Universidade de Brasília

Nominal Commutative Narrowing

Daniella Santaguida Magalhães de Souza

Advisor: Dr. Daniele Nantes Sobrinho

Departamento de Matemática
Universidade de Brasília

Dissertation submitted in partial fulfillment of the requirements for the degree of
Master in Mathematics

Brasília, June 07, 2022

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Matemática

Nominal Commutative Narrowing

por

Daniella Santaguida Magalhães de Souza*

Dissertação apresentada ao Departamento de Matemática da Universidade de Brasília, como parte dos requisitos para obtenção do grau de

MESTRE EM MATEMÁTICA

Brasília, 07 de junho de 2022.

Comissão Examinadora:

____

Profa. Dra. Daniele Nantes Sobrinho - MAT/UnB (Orientadora)

____

Prof. Dr. Mauricio Ayala Rincon – MAT/UnB (Membro)

Daniel Lima Ventura
____

Prof. Dr. Daniel Lima Ventura – UFG (Membro)

* A autora foi bolsista da CAPES durante a elaboração desta dissertação.

Acknowledgements

I would like to thank some people who made this work possible and who play an essential role in my life.

I start by thanking God, who has gifted me with an amazing family that supports and encourages me at every stage of my life. To my father Anderson, thank you for being an example and the biggest encourager of my academic life. To my mother Silvia, thank you for being the solid base of this family and for always doing everything for us. To my brother Thiago, thank you for being an inspiration in the computer science area and also as a person.

Thanks to my friends who kept me sane during the difficult days of research. In particular, a thank you so much to my best friends: Maria Luiza, for making all the difference since the first day of graduation when we met, you know I would not get here today without you, forever “Chris and Greg”; and Fernanda, more than fifteen years of friendship and you are certainly one of the people I admire the most for your dedication, focus and resilience, even though you are in a totally different area. To you both, I am extremely grateful for everything.

To my boyfriend Lucas, there are no words to describe how much you supported me not only throughout my Masters but throughout my life. You always believed in me, even when I did not, and you never let me give up. I continue to learn a lot from you every day. Even after hearing so many “narrowing” words you stayed. I love you and thank you.

All the professors I have had along the way have a space here in this gratitude. But one of them in particular plays a significant role in my academic life. From the first *Calculus I* class until the orientation, you continue to be my inspiration and example of a woman researcher. Thank you very much, Professor Daniele Nantes. This achievement is dedicated to you.

Finally, I would like to thank the Master’s examining board, Professors Maurício Ayala-Rincón and Daniel Ventura, for all the comments and suggestions that definitely enhanced this work. And a special thanks to Professor Maribel Fernández for her collaboration.

Resumo

Modelagem e raciocínio equacional são onipresentes na Matemática e na Ciência da Computação. Técnicas de reescrita têm sido aplicadas com sucesso para formalizar e implementar inferência automatizada em estruturas matemáticas dedutivas. Apresentar teorias equacionais por meio da reescrita dá origem a um mecanismo para decidir a redução equacional da teoria sempre que o sistema de reescrita for terminante e confluente, ou seja, sempre que for convergente. Resolver problemas equacionais é um passo adiante que requer mais esforço do que apenas usar reescrita. De fato, "estreitar" problemas equacionais é uma técnica bem conhecida que adiciona à reescrita o poder necessário para buscar soluções; em outras palavras, adiciona o poder de buscar instâncias das variáveis que ocorrem em um problema equacional que "unifica" as equações.

Por sua vez, a lógica nominal foi desenvolvida para contornar as inconveniências apresentadas quando as variáveis são instanciadas. A abordagem nominal usa átomos nominais em vez de variáveis para evitar a necessidade de renomeação de variáveis ao lidar com equações na abordagem notacional padrão. A sintaxe nominal também inclui permutações de átomos para distinguir algebricamente os átomos evitando colisões e capturas destes.

Neste trabalho, estudamos a reescrita nominal módulo comutatividade. Desenvolvemos o método estreitamento nominal comutativo (*nominal commutative narrowing*) para lidar com o problema de unificação nominal módulo teorias equacionais que incluem comutatividade, o qual não é finitário dependendo da representação das soluções.

Abstract

Equational modelling and reasoning are ubiquitous in Mathematics and Computer Science. Rewriting techniques have been applied successfully to formalize and implement automated inference in mathematical deductive frameworks. Presenting equational theories by rewriting gives rise to a mechanism to decide the equational reduct of the theory whenever the rewriting system is terminating and confluent, i.e., whenever it is convergent. Solving equational problems is a step further that requires more effort than just rewriting. Indeed, “narrowing” equational problems is a well-known technique that adds to rewriting the required power to search for solutions; in other words, it adds the power to search for instantiations of the variables occurring in an equational problem that “unify” the equations.

On its side, the nominal logic has been developed to contour inconveniences presented when variables are instantiated. The nominal approach uses nominal atoms instead of variables to avoid the requirement of variable renaming when dealing with equations in the standard notational approach. The nominal syntax also includes atom permutations to algebraically distinguish atoms avoiding atom collisions and captures.

In this work, we study nominal rewriting modulo commutativity. We develop nominal commutative narrowing to deal with the problem of nominal unification modulo equational theories that include commutativity, which is not finitary depending on the representation of solutions.

Table of contents

Introduction

The E-unification problem is about solving first-order term equations modulo an equational theory E, that is, given an equational theory E and a unification problem $\{s \approx^? t\}$, we want to find a substitution θ such that $s\theta \approx_{\mathsf{E}} t\theta$, for some s and t first-order terms. E-unification is a fundamental tool in logic programming and type assignment systems, and applications may be found at the use of paramodulation, a development of resolution, in automatic theorem proving as well as the computation of complete sets of critical pairs in the Knuth and Bendix completion procedure [25, 17].

Given an equational theory that has an equivalent confluent and terminating term rewrite system (TRS), Fay described a complete unification algorithm for it [11, 17] in the late 70's. The method is based on using the narrowing process by Lankford in 1975 [18]. Narrowing was first mentioned by Slagle in 1974 [21], and Fay gave the first description of the narrowing technique used as a general E-unification procedure in the presence of a term rewrite system in 1978 [11].

Narrowing a term is about finding a substitution, which is the minimal instantiation of the term, such that we are able to apply one rewrite step right after. If we aim to solve an equation $\{s \approx^? t\}$ in an equational theory, that corresponds to a convergent TRS, we may construct all the possible narrowing derivations, iteratively, from the initial terms s and t until we get a new equation $\{s' \approx^? t'\}$ such that s' and t' are syntactically equal and therefore the equation can be trivially solved. Hence the composition of the most general unifier with all the substitutions computed in the narrowing derivations yields an E-unifier of $\{s \approx^? t\}$, see [23, 25].

However, in general this procedure does not terminate. The search procedure may contain various narrowing sequences starting from $\{s \approx^? t\}$, and the more rules in the TRS, the more options for the narrowing sequences. In order to avoid useless computations and to give sufficient conditions for termination, Hullot improved Fay's algorithm [15]. Now, new problems have to be treated: there are several equational theories whose axioms cannot be oriented without loosing the property of finite termination, which is the case of the commutativity axiom, and even though we apply Hullot's algorithm into a convergent set of rules, it does not always terminate.

This limitation of dealing with commutative operators and axioms was circumvented by not using C as a rewrite rule, but taking it into account when applying another rule. This way, we build commutativity into the rewrite process. The main idea is to decompose a set of axioms into a set E of troublesome identities, like C, and a set R of rewrite rules, with the purpose of generating a new rewrite relation $\to_{\mathsf{R}/\mathsf{E}}$, defined on equivalence classes of terms:

$$[s]_{\approx_{\mathsf{E}}} \to_{\mathsf{R}/\mathsf{E}} [t]_{\approx_{\mathsf{E}}} \text{ iff } \exists\, s', t' \,.\, s \approx_{\mathsf{E}} s' \to_{\mathsf{R}} t' \approx_{\mathsf{E}} t.$$

However, if we want to make a one-step reduction from $[s]_{\approx_{\mathsf{E}}}$ with relation to $\to_{\mathsf{R}/\mathsf{E}}$, we would need to investigate the entire equivalence class modulo $\approx_{\mathsf{E}}$ of s. Although, this would ask for all E-equivalence classes to be finite, which is the case for commutativity, but it is not the case for the theory $\mathsf{I} = \{x \oplus 0 \approx x\}$, for instance, $[a]_{\approx_{\mathsf{I}}} = \{a, a \oplus 0, (a \oplus 0) \oplus 0, ((a \oplus 0) \oplus 0) \oplus 0, \ldots\}$, where a and 0 are constants and $\oplus$ is a binary function symbol.

In this work we chose to follow the approach by Jouannaud et. al. [17], which is a generalization of Peterson and Stickel's work [20], and define another relation $\to_{\mathsf{R},\mathsf{E}}$:

$$s \to_{\mathsf{R},\mathsf{E}} t \text{ iff } \exists\, (l \to r) \in \mathsf{R}, s \equiv \mathtt{C}[s'], \theta \,.\, s' \approx_{\mathsf{E}} l\theta \wedge t = \mathtt{C}[r\theta],$$

which means that each rewrite step involves matching modulo $\approx_{\mathsf{E}}$. This allows to avoiding investigating the whole E-equivalence class prior to applying rewriting steps modulo E, and is decidable as long as E-unification is decidable and has a complete set of E-unifiers.

We are interested in extending the above mentioned results to the nominal framework.

