On solving nominal fixpoint equations

Mauricio Ayala-Rincón[1], Washington de Carvalho-Segundo[1], Maribel Fernández[2], and Daniele Nantes-Sobrinho[1]

[1] Departamentos de Matemática e Ciência da Computação
Universidade de Brasília, Brazil
[2] Department of Informatics,
King's College London, UK

Abstract. In nominal syntax, variable binding is specified using atom-abstraction constructors, and alpha-equivalence is formalised using freshness constraints and atom swappings, which implement variable renamings. Composition of swappings gives rise to atom permutations. Algorithms to check equivalence, match and unify nominal terms have been extended to deal with terms where some operators are associative and/or commutative. In the case of nominal C-unification, problems are transformed into finite and complete families of fixpoint equations of the form $\pi.X \approx_? X$, where π is a permutation. To generate nominal C-unifiers, a technique to obtain a sound and complete set of solutions for these equations is needed. In this work we show how complete sets of solutions for nominal fixpoint problems are built and discuss efficient techniques to generate solutions based on algebraic properties of permutations.

1 Introduction

Nominal syntax is an extension of first order syntax, where terms are built using function symbols, abstractions, and two kinds of variables: atoms, which can be abstracted, and unknowns (or simply variables), which behave like first order variables, except for the fact that they can have "suspended atom permutations", which act when the variable is instantiated by a term. Atom abstractions induce an α-equivalence relation on nominal terms, which is axiomatised using a freshness relation between atoms and terms. Nominal unification is unification of nominal terms, and takes into account the α-equivalence relation.

In many application domains, function symbols have equational properties, such as associativity and commutativity, which must be taken into account during the unification process. In previous work [3], we studied α-AC-equivalence of nominal terms, and nominal C-unification [4], that is, nominal unification in languages with commutative operators. The nominal C-unification problem was proved to be NP-complete in [9]. To solve the problem, we provided in [4] a set of simplification rules that generates, for each solvable C-unification problem, a finite set of *fixpoint problems* that are finite sets of *fixpoint equations* together with a freshness context and a substitution. The fixpoint equations generated in the process of solving nominal C-unification problems have the form $\pi.X \approx_? X$, where π is a permutation.

In [4], in addition to introducing the rule-based algorithm to transform nominal C-unification problems into a finite set of equivalent fixpoint problems, we also formalised in Coq its correctness and completeness. Also, we provided a sound method to generate solutions for fixpoint problems showing that infinite independent solutions are possible for a single fixpoint equation, which implies that nominal C-unification is infinitary.

Beyond the extensions of nominal unification, we can find equivariant unification [1, 7] as well as nominal narrowing [5], that are a useful tools in confluence analysis of nominal rewriting systems [2, 8] .

Contribution. The main result is a sound and complete procedure to solve fixpoint problems. More specifically:

- We prove the completeness of solutions for fixpoint equations generated in [4]. The analysis is based on the feasibility of combinations of the atoms in the domain of permutations used in the fixpoint equations in a fixpoint problem, that are built considering the combinatorial properties of the atoms in the permutations and by combining them using the basic elements of the nominal syntax, that is, pairs, abstractions and variables, as well as the commutative and non commutative function symbols in the signature. Variables included in these feasible combinations are new variables that should be restricted through adequate freshness contexts in such a way that atoms in the domain of the permutations should be fresh in these variables. The greedy generation of complete sets of solutions for a fixpoint equation is based on the construction of the so called *extended pseudo-cycles* from permutation cycles in the algebraic representation of permutations as products of *permutation cycles*. Only permutation cycles of length (period) a power of two are considered since permutation cycles of other lengths do not generate feasible (commutative) combinations.
- Furthermore, we work out an interesting improvement that avoids the generation of feasible solutions for different fixpoint equations on the same variable. The improvement is based on the fact that the feasible combinations for permutation cycles of the same length (a power of two) with the same domain, that are not algebraic *factors* of each other would not give rise to feasible common solutions.

Organisation. Section 2 introduces the background about nominal syntax and nominal C-unification. Section 3 proves the soundness and completeness of combinatorial solutions for fixpoint equations. Section 4 presents the improvements of the generator of solutions. Section 5 concludes the paper with future work.

2 Nominal syntax and nominal (α-)C-unification

2.1 Nominal Syntax

Consider countable disjoint sets of variables $\mathcal{X} := \{X, Y, Z, \cdots\}$ and atoms $\mathcal{A} := \{a, b, c, \cdots\}$. A *permutation* π is a bijection on $\mathcal{A}$ with a finite *domain*, where the domain (i.e., the *support*) of π is the set $dom(\pi) := \{a \in \mathcal{A} \mid \pi \cdot a \neq a\}$.

We will assume as in [3] countable sets of function symbols with different equational properties such as associativity, commutativity, idempotence, etc. Function symbols have superscripts that indicate their equational properties; thus, f_k^C will denote the k^{th} function symbol that is commutative and $f_j^{\emptyset}$ the j^{th} function symbol without any equational property.

Definition 1 (Nominal grammar). *Nominal terms are generated by the following grammar.* $s, t := \langle\rangle \mid \bar{a} \mid [a]t \mid \langle s, t\rangle \mid f_k^E\, t \mid \pi.X$

$\langle\rangle$ *denotes the* unit *(that is the empty tuple),* $\bar{a}$ *denotes an* atom term*,* $[a]t$ *denotes an* abstraction *of the atom* a *over the term* t*,* $\langle s, t\rangle$ *denotes a* pair*,* $f_k^E\, t$ *the* application *of* f_k^E *to* t *and,* $\pi.X$ *a* moderated variable *or* suspension*.*

Suspensions of the form $nil.X$ will be represented just by X. The set of variables occurring in a term t will be denoted as $Var(t)$. This notation extends to a set S of terms in the natural way: $Var(S) = \bigcup_{t\in S} Var(t)$.

A *substitution* σ is a mapping from variables to terms such that $X \neq X\sigma$ only for a finite set of variables. This set is called the *domain* of σ and is denoted by $dom(\sigma)$. For $X \in dom(\sigma)$, $X\sigma$ is called the *image* of X by σ. Define the image of σ as $im(\sigma) = \{X\sigma \mid X \in dom(\sigma)\}$. The set of variables occurring in the image of σ is then $Var(im(\sigma))$. A substitution σ with $dom(\sigma) := \{X_0, \cdots, X_n\}$ can be represented as a set of *binds* in the form $\{X_0/t_0, \cdots, X_n/t_n\}$, where for $0 \leq i \leq n$, $X_i\sigma = t_i$. We assume that the *action* of permutations and substitutions on nomins terms have their standard definitions (see e.g. [10], [13]). Since for our purposes the combinatorial properties of permutations on atoms are relevant, in this paper permutations are seen as *products* of *permutation cycles*: for instance, the nominal *swapping* permutation $(a\, b) :: (a\, c) :: (a\, d) :: (e\, f) :: (e\, g)$ is seen as the product of permutation cycles $(a\, b\, c\, d)\,(e\, f\, g)$.

2.2 The relation $\approx_{\{\alpha,C\}}$ and Nominal $\approx_{\{\alpha,C\}}$-unification

In [3], the relation $\approx_\alpha$ was extended to deal with associative and commutative theories. Here we will consider α-equivalence modulo commutativity, denoted $\approx_{\{\alpha,C\}}$. This means that some function symbols in our syntax are commutative.

The inference rules defining freshness and $\approx_{\{\alpha,C\}}$-equivalence are given in Figures 1 and 2. The *difference set* between two permutations π and π' is the set of atoms where the action of π and π' differs: $ds(\pi, \pi') := \{a \in \mathcal{A} \mid \pi \cdot a \neq \pi' \cdot a\}$.

The symbols ∇ and $\varDelta$ are used to denote *freshness contexts* that are sets of constraints of the form $a\#X$, meaning that the atom a is fresh in X. The domain of a freshness context $dom(\varDelta)$ is the set of atoms appearing in it; $\varDelta|_X$ denotes the restriction of $\varDelta$ to the freshness constraints on X: $\{a\#X \mid a\#X \in \varDelta\}$; $dom(\pi)\#X$ and $ds(\pi, \pi')\#X$ denote, respectively, the sets $\{a\#X \mid a \in dom(\pi)\}$ and $\{a\#X \mid a \in ds(\pi, \pi')\}$.

Key properties of the nominal freshness and α-equivalence relations have been extensively explored in previous works [3, 6, 12, 13]. In [4] we also have formalised analogous properties for $\approx_{\{\alpha,C\}}$. Among them we have *freshness preservation*: If

$$\frac{}{\nabla \vdash a \# \langle\rangle}\,(\#\langle\rangle) \quad \frac{}{\nabla \vdash a \# \overline{b}}\,(\#\,\mathbf{atom}) \quad \frac{\nabla \vdash a \# t}{\nabla \vdash a \# f_k^E\; t}\,(\#\,\mathbf{app}) \quad \frac{}{\nabla \vdash a \# [a]t}\,(\#\,\mathbf{a}[\mathbf{a}])$$

$$\frac{\nabla \vdash a \# t}{\nabla \vdash a \# [b]t}\,(\#\,\mathbf{a}[\mathbf{b}]) \quad \frac{(\pi^{-1}\cdot a\#X) \in \nabla}{\nabla \vdash a \# \pi.X}\,(\#\,\mathbf{var}) \quad \frac{\nabla \vdash a \# s \quad \nabla \vdash a \# t}{\nabla \vdash a \# \langle s,t\rangle}\,(\#\,\mathbf{pair})$$

