asymmetric Unification

Let R be a convergent rewrite system. An equation $s =^?_\downarrow t$ is asymmetrically unifiable modulo R if and only if there is a substitution θ such that $\theta(s) \to^!_R \theta(t)$. In other words, θ must be an R-unifier *and* $\theta(t)$ must be in normal form. The asymmetric unification problem modulo R is defined as follows:

[2] Note that every substitution replaces at least one of X, Y, or Z by a and at least one by b

Instance: A set of equations $\{s_1 \ =^?_\downarrow \ t_1, \ldots, s_n \ =^?_\downarrow \ t_n\}$

Question: Does there exist a substitution θ such that $\theta(s_i) \ \rightarrow^!_R \ \theta(t_i)$ for all i?

Lemma 7. *The only asymmetric unifiers for the equations*

$$f(f(a,b), f(X,Y)) \ =^?_\downarrow \ f(X,Y)$$
$$f(f(X,Y), f(a,b)) \ =^?_\downarrow \ f(a,b)$$

are $\{X \mapsto a, \ Y \mapsto b\}$ *and* $\{X \mapsto b, \ Y \mapsto a\}$.

The reduction again uses equations (α_i) and (β_i) for each clause as in the proof of Lemma 5. To restrict the range to $\{a, b\}$ we form the following equation for each variable V, where V' is a new variable:

$$f(f(a,b), f(V,V')) \ =^?_\downarrow \ f(V,V')$$

Lemma 8. *Asymmetric unifiability modulo W is NP-hard even when the input equations and disequations have only two uninterpreted constants.*

Acknowledgements: We wish to thank the referees for their incisive comments which helped considerably to improve the paper.

References

[1] Franz Baader and Tobias Nipkow. *Term Rewriting and All That.* Cambridge University Press, 1999.

[2] Franz Baader and Klaus U. Schulz. Combination techniques and decision problems for disunification. *Theoretical Computer Science*, 142(2):229–255, 1995.

[3] Franz Baader and Wayne Snyder. Unification Theory. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*, pages 440–526. Elsevier Science Publishers BV, 1999.

[4] Sergey Babenyshev and Vladimir V. Rybakov. Unification in linear temporal logic LTL. *Annals of Pure and Applied Logic*, 162(12):991–1000, 2011.

[5] Shreyaben Brahmakshatriya, Sushma Danturi, Kimberly A. Gero, and Paliath Narendran. Unification Problems Modulo a Theory of *Until*. Technical Report SUNYA-CS-13-02, University at Albany — SUNY(USA), http://www.cs.albany.edu/ ncstrl/treports/-Data/README.html, 2013.

[6] Hubert Comon-Lundh and Stéphanie Delaune. The finite variant property: How to get rid of some algebraic properties. In Jürgen Giesl, editor, *RTA*, volume 3467 of *Lecture Notes in Computer Science*, pages 294–307. Springer, 2005.

[7] Serdar Erbatur, Santiago Escobar, Deepak Kapur, Zhiqiang Liu, Christopher Lynch, Catherine Meadows, José Meseguer, Paliath Narendran, Sonia Santiago, and Ralf Sasse. Effective symbolic protocol analysis via equational irreducibility conditions. In Sara Foresti, Moti Yung, and Fabio Martinelli, editors, *ESORICS*, volume 7459 of *Lecture Notes in Computer Science*, pages 73–90. Springer, 2012.

[8] Serdar Erbatur, Santiago Escobar, Deepak Kapur, Zhiqiang Liu, Christopher Lynch, Catherine Meadows, José Meseguer, Paliath Narendran, Sonia Santiago, and Ralf Sasse. Asymmetric unification: A new unification paradigm for cryptographic protocol analysis. In *24th International Conference on Automated Deduction (CADE24)*, 2013. To be presented.

[9] Kousha Etessami. A note on a question of Peled and Wilke regarding stutter-invariant LTL. *Inf. Process. Letters*, 75(6):261–263, 2000.

[10] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979.

[11] Deepak Kapur and Paliath Narendran. Complexity of unification problems with associative-commutative operators. *J. Autom. Reasoning*, 9(2):261–288, 1992.

[12] Paliath Narendran, Frank Pfenning, and Richard Statman. On the unification problem for Cartesian Closed Categories. *J. Symb. Logic*, 62(2):636–647, 1997.

[13] Doron Peled and Thomas Wilke. Stutter-invariant temporal properties are expressible without the next-time operator. *Inf. Process. Letters*, 63(5):243–246, 1997.

[14] Vladimir V. Rybakov. Writing out unifiers in linear temporal logic. *J. Logic and Computation*, 22(5):1199–1206, 2012.

A Unification Algorithm in the Two-Constants Case

Our algorithm is given in terms of inference rules. These are given in order of priority: rule (L1) is applied most eagerly and so on. We assume that every term is in normal form and we treat equations as unordered pairs.

(L1) ***Remove Redundancies*:**

$$\frac{\mathcal{EQ} \uplus \{ s =^? s \}}{\mathcal{EQ}} \qquad s \text{ is any term}$$

(L2) ***Variable Elimination*:**

$$\frac{\mathcal{EQ} \uplus \{ U =^? V \}}{\{U \mapsto V\}(\mathcal{EQ}) \cup \{ U =^? V \}} \qquad \text{if } U \text{ occurs in } \mathcal{EQ}$$

(L3.a) RHS is a

$$\frac{\mathcal{EQ} \uplus \{ s =^? a \}}{\{X \mapsto a\}(\mathcal{EQ} \uplus \{ s =^? a \})} \qquad X \in Var(s)$$

(L3.b) RHS is b

$$\frac{\mathcal{EQ} \uplus \{ s =^? b \}}{\{X \mapsto b\}(\mathcal{EQ} \uplus \{ s =^? b \})} \qquad X \in Var(s)$$

(L4)

$$\frac{\mathcal{EQ} \uplus \{ f(s,a) =^? t \}}{\mathcal{EQ} \cup \{ f(b,a) =^? t \}} \qquad s \neq b$$

(L5)

$$\frac{\mathcal{EQ} \uplus \{ f(s,b) =^? t \}}{\mathcal{EQ} \cup \{ f(a,b) =^? t \}} \qquad s \neq a$$

(L6)

$$\frac{\mathcal{EQ} \uplus \{ f(s_1,s_2) =^? t \}}{\mathcal{EQ} \cup \{ s_2 =^? t \}} \qquad s_2 \neq a \text{ and } s_2 \neq b$$

As mentioned earlier, if none of the above rules is applicable and the set of equations is not in solved form, then we fail. Thus no explicit failure rules are really needed. However, the following failure rules may be applied eagerly for the sake of efficiency:

(F1) $$\frac{\mathcal{EQ} \uplus \{ s =^? t \}}{\textbf{FAIL}} \qquad s,t \text{ ground and } s \neq t$$

(F2) $$\frac{\mathcal{EQ} \uplus \{ f(s,a) =^? f(t,b) \}}{\textbf{FAIL}}$$