Nominal techniques. The nominal syntax [14] emerged to deal with languages that have binders, such as the First Order Logic language, which has existential ($\exists$) and universal ($\forall$) quantifiers that bind variables in their scope, e.g. $\forall x.Q(x)$, for some unary predicate Q; λ-calculus, which has the (λ) abstraction binding variables, as in $\lambda x.xy$; or the π-calculus with the restriction (ν) quantifier, that binds name channels in a process (say P), as in $\nu x.P$. The nominal language is more expressive than all languages above, and its binder ([_]), also called abstraction, binds atoms, as in $[a]f(a,X)$ for a binary function symbol f, and can be "instantiated" to express abstractions in λ-calculus, of quantified first-order formulas, or processes in the π-calculus: one just need to adapt its signature.

Thus, in the nominal syntax, we have two different kinds of objects: *atoms* $(a, b, \ldots)$, that can be abstracted by a binder operator, and *unknowns* $(X, Y, \ldots)$, undefined variables that cannot be abstracted. In addition to the latter, a term may contain *function symbols*

$(f, g, \ldots)$, which are term constructors. That said, in the given examples, $\forall$, P and f are function symbols whilst a is an abstracted atom.

In the nominal setting, equality is established as an α-equivalence relation and we write $s \approx_\alpha t$ to denote that s is equal, modulo renaming of bound atoms, to t. For instance, $\lambda x.xy \approx_\alpha \lambda z.zy$ but $\lambda x.xy \not\approx_\alpha \lambda y.yy$ in the λ-calculus. Even thought $\lambda x.xy$ and $\lambda z.zy$ are syntactically different, they may be considered the "same term", that is, they are in the same α-equivalence class, we just need to choose one representative of the class.

Building upon this scenario, the *Nominal Unification* [24] is developed, which is the problem of solving equations modulo α-equivalence between two nominal terms ($s \approx_\alpha^? t$), and later extended to *Nominal* E*-Unification*, when E is one of the theories A, C and AC [1, 2, 7], which is also the subject of this work, but restricted to the theory C and following the narrowing approach.

Nominal E-unification via nominal narrowing was already investigated when E can be presented by a convergent nominal rewrite system [6]. In this work, we are interested in giving another step towards a more general development, treating the case in which E cannot be oriented as a convergent nominal rewrite system, and one has to deal with nominal rewriting/narrowing modulo some identities, thus extending the works by [17, 25, 10]. In particular, we will provide new developments for the particular theory of commutativity, which already gives some insight about the intricate extensions and properties that need to be addressed when dealing with more complex theories.

Objective. In this dissertation we investigate how nominal equational narrowing works and also its connection to rewriting modulo an equational theory E. In particular, for concrete developments, we study how the theory of commutativity behaves inside the nominal framework and how it influences the narrowing method. We call this study *Nominal Commutative Narrowing*.

We would like to stress that the objective of this dissertation is not to repeat proofs of the established results, so instead, we present the results that are fundamental for our developments and illustrate them by means of examples, remarks and more detailed explanations. The focus here is to give intuition of existing constructions and build up on them, with the aim to achieve new developments.

Contribution. The contributions of the work done in this dissertation consist of the detailed presentation of concepts and results about the development of nominal narrowing that were initially established in [6] and [12]. Furthermore we present some unpublished contributions:

1. We extend Lemma 22 of [12] (here Lemma 1.4) to consider the commutative theory, using the relation $\approx_{\alpha,\mathsf{C}}$, i.e., α-equivalence modulo commutativity.

2. We define nominal C-narrowing relation, written as $\rightsquigarrow_{\mathsf{R},\mathsf{C}}$.

3. We extend the definitions and concepts regarding rewriting modulo C to the nominal framework. For instance, we have nominal versions of the relations $\rightarrow_{\mathsf{R},\mathsf{C}}$ and $\rightarrow_{\mathsf{R}/\mathsf{C}}$ as described earlier in this introduction.

4. We present a version of the Lifting Theorem 2.2 taking into account commutativity:

   - we provide a mapping from nominal C-narrowing sequences to nominal rewriting modulo C sequences, provided that some conditions of freshness constraints are given (Theorem 3.1).

   - we also provide a naive approach to the reverse mapping, from nominal rewriting modulo C to nominal C-narrowing (Theorem 3.2)., provided that a notion of C-coherence (Definition 3.6) is given for nominal frameworks. The relation is necessary for relating $\rightarrow_{\mathsf{R}/\mathsf{C}}$ and $\rightarrow_{\mathsf{R},\mathsf{C}}$, but this has to be further investigated.

Organization. This work is organized in the following way:

- **Chapter 1: Background.** This chapter contains the definitions and properties of the nominal syntax that are necessary for the understanding and development of this work. In Section 1.1, we present the grammar of nominal terms, as well as basic definitions such as signature, permutations and substitutions. Section 1.2 presents the rules for freshness and α-equivalence, and we show them in examples of derivations. In Section 1.3 we recall the concepts of nominal unification: problems, simplification rules, solutions, as well as the formal definitions of nominal unification and nominal matching problems, concluding with the definition of most general unifier and also exhibiting some interesting examples of nominal unification problems. In Section 1.4 we present extensions of the concept of α-equivalence to α-equivalence modulo C, adding commutativity into the equational reasoning. We also show the adaptations of the system of derivation rules using $\approx_{\alpha,\mathsf{C}}$ and check how it works in an example. Later, we extend Lemma 22 from [12] considering the commutative theory (Lemma 1.4). Finally, in Subsection 1.4.1 we add commutativity into nominal unification problems, bringing format of solutions via the use of triples of the form (Δ, θ, Pr) together with the algorithm rules for nominal C-unification, complementing with examples.

- **Chapter 2: Nominal Narrowing.** We present the definitions and results regarding nominal narrowing. In Section 2.1, we recall definitions given by [6] such as judgements, equational theory, rewrite theory, nominal rewriting, with properties of confluence and equivariance; and we also present the concepts of nominal algebra equality and closed nominal rewriting in Subsection 2.1.1. In Section 2.2, we present our definition of nominal narrowing and some examples, together with the definition of normalized substitution with the aim of presenting the Nominal Lifting Theorem, in Subsection 2.2.1, with an example of the theorem in action. Finally, in Section 2.3, we explain that for a closed nominal equational theory, we can prove soundness and completeness of the nominal narrowing process for nominal unification.

- **Chapter 3: Nominal Commutative Narrowing.** This chapter focuses on showing all requirements to extend the lifting result for narrowing modulo C, i.e., to establish the correspondence between nominal C-narrowing and nominal rewriting modulo C. In Section 3.1, we define equational nominal rewrite system, where we split a theory T into a set R of rules and a set E of equations. We also bring some definitions from Jouannaud et. al [17] in order to present our definition of nominal rewriting modulo C, together with C-confluence and C-termination, and at last but not least nominal narrowing modulo C. Section 3.2 presents our extension of the Lifting Theorem 2.2, making a correspondence between the relations $\rightsquigarrow_{\mathsf{R,C}}$ and $\rightarrow_{\mathsf{R,C}}$. First from $\rightsquigarrow_{\mathsf{R,C}}$ to $\rightarrow_{\mathsf{R,C}}$ we managed to provide the complete proof; the other direction though, $\rightarrow_{\mathsf{R,C}}$ to $\rightsquigarrow_{\mathsf{R,C}}$, requires an extra property, called C-coherence, that is fundamental for C-confluence and this needs to be further investigated. We provided some proof sketches.

- **Chapter 4: Conclusion and Future Work.** We conclude the work with the main considerations of the development of this dissertation and we also propose some tasks for future work.

Chapter 1

Background

This chapter contains standard definitions, notations and basic results about nominal techniques that are necessary for the understanding of this work. Section 1.4 contains our first contribution: in Lemma 1.4, we prove an extension of a result that relates α-equivalence derivations with normalized freshness constraints (Lemma 22, in [12]), but taking into account the relation $\approx_{\alpha,\mathsf{C}}$. Subsection 1.4.1 contains established results in the context of nominal reasoning modulo commutativity: we present the rule-based algorithm for nominal C-unification as well as some properties and examples. More details can be found in [1]. The notations are mostly consistent with [6, 12].

1.1 Nominal Syntax

A *nominal signature* Σ is a set of *function symbols* $f, g, \ldots$, each with a fixed *arity* $n \geq 0$. Fix countably infinite sets $\mathcal{X}$ of *term variables* $X, Y, Z, \ldots$ and $\mathcal{A}$ of *atoms* $a, b, c, d, \ldots$. Variables represent meta-level *unknowns* and atoms represent object-level variable symbols. We assume that Σ, $\mathcal{X}$ and $\mathcal{A}$ are pairwise disjoint. As usual, we will write $a \equiv a$ and $X \equiv X$ to denote syntactic identity of atoms and unknowns. We will omit the explicit signature Σ and the arity of its function symbols when it is clear from the context.

A *swapping* is a pair of atoms, written $(a\ b)$, that maps a to b, b to a and all other atoms c to themselves. Although $(a\ b)$ and $(b\ a)$ are different swappings, they make the same action over terms. A *permutation* π is a bijection on atoms, with finite domain. We call $\mathtt{Id}$ the *identity permutation*. Also, $\pi \circ \pi'$ denotes *functional composition* of permutations and π^{-1} denotes the *inverse* of π. We call a pair of a permutation π and a variable X a *moderated variable*, written $\pi \cdot X$, and we say that π is *suspended* on X.

Definition 1.1. (Nominal term) *Nominal terms* are generated inductively by the grammar

$$t ::= a \mid \pi \cdot X \mid [a]t \mid f(t_1, \dots, t_n).$$

Terms are called, respectively, *atoms*, *suspensions*, *abstractions* and *function applications*. Notice that X is not a term, but $\mathtt{Id} \cdot X$ is. We abbreviate $\mathtt{Id} \cdot X$ as X when there is no ambiguity. We also write $t \equiv t$ to denote syntactic identity of terms.