Fig. 1. Rules for the relation $\#$

$$\frac{}{\nabla \vdash \langle\rangle \approx_{\alpha} \langle\rangle}\,(\approx_{\{\alpha,\mathbf{C}\}}\langle\rangle) \qquad \frac{}{\nabla \vdash \overline{a} \approx_{\{\alpha,C\}} \overline{a}}\,(\approx_{\{\alpha,\mathbf{C}\}}\ \mathbf{atom})$$

$$\frac{\nabla \vdash s \approx_{\{\alpha,C\}} t}{\nabla \vdash f_k^E\; s \approx_{\{\alpha,C\}} f_k^E\; t},\ \ E \neq C \text{ or both } s \text{ and } t \text{ are not pairs } (\approx_{\{\alpha,\mathbf{C}\}}\ \mathbf{app})$$

$$\frac{\nabla \vdash s_0 \approx_{\{\alpha,C\}} t_i,\ \ \nabla \vdash s_1 \approx_{\{\alpha,C\}} t_{(i+1)\,mod\,2}}{\nabla \vdash f_k^C\;\langle s_0,s_1\rangle \approx_{\{\alpha,C\}} f_k^C\;\langle t_0,t_1\rangle},\ \ i=0,1\ \ (\approx_{\{\alpha,\mathbf{C}\}}\ \mathbf{C})$$

$$\frac{\nabla \vdash s \approx_{\{\alpha,C\}} t}{\nabla \vdash [a]s \approx_{\{\alpha,C\}} [a]t}\,(\approx_{\{\alpha,\mathbf{C}\}}\ [\mathbf{aa}]) \qquad \frac{\nabla \vdash s \approx_{\{\alpha,C\}} (a\,b)\cdot t \quad \nabla \vdash a \# t}{\nabla \vdash [a]s \approx_{\{\alpha,C\}} [b]t}\,(\approx_{\{\alpha,\mathbf{C}\}}\ [\mathbf{ab}])$$

$$\frac{ds(\pi,\pi')\#X \subseteq \nabla}{\nabla \vdash \pi.X \approx_{\{\alpha,C\}} \pi'.X}\,(\approx_{\{\alpha,\mathbf{C}\}}\ \mathbf{var}) \quad \frac{\nabla \vdash s_0 \approx_{\{\alpha,C\}} t_0 \quad \nabla \vdash s_1 \approx_{\{\alpha,C\}} t_1}{\nabla \vdash \langle s_0,t_0\rangle \approx_{\{\alpha,C\}} \langle s_1,t_1\rangle}\,(\approx_{\{\alpha,\mathbf{C}\}}\ \mathbf{pair})$$

Fig. 2. Rules for the relation $\approx_{\{\alpha,C\}}$

$\nabla \vdash a \# s$ and $\nabla \vdash s \approx_{\{\alpha,C\}} t$, then $\nabla \vdash a \# t$; *equivariance*: for all permutations π, if $\nabla \vdash s \approx_{\{\alpha,C\}} t$ then $\nabla \vdash \pi \cdot s \approx_{\{\alpha,C\}} \pi \cdot t$; and, *equivalence*: $\_ \vdash \_ \approx_{\{\alpha,C\}} \_$ is an equivalence relation, indeed.

Definition 2 (Nominal unification problem). *A* nominal unification problem *is a pair $\langle \Delta, P\rangle$, where Δ is a* freshness context *and P is a finite set of* equations *and* freshness constraints *of the form $s \approx_? t$ and $a\#_?s$, respectively, where $\approx_?$ is symmetric, s and t are terms and a is an atom. Nominal terms in the equations preserve the syntactic restriction that commutative symbols are only applied to tuples.*

A formalised sound and complete rule-based algorithm was presented in [4], that transforms a nominal unification problem, say $\langle \Delta, P\rangle$, with commutative function symbols into a finite set of fixpoint problems that consist exclusively of equations of the form $\pi.X \approx_? X$. The transformation starts from the triple $\mathcal{P} = \langle \Delta, id, P\rangle$, where id denotes the substitution identity, and the rules act over triples building a finite set of fixpoint problems of the form $\mathcal{Q}_i = \langle \nabla_i, \sigma_i, Q_i\rangle$, for $0 \leq i \leq n$, where for each i, ∇_i is a freshness context, σ_i a substitution, and Q_i consists only of fixpoint equations.

For ∇ and ∇' freshness contexts and σ and σ' substitutions, $\nabla' \vdash \nabla\sigma$ denotes that $\nabla' \vdash a \# X\sigma$ holds for each $(a\#X) \in \nabla$; $\nabla \vdash \sigma \approx \sigma'$ denotes that $\nabla \vdash X\sigma \approx_{\{\alpha,C\}} X\sigma'$ for all X (in $dom(\sigma) \cup dom(\sigma')$).

Definition 3 (Solution for a triple or problem). *A* solution *for a triple $\mathcal{P} = \langle \Delta, \delta, P\rangle$ is a pair $\langle \nabla, \sigma\rangle$, where the following conditions are satisfied:*

1. $\nabla \vdash \Delta\sigma$;
2. if $a\#_? t \in P$ then $\nabla \vdash a \# t\sigma$;
3. if $s \approx_? t \in P$ then $\nabla \vdash s\sigma \approx_{\{\alpha,C\}} t\sigma$;
4. there exists λ such that $\nabla \vdash \delta\lambda \approx \sigma$.

A *solution for a unification problem* $\langle \Delta, P\rangle$ is a solution for the associated triple $\langle \Delta, id, P\rangle$. The *solution set* for a problem or triple $\mathcal{P}$ is denoted by $\mathcal{U}_C(\mathcal{P})$.

Definition 4 (More general solution and complete set of solutions). *For $\langle \nabla, \sigma\rangle$ and $\langle \nabla', \sigma'\rangle$ in $\mathcal{U}_C(\mathcal{P})$, we say that $\langle \nabla, \sigma\rangle$ is* more general *than $\langle \nabla', \sigma'\rangle$, denoted $\langle \nabla, \sigma\rangle \dot{\preccurlyeq} \langle \nabla', \sigma'\rangle$, if there exists a substitution λ satisfying $\nabla' \vdash \sigma\lambda \approx \sigma'$ and $\nabla' \vdash \nabla\lambda$. A subset $\mathcal{V}$ of $\mathcal{U}_C(\mathcal{P})$ is said to be a* complete set of solutions *of $\mathcal{P}$ if for all $\langle \nabla', \sigma'\rangle \in \mathcal{U}_C(\mathcal{P})$, there exists $\langle \nabla, \sigma\rangle$ in $\mathcal{V}$ that is more general than $\langle \nabla', \sigma'\rangle$.*

Example 1. The problem $\mathcal{P} = \langle \{b\#Y\}, \{+\langle (a\,b).X, Y\rangle \approx_? +\langle (b\,c).Y, X\rangle\}\rangle$ would be transformed by the algorithm in [4] into the set of fixpoint problems $\{\mathcal{Q}_1, \mathcal{Q}_2\}$, where $\mathcal{Q}_1 = \langle \{b\#Y\}, \{Y/X\}, \{(a\,c\,b).X \approx_? X\}\rangle$ and $\mathcal{Q}_2 = \langle \{b\#Y\}, id, \{(a\,b).X \approx_? X, (b\,c).Y \approx_? Y\}\rangle$. These fixpoint problems are generated by considering '+' to be a commutative symbol and by inversions on the permutations in the suspended variables.

The results in [4] include formalisations in Coq of theorems related with the following properties: **termination**: there are no possible infinite chains of applications of the unification transformation rules; **soundness**: for each possible transformation from $\mathcal{P}$ to $\mathcal{Q}$, one has that $\mathcal{U}_C(\mathcal{Q}) \subseteq \mathcal{U}_C(\mathcal{P})$; **unsolvability**: if $\mathcal{Q} = \langle \Delta, \sigma, Q\rangle$ cannot be simplified and Q contains non fixpoint equations or freshness constraints then $\mathcal{U}_C(\mathcal{Q}) = \emptyset$; and, **completeness**: if the unification problem $\mathcal{P} = \langle \nabla, id, P\rangle$ is transformed into the finite set of fixpoint problems $\mathcal{Q}_i$, for $1 \leq i \leq n$, then $\mathcal{U}_C(\mathcal{P}) = \bigcup_{i=1}^{n} \mathcal{U}_C(\mathcal{Q}_i)$.

Example 2. (Continuing example 1) The unification algorithm requires a mechanism to enumerate solutions of the fixpoint problems. Solutions in $\mathcal{U}(\mathcal{Q}_1)$ are built using the substitution $\{Y/X\}$ and solutions for the *single* fixpoint problem $\langle \{b\#Y\}, \{(a\,c\,b).X \approx_? X\}\rangle$. One such solution is $\langle \{b\#Y, a\#X, b\#X, c\#X\}, id\rangle$; therefore, $\langle \{b\#Y, a\#X, b\#X, c\#X\}, \{Y/X\}\rangle$ is a solution for $\mathcal{P}$. Solutions in $\mathcal{U}(\mathcal{Q}_2)$ are built by combining solutions for the associated single fixpoint problems $\langle \{b\#Y\}, \{(a\,b).X \approx_? X\}\rangle$ and $\langle \{b\#Y\}, \{(b\,c).Y \approx_? Y\}\rangle$. Assuming '$\star$' is a commutative operator in the signature, an admissible solution (among other infinite possibilities) for the former problem would be $\langle \{b\#Y\}, \{X/(\overline{a} + \overline{b}) \star (\overline{a} + \overline{b})\}\rangle$. Notice that since the freshness constraint $b\#Y$ belongs to $\mathcal{P}$, no possible combinations as the one given for X are allowed, and the unique possible solution for $\langle \{b\#Y\}, \{(b\,c).Y \approx_? Y\}\rangle$ is $\langle \{b\#Y\} \cup \{b\#Y, c\#Y\}, id\rangle$. Therefore, $\langle \{b\#Y, c\#Y\}, \{X/[d]\langle (\overline{a} + \overline{b}), Y\rangle \star [d]\langle (\overline{a} + \overline{b}), Y\rangle\}\rangle$ is also a solution for $\mathcal{P}$.

2.3 Solutions of fixpoint problems through extended pseudo-cycles

Given a permutation π, the set of solutions of a fixpoint equational problem $\langle \nabla, \{\pi.X \approx_? X\}\rangle$ is built according to the recursive definition of (*unitary*) *extended pseudo-cycles* below ([4]).

Definition 5 (Extended Pseudo-cycle). *Let $\pi.X \approx_? X$ be a fixpoint equation. The* extended pseudo-cycles *(*`epc`*, for short) κ' of π are inductively defined from its permutation cycles as follows:*

1. *$\kappa' = (Y)$, for any variable not occurring in the problem, is an extended pseudo-cycle;*
2. *$\kappa' = (\overline{a_0} \cdots \overline{a_{k'-1}})$ is an* extended pseudo-cycle *w.r.t. $(a_0 \ \cdots a_{k'-1})$ a permutation cycle in π such that $k' = 2^l$, for $l > 0$, called* trivial pseudo-cycle *of π.*
3. *$\kappa' = (A_0 \ \ldots A_{k'-1})$ is an* extended pseudo-cycle *w.r.t. π, if the following conditions are simultaneously satisfied:*
   (a) i. *each element of κ' is of the form $B_i \star B_j$, where $\star$ is a commutative function symbol in the signature, and B_i, B_j are different elements of κ, an* extended pseudo-cycle *w.r.t. π. κ' will be called an* extended first-instance pseudo-cycle *of κ w.r.t. π, or*
   ii. *each element of κ' is of the form $B_i \star C_j$ for any commutative symbol $\star$, where B_i and C_j are elements of κ and κ''* extended pseudo-cycles *w.r.t. π, that might be the same, but not being κ' an extended first-instance pseudo-cycle, or*
   iii. *each element of κ' is of the form $\langle B_i, C_j\rangle$, where B_i and C_j are elements of κ and κ''* extended pseudo-cycles *w.r.t. π, that might be the same, or*
   iv. *either each element of κ' is of the form $g\,B_i$ or each element is of the form $[e']\,B_i$, where g is a non commutative function symbol in the signature and $e' \notin dom(\pi)$, and each B_i is an element of an extended pseudo-cycle w.r.t π.*
   v. *each element of κ' is of the form $[a_j]B_i$, where a_j and B_i are resp. atoms in κ and elements in κ', a permutation cycle of π and an* extended pseudo-cycle *w.r.t. π.*
   (b) *For $\nabla = \cup_{Y \in Var(\kappa')} dom(\pi)\#Y$,*
   i. *it does not hold that $\nabla \vdash A_i \approx_{\{\alpha,C\}} A_j$ for $i \neq j$, $0 \leq i, j \leq k'-1$, where k' is the length of κ' (that is, the number of elements in κ'), and*
   ii. *for each $0 \leq i < k'-1$ one has $\nabla \vdash \pi(A_i) \approx_{\{\alpha,C\}} A_{i+1}$, where $i+1$ abbreviates $i+1$ modulo k'.*

Extended pseudo-cycles built using only items 2 and 3.a.i are called pseudo-cycles. *Extended pseudo-cycles of length 1 are called* unitary.

