32nd International Workshop on Unification (UNIF 2018), Informal Proceedings
Oxford, 7th July, 2018

Towards Generalization Methods for Purely Idempotent Equational Theories *

David M. Cerna and Temur Kutsia

Research Institute for Symbolic Computation (RISC)
Johannes Kepler University, Linz
{David.Cerna , Temur.Kutsia}@risc.jku.at

Abstract

In *Generalisation de termes en theorie equationnelle. Cas associatif-commutatif*, a pair of terms was presented over the language $\{f(,),g(,),a,b\}$, where f and g are interpreted over an idempotent equational theory, i.e. $g(x,x) = x$ and $f(x,x) = x$, resulting in an infinite set of generalizations. While this result provides an answer to the complexity of the idempotent generalization problem for arbitrarily idempotent equational theories (theories with two or more idempotent functions) the complexity of generalization for equational theories with a single idempotent function symbols was left unsolved. We show that the two idempotent function symbols example can be encoded using a single idempotent function and two uninterpreted constants thus proving that idempotent generalization, even with a single idempotent function symbol, can result in an infinite set of generalizations. Based on this result we discuss approaches to handling generalization within idempotent equational theories.

1 Introduction

Anti-unification or term generalization algorithms aim at computing generalizations for given terms. A generalization of t and s is a term r such that t and s are substitution instances of r, i.e. there exists σ and μ such that $r\sigma = t$ and $r\mu = s$. Interesting generalizations are those that are least general (lggs). However, it is not always possible to have a unique lgg. In these cases the task is either to compute a minimal complete set of generalizations, or to impose restrictions so that uniqueness is guaranteed.

In particular, we consider anti-unification problems which allow equational interpretations of the function symbols and constants present in the term signature. This is known as equational anti-unification or E-generalization. When the equational theory does not interpret any of the function symbols or constants in the term signature the resulting generalizations are referred to as syntactic. For most of the commonly considered equational theories the minimal complete set of generalizations is finite, for example, theories including commutativity and associative function symbols discussed in [1]. However, as pointed out in [8], this need not be the case. A pair of terms constructed from the signature $\{f(\cdot,\cdot),g(\cdot,\cdot),a,b\}$ where f and g are interpreted as idempotent functions resulted in an infinite set of generalizations, though it was not shown to be the minimal complete set. While the case of two idempotent function symbols was addressed in [8], the case of generalization for terms constructed from a signature with a single

*This research is supported by the FWF project P28789-N32.

idempotent function symbol, i.e. $\{f(\cdot,\cdot),a,b\}$ was not discussed. This gap implies an interesting question concerning modular generalization algorithms like the ones discussed in [1].

The result reported in this paper has been motivated by its influence on developing a combination method for signature-disjoint generalization theories. Namely, as shown by Pottier in [8], anti-unification problems with two idempotent function symbols may have infinitely many incomparable generalizations. If anti-unification problems with one idempotent symbol had only finitely many incomparable solutions, it would be a serious problem for the prospect of developing a combination method: finitary generalization algorithms would have been impossible to combine. However, our result shows that it is not the case.

Combination methods for unification algorithms, constraint solvers, and decision procedures have been studied in detail [2, 3, 5, 4, 6, 9, 10]. Though surprisingly, it has been shown that term generalization when the signature contains a function which is associative, commutative and idempotent is finitary. This follows from Theorem 2 of [7]. Such varying results provide motivations for investigating term generalization as discussed in this paper and removes an obstacle to study such methods for generalization algorithms as well.

2 Preliminaries

We now outline the basic concepts needed to understand term generalizations and the results outlined in later sections. Our term language $\mathscr{L}$ is built from a finite signature of function and constant symbols Σ and a countable set variable symbols $\mathscr{V}$. Function symbols have a fixed arity, i.e. the number of arguments, and constant symbols are essentially function symbols with arity zero. If necessary, we denote the set of variables of a term t by $\mathrm{Vars}(t)$.

Each symbol $f \in \Sigma$ in the signature has an associated equational theory $Ax(f)$. When $Ax(f)$ is empty the function or constant symbol is left uninterpreted. We will only consider in this work function symbols f interpreted as *idempotent*, $Ax(f) = \{\mathbf{I}\}$, that is binary functions such that $f(x,x) = x$.

When two terms s and t are equivalent over an equational theory $\mathscr{E}$ we write $s =_{\mathscr{E}} t$. In this work we will only consider the equational theory $\mathbf{I}_F$ where F is a set of function symbols interpreted as idempotent.