We write $V(t)$ for the set of variables occurring in t and $A(t)$ for the set of atoms mentioned in t. Terms without variables are called *ground terms*, that is, $V(t) = \emptyset$, but they may still contain atoms. Abstractions like $[a]t$ mean that a is bound in t (scope of $[a]$). Occurrences of a are said to be *abstracted* if they occur in the scope of t, and are called *free* otherwise.

When convenient we will write $f(\bar{t})_n$ to denote $f(t_1, \dots, t_n)$.

Example 1.1. The following are all examples of nominal terms when we consider the signature $\Sigma = \{\mathtt{app} : 2, \mathtt{lam} : 1, \mathtt{add} : 2, \mathtt{succ} : 1, \mathtt{let} : 2, \mathtt{map} : 2, f : 1, \mathtt{nil} : 0\}$:

$$\mathtt{app}(\mathtt{lam}([a]a), X) \qquad \mathtt{add}(\mathtt{succ}(X), Y) \qquad (a\ b) \cdot X \qquad \mathtt{let}([a]a, b) \qquad \mathtt{map}\ f\ \mathtt{nil}$$

A *position* $\mathtt{C}$ is defined as a pair $(s, \_)$ of a term and a distinguished variable $\_ \in \mathcal{X}$ that occurs exactly once in s. We write $\mathtt{C}[s']$ for $\mathtt{C}[\_ \mapsto s']$ and if $s \equiv \mathtt{C}[s']$, we say that s' is a subterm of s with position $\mathtt{C}$. The root position will be denoted by $\mathtt{C} = [\_]$.

Definition 1.2. (Permutation action) A *permutation action* of π on a term t is defined by induction on the term structure

$$\pi \cdot a = \pi(a) \qquad \pi \cdot (\pi' \cdot X) = (\pi \circ \pi') \cdot X$$

$$\pi \cdot [a]t = [\pi \cdot a](\pi \cdot t) \qquad \pi \cdot f(t_1, \dots, t_n) = f(\pi \cdot t_1, \dots, \pi \cdot t_n).$$

Example 1.2. By the definition above, and adding the function symbol $\{h : 3\}$ to Σ from Example 1.1,

$$(a\ b)(c\ d) \cdot \big(\mathtt{lam}([a]\mathtt{lam}([d]h(c, b, X)))\big) \equiv \mathtt{lam}([b]\mathtt{lam}([c]h(d, a, (a\ b)(c\ d) \cdot X))).$$

Definition 1.3. (Meta-action of a permutation) The *meta-action* of π on t, written t^{π}, is defined by

$$a^{\pi} = \pi(a) \qquad (\theta \cdot X)^{\pi} = \theta^{\pi} \cdot X$$

$$([a]t)^{\pi} = [a^{\pi}]t^{\pi} \qquad f(t_1 \dots, t_n)^{\pi} = f(t_1^{\pi}, \dots, t_n^{\pi}),$$

where $\mathtt{Id}^{\pi} = \mathtt{Id}$ and $((a\ b) \cdot \theta)^{\pi} = (\pi(a)\ \pi(b)) \cdot \theta^{\pi}$.

The meta-action of permutations affects only atoms in terms [5]. It does not suspend on variables, as permutation action does.

Definition 1.4. (Substitution) A *substitution* θ is a mapping from a finite set of variables to terms. The *substitution action* $t\theta$ is defined as follows

$$a\theta = a \qquad (\pi \cdot X)\theta = \pi \cdot (X\theta)$$

$$([a]t)\theta = [a](t\theta) \qquad f(t_1, \ldots, t_n)\theta = f(t_1\theta, \ldots, t_n\theta).$$

The domain of a substitution θ is written as $\mathtt{dom}(\theta)$, and the image is denoted as $\mathtt{Im}(\theta)$. Therefore, if $X \notin \mathtt{dom}(\theta)$ then $X\theta = X$. Also if we restrict the domain to a certain set V of variables, where the substitution will map X to $X\theta$, if $X \in V$, and to X, otherwise, then we call $\theta|_V$ the *restriction of* θ *to* V.

The composition of two substitutions θ_1 and θ_2 will be written as $\theta_1\theta_2$.

Note that substitution does not avoid capture of free atoms.

Lemma 1.1. Substitution and permutation commute, that is, $\pi \cdot (t\theta) = (\pi \cdot t)\theta$.

Proof. The proof is by induction on the structure of t.

Base Case.

- If $t \equiv a$: the result is trivial since the substitution does not affect atoms;
- If $t \equiv \pi' \cdot X$:

$$\begin{array}{rcl} \pi \cdot ((\pi' \cdot X)\theta) & = & \pi \cdot (\pi' \cdot (X\theta)) \\ & = & (\pi \circ \pi') \cdot (X\theta) \\ & = & ((\pi \circ \pi') \cdot X)\theta \\ & = & (\pi \cdot (\pi' \cdot X))\theta; \end{array}$$

Inductive Step.

- If $t \equiv [a]t'$:

$$\begin{array}{rcl} \pi \cdot (([a]t')\theta) & = & \pi \cdot ([a](t'\theta)) \\ & = & [\pi \cdot a](\pi \cdot (t'\theta)) \\ & \overset{IH}{=} & [\pi \cdot a]((\pi \cdot t')\theta) \\ & = & ([\pi \cdot a](\pi \cdot t'))\theta \\ & = & (\pi \cdot ([a]t'))\theta; \end{array}$$

- If $t \equiv f(t_1, \ldots, t_n)$:

$$\begin{array}{rcl} \pi \cdot (f(t_1, \ldots, t_n)\theta) & = & \pi \cdot (f(t_1\theta, \ldots, t_n\theta)) \\ & = & f(\pi \cdot (t_1\theta), \ldots, \pi \cdot (t_n\theta)) \\ & \overset{IH}{=} & f((\pi \cdot t_1)\theta, \ldots, (\pi \cdot t_n)\theta) \\ & = & f(\pi \cdot t_1, \ldots, \pi \cdot t_n)\theta \\ & = & (\pi \cdot f(t_1, \ldots, t_n))\theta. \end{array}$$

□

1.2 Rules for Freshness and α-equivalence

Whenever we are talking about nominal terms and how we relate them, we need to understand first the notions of *freshness* (denoted by the predicate #) and α*-equivalence* (denoted by the predicate $\approx_\alpha$):

- Intuitively, $a\#t$ means that a does not occur free in t (read "a fresh in t").
- Intuitively, $s \approx_\alpha t$ means that s and t are α-equivalent, that is, they are the same term written with a different choice of bound names.
- α-equivalence is defined using swappings and freshness.

Here $a\#t$ and $s \approx_\alpha t$ represent constraints. The inference rules defining freshness and α-equivalence are given in Figure 1.1 and Figure 1.2. A *freshness context* is a set of freshness constraints of the form $a\#X$, and usually denoted by Δ and ∇.

We call a freshness constraint $a\#s$ *reduced* when it is of the form $a\#a$ or $a\#X$. If there are no constraints of the form $a\#a$ in these contexts, we call them consistent. Also, a *freshness judgement* is a tuple of the form $\Delta \vdash a\#t$ and an α*-equivalence judgement* is a tuple of the form $\Delta \vdash s \approx_\alpha t$. We will write $\Delta \vdash (\phi_1, \ldots, \phi_n)$ for the judgements $\Delta \vdash \phi_1, \ldots, \Delta \vdash \phi_n$.

In Figure 1.2 we use the *difference set* of two permutations $ds(\pi, \pi') := \{n \mid \pi \cdot n \neq \pi' \cdot n\}$. So $ds(\pi, \pi')\#X$ represents the set of constraints $\{n\#X \mid n \in ds(\pi, \pi')\}$. For example, if $\pi = (a\ b)(c\ d)$ and $\pi' = (c\ b)$, then $ds(\pi, \pi') = \{a, b, c, d\}$, and $ds(\pi, \pi')\#X = \{a\#X, b\#X, c\#X, d\#X\}$.

Remark 1.1. The rule ($\approx_\alpha$ [ab]) in Figure 1.2 is equivalent to a rule with premises $\Delta \vdash (b\ a) \cdot s \approx_\alpha t$ and $\Delta \vdash b\#s$.