Remark 1. Pseudo-cycles are built just from atom terms in $dom(\pi)$ and commutative function symbols, while extended pseudo-cycles consider all nominal syntactic elements including new variables, and also non commutative function symbols.

Example 3. Let $\kappa = (a\,b\,c\,d)$ be a permutation cycle of π. Assume, $e \notin dom(\pi)$, $\star$ and $\oplus$ are commutative symbols, f and g non commutative symbols and Y a new variable. The following are pseudo-cycles: $(\overline{a} \star \overline{d} \quad \overline{b} \star \overline{a} \quad \overline{c} \star \overline{b} \quad \overline{d} \star \overline{c})$, $(\overline{a}\star\overline{c}\oplus\overline{b}\star\overline{d})$, etc. The following are `epc`'s: $(f\langle\overline{a},\overline{b}\rangle \ \ f\langle\overline{b},\overline{c}\rangle \ \ f\langle\overline{c},\overline{d}\rangle \ \ f\langle\overline{d},\overline{a}\rangle)$, $([e]\overline{a}\star\overline{c} \ \ [e]\overline{b}\star\overline{d})$, $(g\langle f\overline{a},[e]\overline{a}\rangle \ \ g\langle f\overline{b},[e]\overline{b}\rangle \ \ g\langle f\overline{c},[e]\overline{c}\rangle \ \ g\langle f\overline{d},[e]\overline{d}\rangle)$, $(\langle t, f\langle g\langle f\overline{a},[e]\overline{b}\rangle, Y\rangle \oplus f\langle g\langle f\overline{c},[e]\overline{d}\rangle, Y\rangle\rangle \star \langle t, f\langle g\langle f\overline{b},[e]\overline{c}\rangle, Y\rangle \oplus f\langle g\langle f\overline{d},[e]\overline{a}\rangle, Y\rangle\rangle\rangle)$, etc.

A relevant aspect in this construction (Def. 5) is that case 3.a.i allows to build `epc`'s that have half the length of the `epc`'s to which this case is applied. Therefore, we can only produce unitary `epc`'s from a permutation cycle that has length 2^k, for a positive integer k. Considering that the last step transforms an `epc` of the length two of the form (A_0, A_1) such that $\nabla \vdash \pi(A_0) \approx_{\{\alpha,C\}} A_1$ and $\nabla \vdash \pi(A_1) \approx_{\{\alpha,C\}} A_0$ (3.b.ii), the resulting unitary `epc`, $(A_0 \star A_1)$ satisfies $\nabla \vdash \pi(A_0 \star A_1) \approx_{\{\alpha,C\}} A_0 \star A_1$. Which means that this kind of solutions might be generated only from permutation cycles that have period a power of two. Indeed, it holds that only cycles of length a power of two might generate combinations that are feasible solutions.

Another relevant aspect of this construction is that although, we are using the relation $\approx_{\{\alpha,C\}}$, by the type of nominal terms involved in the extended pseudo-cycles only $\approx_C$ is necessary, except for considerations related with the freshness constraints (on new variables); hence, the invariant 3.b.ii can be seen as $\pi(A_i) \approx_C A_{i+1}$, where $i+1$ is read modulo the length of the `epc`.

3 Soundness and completeness of combinatorial solutions for fixpoint equations

Definition 6 (Generated solutions of fixpoint equations). *Given a solution triple $\mathcal{P} = \langle \nabla, \sigma, P \rangle$, such that $\pi.X \approx_? X \in P$. The set of* generated solutions *for $\langle \nabla, \pi.X \approx_? X \rangle$, denoted as $\langle \nabla, \pi.X \approx_? X \rangle_{Sol_G}$, includes all solutions of the form $\langle \nabla', \{X/s\} \rangle$ where (s) is an extended unitary pseudo-cycle of π such that $\nabla' \vdash dom(\nabla|_X)\#s$, where $\nabla' = \nabla \cup_{Y \in Var(s)} (dom(\nabla|_X)\#Y \cup dom(\pi)\#Y)$.*

In the following we are using $\langle \nabla, \pi.X \approx_? X \rangle_{Sol_C}$ to denote the solutions set of the fixpoint problem $\langle \nabla, \pi.X \approx_? X \rangle$.

Theorem 1 (Soundness of eps solutions). *Each generated solution of the form $\langle \nabla, \{X/s\} \rangle$ in $\langle \nabla, \pi.X \approx_? X \rangle_{Sol_G}$ is a solution in $\langle \nabla, \pi.X \approx_? X \rangle_{Sol_C}$.*

Proof. The proof follows the lines of reasoning used for non trivial unitary pseudo-cycles. By construction, the invariant that the elements of an extended pseudo-cycle of length l, $\kappa' = (e_0 \dots e_{l-1})$, satisfy the property $\nabla' \vdash \pi(e_i) \approx_{\{\alpha,C\}} e_{i+1}$, where $i+1$ abbreviates $i+1$ modulo l, and $\nabla' = \cup_{Y \in Var(\kappa')} dom(\pi)\#Y$, holds. The only case in which the length of an extended pseudo-cycle decreases is 3.a.i. Thus, when this case applies to a binary pseudo-cycle, say $(s_0\ s_1)$, an extended unitary pseudo-cycle (s) is built, being this of the form $(s_0 \oplus s_1)$ for a commutative function symbol $\oplus$. Since by the invariant we have that $\nabla' \vdash \pi(s_i) \approx_{\{\alpha,C\}} s_{i+1}$, for $i = 0, 1$, we have that $\nabla' \vdash \pi(s_0 \oplus s_1) \approx_{\{\alpha,C\}} s_0 \oplus s_1$; thus, we have that $\nabla' \vdash \pi(s) \approx_{\{\alpha,C\}} s$. In further steps in the construction of extended pseudo-cycles, new unitary pseudo cycles (t') might be built from unitary extended pseudo-cycles (t) applying cases 3.a.ii, iii, iv and v, that, can easily be checked, preserve the property $\nabla' \vdash \pi(t') \approx_{\{\alpha,C\}} t'$, for $\nabla' = \cup_{Y \in Var(t')} dom(\pi)\#Y$, if $\nabla' \vdash \pi(t) \approx_{\{\alpha,C\}} t$, for $\nabla' = \cup_{Y \in Var(t)} dom(\pi)\#Y$. Therefore all unitary non-trivial extended pseudo-cycles give a correct solution of the form $\langle \nabla', \{X/s\}\rangle$ of the problem $\langle \emptyset, \pi.X \approx_? X\rangle$. Hence, if in addition, we have that $\nabla' \cup \Delta \vdash \mathit{dom}(\Delta|_X)\#s$, then $\langle \nabla' \cup \Delta, \{X/s\}\rangle$ is a solution of $\langle \Delta, \pi.X \approx_? X\rangle$, whenever $\Delta \vdash \mathit{dom}(\Delta|_X)\#s$. □

Assuming the symbols in the signature are denumerable, it is possible to enumerate the unitary extended pseudo-cycles and thus the *generated solutions.* This can be done as usual, enumerating first all possible unitary pseudo-cycles with an element of length bounded by a small natural, say twice the length of π, and using only the first $|\pi|$ symbols in the signature and atoms in $\mathit{dom}(\pi)$; then, this length is being increased generating all extended unitary pseudo-cycles with elements of length $|\pi|+1$ and using only the first $|\pi+1|$ symbols in the signature and atoms in $\mathit{dom}(\pi)$ and so on.

The following technical result is proved by induction (see Appendix A) and is used in the proof of completeness of generated solutions for fixpoint problems.

Lemma 1 (Extended pseudo-cycle correspondence between π and π^2). *For $k \geq 1$, $(A_0 \cdots A_{2^k-1})$ is an extended pseudo-cycle for π if, and only if, there exist $(B_0 \cdots B_{2^{k-1}-1})$ and $(C_0 \cdots C_{2^{k-1}-1})$* epc *'s for π^2 with a substitution σ such that atoms in its image belong to $\mathit{dom}(\pi)\backslash \mathit{dom}(\pi^2)$, and for $0 \leq j \leq 2^{k-1}-1$ one has $B_j\sigma \approx_{\{\alpha,C\}} A_{2j}$ and $C_j\sigma \approx_{\{\alpha,C\}} A_{2j+1}$.*

Example 4. For $(a\,b)$ and $(c\,d\,e\,f)$, permutation cycles of π, one has that (a), (b), $(c\,e)$ and $(d\,f)$ are permutation cycles of π^2, and also, $a, b \in \mathit{dom}(\pi)\backslash \mathit{dom}(\pi^2)$. Therefore $((\overline{c}*\overline{e})+\overline{a})\star((\overline{d}*\overline{f})+\overline{b})$ and $((\overline{c}*\overline{e})+Y)\star((\overline{d}*\overline{f})+Y')$ are respectively unitary epc 's of π and π^2. Then:

- $\langle \Delta, \{X/((\overline{c}*\overline{e})+\overline{a})\star((\overline{d}*\overline{f})+\overline{b})\}\rangle \in \langle \Delta, \pi.X \approx_? X\rangle_{Sol_G}$ iff
- $\langle \Delta', \{X/((\overline{c}*\overline{e})+Y)\star((\overline{d}*\overline{f})+Y')\}\rangle \in \langle \Delta, \pi^2.X \approx_? X\rangle_{Sol_G}$,

where $\Delta' = \Delta \cup \mathit{dom}(\pi^2)\#Y, Y' \cup \mathit{dom}(\Delta|_X)\#Y, Y'$. So the σ of Lemma 1 will be $\{Y/\overline{a}, Y'/\overline{b}\}$, so that $((\overline{c}*\overline{e})+\overline{a}\;\;(\overline{d}*\overline{f})+\overline{b})$ is an epc of π, $((\overline{c}*\overline{e})+Y)$ and $((\overline{d}*\overline{f})+Y')$ are epc 's of π^2, with $((\overline{c}*\overline{e})+Y)\sigma = (\overline{c}*\overline{e})+\overline{a}$ and $((\overline{d}*\overline{f})+Y')\sigma = (\overline{d}*\overline{f})+\overline{b}$.