A *Substitution* is a finite set of pairs $\{X_1 \mapsto t_1, \ldots, X_n \mapsto t_n\}$ where X_i is a variable, t_i is a term, and the X's are pairwise distinct variables. The notions of substitution *domain* and *range* are also standard and are denoted, respectively, by Dom and Ran.

We use postfix notation for substitution applications, writing $t\sigma$ instead of $\sigma(t)$. As usual, the application $t\sigma$ affects only the occurrences of variables from $\mathrm{Dom}(\sigma)$ in t. The *composition* of σ and ϑ is written as juxtaposition $\sigma\vartheta$ and is defined as $x(\sigma\vartheta) = (x\sigma)\vartheta$ for all x.

A substitution σ_1 is *more general* than σ_2, written $\sigma_1 \preceq \sigma_2$, if there exists ϑ such that $X\sigma_1\vartheta = X\sigma_2$ for all $X \in \mathrm{Dom}(\sigma_1) \cup \mathrm{Dom}(\sigma_2)$. The strict part of this relation is denoted by $\prec$. The relation $\preceq$ is a partial order and generates the equivalence relation which we denote by $\simeq$. We overload $\preceq$ by defining $s \preceq t$ if there exists a substitution σ such that $s\sigma = t$. The focus of this work is generalization in the presence of equational axioms thus we need a more general concept of ordering of substitutions/terms by their generality. We say for two terms s,t are $s =_{\mathscr{E}} t$ if they are equivalent modulo $\mathscr{E}$. Under this notion of equality we can say that a substitution σ_1 is *more general modulo an equational theory* $\mathscr{E}$ than σ_2 written $\sigma_1 \preceq_{\mathscr{E}} \sigma_2$ if there exists ϑ such that $X\sigma_1\vartheta =_{\mathscr{E}} X\sigma_2$ for all $X \in \mathrm{Dom}(\sigma_1) \cup \mathrm{Dom}(\sigma_2)$ Note that $\prec$ and $\simeq$ and the term extension are generalized accordingly. From this point on we will use the ordering relation modulo an equational theory when discussing generalization.

A term t is called a *generalization* or an *anti-instance* modulo an equational theory $\mathscr{E}$ of two terms t_1 and t_2 if $t \preceq_{\mathscr{E}} t_1$ and $t \preceq_{\mathscr{E}} t_2$. It is the *least general generalization* (lgg in short), aka a *most specific anti-instance,* of t_1 and t_2, if there is no generalization s of t_1 and t_2 which satisfies $t \prec_{\mathscr{E}} s$.

An *anti-unification problem* (Briefly AUP) is a triple $X : t \triangleq s$ where t and s are terms constructed from the signature Σ, and X does not occur in t and s. The variable X is called a *generalization variable.*

Generalization variables are written with capital letters $X, Y, Z, \ldots$. Note that generalization variables are not used explicitly in this work but they sere syntactic purpose in most algorithms defined in literature, thus we keep them to conform with common syntactic expressions. The size of a set of AUPs is defined as $|\{X_1 : t_1 \triangleq s_1, \ldots, X_n : t_n \triangleq s_n\}| = \sum_{i=1}^{n} |t_i| + |s_i|$. A *generalization* of an AUP $X : t \triangleq s$ is a term r such that there exists substitutions σ_1 and σ_2 such that $\mathrm{Dom}(\sigma_1) = \mathrm{Dom}(\sigma_2) = \mathscr{V}(r)$ and $r\sigma_1 = t$ and $r\sigma_2 = s$

A generalization r of $X : t \triangleq s$ is *least general* (or *most specific*) modulo an equational theory $\mathscr{E}$ if there is no generalization r' of $X : t \triangleq s$ such that $r \prec_{\mathscr{E}} r'$.

3 Idempotent Generalization with two symbols

We will now consider an alphabet $\Sigma = \{f(\cdot,\cdot), g(\cdot,\cdot), a, b\}$. Both f and g are idempotent function symbols (our equational theory is $\mathscr{E} = \mathbf{I}_{f,g}$), that is $f(x,x) =_{\mathbf{I}_{f,g}} x$ and $g(x,x) =_{\mathbf{I}_{f,g}} x$. Now we consider the following generalization problem:

$$X : f(a,b) \triangleq g(a,b) \tag{1}$$

The seemingly simple generalization problem of Equation 1 results in an infinite set of least general generalizations. This follows from the production of the first two least general generalizers $g(f(a,x), f(y,b))$ and $f(g(a,x), g(y,b))$ which we refer to as G_1 and G_2, respectively. It is quite simple to check that these two terms are indeed generalizers and are least general. In [8], an infinite set of generalizations was produced by the following recursive construction:

$$S_0 = G_1 \qquad\qquad S_{n+1} = f(G_1, g(S_n, G_2)) \tag{2}$$

Notice that the generalizer produced at each step is least general and is not comparable with the generalizers produced at the previous step and thus, the construction produces an infinite sequence of incomparable least general generalizers. However this is not the minimal complete set being that the construction is limited to repeated use of $\{G_1, G_2\}$. Any previously constructed generalizer can be used. Essentially, let $h \in \{f, g\}$ and $\mathscr{S}$ the set of least general generalizations of Equation 1, then $h(S_1, S_2)$ is a least general generalizations of Equation 1 when S_1 is distinct from S_2. We elucidate this construction further after presenting our solution to the generalization problem for one idempotent function symbol.