Example 1.3. We can derive $\Delta \vdash a\#f([a]X, (a\ b) \cdot Y)$ with $\Delta = \{b\#Y\}$ and observing that $(a\ b)^{-1} \cdot a \equiv (b\ a) \cdot a \equiv b$:

$$\frac{}{\Delta \vdash a\#b}\,(\#\ \text{atom}) \qquad\qquad \frac{\Delta \vdash a\#t_1 \ \cdots\ \Delta \vdash a\#t_n}{\Delta \vdash a\#f(t_1,\cdots,t_n)}\,(\#\ \text{app})$$

$$\frac{}{\Delta \vdash a\#[a]t}\,(\#\ \text{a[a]}) \qquad \frac{\Delta \vdash a\#t}{\Delta \vdash a\#[b]t}\,(\#\ \text{a[b]}) \qquad \frac{(\pi^{-1}\cdot a\#X) \in \Delta}{\Delta \vdash a\#\pi\cdot X}\,(\#\ \text{var})$$

Fig. 1.1 Rules for #

$$\frac{}{\Delta \vdash a \ \approx_\alpha\ a}\,(\approx_\alpha \text{ atom}) \qquad\qquad \frac{\Delta \vdash s_1 \ \approx_\alpha\ t_1 \ \cdots\ \Delta \vdash s_n \ \approx_\alpha\ t_n}{\Delta \vdash f(s_1,\cdots,s_n) \ \approx_\alpha\ f(t_1,\cdots,t_n)}\,(\approx_\alpha \text{ app})$$

$$\frac{\Delta \vdash s \ \approx_\alpha\ t}{\Delta \vdash [a]s \ \approx_\alpha\ [a]t}\,(\approx_\alpha \text{ [aa]}) \qquad\qquad \frac{\Delta \vdash s \ \approx_\alpha\ (a\ b)\cdot t \qquad \Delta \vdash a\#t}{\Delta \vdash [a]s \ \approx_\alpha\ [b]t}\,(\approx_\alpha \text{ [ab]})$$

$$\frac{ds(\pi,\pi')\#X \in \Delta}{\Delta \vdash \pi\cdot X \ \approx_\alpha\ \pi'\cdot X}\,(\approx_\alpha \text{ var})$$

Fig. 1.2 Rules for $\approx_\alpha$

$$\frac{\dfrac{}{\Delta \vdash a\#[a]X}\,(\#\ \text{a[a]}) \qquad \dfrac{b\#Y \in \Delta}{\Delta \vdash a\#(a\ b)\cdot Y}\,(\#\ \text{var})}{\Delta \vdash a\#f([a]X,(a\ b)\cdot Y)}\,(\#\ \text{app}).$$

With $\Delta' = \{a\#X\}$ we can deduce $\Delta' \vdash a\#h([b]X,c)$:

$$\frac{\dfrac{\dfrac{a\#X \in \Delta'}{\Delta' \vdash a\#X}\,(\#\ \text{var})}{\Delta' \vdash a\#[b]X}\,(\#\ \text{a[b]}) \qquad \dfrac{}{\Delta' \vdash a\#c}\,(\#\ \text{atom})}{\Delta' \vdash a\#h([b]X,c)}\,(\#\ \text{app}).$$

Also we can derive $\nabla \vdash \mathtt{lam}([d]\mathtt{app}(\mathtt{lam}([d]X),c)) \approx_\alpha \mathtt{lam}([d]\mathtt{app}(\mathtt{lam}([c]X),c))$ with $\nabla = \{c\#X, d\#X\}$, noting that $\pi := (c\ d)$ and $ds(\pi,\mathtt{Id}) = \{c,d\}$:

$$\frac{\dfrac{\dfrac{\dfrac{\dfrac{ds(\mathtt{Id},(d\ c)) = \{c,d\#X\} \in \nabla}{\nabla \vdash X \approx_\alpha (d\ c)\cdot X}\,(\approx_\alpha \text{ var}) \qquad \dfrac{d\#X \in \nabla}{\nabla \vdash d\#X}\,(\#\ \text{var})}{\nabla \vdash [d]X \approx_\alpha [c]X}\,(\approx_\alpha \text{ [dc]})}{\nabla \vdash \mathtt{lam}([d]X) \approx_\alpha \mathtt{lam}([c]X)}\,(\approx_\alpha \text{ app}) \qquad \dfrac{}{\nabla \vdash c \approx_\alpha c}\,(\approx_\alpha \text{ atom})}{\nabla \vdash \mathtt{app}(\mathtt{lam}([d]X),c) \approx_\alpha \mathtt{app}(\mathtt{lam}([c]X),c)}\,(\approx_\alpha \text{ app})}{\dfrac{\nabla \vdash [d]\mathtt{app}(\mathtt{lam}([d]X),c) \approx_\alpha [d]\mathtt{app}(\mathtt{lam}([c]X),c)}{\nabla \vdash \mathtt{lam}([d]\mathtt{app}(\mathtt{lam}([d]X),c)) \approx_\alpha \mathtt{lam}([d]\mathtt{app}(\mathtt{lam}([c]X),c))}\,(\approx_\alpha \text{ app}).}\,(\approx_\alpha \text{ [dd]})$$

1.3 Nominal Unification

Unification is a mechanism used to find out if two terms can be equal using a proper substitution. In the nominal syntax, the notion of syntactic equality is replaced by α-equality, therefore Nominal Unification uses this notion intrinsically. In order to understand this method, we recall some definitions from [24].

Definition 1.5. (Problem) A *problem Pr* is a set of constraints and we write $\Delta \vdash Pr$ when for all $P \in Pr$ there is a derivation proof using the rules in Figures 1.1 and 1.2, taking elements of the context Δ as assumptions. $\Delta \vdash Pr$ is read as "Δ *entails Pr*". If Δ entails P because $P \in \Delta$, we say the derivation is trivial.

Here we bring an algorithm based on simplification rules acting on problems, since the rules for freshness and for α-equivalence give us terms above the line simpler than the terms below.

Definition 1.6. (Simplification rules) Here are the *simplification rules* for problems:

$$\boxed{\begin{array}{rcll} a\#b, Pr & \Longrightarrow & Pr & \\ a\#f(t_1,\cdots,t_n), Pr & \Longrightarrow & a\#t_1,\cdots,a\#t_n, Pr & \\ a\#[b]t, Pr & \Longrightarrow & a\#t, Pr & \\ a\#[a]t, Pr & \Longrightarrow & Pr & \\ a\#\pi\cdot X, Pr & \Longrightarrow & \pi^{-1}\cdot a\#X, Pr & \pi \neq Id \end{array}}$$

$$\boxed{\begin{array}{rcl} a \approx_\alpha a, Pr & \Longrightarrow & Pr \\ f(s_1,\cdots,s_n) \approx_\alpha f(t_1,\cdots,t_n), Pr & \Longrightarrow & s_1 \approx_\alpha t_1,\cdots,s_n \approx_\alpha t_n, Pr \\ [a]s \approx_\alpha [a]t, Pr & \Longrightarrow & s \approx_\alpha t, Pr \\ [a]s \approx_\alpha [b]t, Pr & \Longrightarrow & s \approx_\alpha (a\ b)\cdot t, a\#t, Pr \\ \pi\cdot X \approx_\alpha \pi'\cdot X, Pr & \Longrightarrow & ds(\pi,\pi')\#X, Pr \end{array}}$$

Fig. 1.3 Simplification Rules

When a problem Pr' is obtained from another problem Pr using one of these simplification rules, we write $Pr \Longrightarrow Pr'$. For the transitive and reflexive closure of $\Longrightarrow$, we use the notation $\overset{*}{\Longrightarrow}$.

If we manage to reduce a set of constraints to the empty set, then the problem holds. Otherwise, we need to take this set of reduced constraints as assumptions to derive them.

Example 1.4. Because the simplification rules make the reverse role of the derivation rules, from Example 1.3 we conclude the following:

$$a\#f([a]X,(a\ b)\cdot Y) \overset{*}{\Longrightarrow} b\#Y \qquad\qquad a\#h([b]X,c) \overset{*}{\Longrightarrow} a\#X$$

$$\mathtt{lam}([d]\mathtt{app}(\mathtt{lam}([d]X),c)) \approx_\alpha \mathtt{lam}([d]\mathtt{app}(\mathtt{lam}([c]X),c)) \overset{*}{\Longrightarrow} c\#X, d\#X$$

We bring some important results from Fernández and Gabbay's paper [12]. The proofs will be omitted and can be found in their work.

Lemma 1.2. The relation $\Longrightarrow$ is confluent and strongly normalizing.

Proof. The proof can be found in [12], Lemma 11. □

Let $\langle Pr\rangle_{nf}$ denote the normal form of the problem Pr, that is, $Pr \Longrightarrow^* Pr' = \langle Pr\rangle_{nf}$ and there is no Pr'' such that $Pr' \Longrightarrow Pr''$.

Lemma 1.3. Consider Pr and Pr' problems.

(1) $\langle Pr \cup Pr'\rangle_{nf} = \langle Pr\rangle_{nf} \cup \langle Pr'\rangle_{nf}$. If $Pr \subseteq Pr'$ then $\langle Pr\rangle_{nf} \subseteq \langle Pr'\rangle_{nf}$.

(2) Assume $Pr \overset{*}{\Longrightarrow} Pr'$. Then $\Gamma \vdash Pr$ if and only if $\Gamma \vdash Pr'$.

(3) $\Gamma \vdash Pr$ if and only if $\Gamma \vdash \langle Pr\rangle_{nf}$.

Proof. The proof can be found in [12], Corollary 12 and Lemma 15. □

We can also enrich the simplification rules with the instantiating rules, in order to solve unification problems, which we will define ahead.

Instantiating rules:

$$\boxed{\begin{array}{llll} \pi \cdot X \;{}_?{\approx}_?\; t, Pr & \overset{X\mapsto\pi^{-1}\cdot t}{\Longrightarrow} & Pr[X \mapsto \pi^{-1}\cdot t] & (X \notin V(t)) \\ t \;{}_?{\approx}_?\; \pi \cdot X, Pr & \overset{X\mapsto\pi^{-1}\cdot t}{\Longrightarrow} & Pr[X \mapsto \pi^{-1}\cdot t] & (X \notin V(t)) \end{array}}$$

Fig. 1.4 Instantiating Rules

These rules are also called *occurs check.*

Now back to the definitions.

Definition 1.7. (Solution) A *solution* for a problem Pr is a pair (Δ, θ) such that Δ entails $Pr\theta$, where $Pr\theta$ denotes the substitution θ applied to terms in Pr.