Theorem 2 (Completeness of generated solutions for fixpoint problems). *Let $\langle \Delta, \pi.X \approx_? X\rangle$ be a fixpoint problem. If $\langle \nabla, \{X/s\}\rangle \in \langle \Delta, \pi.X \approx_? X\rangle_{Sol_C}$ then there exists a more general extended solution, that is, $\langle \nabla', \{X/t\}\rangle \in \langle \Delta, \pi.X \approx_? X\rangle_{Sol_G}$ such that $\langle \nabla', \{X/t\}\rangle \dot{\preceq} \langle \nabla, \{X/s\}\rangle$.*

Proof. Since $\langle \nabla, \{X/s\}\rangle \in \langle \Delta, \pi.X \approx_? X\rangle_{Sol_C}$, it follows that $\nabla \vdash \Delta\{X/s\}$ and $\nabla \vdash \pi(s) \approx_{\{\alpha,C\}} s$. The proof is by induction on the structure of s.
Base Case. This case will be split in two parts.

1. $s = \overline{a}$.
   The pair $\langle \nabla, \{X/\overline{a}\}\rangle$ is a solution only if $a \notin \mathit{dom}(\Delta|_X) \cup \mathit{dom}(\pi)$, then $\emptyset \vdash \pi \cdot \overline{a} = \overline{a}$. Let Y be a new variable and $\nabla' = \mathit{dom}(\Delta|_X)\#Y \cup \mathit{dom}(\pi)\#Y$, then $\langle \nabla', \{X/Y\}\rangle$ is a generated solution. Let $\sigma = \{Y/\overline{a}\}$, notice that $\nabla \vdash \nabla'\sigma$ and $Y\sigma = \overline{a}$. Therefore, $\langle \nabla', \{X/Y\}\rangle \dot{\preceq} \langle \nabla, \{X/\overline{a}\}\rangle$.

2. $s = \pi'.Y$ and $dom(\pi)\#\pi'.Y$.
   Notice that $\langle\nabla, \{X/\pi'.Y\}\rangle \in \langle\varDelta, \pi.X \approx_? X\rangle_{Sol_C}$ only if $\nabla \vdash dom(\varDelta|_X)\#\pi'.Y, dom(\pi)\#\pi'.Y$, that is, if $\nabla \vdash (\pi')^{-1} \cdot dom(\varDelta|_X)\#Y$ and $\nabla \vdash (\pi')^{-1} \cdot dom(\pi)\#Y$, so that

$$\varDelta \cup ((\pi')^{-1} \cdot dom(\varDelta|_X) \cup (\pi')^{-1} \cdot dom(\pi))\#Y \subset \nabla$$

   Let $\langle\nabla', \{X/Z\}\rangle \in \langle\varDelta, \pi.X \approx_? X\rangle_{Sol_G}$ with $\nabla' = \varDelta\cup(dom(\pi)\cup dom(\varDelta|_X))\#Z$, Consider the substitution $\sigma = \{Z/\pi'.Y\}$, then $\nabla \vdash Z\sigma \approx_{\{\alpha,C\}} \pi'.Y$ and $\nabla'\sigma = \varDelta\sigma \cup (dom(\pi) \cup dom(\varDelta|_X))\#Z\sigma = \varDelta \cup (\pi')^{-1} \cdot dom(\pi)\#Y \cup (\pi')^{-1} \cdot dom(\varDelta|_X)\#Y$, so $\nabla \vdash \nabla'\sigma$. Therefore, $\langle\nabla', \{X/Z\}\rangle\dot{\preceq}\langle\nabla, \{X/\pi'.Y\}\rangle$.

Induction Step.

1. $s = \langle s_1, s_2\rangle$
   In this case $\nabla \vdash \pi(\langle s_1, s_2\rangle) \approx_{\{\alpha,C\}} \langle s_1, s_2\rangle$, that is, $\nabla \vdash \langle\pi(s_1), \pi(s_2)\rangle \approx_{\{\alpha,C\}} \langle s_1, s_2\rangle$, which implies in $\nabla \vdash \pi(s_i) \approx_{\{\alpha,C\}} s_i$, for $i = 1, 2$.
   By i.h. and Definitions 5 and 6, there exist $\langle\nabla_1', \{X/t_1\}\rangle, \langle\nabla_2', \{X/t_2\}\rangle \in \langle\varDelta, \pi \cdot X \approx_? X\rangle_{Sol_G}$ s.t. (t_1), (t_2) and $(\langle t_1, t_2\rangle)$ are unitary `epc`'s w.r.t. π. Furthermore $\langle\nabla_i', \{X/t_i\}\rangle\dot{\preceq}\langle\nabla, \{X/s_i\}\rangle$, i.e., there exist substitutions λ_i s.t. $\nabla \vdash \nabla_i\lambda_i$ and $\nabla \vdash t_i\lambda_i \approx s_i$, for $i = 1, 2$. One can choose (t_1) and (t_2) s.t. $Vars(t_1) \cap Vars(t_2) = \emptyset$ and $dom(\lambda_i) \cap Vars(s_j) = \emptyset$, for $i, j = 1, 2$. Then, $\nabla \vdash \langle t_1, t_2\rangle\lambda_1\lambda_2 \approx_{\{\alpha,C\}} \langle s_1, s_2\rangle$, and $\nabla \vdash (\nabla_1 \cup \nabla_2)\lambda_1\lambda_2$, that is, $\langle\nabla_1 \cup \nabla_2, \{X/\langle t_1, t_2\rangle\}\rangle\dot{\preceq}\langle\nabla, \{X/\langle s_1, s_2\rangle\}\rangle$.
2. $s = fs'$
   Since $\nabla \vdash \pi \cdot fs' \approx_{\{\alpha,C\}} fs'$, it follows that $\nabla \vdash f(\pi(s')) \approx_{\{\alpha,C\}} fs'$ and therefore, $\nabla \vdash \pi(s') \approx_{\{\alpha,C\}} s'$. By i.h. and Defs. 5 and 6, there exist $\langle\nabla', \{X/t'\}\rangle \in \langle\varDelta, \pi \cdot X \approx_? X\rangle_{Sol_G}$ such that (t') and (ft') are unitary `epc`'s w.r.t. π. Furthermore $\langle\nabla', \{X/t'\}\rangle\dot{\preceq}\langle\nabla, \{X/s'\}\rangle$, that is, there exist a substitution σ such that $\nabla \vdash \nabla'\sigma$ and $\nabla \vdash t'\sigma \approx_{\{\alpha,C\}} s'$, and since $\nabla \vdash ft'\sigma \approx_{\{\alpha,C\}} f(t'\sigma) \approx_{\{\alpha,C\}} fs'$ and adding f at the top of t' does not change the variables of t', therefore, $\langle\nabla', \{X/ft'\}\rangle \in \langle\varDelta, \pi \cdot X \approx_? X\rangle_{Sol_G}$ and $\langle\nabla', \{X/ft'\}\rangle\dot{\preceq}\langle\nabla, \{X/fs'\}\rangle$.
3. $s = [e]s'$.
   (a) $e \notin dom(\pi)$
   Since $\nabla \vdash \pi([e]s') \approx_{\{\alpha,C\}} [e]s'$, it follows that $\nabla \vdash \pi(s') \approx_{\{\alpha,C\}} s'$, that is, $\langle\nabla, X/s'\rangle$ is a solution for $\langle\varDelta, \pi \cdot X \approx_? X\rangle$. By i.h. and Defs. 5 and 6, there exist $\langle\nabla', \{X/t'\}\rangle \in \langle\varDelta, \pi \cdot X \approx_? X\rangle_{Sol_G}$ such that (t') and $([e]t')$ are unitary `epc`'s w.r.t. π. Furthermore $\langle\nabla', \{X/t'\}\rangle\dot{\preceq}\langle\nabla, \{X/s'\}\rangle$, that is, there exist a substitution σ such that $\nabla \vdash \nabla'\sigma$ and $\nabla \vdash t'\sigma \approx_{\{\alpha,C\}} s'$, therefore, $\langle\nabla', \{X/[e]t'\}\rangle \in \langle\varDelta, \pi \cdot X \approx_? X\rangle_{Sol_G}$ and $\langle\nabla', \{X/[e]t'\}\rangle\dot{\preceq}\langle\nabla, \{X/[e]s'\}\rangle$.
   (b) $e \in dom(\pi)$.
   By hypothesis, $\nabla \vdash \pi([e]s') \approx_{\{\alpha,C\}} [e]s'$, that is, $\nabla \vdash [\pi \cdot e](\pi(s')) \approx_{\{\alpha,C\}} [e]s'$, and $\nabla \vdash \pi(s') \approx_{\{\alpha,C\}} (\pi \cdot e\ \ e)(s')$ only if $\nabla \vdash (\pi \cdot e)\#s'$.
   Notice that e occurs in s' iff $\pi \cdot e$ occurs in s'. Therefore, for $\nabla \vdash e\#s'$, it follows that $\nabla \vdash \pi(s') \approx_{\{\alpha,C\}} s'$ and the result follows by induction hypothesis.