4 Idempotent Generalization with a single function symbol

We will now consider an alphabet $\Sigma = \{h(\cdot,\cdot), a, b\}$ where h is an idempotent function symbol (our equational theory is $\mathscr{E} = \mathbf{I}_h$). Our goal is to show that a term signature with a single binary function symbol interpreted as idempotent also allows the construction of AUPs with infinitely many lggs. We solve this problem by encoding the two symbol case into the one symbol case. Essentially we write $f(\cdot,\cdot)$ as $h(a, h(\cdot,\cdot))$ and $g(\cdot,\cdot)$ as $h(b, h(\cdot,\cdot))$. Thus, the generalization problem from the previous section is now:

$$h(a, h(a,b)) \triangleq h(b, h(a,b)) \tag{3}$$

The reader might notice right away that this has a solution $h(x, h(a,b))$, however, this solution isn't of much interest to us because we cannot produce an infinite construction using it alone, but it can be considered as one of the least generalizers within the construction. Also, it happens to be the case that there are two additional least general generalizers which are incomparable to it. These generalizers, which are incomparable to $h(x, h(a,b))$, are as follows:

$$G_1' = h(h(x, h(x,b)), h(a, h(x,b))) \qquad G_2' = \qquad h(f(x, h(a,x)), h(h(x,b), h(a,b)))$$

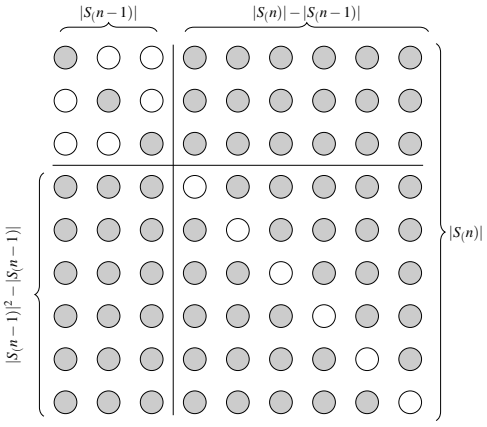

Figure 1: Geometric proof of Theorem 1 for $|A| = 3$.

Notice that these generalizers are even simpler than those produced in the previous example given that the domain of the substitutions contain a single variable x. Furthermore, this variable is substituted by one of the two constants. Using the recursive construction outlined in Equation 2, replacing G_1 and G_2 by G'_1 and G'_2 we produce a similar infinite set as in the Pottier example.

Concerning the construction of all least general generalizations constructable from the set $\{h(x,h(a,b)), G'_1, G'_2\}$ consider the following theorem.

Theorem 1. *Let A be a finite set, $P(S,S') = \{(a,b) | a \in S \ , \ b \in S' \ , \ a \neq b\}$, and S_n the following recursive set construction:$S_0 = \{\emptyset\}$, $S_1 = A$, and $S_{(n+1)} = S_n \cup P(S_n, S_n)$. Then for $n \geq 1$, $|S_{(n+1)}| = |S_n|^2 - |S_{(n-1)}|^2 + |S_{(n-1)}|$.*

Proof. Let us consider the case of $S_2 = S_1 \cup P(S_1,S_1)$ we know that $|S_1| = |A|$ and that $|P(S_1,S_1)| = |A|^2 - |A|$ because $(a,a) \notin P(S_1,S_1)$ for $a \in A$. Thus, $|S_2| = |A|^2$ which is precisely given by the formula in the theorem $|S_2| = |S_1|^2 - |S_0|^2 + |S_0| = |A|^2 - 1 + 1 = |A|^2$. For the induction hypothesis, let us assume the theorem holds for S_n and show that it holds for $S_{(n+1)}$. We know that S_n contains $S_{(n-1)}$ by definition and thus we can consider the subsets of S_n, $S_{(n-1)}$ and $S_n \setminus S_{(n-1)}$, Note that the elements of $P(S_{(n-1)}, S_{(n-1)})$ are already members of $S_n \setminus S_{(n-1)}$ and thus do not need to be considered. But we do need to consider the following cases $P(S_{(n-1)}, S_n \setminus S_{(n-1)})$, $P(S_n \setminus S_{(n-1)}, S_{(n-1)})$, $P(S_n \setminus S_{(n-1)}, S_n \setminus S_{(n-1)})$ which have size $|S_{(n-1)}|(|S_n| - |S_{(n-1)}|)$, $|S_{(n-1)}|(|S_n| - |S_{(n-1)}|)$, $(|S_n| - |S_{(n-1)}|)^2 - (|S_n| - |S_{(n-1)}|)$, respectively. Thus, we get that the size $|S_{(n+1)}|$ is the following:

$$2 \cdot |S_{(n-1)}|(|S_n| - |S_{(n-1)}| + (|S_n| - |S_{(n-1)}|)^2 - (|S_n| - |S_{(n-1)}|) + |S_n| =$$

$$2|S_{(n-1)}||S_n| - 2|S_{(n-1)}|^2 + (|S_n|^2 - 2|S_n||S_{(n-1)}| + |S_{(n-1)}|^2 + |S_{(n-1)}| = |S_n|^2 - |S_{(n-1)}|^2 + |S_{(n-1)}|$$

Proving the induction step. See Figure 1 for a geometric proof of the theorem. □

Concerning the $O(2^{2^n})$ growth rate in terms of the initial set size $|S_1|$, consider the ratio between the smaller square's area $|S(n-1)|$ and the larger square's $(|S(n-1)|^2 - |S(n-1)|)^2$ which is precisely $1 : O(|S_{n-1}|^2)$. Iterating this provides $O(2^{2^n})$ growth rate.

5 Conclusion

We have shown that even a simple term signature with a single binary function interpreted as idempotent results in an infinite set of generalizations. Theorem 1 provides information concerning the growth of the set of least general generalizations in terms of the number of nestings of idempotent function symbols. Further analysis provides a growth rate of $O(2^{2^n})$ in terms of the number of nested function symbols. This implies that the minimal complete set of generalizations is at least as large as this construction and thus infinite. However, we have not provided a precise construction of the minimal complete set of generalizations, only a lower bound. In future work we plan to investigate the construction of the minimal complete set of generalizations and hopefully find a precise expression of its construction as well as an understanding of modular algorithms for idempotent generalization.

6 References

References

[1] M. Alpuente, S. Escobar, J. Espert, and J. Meseguer. A modular order-sorted equational generalization algorithm. *Inf. Comput.*, 235:98–136, 2014.

[2] F. Baader and K. U. Schulz. Combination techniques and decision problems for disunification. *Theor. Comput. Sci.*, 142(2):229–255, 1995.

[3] F. Baader and K. U. Schulz. Unification in the union of disjoint equational theories:combining decision procedures. *J. Symb. Comput.*, 21(2):211–243, 1996.

[4] F. Baader and K. U. Schulz. Combining constraint solving. In H. Comon, C. Marché, and R. Treinen, editors, *Constraints in Computational Logics: Theory and Applications, International Summer School, CCL'99 Gif-sur-Yvette, France, September 5-8, 1999, Revised Lectures*, volume 2002 of *Lecture Notes in Computer Science*, pages 104–158. Springer, 1999.

[5] P. Chocron, P. Fontaine, and C. Ringeissen. A gentle non-disjoint combination of satisfiability procedures. In *IJCAR 2014*, pages 122–136, 2014.

[6] S. Erbatur, D. Kapur, A. M. Marshall, P. Narendran, and C. Ringeissen. Unification and matching in hierarchical combinations of syntactic theories. In C. Lutz and S. Ranise, editors, *Frontiers of Combining Systems - 10th International Symposium, FroCoS 2015, Wroclaw, Poland, September 21-24, 2015. Proceedings*, volume 9322 of *Lecture Notes in Computer Science*, pages 291–306. Springer, 2015.

[7] B. Konev and T. Kutsia. Anti-unification of concepts in description logic EL. In C. Baral, J. P. Delgrande, and F. Wolter, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Fifteenth International Conference, KR 2016, Cape Town, South Africa, April 25-29, 2016.*, pages 227–236. AAAI Press, 2016.

[8] L. Pottier. Generalisation de termes en theorie equationnelle. cas associatif-commutatif. Research Report 1056, INRIA, 1989.

[9] M. Schmidt-Schauß. Unification in a combination of arbitrary disjoint equational theories. *J. Symb. Comput.*, 8(1/2):51–99, 1989.

[10] C. Tinelli and C. Ringeissen. Unions of non-disjoint theories and combinations of satisfiability procedures. *Theor. Comput. Sci.*, 290(1):291–353, 2003.