An *unification problem Pr* is a problem as defined in Definition 1.5 but replacing α-equality constraints $s \approx_\alpha t$ by *unification constraints* $s \;{}_?{\approx}_?\; t$. Now, in order to solve nominal unification problems, it is necessary to look into the freshness contexts of each term. So we define *term-in-context* as a pair $\Delta \vdash s$ of a freshness context and a term.

Definition 1.8. (Nominal unification problem) An *unification problem (in context)* is a pair $(\nabla \vdash l) \;{}_?{\approx}_?\; (\Delta \vdash s)$. Here Δ, ∇ are freshness contexts and l, s are nominal terms. The *solution* to this unification problem, if it exists, is a pair (Δ', θ) that solves the problem $Pr = \{\Delta, \nabla, l \approx_\alpha s\}$, that is,

1. $\Delta' \vdash \Delta\theta$;

2. $\Delta' \vdash \nabla\theta$;

3. $\Delta' \vdash l\theta \approx_\alpha s\theta$.

Sometimes we will be interested in solving a "simpler" nominal unification problem in which the substitution applies to only one side.

Definition 1.9. (Nominal matching problem) A *nominal matching problem* is a pair of terms-in-context $(\nabla \vdash l) \;{}_?{\approx}\; (\Delta \vdash s)$ where $V(\nabla \vdash l) \cap V(\Delta \vdash s) = \emptyset$. A *solution* to this problem is a substitution θ such that $\Delta \vdash \nabla\theta$ and $\Delta \vdash l\theta \approx_\alpha s$ and $\mathtt{dom}(\theta) \subseteq V(\nabla \vdash l)$.

Example 1.5. The solution to the unification problem $(\emptyset \vdash [a][b]X) \;{}_?{\approx}_?\; (\emptyset \vdash [b][a]Y)$ is the pair $(\emptyset, [X \mapsto (a\ b) \cdot Y])$:

$$\begin{aligned}
(\emptyset \vdash [a][b]X) \;{}_?{\approx}_?\; (\emptyset \vdash [b][a]Y) \quad &\Longrightarrow \quad \{\emptyset, \emptyset, [b]X \;{}_?{\approx}_?\; (a\ b) \cdot ([a]Y),\ a\#[a]Y\} \\
&\Longrightarrow \quad \{\emptyset, \emptyset, [b]X \;{}_?{\approx}_?\; [b](a\ b) \cdot Y\} \\
&\Longrightarrow \quad \{\emptyset, \emptyset, X \;{}_?{\approx}_?\; (a\ b) \cdot Y\} \\
&\overset{X \mapsto (a\ b) \cdot Y}{\Longrightarrow} \quad \emptyset.
\end{aligned}$$

Example 1.6. On the other hand, a unification problem may not have a solution. See for example $(\emptyset \vdash \mathtt{lam}([a](\mathtt{lam}([b]\mathtt{app}(X,b))))) \;{}_?{\approx}_?\; (\emptyset \vdash \mathtt{lam}([b](\mathtt{lam}([a]\mathtt{app}(a,X)))))$:

$$\begin{aligned}
&\mathtt{lam}([a](\mathtt{lam}([b]\mathtt{app}(X,b)))) \;{}_?{\approx}_?\; \mathtt{lam}([b](\mathtt{lam}([a]\mathtt{app}(a,X)))) \quad \Longrightarrow^2 \\
&\qquad \Longrightarrow^2 \{\mathtt{lam}([b]\mathtt{app}(X,b)) \;{}_?{\approx}_?\; (a\ b) \cdot \big(\mathtt{lam}([a]\mathtt{app}(a,X))\big),\ a\#\mathtt{lam}([a]\mathtt{app}(a,X))\} \\
&\qquad \Longrightarrow \{\mathtt{lam}([b]\mathtt{app}(X,b)) \;{}_?{\approx}_?\; \mathtt{lam}([b]\mathtt{app}(b,(a\ b) \cdot X))\} \\
&\qquad \Longrightarrow^2 \{\mathtt{app}(X,b) \;{}_?{\approx}_?\; \mathtt{app}(b,(a\ b) \cdot X)\} \\
&\qquad \Longrightarrow \{X \;{}_?{\approx}_?\; b,\ b \;{}_?{\approx}_?\; (a\ b) \cdot X\} \\
&\qquad \overset{X \mapsto b}{\Longrightarrow} \{b \;{}_?{\approx}_?\; (a\ b) \cdot b\} = \{b \;{}_?{\approx}_?\; a\} \quad \not\Longrightarrow \qquad \text{(no solution)}.
\end{aligned}$$

Example 1.7. Let $\Sigma = \{\forall, \exists, \neg, \wedge, \vee\}$ be the signature for first-order logic. For this example we are not considering initially that disjunction ($\vee$) and conjunction ($\wedge$) are commutative operators, this property will be treat later. Consider the following unification problem

$$(a\#P \vdash (\forall[a]Q) \vee P) \;{}_?{\approx}_?\; (c\#P_0 \vdash (\forall[b]Q_0) \vee (b\ c) \cdot P_0).$$

Its solution is a pair (Δ', θ) that is a solution to the problem

$$S = \{a\#P, c\#P_0, (\forall[a]Q) \vee P \,{}_?{\approx}_?\, (\forall[b]Q_0) \vee (b\ c) \cdot P_0\}.$$

$$\begin{aligned}
&\{a\#P, c\#P_0, (\forall[a]Q) \vee P \,{}_?{\approx}_?\, (\forall[b]Q_0) \vee (b\ c) \cdot P_0\} \quad \Longrightarrow \\
&\qquad \Longrightarrow \{a\#P, c\#P_0,\ (\forall[a]Q) \,{}_?{\approx}_?\, (\forall[b]Q_0)\ ,\ P \,{}_?{\approx}_?\, (b\ c) \cdot P_0\} \\
&\qquad \overset{P \mapsto (b\ c) \cdot P_0}{\Longrightarrow} \{a\#(b\ c) \cdot P_0, c\#P_0,\ (\forall[a]Q) \,{}_?{\approx}_?\, (\forall[b]Q_0)\} \\
&\qquad \Longrightarrow \{a\#P_0, c\#P_0,\ [a]Q \,{}_?{\approx}_?\, [b]Q_0\} \\
&\qquad \Longrightarrow \{a\#P_0, c\#P_0,\ Q \,{}_?{\approx}_?\, (a\ b) \cdot Q_0,\ a\#Q_0\} \\
&\qquad \overset{Q \mapsto (a\ b) \cdot Q_0}{\Longrightarrow} \{a\#P_0, c\#P_0, a\#Q_0\}
\end{aligned}$$

Therefore the solution is $(\Delta', \theta) = (\{a\#P_0, c\#P_0, a\#Q_0\}, [P \mapsto (b\ c) \cdot P_0, Q \mapsto (a\ b) \cdot Q_0])$.

We call a context *consistent* if it is a set of consistent reduced freshness constraints.

Definition 1.10. (More general solution) Let Δ_1, Δ_2 be consistent contexts and θ_1, θ_2 substitutions. We write $(\Delta_1, \theta_1) \leq (\Delta_2, \theta_2)$ whenever there exists some θ' such that: $\forall X, \Delta_2 \vdash X\theta_1\theta' \approx_\alpha X\theta_2$ and $\Delta_2 \vdash \Delta_1\theta'$. Here the relation $\leq$ is a partial order.

A *principal solution* or *most general unifier*, $mgu(Pr)$, is a least solution according to $\leq$.

Example 1.8 (Cont. Example 1.5)**.** Notice that $(\{c\#Z\}, [X \mapsto f((a\ b) \cdot Z), Y \mapsto f(Z)])$ is also a solution for the problem $(\emptyset \vdash [a][b]X) \,{}_?{\approx}_?\, (\emptyset \vdash [b][a]Y)$, but it is not a most general one. In fact, $[X \mapsto f((a\ b) \cdot Z), Y \mapsto f(Z)] = [X \mapsto (a\ b) \cdot Y][Y \mapsto f(Z)]$. Therefore, we can write $(\emptyset, [X \mapsto (a\ b) \cdot Y]) \leq (\{c\#Z\}, [X \mapsto f((a\ b) \cdot Z), Y \mapsto f(Z)])$.

1.4 α-equivalence modulo C

In this work, we want to see what happens when we add commutativity in the equational reasoning. We start by adjusting the notation of α-equivalence in the rules in Figure 1.2.

For α-equivalence modulo commutativity, denoted by $\approx_{\alpha,\mathsf{C}}$, we have the same rules for $\approx_\alpha$ (Figure 1.2), where we replace $\approx_\alpha$ for $\approx_{\alpha,\mathsf{C}}$, and also replace the rule ($\approx_\alpha$ app) by the rules in Figure 1.5. This means that in our signature will appear some function symbols which are commutative, and denoted by $f \in \mathsf{C}$, i.e., $\Sigma = \Sigma' \cup \mathsf{C}$, where the function symbols in Σ' are uninterpreted, i.e., they do not satisfy an equational theory.