4. $s = s_1 \oplus s_2$
   This case has two parts:
   (a) $\nabla \vdash \pi(s_1) \approx_{\{\alpha,C\}} s_1$ and $\nabla \vdash \pi(s_2) \approx_{\{\alpha,C\}} s_2$.
   By i.h. and Definitions 5 and 6, there exist $\langle \nabla'_1, \{X/t_1\}\rangle, \langle \nabla'_2, \{X/t_2\}\rangle \in \langle \mathit{\Delta}, \pi \cdot X \approx_? X\rangle_{Sol_G}$ s.t. (t_1), (t_2) and $(t_1 \oplus t_2)$ are unitary epc 's w.r.t. π. Furthermore $\langle \nabla'_i, \{X/t_i\}\rangle \dot{\preceq} \langle \nabla, \{X/s_i\}\rangle$, i.e., there exist substitutions λ_i s.t. $\nabla \vdash \nabla_i \lambda_i$ and $\nabla \vdash t_i \lambda_i \approx s_i$, for $i = 1, 2$. One can choose (t_1) and (t_2) s.t. $Vars(t_1) \cap Vars(t_2) = \emptyset$ and $dom(\lambda_i) \cap Vars(s_j) = \emptyset$, for $i, j = 1, 2$. Then, $\nabla \vdash (t_1 \oplus t_2)\lambda_1\lambda_2 \approx_{\{\alpha,C\}} (s_1 \oplus s_2)$, and $\nabla \vdash (\nabla_1 \cup \nabla_2)\lambda_1\lambda_2$, that is, $\langle \nabla_1 \cup \nabla_2, \{X/t_1 \oplus t_2\}\rangle \dot{\preceq} \langle \nabla, \{X/s_1 \oplus s_2\}\rangle$.
   (b) $\nabla \vdash \pi(s_1) \approx_{\{\alpha,C\}} s_2$ and $\nabla \vdash \pi(s_2) \approx_{\{\alpha,C\}} s_1$.
   Notice that $\nabla \vdash \pi^2(s_1) \approx_{\{\alpha,C\}} \pi(s_2) \approx_{\{\alpha,C\}} s_1$ and $\nabla \vdash \pi^2(s_2) \approx_{\{\alpha,C\}} \pi(s_1) \approx_{\{\alpha,C\}} s_2$. Therefore, $\langle \nabla, \{X/s_1\}\rangle$ and $\langle \nabla, \{X/s_2\}\rangle$ are solutions of $\langle \mathit{\Delta}, \pi^2.X \approx_? X\rangle$. By IH, there exist $\langle \nabla_1, \{X/t_1\}\rangle, \langle \nabla_2, \{X/t_2\}\rangle \in \langle \mathit{\Delta}, \pi^2 \cdot X \approx_? X\rangle_{Sol_G}$ such that $\langle \nabla_i, \{X/t_i\}\rangle \dot{\preceq} \langle \nabla, \{X/s_i\}\rangle$. Then there exist substitutions λ_i s.t. $\nabla \vdash \nabla_i \lambda_i$ and $\nabla \vdash t_i \lambda_i \approx_{\{\alpha,C\}} s_i$, for $i = 1, 2$. One can choose (t_1) and (t_2) s.t. $Vars(t_1) \cap Vars(t_2) = \emptyset$ and $dom(\lambda_i) \cap Vars(s_j) = \emptyset$, for $i, j = 1, 2$.
   Therefore, $\langle \nabla_1 \cup \nabla_2, X/t_1 \oplus t_2\rangle \in \langle \mathit{\Delta}, \pi^2 \cdot X \approx_? X\rangle_{Sol_G}$ and $\langle \nabla_1 \cup \nabla_2, X/t_1 \oplus t_2\rangle \dot{\preceq} \langle \nabla, X/s_1 \oplus s_2\rangle$, via substitution $\lambda = \lambda_1\lambda_2$.
   Notice that $\nabla \vdash (\pi \cdot t_1)\lambda \approx_{\{\alpha,C\}} \pi \cdot s_1 \approx_{\{\alpha,C\}} s_2 \approx_{\{\alpha,C\}} t_2\lambda$ and analogously, $\nabla \vdash (\pi \cdot t_2)\lambda \approx_{\{\alpha,C\}} t_1\lambda$. Hence, $\langle \nabla, \lambda\rangle$ is a solution for the C-unification problems $\pi \cdot t_1 \approx^?_{\{\alpha,C\}} t_2$ and $\pi \cdot t_2 \approx^?_{\{\alpha,C\}} t_1$. Let $\langle \nabla', \lambda'\rangle$ be a mgu for $\langle \nabla, \lambda\rangle$ such that $dom(\lambda') \subseteq dom(\pi) \backslash dom(\pi^2)$. Since $(\pi \cdot t_1)$ and $(\pi \cdot t_2)$ are unitary epc 's of π^2, it follows by Lemma 1, that $(t_1\lambda'\ t_2\lambda')$ is an epc for π.
   By Definition 5, $(t_1\lambda' \oplus t_2\lambda')$ is a unitary epc for π, such that $\langle \nabla', X/t_1\lambda' \oplus t_2\lambda'\rangle \in \langle \mathit{\Delta}, \pi \cdot X \approx_? X\rangle_{Sol_G}$ and $\langle \nabla', X/t_1\lambda' \oplus t_2\lambda'\rangle \dot{\preceq} \langle \nabla, X/s_1 \oplus s_2\rangle$.

□

Remark 2. Notice that to build a most general C-unifiers in the proof of Lemma 2 (case 4.b) and Def. 7 one can use the algorithm proposed by Siekmann [11], which provides a finite, minimal and complete set of C-unifiers.

Definition 7 (General C-matchers). *Let s_i, for $i = 1..k$, be nominal terms. A* most general C-matcher *of these terms, if it exists, is the image of a most general C-unifier restricted to the domain Z, say $\{Z/t\}$, of the C-unification problem $\{s_i =_? Z\}_{i=1..k}$, where Z is a new variable.*

Remark 3. Alternatively, Definition 5 could be restricted to ground terms (by removing the first case in the construction of extended pseudo-cycles), and then instead of computing C-matchers via C-unification, one could use an α-C-equivalence checker (for example, the one specified in [3]). This would also simplify case iv in Definition 5, since it would be sufficient to consider just one atom e' not in $dom(\pi)$.

Definition 8 (Generated solutions for a variable). *Let the fixpoint problems for X in $\mathcal{P}$ be given by $\langle \nabla, \pi_i.X \approx_? X\rangle$, for $\pi_i \in \mathit{\Pi}_X$, and such that $|\mathit{\Pi}_X| = k$. If there exist solutions $\langle \nabla_i, \{X/t_i\}\rangle \in \langle \nabla, \pi_i.X \approx_? X\rangle_{Sol_G}$ and for:*

- *δ, a most general C-matcher of terms $\{t_i\}_{i=1..k}$ with X as new variable and*
- *$\nabla'' := \nabla \cup_{i=1}^{k} \nabla_i$, and $\nabla' := \nabla'' \cup_{Y \in dom(\delta)} \cup_{Y' \in Var(Y\delta)} dom(\nabla''|_Y)\#Y'$;*

it holds that for all $Y \in dom(\delta)$, $\nabla' \vdash dom(\nabla''|_Y)\#Y\delta$; then in this case we say that $\langle \nabla', \{X/X\delta\}\rangle$ is a generated solution *for X. The set of all generated solutions is denoted by $[X]_{\mathcal{P}_G}$.*

Example 5. Let $P_i := \pi_i.X \approx_? X$, for $i = 1..3$, be fixpoint equations for $\pi_1 = (a\ b\ c\ d)$, $\pi_2 = (a\ c)$ and $\pi_3 = (b\ d)$ and suppose that $\mathcal{P} := \langle \nabla, \sigma, P\rangle$ is a successful leaf such that P_i for $i = 1..3$ are the fixpoint equations for X in P.

1. $\langle a, b, c, d\#Y, \delta_1 := \{X/((a * c) * (b * d)) \oplus Y\}\rangle \in \langle \nabla, P_1\rangle_{Sol_G}$;
2. $\langle a, c\#Y', Y'', \delta_2 := \{X/((a * c) * Y') \oplus Y''\}\rangle \in \langle \nabla, P_2\rangle_{Sol_G}$; and
3. $\langle b, d\#Y_1', Y_1'', \delta_3 := \{X/((b * d) * Y_1') \oplus Y_1''\}\rangle \in \langle \nabla, P_3\rangle_{Sol_G}$.

Notice that $\delta = \{X/((a{*}c){*}(b{*}d)){\oplus}Y'',\ Y'/(b{*}d),\ Y_1'/(a{*}c),\ Y/Y'', Y_1''/Y''\}$ is a most general C-unifier of terms $\{t_i := X\delta_i\}$ with variable X.

According to the definiton, the set of initial freshness constraints is given as $\nabla'' = \nabla \cup \{a, b, c, d\#Y,\ a, c\#Y', Y'',\ b, d\#Y_1', Y_1''\}$. Notice that $Y'' \in Var(im(\delta))$, have to satisfy the constraints on Y_1'', Y and X, that is, $a, b, c, d\#Y''$ is a new constraint on Y'', inherited from the constraints of the variables in the domain of δ.

For $\nabla' = \nabla'' \cup \{a, b, c, d\#Y''\}$, it holds that $\nabla' \vdash dom(\nabla''|_Z)\#Z\delta$, for all $Z \in dom(\delta)$. Therefore, $\langle \nabla', X/X\delta\rangle$ belongs to $[X]_{\mathcal{P}_G}$.

Now we prove that the set of generated solutions $[X]_{\mathcal{P}_G}$ is correct.

Corollary 1 (Soundness and completeness of generated solutions for a variable). *Let $\mathcal{P}$ be fixpoint problem. Any solution in $[X]_{\mathcal{P}_G}$ is a solution of each fixpoint equation for X in $\mathcal{P}$. If $\langle \nabla, X/t\rangle$ is a solution for each fixpoint equation for X in $\mathcal{P}$ then there exists $\langle \nabla', X/t'\rangle \in [X]_{\mathcal{P}_G}$ such that $\langle \nabla', X/t'\rangle \dot{\preceq} \langle \nabla, X/t\rangle$*

Proof (sketch). (*Soundness*) By Lemma 1 and Def. 6 each solution $\langle \nabla_i, \{X/t_i\}\rangle$ in $\langle \nabla, \pi_i.X \approx_? X\rangle_{Sol_G}$ is a correct solution for $\langle \nabla, \pi_i.X \approx_? X\rangle$, for $\pi_i \in \mathit{\Pi}_X$. Suppose $\langle \nabla', \{X/X\delta\}\rangle$ belongs to $[X]_{\mathcal{P}_G}$. Since δ is a C-unifier of terms t_i with variable X, we have that $X\delta \approx_C t_i\delta$, and also that $\nabla_i \vdash \pi.t_i \approx_{\{\alpha,C\}} t_i$. Thus, $\nabla' \vdash \pi.t_i\delta \approx_{\{\alpha,C\}} t_i\delta$ since by definition we also have that $\nabla' \vdash dom(\nabla|_X)\#X\delta$, because by construction for all $Y \in Var(X\delta)$, ∇' includes the freshness constraints $dom(\nabla''|_X)\#Y$ and ∇'' is an extension of ∇.
(*Completeness*) Let $\{\langle \mathit{\Delta}, \pi_i.X \approx_? X\rangle\}_{\pi_i \in \mathit{\Pi}_X}$ be a family of all fixpoint equations on X in $\mathcal{P}$ and $\langle \nabla, X/t\rangle$ a solution for all of them, that is, $\nabla \vdash \mathit{\Delta}\{X/t\}$ and $\nabla \vdash \pi_i \cdot t \approx_{\{\alpha,C\}} t$, for all $\pi_i \in \mathit{\Pi}_X$. Since $\langle \nabla, X/t\rangle \in \langle \mathit{\Delta}, \pi_i.X \approx_? X\rangle_{Sol_C}$, by Theorem 2, there exist a most general solution $\langle \nabla_i, X/t_i\rangle \in \langle \mathit{\Delta}, \pi.X \approx_? X\rangle_{Sol_G}$. Let $\langle \nabla', \delta\rangle$ be a most general C-matcher of the terms $\{t_i\}$. By the construction in Definition 8 follows the result. □

Example 6. Let $\pi = (a\ b\ c\ d\ e\ f\ g\ h)$ then $\pi^2 = (a\ c\ e\ g)(b\ d\ f\ h)$. There are solutions of $\langle \emptyset, \pi^2 \cdot X \approx_? X\rangle$ that are not solutions of $\langle \emptyset, \pi.X \approx_? X\rangle$:

- $\langle \emptyset, X/(\overline{a} \oplus \overline{e}) \oplus (\overline{c} \oplus \overline{g})\rangle, \langle \emptyset, X/(\overline{b} \star \overline{f}) \oplus (\overline{d} \star \overline{h})\rangle \in \langle \emptyset, \pi^2 \cdot X \approx_? X\rangle_{Sol_G}$;
- $\langle \emptyset, X/((\overline{a} \oplus \overline{e}) \oplus (\overline{c} \oplus \overline{g})) \oplus ((\overline{b} \star \overline{f}) \oplus (\overline{d} \star \overline{h}))\rangle \in \langle \emptyset, \pi^2 \cdot X \approx_? X\rangle_{Sol_G}$

but none of them is a solution for $\langle \emptyset, \pi.X \approx_? X\rangle$.