Example 1.9. Let $f \notin \mathsf{C}$ and $g \in \mathsf{C}$ be function symbols of Σ. Considering $\Delta = \{a\#X, b\#X\}$, we can deduce $\Delta \vdash g(f((a\ b) \cdot X, a), b) \approx_{\alpha,\mathsf{C}} g(b, f(X, a))$:

$$\frac{\Delta \vdash s_1 \approx_{\alpha,\mathtt{C}} t_1 \cdots \Delta \vdash s_n \approx_{\alpha,\mathtt{C}} t_n}{\Delta \vdash f(s_1,\cdots,s_n) \approx_{\alpha,\mathtt{C}} f(t_1,\cdots,t_n)}, \text{ if } f \notin \mathtt{C}\ (\approx_{\alpha,\mathtt{C}} \text{app})$$

$$\frac{\Delta \vdash s_0 \approx_{\alpha,\mathtt{C}} t_i \qquad \Delta \vdash s_1 \approx_{\alpha,\mathtt{C}} t_{(i+1)mod2}}{\Delta \vdash f(s_0,s_1) \approx_{\alpha,\mathtt{C}} f(t_0,t_1)}, i = 0,1\ (\approx_{\alpha,\mathtt{C}} \mathtt{C})$$

Fig. 1.5 Additional rules for $\approx_{\alpha,\mathtt{C}}$

$$\frac{\dfrac{\dfrac{ds((a\,b),\mathtt{Id}) = \{a, b\#X\} \in \Delta}{\Delta \vdash (a\,b)\cdot X \approx_{\alpha,\mathtt{C}} X}\,(\approx_{\alpha,\mathtt{C}} \text{var}) \qquad \dfrac{}{\Delta \vdash a \approx_{\alpha,\mathtt{C}} a}\,(\approx_{\alpha,\mathtt{C}} \text{atom})}{\Delta \vdash f((a\,b)\cdot X, a) \approx_{\alpha,\mathtt{C}} f(X,a)}\,(\approx_{\alpha,\mathtt{C}} \text{app}) \qquad \dfrac{}{\Delta \vdash b \approx_{\alpha,\mathtt{C}} b}\,(\approx_{\alpha,\mathtt{C}} \text{atom})}{\Delta \vdash g(f((a\,b)\cdot X, a), b) \approx_{\alpha,\mathtt{C}} g(b, f(X,a))}\,(\approx_{\alpha,\mathtt{C}} \mathtt{C}).$$

Notice that the application of g changes the order of the arguments, due to commutativity. Here the term on the left of the $\approx_{\alpha,\mathtt{C}}$, $g(f((a\,b)\cdot X,a),b)$, has atom b in the second argument and the term on the right, $g(b, f(X,a))$, has atom b in the first argument.

Definition 1.11. Similarly to Definition 1.5, we define a *problem* Pr to be now a set of constraints of the form $a\#X$ and $s \approx_{\alpha,\mathtt{C}} t$, and we write $\Delta \vdash Pr$ when for all $P \in Pr$ there is a derivation proof using the rules in Figures 1.1 and 1.2 (replacing $\approx_\alpha$ for $\approx_{\alpha,\mathtt{C}}$ and the rule ($\approx_\alpha$ app) by the rules in Figure 1.5), taking elements of the context Δ as assumptions.

We now extend a lemma presented in [12] to consider the commutative theory, i.e., we use the relation $\approx_{\alpha,\mathtt{C}}$. Here, $\Delta\theta$ consists of the set of constraints $\{a\#X\theta \mid a\#X \in \Delta\}$ and $\langle\Delta\theta\rangle_{nf}$ consists of the freshness context obtained after applying the rules of Figure 1.3 in $\Delta\theta$, since commutativity does not interfere with freshness constraints.

Lemma 1.4. (Compatibility of $\vdash$ by substitutions) Suppose Δ and $\Delta\theta$ are consistent.

1. If $\Delta \vdash a\#t$ then $\langle\Delta\theta\rangle_{nf} \vdash a\#(t\theta)$.

2. If $\Delta \vdash s \approx_{\alpha,\mathtt{C}} t$ then $\langle\Delta\theta\rangle_{nf} \vdash (s\theta) \approx_{\alpha,\mathtt{C}} (t\theta)$.

3. If $\Delta \vdash Pr$ then $\langle\Delta\theta\rangle_{nf} \vdash Pr\theta$.

Proof. We work by induction on the derivation of $\Delta \vdash a\#t$ or $\Delta \vdash s \approx_{\alpha,\mathtt{C}} t$.

1. We consider all rules in Figure 1.1, by analyzing the last rule applied in $\Delta \vdash a\#t$.

    - Suppose the derivation concludes with (# atom). Then we have the trivial derivation

$$\frac{}{\Delta \vdash a\#b}\ (\#\ \text{atom})$$

Notice that applying the rule (# atom) again and that $a\#b\theta \equiv a\#b$ we have

$$\frac{}{\langle \Delta\theta \rangle_{nf} \vdash a\#b}\ (\#\ \text{atom})$$

and the result follows.

- Suppose the derivation concludes with (# var), thus $\Delta \vdash a\#\pi \cdot X$, and consequently, $\pi^{-1} \cdot a\#X \in \Delta$. Therefore, $\pi^{-1} \cdot a\#X\theta \in \Delta\theta$. By Lemma 1.3 we have $\langle \Delta\theta \cup \{\pi^{-1} \cdot a\#X\theta\}\rangle_{nf} = \langle \Delta\theta\rangle_{nf} \cup \langle\{\pi^{-1} \cdot a\#X\theta\}\rangle_{nf}$. Since $\{\pi^{-1} \cdot a\#X\theta\} \subseteq \Delta\theta$, one has $\langle\{\pi^{-1} \cdot a\#X\theta\}\rangle_{nf} \subseteq \langle \Delta\theta\rangle_{nf}$. Thus, $\langle \Delta\theta\rangle_{nf} \vdash \langle\{\pi^{-1} \cdot a\#X\theta\}\rangle_{nf}$. From Lemma 1.3 $\langle \Delta\theta\rangle_{nf} \vdash \pi^{-1} \cdot a\#X\theta$.
- Suppose the derivation concludes with (# a[a]). Then we have the trivial derivation

$$\frac{}{\Delta \vdash a\#[a]t}\ (\#\ \text{a[a]})$$

Notice that applying the rule (# a[a]) again and that $[a](t\theta) \equiv ([a]t)\theta$ we have

$$\frac{}{\langle \Delta\theta \rangle_{nf} \vdash a\#[a](t\theta)}\ (\#\ \text{a[a]})$$

- Suppose the derivation concludes with (# a[b]). Then $t = [b]t'$ and there exists a derivation Π such that

$$\frac{\begin{array}{c}\Pi \\ \Delta \vdash a\#t'\end{array}}{\Delta \vdash a\#[b]t'}\ (\#\ \text{a[b]})$$

By the induction hypothesis, there exists a derivation Π' of $\langle \Delta\theta\rangle_{nf} \vdash a\#t'\theta$. Now we can apply (# a[b]) again and obtain

$$\frac{\begin{array}{c}\Pi' \\ \langle \Delta\theta\rangle_{nf} \vdash a\#t'\theta\end{array}}{\langle \Delta\theta\rangle_{nf} \vdash a\#[b](t'\theta)}\ (\#\ \text{a[b]})$$

Observing that $[b](t'\theta) \equiv ([b]t')\theta = t\theta$, the result follows.

- Suppose the derivation concludes with (# app), that is, $t = f(t_1,\ldots,t_n)$ and $\Delta \vdash a\#f(t_1,\ldots,t_n)$. Thus, there exist derivations $\Pi_1,\ldots,\Pi_n$ such that

$$\frac{\begin{array}{ccc}\Pi_1 & & \Pi_n \\ \Delta \vdash a\#t_1 & \cdots & \Delta \vdash a\#t_n\end{array}}{\Delta \vdash a\#f(t_1,\ldots,t_n)}\ (\#\ \text{app})$$

By the induction hypothesis, there exist derivations $\Pi'_1,\ldots,\Pi'_n$ for $\langle \Delta\theta\rangle_{nf} \vdash a\#(t_1\theta)$, $\cdots$, $\langle \Delta\theta\rangle_{nf} \vdash a\#(t_n\theta)$, respectively. Now we can apply the rule (# app) again and obtain

$$\dfrac{\begin{matrix}\Pi'_1 & & \Pi'_n \\ \langle\Delta\theta\rangle_{nf} \vdash a\#(t_1\theta) & \cdots & \langle\Delta\theta\rangle_{nf} \vdash a\#(t_n\theta)\end{matrix}}{\langle\Delta\theta\rangle_{nf} \vdash a\#f(t_1\theta,\ldots,t_n\theta)}\ (\#\ \mathrm{app})$$

Since $f(t_1\theta,\ldots,t_n\theta) \equiv f(t_1,\ldots,t_n)\theta$, we obtain $\langle\Delta\theta\rangle_{nf} \vdash a\#f(t_1,\ldots,t_n)\theta$, and the result follows.

2. We consider all rules in Figure 1.2 and Figure 1.5, by analyzing the last rule applied in $\Delta \vdash s \approx_{\alpha,\mathsf{C}} t$.

   - Suppose the derivation concludes with ($\approx_{\alpha,\mathsf{C}}$ atom). Then we have the trivial derivation

     $$\dfrac{}{\Delta \vdash a \approx_{\alpha,\mathsf{C}} a}\ (\approx_{\alpha,\mathsf{C}}\ \mathrm{atom})$$

     Notice that applying the rule ($\approx_{\alpha,\mathsf{C}}$ atom) again and that $a\theta \approx_{\alpha,\mathsf{C}} a\theta \equiv a \approx_{\alpha,\mathsf{C}} a$ we have

     $$\dfrac{}{\langle\Delta\theta\rangle_{nf} \vdash a \approx_{\alpha,\mathsf{C}} a}\ (\approx_{\alpha,\mathsf{C}}\ \mathrm{atom})$$

     and the result follows.

   - Suppose the derivation concludes with ($\approx_{\alpha,\mathsf{C}}$ var), thus $\Delta \vdash \pi \cdot X \approx_{\alpha,\mathsf{C}} \pi' \cdot X$, and consequently, $ds(\pi,\pi')\#X \in \Delta$. Therefore, for all $a \in ds(\pi,\pi')$, we have that $a\#X\theta \in \Delta\theta$. By Lemma 1.3 we have $\langle\Delta\theta \cup \{a\#X\theta\}\rangle_{nf} = \langle\Delta\theta\rangle_{nf} \cup \langle\{a\#X\theta\}\rangle_{nf}$. Since $\{a\#X\theta\} \subseteq \Delta\theta$, one has $\langle\{a\#X\theta\}\rangle_{nf} \subseteq \langle\Delta\theta\rangle_{nf}$. Thereby, $\langle\Delta\theta\rangle_{nf} \vdash \langle\{a\#X\theta\}\rangle_{nf}$, for all $a \in ds(\pi,\pi')$. From Lemma 1.3 $\langle\Delta\theta\rangle_{nf} \vdash a\#X\theta$, and consequently $a\#X\theta \in \langle\Delta\theta\rangle_{nf}$, for all $a \in ds(\pi,\pi')$. Applying the rule ($\approx_{\alpha,\mathsf{C}}$ var) again the result follows.

   - Suppose the derivation concludes with ($\approx_{\alpha,\mathsf{C}}$ [aa]). Then $s = [a]s'$, $t = [a]t'$ and there exists a derivation Π such that

     $$\dfrac{\begin{matrix}\Pi \\ \Delta \vdash s' \approx_{\alpha,\mathsf{C}} t'\end{matrix}}{\Delta \vdash [a]s' \approx_{\alpha,\mathsf{C}} [a]t'}\ (\approx_{\alpha,\mathsf{C}}\ [\mathrm{aa}])$$

     By the induction hypothesis, there exists a derivation Π' of $\langle\Delta\theta\rangle_{nf} \vdash s'\theta \approx_{\alpha,\mathsf{C}} t'\theta$. Now we can apply ($\approx_{\alpha,\mathsf{C}}$ [aa]) again and obtain

     $$\dfrac{\begin{matrix}\Pi' \\ \langle\Delta\theta\rangle_{nf} \vdash s'\theta \approx_{\alpha,\mathsf{C}} t'\theta\end{matrix}}{\langle\Delta\theta\rangle_{nf} \vdash [a](s'\theta) \approx_{\alpha,\mathsf{C}} [a](t'\theta)}\ (\approx_{\alpha,\mathsf{C}}\ [\mathrm{aa}])$$

Observing that $[a](s'\theta) \equiv ([a]s')\theta = s\theta$ and $[a](t'\theta) \equiv ([a]t')\theta = t\theta$, the result follows.

- Suppose the derivation concludes with ($\approx_{\alpha,\mathsf{C}}$ [ab]). So $s = [a]s'$, $t = [b]t'$ and there exist derivations Π_1 and Π_2 such that

$$\dfrac{\begin{array}{cc}\Pi_1 & \Pi_2\\ \Delta \vdash s' \approx_{\alpha,\mathsf{C}} (a\ b)\cdot t' & \Delta \vdash a\#t'\end{array}}{\Delta \vdash [a]s' \approx_{\alpha,\mathsf{C}} [b]t'}\ (\approx_{\alpha,\mathsf{C}}\ [\mathrm{ab}])$$

By the induction hypothesis, there exists a derivation Π'_1 of $\langle\Delta\theta\rangle_{nf} \vdash s'\theta \approx_{\alpha,\mathsf{C}} (a\ b)\cdot t'\theta$ and by the first part of this Lemma there exists a derivation Π'_2 of $\langle\Delta\theta\rangle_{nf} \vdash a\#t'\theta$. Now we can apply ($\approx_{\alpha,\mathsf{C}}$ [ab]) again and obtain

$$\dfrac{\begin{array}{cc}\Pi'_1 & \Pi'_2\\ \langle\Delta\theta\rangle_{nf} \vdash s'\theta \approx_{\alpha,\mathsf{C}} (a\ b)\cdot t'\theta & \langle\Delta\theta\rangle_{nf} \vdash a\#t'\theta\end{array}}{\langle\Delta\theta\rangle_{nf} \vdash [a](s'\theta) \approx_{\alpha,\mathsf{C}} [b](t'\theta)}\ (\approx_{\alpha,\mathsf{C}}\ [\mathrm{ab}])$$

Observing that $[a](s'\theta) \equiv ([a]s')\theta = s$ and $[b](t'\theta) \equiv ([b]t')\theta = t$, and using Lemma 1.1, i.e., $((a\ b)\cdot t')\theta \equiv (a\ b)\cdot(t'\theta)$, the result follows.

- Suppose the derivation concludes with ($\approx_{\alpha,\mathsf{C}}$ app). Then $s = f(s_1,\dots,s_n)$, $t = f(t_1,\dots,t_n)$ and there exist derivations Π_1, ... , Π_n of $\Delta \vdash s_1 \approx_{\alpha,\mathsf{C}} t_1$, $\cdots$, $\Delta \vdash s_n \approx_{\alpha,\mathsf{C}} t_n$, respectively, such that

$$\dfrac{\begin{array}{ccc}\Pi_1 & & \Pi_n\\ \Delta \vdash s_1 \approx_{\alpha,\mathsf{C}} t_1 & \cdots & \Delta \vdash s_n \approx_{\alpha,\mathsf{C}} t_n\end{array}}{\Delta \vdash f(s_1,\dots,s_n) \approx_{\alpha,\mathsf{C}} f(t_1,\dots,t_n)}\ , f \notin \mathsf{C}\ (\approx_{\alpha,\mathsf{C}}\ \mathrm{app})$$

By the induction hypothesis, there exist Π'_1, ... , Π'_n for $\langle\Delta\theta\rangle_{nf} \vdash s_1\theta \approx_{\alpha,\mathsf{C}} t_1\theta$, $\cdots$, $\langle\Delta\theta\rangle_{nf} \vdash s_n\theta \approx_{\alpha,\mathsf{C}} t_n\theta$, respectively. Now we can apply the rule ($\approx_{\alpha,\mathsf{C}}$ app) again and obtain

$$\dfrac{\begin{array}{ccc}\Pi'_1 & & \Pi'_n\\ \langle\Delta\theta\rangle_{nf} \vdash s_1\theta \approx_{\alpha,\mathsf{C}} t_1\theta & \cdots & \langle\Delta\theta\rangle_{nf} \vdash s_n\theta \approx_{\alpha,\mathsf{C}} t_n\theta\end{array}}{\langle\Delta\theta\rangle_{nf} \vdash f(s_1\theta,\dots,s_n\theta) \approx_{\alpha,\mathsf{C}} f(t_1\theta,\dots,t_n\theta)}\ , f \notin \mathsf{C}\ (\approx_{\alpha,\mathsf{C}}\ \mathrm{app})$$

Observing that $f(s_1\theta,\dots,s_n\theta) \equiv f(s_1,\dots,s_n)\theta = s\theta$ and also $f(t_1\theta,\dots,t_n\theta) \equiv f(t_1,\dots,t_n)\theta = t\theta$, the result follows.

- Suppose the derivation concludes with ($\approx_{\alpha,\mathsf{C}}$ C). Then $s = f(s_0,s_1)$, $t = f(t_0,t_1)$ and there exist derivations Π_1 and Π_2 of $\Delta \vdash s_0 \approx_{\alpha,\mathsf{C}} t_i$ and $\Delta \vdash s_1 \approx_{\alpha,\mathsf{C}} t_{(i+1)mod2}$, respectively, $i = 0,1$, such that

$$\dfrac{\begin{array}{cc}\Pi_1 & \Pi_2\\ \Delta \vdash s_0 \approx_{\alpha,\mathsf{C}} t_i & \Delta \vdash s_1 \approx_{\alpha,\mathsf{C}} t_{(i+1)mod2}\end{array}}{\Delta \vdash f(s_0,s_1) \approx_{\alpha,\mathsf{C}} f(t_0,t_1)}, i=0,1\ (\approx_{\alpha,\mathsf{C}}\ \mathsf{C})$$

By the induction hypothesis, there exist derivations Π'_1 and Π'_2 for $\langle\Delta\theta\rangle_{nf} \vdash s_0\theta \approx_{\alpha,\mathsf{C}} t_i\theta$ and $\langle\Delta\theta\rangle_{nf} \vdash s_1\theta \approx_{\alpha,\mathsf{C}} t_{(i+1)mod2}\theta$, respectively. Now we can apply the rule ($\approx_{\alpha,\mathsf{C}}$ C) again and obtain

$$\dfrac{\begin{array}{cc}\Pi'_1 & \Pi'_2\\ \langle\Delta\theta\rangle_{nf} \vdash s_0\theta \approx_{\alpha,\mathsf{C}} t_i\theta & \langle\Delta\theta\rangle_{nf} \vdash s_1\theta \approx_{\alpha,\mathsf{C}} t_{(i+1)mod2}\theta\end{array}}{\langle\Delta\theta\rangle_{nf} \vdash f(s_0\theta,s_1\theta) \approx_{\alpha,\mathsf{C}} f(t_0\theta,t_1\theta)}, i=0,1\ (\approx_{\alpha,\mathsf{C}}\ \mathsf{C})$$

Observing that $f(s_0\theta,s_1\theta) \equiv f(s_0,s_1)\theta = s$ and $f(t_0\theta,t_1\theta) \equiv f(t_0,t_1)\theta = t$, the result follows.

3. Since Pr is a set of freshness or α, C-equivalence constraints, by items (1) and (2), the result follows.

□

Notice that by Lemma 1.2 the relation $\Longrightarrow$ is terminating, which guarantees the existence of $\langle Pr\rangle_{nf}$.