However there exist solutions in the intersection of both problems, for instance, $\langle \emptyset, X/((\overline{a} \oplus \overline{e}) \oplus (\overline{c} \oplus \overline{g})) * (X/(\overline{b} \oplus \overline{f}) \oplus (\overline{d} \oplus \overline{h}))\rangle$.

Definition 9 (Generated Solutions for fixpoint problems). *Let $\mathcal{P}$ be a fixpoint problem. The set of generated solutions for $\mathcal{P}$ denoted as $[\mathcal{P}]_{Sol_G}$ is defined as the set that contains all solutions of the form*

$$\left\langle \bigcup_{X \in Var(P)} \nabla_X, \quad \bigcup_{X \in Var(P)} \{X/s_X\} \right\rangle, \text{ where each } \langle \nabla_X, \{X/s_X\}\rangle \in [X]_{\mathcal{P}_G}.$$

Corollary 2 (Soundness and completeness of generated solutions for fixpoint problems). *Let $\mathcal{P}$ be a fixpoint problem. Any solution in the set of solutions $[\mathcal{P}]_{Sol_G}$ is a correct solution of $\mathcal{P}$. For any $\langle \nabla, \delta\rangle$ solution of $\mathcal{P}$ there exist a pair $\langle \nabla', \sigma\rangle \in [\mathcal{P}]_{Sol_G}$ such that $\langle \nabla', \sigma\rangle \dot{\preceq} \langle \nabla, \delta\rangle$.*

Proof (Sketch). (Soundness) By previous corollary, a solution of $\mathcal{P}$ is of the form $\left\langle \bigcup_{X \in Var(P)} \nabla_X, \bigcup_{X \in Var(P)} \{X/s_X\} \right\rangle$, where each $\langle \nabla_X, \{X/s_X\}\rangle$ is a correct solution for all fixpoint equations in $\mathcal{P}$ for the variable X, this completes the soundness proof. *(Completeness)* Let $\mathcal{P} = \{\langle \varDelta, \{\pi_{i_1} \cdot X_1 \approx_? X_1\}_{\pi_{i_1} \in \varPi_{X_1}}, \ldots, \{\pi_{i_k} \cdot X_k \approx_? X_k\rangle_{\pi_k \in \varPi_{X_k}}\}$ and $\langle \nabla, \delta\rangle$ be a solution of $\mathcal{P}$. Then, $\langle \nabla, \delta\rangle \in \langle \varDelta, \pi_{i_j}.X_j \approx_? X_j\rangle_{Sol_C}$, for all i and j. By Corollary 1, there exist most general $\langle \nabla_j, X/t_j\rangle \in [X_j]_{\mathcal{P}_G}$, for $j = 1, \ldots, k$, i.e., $\langle \nabla_j, X/t_j\rangle \dot{\preceq} \langle \nabla, \delta\rangle$ and . Therefore, $\langle \cup_j \nabla_j, \cup_j \{X/t_j\}\rangle \dot{\preceq} \langle \nabla, \delta\rangle$ and $\langle \cup_j \nabla_j, \cup_j \{X/t_j\}\rangle \in [\mathcal{P}]_{Sol_G}$. □

A greedy procedure for the generation of solutions in $[X]_{\mathcal{P}}$ proceeds as follows. Follow the construction of generated solutions in Definition 6 for each fixpoint problem $\langle \nabla, \pi_i.X \approx_? X\rangle$ in P, where $\pi_i \in \varPi_X$, as given in Lemma 1; for each generated solution $\langle \nabla', \{X/s\}\rangle$ build the freshness context $\nabla'' = \nabla' \cup \bigcup_{Y \in Var(s)} dom(\nabla|_X)\#Y \cup dom(\varPi_X)\#Y$ and check whether $\langle \nabla'', \{X/s\}\rangle$ is a solution for all $\langle \nabla, \pi_i.X \approx_? X\rangle$, for $\pi_i \in \varPi_X$. Here, $dom(\varPi_X)\#Y$ abbreviates $\cup_{\pi_i \in \varPi_X} dom(\pi_i)\#Y$.

4 Improvements in the generation of solutions

The greedy procedure might be improved eliminating generation of solution of non interesting permutation cycles in $\varPi_X$, according to the observations below.

In first place, notice that according to the theory of pseudo-cycles, we are interested in building solutions with atoms that occur only in permutation cycles of length a power of two in all permutations $\pi \in \varPi_X$.

In second place, notice that if there exist permutation cycles of length a power of two $\kappa_i \in \pi_i$ and $\kappa_j \in \pi_j$, for $\pi_i, \pi_j \in \mathit{\Pi}_X$, such that $dom(\pi_i) \cap dom(\pi_j) \neq \emptyset$, $dom(\pi_i) \setminus dom(\pi_j) \neq \emptyset$ and $dom(\pi_j) \setminus dom(\pi_i) \neq \emptyset$, then there might not be possible solutions with occurrences of atom terms in the domain of π_i and/or π_j for the fixpoint equations related with permutations π_i and π_j. The simplest example is given by permutation cycles $(a\,b)$ and $(a\,c)$. The precise relation between permutation cycles that allows for construction of solutions for all permutations in $\mathit{\Pi}_X$ is given in the next definition.

Definition 10 (Permutation factor). *A permutation π is said to be an n-*factor *of a permutation π' whenever there exists n such that $\pi^n = \pi'$.*

Example 7. Let $\pi = (a\,b\,c\,d\,e\,f\,g\,h)$. The odd powers of π, π^1, $\pi^3 = (a\,d\,g\,b\,e\,h\,c\,f)$, $\pi^5 = (a\,f\,c\,h\,e\,b\,g\,d)$ and $\pi^7 = (a\,h\,g\,f\,e\,d\,c\,b)$ are the only factors of π.

Remark 4. For κ a permutation cycle of length 2^k, their factors corresponding to permutation cycles of the same length, are exactly the permutations cycles κ^p, for p odd such that $0 < p < 2^k$; also, if λ is a p-factor of κ then λ is the q-factor of κ, where q is the minimum odd number such that $0 < q < 2^k$ and $p \cdot q = 1$ modulo 2^k. For instance, if κ is a permutation cycle of length 2^4, κ^3, κ^5, κ^7, etc, are respectively the 11- 13- and 7-factors, etc, of κ.

The key observation about permutation cycles κ and λ, of respective lengths 2^k and 2^l, for $k \geq l \geq 0$, such that, $\kappa^{2^{k-l}}$ contains a permutation cycle, say ν, that is a p-factor of λ, is that this happens if and only if regarding elements in $dom(\lambda)$, possible generated solutions from both permutation cycles coincide. Indeed, first, notice that either $l = 0$ and then $\nu = \lambda$ or $l > 0$ and $\lambda^{2^{l-1}}$ consists of 2^{l-1} permutation cycles of length two; second, observe that if $l > 0$, then $\lambda^{2^{l-1}} = \nu^{p \cdot 2^{l-1}} = \nu^{2^{l-1}}$, since p is an odd number (such that $0 < p < 2^l$). Moreover, notice that $\kappa^{2^{k-l}}|_{dom(\lambda)} = \nu$, that implies that $\kappa^{2^{k-1}}|_{dom(\lambda)} = \nu^{2^{l-1}}$. Thus, the permutation cycles of length two generated from κ and λ, restricted to $dom(\lambda)$ are the same, which implies that commutative combinations built (according to Def. 5) regarding to the elements in $dom(\lambda)$ are the same.

Example 8. Consider $\kappa = (a\,b\,c\,d\,e\,f\,g\,h)$ and $\lambda = (a\,g\,e\,c)$. Notice that $\kappa^2 = (a\,c\,e\,g)(b\,d\,f\,h)$ and λ is a 3-factor of $\nu = (a\,c\,e\,g)$. Then $\lambda^2 = \nu^{3 \cdot 2} = \nu^2 = (a\,e)(c\,g)$. Also, notice that the unitary extended pseudo-cycles built from λ and ν are the same.

Definition 11 (Permutation cycles in the top of $\mathit{\Pi}_X$). *Let $\mathit{\Pi}_X$ be the set of permutations for fixpoint equations on the variable X in a fixpoint problem. A permutation cycle $\kappa \in \pi \in \mathit{\Pi}_X$* is in the top *of $\mathit{\Pi}_X$, whenever for all atoms $a \in dom(\kappa)$ and all $\pi' \in \mathit{\Pi}_X$, if $a \in dom(\pi')$, and a is an element in a permutation cycle λ in π', then there exists a natural m such that the permutation cycle of the element a in π^{2^m}, say ν, is a factor of the permutation cycle λ.*

Example 9. Consider the permutations $\pi_1 = (a\,b\,c\,d\,e\,f\,g\,h)$, $\pi_2 = (a\,g\,e\,c)(b\,f)$ and $\pi_3 = (a\,e)(c\,g)(d\,h)$. The permutation cycle π_1 is in the top of the set of permutations; indeed, notice that all permutation cycles in all permutations appear as a factor in powers of two of π_1: $\pi_1^0 = (a\,b\,c\,d\,e\,f\,g\,h)$; $\pi_1^2 = (a\,c\,e\,g)(b\,d\,f\,h)$; $\pi_1^4 = (a\,e)(c\,g)(b\,f)(d\,h)$; $\pi_1^8 = (a)(e)(c)(g)(b)(f)(d)(h)$.

Theorem 3 (Atoms of interest in fixpoint problems on a variable). *Let Π_X be the set of permutations for fixpoint equations on the variable X in a fixpoint problem. Only the set of atoms in the domain of permutation cycles in the top of Π_X might occur in solutions of all fixpoint equations on X.*

Proof. Only atoms that are in permutation cycles of length a power of two in all permutations $\pi \in \Pi_X$ might occur in solutions of all fixpoint equations on X. Suppose a is an atom that only occurs in permutation cycles of length a power of two for all $\pi \in \Pi_X$ and let κ be a permutation cycle in Π_X of maximal length, say 2^k, with $a \in \mathit{dom}(\kappa)$. Suppose λ is a permutation cycle in ϕ, for some $\phi \in \Pi_X$, with $a \in \mathit{dom}(\lambda)$ and let 2^l be the length of λ. Only if λ is a factor of a permutation cycle in $\pi^{2^{k-l}}$, say ν such that $\nu^p = \lambda$, the extended pseudo-cycles built from λ (and from κ) will maintain the invariants required, restricted to the atoms in $\mathit{dom}(\lambda)$, that is for an extended pseudo-cycle built from λ of the form $(A_0\ \ldots\ A_{2^m-1})$, where $m \leq l$, $\phi(A_i) \approx_C A_{i+1}$ and $\phi^{2^{l-m}}(A_i) \approx_C A_i$, where $i+1$ reads modulo 2^m. This also holds for λ. Hence, since ν is a p-factor of λ (and also, $\pi^{2^{k-l}}|_{\mathit{dom}(\lambda)} = \nu$), one has that $\nu^p(A_i) \approx_C A_{i+1}$ and $\nu^{p\cdot 2^{l-m}}(A_i) \approx_C A_i$. If the extended pseudo-cycle is of length two, that is it is of the form $(A_0\,A_1)$, we have $m = 1$ and $\nu^{p\cdot 2^{l-1}}(A_i) \approx_C A_i$, for $i = 0, 1$, and since p is odd, this implies that $\nu^{2^{l-1}}(A_i) \approx_C A_i$, for $i = 0, 1$. This condition also holds for π, since $(\pi^{2^{k-l}}|_{\mathit{dom}(\nu)})^{2^{l-1}} = (\nu)^{2^{l-1}}$; hence, $\pi^{2^{k-1}}(A_i) = A_{i+1}$, for $i = 0, 1$. If κ is not a permutation cycle in the top of Π_X, then there exists some permutation cycle $\lambda \in \phi \in \Pi_X$, such that $a \in \mathit{dom}(\kappa) \cap \mathit{dom}(\lambda)$, 2^l is the length of λ, but the permutation cycle of length 2^l in $\kappa^{2^{k-l}}$, say ν, such that $a \in \mathit{dom}(\nu)$ is not a factor of λ. Thus, since $\nu^{2^{l-1}} \neq \lambda^{2^{l-1}}$ atoms in the domains of ν and λ cannot be combined uniformly to build common solutions for κ and λ (i.e., for π and ψ).

To finish we show how a common solution can be built when κ is in the top of Π_X. Suppose that (A) is a unitary extended pseudo cycle built from λ by successive applications of case 3.a.i. of Definition 5 halving in each step the length of the pseudo-cycle. We have that $\lambda(A) = A$. It is possible to generate an extended pseudo-cycle for κ of the form $(A\,\kappa(A)\,\kappa^2(A)\ \ldots\ \kappa^{2^{k-l}-1}(A))$. From this extended pseudo-cycle it is possible to build a unitary extended pseudo-cycle by successive applications of case 3.a.i. of Definition 5, first obtaining $(A\star_1 \kappa^{2^{K-l-1}}(A)\ \kappa(A)\star_1\kappa^{2^{k-l-1}+1}(A)\ldots\ \kappa^{2^{k-l-1}-1}(A)\star_1\kappa^{2^{k-l}-1}(A))$, and so on until a unitary extended pseudo-cycle of the form $((\cdots((A\star_1 B_1)\star_2 B_2)\cdots)\star_{k-l} B_{k-l})$ is obtained where the B_i's, for $1 \leq i \leq k-l$ are adequate combinations of the terms $\kappa(A), \ldots \kappa^{2^{k-l}-1}(A)$ according to the constructions of extended pseudo-cycles. From this extended pseudo-cycle one has the solution for $\pi.X \approx_? X$ of the form $\langle\emptyset, \{X/(\cdots((A\star_1 B_1)\star_2 B_2)\cdots)\star_{k-l} B_{k-l}\}\rangle$, where $\star_j$, for $j = 1, \ldots, l$ are commutative symbols. Using the unitary cycle (A) for λ and cases 1 and 3.a.ii of Definition 5 one can generate the unitary pseudo-cycle $((\cdots((A\star_1 Y_1)\star_2$

$Y_2) \cdots) \star_{k-l} Y_{k-l})$ which gives the solution $\langle \nabla, \{X/(\cdots((A \star_1 Y_1) \star_2 Y_2) \cdots) \star_{k-l} Y_{k-l}\}\rangle$ for λ, where $\nabla = \{dom(\lambda)\#Y_j | 1 \leq j \leq l\}$. The C-unification problem $\langle \nabla, X \approx_? (\cdots((A \star_1 B_1) \star_2 B_2) \cdots) \star_{k-l} B_{k-l}, X \approx_? (\cdots((A \star_1 Y_1) \star_2 Y_2) \cdots) \star_{k-l} Y_{k-l}\}\rangle$ unifies with solution $\langle \emptyset, \{X/(\cdots((A \star_1 B_1) \star_2 B_2) \cdots) \star_{k-l} B_{k-l}\}\rangle$ which is a common solution for π and ϕ.

Example 10. (Continuing example 9) First, notice that the permutation cycle $\pi_1 = (a\,b\,c\,d\,e\,f\,g\,h)$ is not in the top of $(a\,d\,e\,b\,g\,h\,c\,f)$; also, π_1 is neither in the top of $(a\,b\,c\,d)$ nor in the top of $(a\,i)$. Since π_1 is not a factor of π_2, solutions generated from the extended pseudo-cycle $(\bar{a}\,\bar{d}\,\bar{e}\,\bar{b}\,\bar{g}\,\bar{h}\,\bar{c}\,\bar{f})$ might not be solutions built for π_1; for instance, consider the unitary extended pseudo-cycle built for π_2, $(((\bar{a} \star \bar{g}) \diamond (\bar{e} \star \bar{c})) \oplus ((\bar{d} \star \bar{h}) \diamond (\bar{b} \star \bar{f}))$, which is not a solution for π_1, since not $\pi_1((\bar{a} \star \bar{g}) \diamond (\bar{e} \star \bar{c})) \approx_C (\bar{d} \star \bar{h}) \diamond (\bar{b} \star \bar{f})$. Also, for the extended pseudo cycle $(\bar{a}\,\bar{b}\,\bar{c}\,\bar{d})$: the permutation cycles in π_1^2 are $(a\,c\,e\,g)$ and $(b\,d\,f\,h)$, which give different solutions. For $(\bar{a}\,\bar{i})$, the permutation cycle $(a\,e)$ in π_i^4 will produce different solutions.

Now consider solutions of fixpoint equations $\pi_i.X \approx_? X$, for $i = 1, 2, 3$, where $\mathit{\Pi}_X$ consists of the permutations $\pi_1 = (a\,b\,c\,d\,e\,f\,g\,h)$, $\pi_2 = (a\,g\,e\,c)(b\,f)$ and $\pi_3 = (a\,e)(c\,g)(d\,h)$. In this case, we have seen (Example 9) that π_1 is a permutation cycle in the top of $\mathit{\Pi}_X$. Among the solutions generated for $\pi_i.X \approx_? X$, for $i = 1, 2, 3$ through extended pseudo-cycles we have, respectively:
$\langle \nabla_1, \{X/s_1 = ((\bar{a} + \bar{e}) \star (\bar{c} + \bar{g})) \oplus ((\bar{b} + \bar{f}) \star (\bar{d} + \bar{h})\}\rangle$,
$\langle \nabla_2, \{X/s_2 = ((\bar{a} + \bar{e}) \star (\bar{c} + \bar{g})) \oplus ((\bar{b} + \bar{f}) \star Y\}\rangle$ and
$\langle \nabla_3, \{X/s_3 = ((\bar{a} + \bar{e}) \star (\bar{c} + \bar{g})) \oplus (Z \star (\bar{d} + \bar{h})\}\rangle$,
where $\nabla_1 = \emptyset$, $\nabla_2 = \{a\#Y, b\#Y, c\#Y, e\#Y, f\#Y, g\#Y\}$ and $\nabla_3 = \{a\#Z, c\#Z, d\#Z, e\#Z, g\#Z, h\#Z\}$, and the symbols $\oplus, \star$ and $+$ are commutative. The C-unification problem $\langle \nabla_1 \cup \nabla_2 \cup \nabla_3, \{X \approx_? s_1, X \approx_? s_2, X \approx_? s_3\}\rangle$ has solution $\{X/s_1, Y/\bar{d} + \bar{h}, Z/\bar{b} + \bar{f}\}$ with the respective freshness constraints; thus, restricting this solution to the freshness constraints on X we have the common solution $\langle \emptyset, \{X/s_1\}\rangle$.

The greedy generation algorithm can then be improved by generating solutions only for the atoms in permutation cycles in the top of $\mathit{\Pi}_X$.

5 Conclusions and future work

We presented a procedure to generate solutions of fixpoint nominal C-unification problems modulo commutativity. The procedure is proved to be sound and complete. This piece of work is relevant to provide a sound and complete procedure to generate solutions of nominal C-unification problems which consists of an initial phase in which nominal C-unification problems are transformed into an equivalent finite set of fixpoint problems, as described in [4] and the second phase that consist of the generation of potentially infinite set of independent solutions, presented in this paper, based on combinatorial properties of permutations.

Additional improvements of the generation procedure should be investigated exhaustively, as well as possible extensions of nominal unification, nominal matching, nominal narrowing modulo other equational theories of interest.

References

[1] T. Aoto and K. Kikuchi. *A Rule-Based Procedure for Equivariant Nominal Unification.* In *Pre-proc. of Higher-Order Rewriting (HOR)*, pages 1–5, 2016.

[2] T. Aoto and K. Kikuchi. *Nominal Confluence Tool.* In *Proc. of 8th Int. Joint Conf.: Automated Reasoning (IJCAR)*, volume 9706 of *LNCS*, pages 173–182. Springer, 2016.

[3] M. Ayala-Rincón, W. Carvalho-Segundo, M. Fernández, and D. Nantes-Sobrinho. *A Formalisation of Nominal Equivalence with Associative-Commutative Function Symbols.* In *Pre-proc. of Logical and Semantic Frameworks with Applications (LSFA)*, to appear in *ENTCS*, pages 78–93, 2016.

[4] M. Ayala-Rincón, W. Carvalho-Segundo, M. Fernández, and D. Nantes-Sobrinho. *Nominal C-Unification.* Available at `ayala.mat.unb.br/publications.htlm`, 2017.

[5] M. Ayala-Rincón, M. Fernández, and D. Nantes-Sobrinho. *Nominal Narrowing.* In *Proc. of 1st Int. Conf. on Formal Structures for Computation and Deduction (FSCD)*, volume 52 of *LIPIcs*, pages 1–16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016.

[6] M. Ayala-Rincón, M. Fernández, and A. C. Rocha-oliveira. *Completeness in PVS of a Nominal Unification Algorithm.* *ENTCS*, 323:57–74, 2016.

[7] J. Cheney. Equivariant unification. *J. of Automated Reasoning*, 45:267–300, 2010.

[8] M. Fernández and M. J. Gabbay. *Nominal Rewriting.* *Information and Computation*, 205(6):917–965, 2007.

[9] T. Kutsia, J. Levy, M. Schmidt-Schauß, and M. Villaret. *Nominal Unification of Higher Order Expressions with Recursive Let.* In *Proc. of 26th Int. Sym. on Logic-Based Program Synthesis and Transformation (LOPSTR)*, pages 1–15, 2016. To appear in *LNCS.*

[10] A. M. Pitts. Nominal Logic, a First Order Theory of Names and Binding. *Information and Computation*, 186(2):165–193, 2003.

[11] J. H. Siekmann. *Unification of Commutative Terms.* In *Proc. of An Int. Symp. on Symbolic and Algebraic Manipulation*, volume 72 of *LNCS*, pages 22–29. Springer, 1979.

[12] C. Urban. *Nominal Unification Revisited.* In *Proc. of Int. Work. on Unification (UNIF)*, volume 42 of *EPTCS*, pages 1–11, 2010.

[13] C. Urban, A. M. Pitts, and M. J. Gabbay. *Nominal Unification.* *Theoretical Computer Science*, 323(1-3):473497, 2004.

A Technical Proof

Lemma 1 (Extended pseudo-cycle correspondence between π and π^2). *For $k \geq 1$, $(A_0 \cdots A_{2^k-1})$ is an extended pseudo-cycle for π if, and only if, there exist $(B_0 \cdots B_{2^{k-1}-1})$ and $(C_0 \cdots C_{2^{k-1}-1})$* `epc` *'s for π^2 with a substitution σ such that atoms in its image belong to $dom(\pi) \backslash dom(\pi^2)$, and for $0 \leq j \leq 2^{k-1}-1$ one has $B_j\sigma \approx_{\{\alpha,C\}} A_{2j}$ and $C_j\sigma \approx_{\{\alpha,C\}} A_{2j+1}$.*

Proof. The proof is by induction in the construction of the extended pseudo-cycles.

Base Case.

1. Case 1 of Def. 5 does not apply since $k \geq 1$.
2. Let $(\overline{a}\ \overline{b})$ be a trivial `epc` for π, and let $(Y\ Y')$ be an `epc` for π^2. Notice that $dom(\pi^2) = dom(\pi) \backslash \{a \mid a \in \text{2-cycle of } \pi\}$.

Induction Step.

1. By Def. 5(3.a.i), from an `epc` for π, $(A_0 \cdots A_{2^k-1})$ one can build another `epc` in steps A and/or B. By i.h., there exist `epc` 's for π^2: $(B_0 \cdots B_{2^{k-1}-1})$ and $(C_0 \cdots C_{2^{k-1}-1})$ and a substitution σ such that $B_j\sigma \approx_{\{\alpha,C\}} A_{2j}$ and $C_j\sigma \approx_{\{\alpha,C\}} A_{2j+1}$. We consider two cases, depending on whether the `epc` obtained has length equal to 2^k or 2^{k-1}:
   (a) $(A_0 \star A_j \cdots A_{2^k-1} \star A_{2^k-1+j})$ with $j \neq 2^{k-1}$, then the length of the new `epc` does not change. From the two `epc` 's for π^2 we build the `epc` 's in the following way:
      - j is even: $(B_0 \star B_{\frac{j}{2}} \cdots B_{2^{k-1}-1} \star B_{2^{k-1}+\frac{j}{2}-1})$ and $(C_0 \star C_{\frac{j}{2}} \cdots C_{2^{k-1}-1} \star C_{2^{k-1}+\frac{j}{2}-1})$. Notice that the conditions of σ are preserved, for instance, $(B_0 \star B_{\frac{j}{2}})\sigma = A_0 \star A_j$.
      - j is odd: $(B_0 \star C_{\frac{j-1}{2}} \cdots B_{2^{k-1}-1} \star C_{2^{k-1}+\frac{j-1}{2}-1})$ and $(C_0 \star B_{\frac{j+1}{2}} \cdots C_{2^{k-1}-1} \star B_{2^{k-1}+\frac{j+1}{2}-1})$. Notice that $(B_0 \star C_{\frac{j-1}{2}})\sigma = B_0\sigma \star C_{\frac{j-1}{2}}\sigma = A_0 \star A_j$, similarly, one can check that the two `epc` 's satisfy the conditions on σ.

   (b) $(A_0 \star A_{2^{k-1}} \cdots A_{2^{k-1}-1} \star A_{2^k-1})$ with $j = 2^{k-1}$. From the `epc` 's for π^2 we build $(B_0 \star B_{2^{k-2}} \cdots B_{2^{k-2}-1} \star B_{2^{k-1}-1})$ and $(C_0 \star C_{2^{k-2}} \cdots C_{2^{k-2}-1} \star C_{2^{k-1}-1})$.
2. By Def. 5(3.a.ii), from `epc` 's for π, $(A_0 \cdots A_{2^k-1})$ and $(A'_0 \cdots A'_{2^{k'}-1})$, we build an `epc` for π, $(A_0 \star A'_j \cdots A_{2^k-1} \star A'_{2^{k'}-1})$, for some $0 \leq j \leq 2^k - 1$ (the case $0 \leq j \leq 2^{k'} - 1$ is analogous).
   By i.h., there exist `epc` 's for π^2, $(B_0 \cdots B_{2^{k-1}-1})$ and $(C_0 \cdots C_{2^{k-1}-1})$ for $(A_0 \cdots A_{2^k-1})$, and $(B'_0 \cdots B'_{2^{k'-1}-1})$ and $(C'_0 \cdots C'_{2^{k'-1}-1})$ for $(A'_0 \cdots A'_{2^{k'}-1})$, satisfying the conditions for σ, σ' respectively. We can choose B_i, C_i, B'_j, C'_j such that $var(B_i, C_i) \cap var(B'_j, C'_j) = \emptyset$. Then we have consider two cases:
   - for a j even: take $(B_0 \star B'_{\frac{j}{2}} \cdots B_{2^{k-1}-1} \star B'_{2^{k'-1}+\frac{j}{2}-1})$ and $(C_0 \star C'_{\frac{j}{2}} \cdots C_{2^{k-1}-1} \star C_{2^{k'-1}+\frac{j}{2}-1})$.

   - for a j odd: $(B_0 \star C'_{\frac{j-1}{2}} \cdots B_{2^{k-1}-1} \star C'_{2^{k'-1}+\frac{j-1}{2}-1})$ and $(C_0 \star B'_{\frac{j-1}{2}}) \cdots C_{2^{k-1}-1} \star B'_{2^{k'-1}+\frac{j-1}{2}-1})$.

   Its straightforward to check that the conditions for these epc of π^2 hold for $\sigma \cup \sigma'$.

3. By Def. 5(3.a.iii), from epc 's for π, $(A_0 \cdots A_{2^k-1})$ and $(A'_0 \cdots A'_{2^{k'}-1})$, we build the epc of π: $(\langle A_0, A'_j \rangle \cdots \langle A_{2^k-1}, A'_{2^{k'}+j-1} \rangle)$.

   By i.h., there exist epc 's for π^2, $(B_0 \cdots B_{2^{k-1}-1})$ and $(C_0 \cdots C_{2^{k-1}-1})$ for $(A_0 \cdots A_{2^k-1})$, and $(B'_0 \cdots B'_{2^{k'-1}-1})$ and $(C'_0 \cdots C'_{2^{k'-1}-1})$ for $(A'_0 \cdots A'_{2^{k'}-1})$, satisfying the conditions for σ, σ' resp. Similarly to the previous case, one can check that the result holds, depending on whether j is even or odd.
4. By Def. 5(3.a.iv), from $(A_0 \cdots A_{2^k-1})$ an epc for π, one can build either $([e]A_0 \cdots [e]A_{2^k-1})$ or $(g\,A_0 \cdots g\,A_{2^k-1})$. By i.h., there exist epc 's for π^2: $(B_0 \cdots B_{2^{k-1}-1})$ and $(C_0 \cdots C_{2^{k-1}-1})$ and a substitution σ satisfying the requirements. Its clear that $([e]B_0 \cdots [e]B_{2^{k-1}-1})$ and $([e]C_0 \cdots [e]C_{2^{k-1}-1})$ are epc 's in π^2, for $([e]A_0 \cdots [e]A_{2^k-1})$, and similarly, $(g\,B_0 \cdots g\,B_{2^{k-1}-1})$ and $(g\,C_0 \cdots g\,C_{2^{k-1}-1})$ are epc 's in π^2, for $(g\,A_0 \cdots g\,A_{2^k-1})$.
5. By Def. 5(3.a.v), from $(A_0 \cdots A_{2^k-1})$ and $(a_0 \cdots a_{2^l-1})$, resp. an epc and a permutation cycle of π, we build $([a_j]A_0 \cdots [a_{2^l+j-1}]A_{2^k-1})$, another epc for π. By i.h., we have for π^2 epc 's $(B_0 \cdots B_{2^{k-1}-1})$ and $(C_0 \cdots C_{2^{k-1}-1})$ and a substitution σ satisfying the necessary conditions.
   - Case $l = 1$, then $(a_0 \cdots a_{2^l-1}) = (a\,b)$, take $([a]B_0 \cdots [a]B_{2^{k-1}-1})$ and $([b]C_0 \cdots [b]C_{2^{k-1}-1})$.
   - Otherwise, $(a_0\,a_2 \cdots a_{2^l-2})$ and $(a_1\,a_3 \cdots a_{2^l-1})$ are permutation cycles of π^2, and the we take $([a_j]B_0 \cdots [a_{j-2}]B_{2^{k-1}-1})$ and $([a_{j+1}]C_0 \cdots [a_{j-1}]C_{2^{k-1}-1})$.

   In both cases σ satisfy the necessary conditions.

□