1.4.1 Nominal Commutative Unification

A nominal unification problem has a unique most general unifier. Nonetheless, adding at least one commutative operator into the signature, say $+$, can generate infinite independent solutions, as it was shown in [1].

In this section, we will briefly present the approach proposed by Ayala-Rincón et. al. [2]: we will present the new set of inference rules that will transform a nominal C-unification problem into a triple of the form (Δ,δ,Pr), composed of a set of freshness constraints, a substitution and a family of problems consisting exclusively of fixed point equations of the form $\pi \cdot X\ {}_?{\approx}_?\ X$. This approach provides a finite representation to the complete set of C-unifiers. The related proofs are omitted, since they are out of the scope of this work. We will just present the main notions, rules, results and some examples.

Example 1.10. Consider the problem $Pr = \{\emptyset, \{[a][b]X \approx_{\alpha,\mathsf{C}} [b][a]X\}\}$. Using the standard unification algorithm given by the simplification and instantiating rules, in Definition 1.6, we get the solution $(\{a\#X, b\#X\}, \mathtt{Id})$ to the reduced problem $\{\emptyset, \{X \approx_{\alpha,\mathsf{C}} (a\ b)\cdot X\}\}$, that is the `Id` substitution is a solution whenever a and b do not occur in any instance of X. However, this is not the only solution.

In fact, a and b may occur in X and we still have a solution. Notice that $(\emptyset,[X\mapsto a+b])$ is a solution for Pr when $+\in \mathsf{C}$: $X[X\mapsto a+b]=a+b\approx_{\alpha,\mathsf{C}} b+a=((a\ b)\cdot X)[X\mapsto a+b]$. Other solutions can be found, such as $[X\mapsto (a+b)+(a+b)]$, $[X\mapsto [e](a+b)]$, ...

That said, we need to build a new algorithm for nominal C-unification. This new method will act over triples of the form (Δ,θ,Pr), where θ is a substitution. For a given unification problem (Δ,Pr), we associate the triple $(\Delta,\mathtt{Id},Pr)$. We will denote the triples by $\mathcal{P},\mathcal{Q},\mathcal{S},\cdots$.

With these new information, it is necessary to adapt the definitions and notations from the previous section.

Definition 1.12. (C-solution) A C-*solution* for a triple $\mathcal{P}=(\Delta,\delta,Pr)$ is a pair (Δ',θ) where the following conditions are satisfied:

1. $\Delta'\vdash\Delta\theta$;
2. $\Delta'\vdash a\#t\theta$, if $a\#t\in Pr$;
3. $\Delta'\vdash s\theta\approx_{\alpha,\mathsf{C}} t\theta$, if $s\approx_{\alpha,\mathsf{C}} t\in Pr$;
4. there is a substitution θ' such that $\Delta'\vdash\delta\theta'\approx_{\alpha,\mathsf{C}}\theta$.

If there is no (Δ',θ) then we say that the problem $\mathcal{P}$ is *unsolvable*. Also $\mathcal{U}_{\mathsf{C}}(\mathcal{P})$ denotes the set of all C-solutions of the triple $\mathcal{P}$.

Definition 1.13. (Nominal C-unification problem) A *nominal* C-*unification problem (in context)* is a pair $(\nabla\vdash l)\ {}_?\overset{\mathsf{C}}{\approx}_?\ (\Delta\vdash s)$. The pair (Δ',θ) is an C-solution, or C-unifier, of $(\nabla\vdash l)\ {}_?\overset{\mathsf{C}}{\approx}_?\ (\Delta\vdash s)$ iff (Δ',θ) is a C-solution of the triple $\mathcal{P}=(\{\nabla,\Delta\},\mathtt{Id},\{l\approx_{\alpha,\mathsf{C}} s\})$, that is, conditions (1)-(4) of Definition 1.12 are satisfied.

$\mathcal{U}_{\mathsf{C}}(\nabla\vdash l,\Delta\vdash s)$ denotes the set of all C-solutions of $(\nabla\vdash l)\ {}_?\overset{\mathsf{C}}{\approx}_?\ (\Delta\vdash s)$. If ∇ and Δ are empty we write $\mathcal{U}_{\mathsf{C}}(l,s)$ for the set of C-unifiers of l and s.

Definition 1.14 (More general C-solution and complete set of C-solutions)**.** Consider (Δ_1,θ_1) and (Δ_2,θ_2) solutions in $\mathcal{U}_{\mathsf{C}}(\mathcal{P})$. We say that (Δ_1,θ_1) is *more general than* (Δ_2,θ_2), written $(\Delta_1,\theta_1)\leq_{\mathsf{C}}(\Delta_2,\theta_2)$, if there exists a substitution θ' such that $\Delta_2\vdash\theta_1\theta'\approx_{\alpha,\mathsf{C}}\theta_2$ and $\Delta_2\vdash\Delta_1\theta'$.

The ordering $\leq_{\mathsf{C}}$ is the extension of $\leq$ with respect to C, and we write $\leq^V$ for the restriction of $\leq$ to a set V of variables.

A subset $\mathcal{V}\in\mathcal{U}_{\mathsf{C}}(\mathcal{P})$ is said to be a *complete set of* C-*solutions of* $\mathcal{P}$ if for all $(\Delta_1,\theta_1)\in\mathcal{U}_{\mathsf{C}}(\mathcal{P})$, there exists $(\Delta_2,\theta_2)\in\mathcal{V}$ such that $(\Delta_2,\theta_2)\leq_{\mathsf{C}}(\Delta_1,\theta_1)$.

$$
\begin{array}{lrcl}
(\#\mathrm{ab}) & (\Delta,\theta,Pr\uplus\{a\#b\}) & \Longrightarrow & (\Delta,\theta,Pr)\\
(\#\mathrm{app}) & (\Delta,\theta,Pr\uplus\{a\#f(t_1,\cdots,t_n)\}) & \Longrightarrow & (\Delta,\theta,Pr\cup\{a\#t_1,\cdots,a\#t_n\})\\
(\#\mathrm{a[a]}) & (\Delta,\theta,Pr\uplus\{a\#[a]t\}) & \Longrightarrow & (\Delta,\theta,Pr)\\
(\#\mathrm{a[b]}) & (\Delta,\theta,Pr\uplus\{a\#[b]t\}) & \Longrightarrow & (\Delta,\theta,Pr\cup\{a\#t\})\\
(\#\mathrm{var}) & (\Delta,\theta,Pr\uplus\{a\#\pi\cdot X\}) & \Longrightarrow & (\{(\pi^{-1}\cdot a)\#X\}\cup\Delta,\theta,Pr)
\end{array}
$$

Fig. 1.6 Rules for #

$$
\begin{array}{lrcl}
(\approx_{\alpha,\mathsf{C}}\ \mathrm{refl}) & (\Delta,\theta,Pr\uplus\{s\approx_{\alpha,\mathsf{C}} s\}) & \Longrightarrow & (\Delta,\theta,Pr)\\
(\approx_{\alpha,\mathsf{C}}\ \mathrm{app}) & (\Delta,\theta,Pr\uplus\{f(\overline{s})_n\approx_{\alpha,\mathsf{C}} f(\overline{t})_n\}) & \Longrightarrow & (\Delta,\theta,Pr\cup\bigcup\{s_i\approx_{\alpha,\mathsf{C}} t_i\})\\
(\approx_{\alpha,\mathsf{C}}\ C) & (\Delta,\theta,Pr\uplus\{f^{\mathsf{C}}s\approx_{\alpha,\mathsf{C}} f^{\mathsf{C}}t\}) & \Longrightarrow & (\Delta,\theta,Pr\cup\{s\approx_{\alpha,\mathsf{C}} v\}),\text{where } s=(s_0,s_1)\\
 & & & \text{and } t=(t_0,t_1), v=(t_i,t_{(i+1)mod2}), i=0,1\\
(\approx_{\alpha,\mathsf{C}}\ [\mathrm{aa}]) & (\Delta,\theta,Pr\uplus\{[a]s\approx_{\alpha,\mathsf{C}} [a]t\}) & \Longrightarrow & (\Delta,\theta,Pr\cup\{s\approx_{\alpha,\mathsf{C}} t\})\\
(\approx_{\alpha,\mathsf{C}}\ [\mathrm{ab}]) & (\Delta,\theta,Pr\uplus\{[a]s\approx_{\alpha,\mathsf{C}} [b]t\}) & \Longrightarrow & (\Delta,\theta,Pr\cup\{s\approx_{\alpha,\mathsf{C}} (a\ b)\cdot t, a\#t\})\\
(\approx_{\alpha,\mathsf{C}}\ \mathrm{inst}) & (\Delta,\theta,Pr\uplus\{\pi\cdot X\approx_{\alpha,\mathsf{C}} t\}) & \Longrightarrow & (\Delta,\theta',Pr[X\mapsto\pi^{-1}\cdot t]\cup\bigcup\limits_{\substack{Y\in dom(\theta'),\\ a\#Y\in\Delta}}\{a\#Y\theta'\}),\\
 & & & \text{let } \theta':=\theta[X\mapsto\pi^{-1}\cdot t],\\
 & & & \text{if } X\notin Var(t)\\
(\approx_{\alpha,\mathsf{C}}\ \mathrm{inv}) & (\Delta,\theta,Pr\uplus\{\pi\cdot X\approx_{\alpha,\mathsf{C}} \pi'\cdot X\}) & \Longrightarrow & (\Delta,\theta,Pr\cup\{(\pi')^{-1}\circ\pi\cdot X\approx_{\alpha,\mathsf{C}} X\})\\
 & & & \text{if } \pi'\neq \mathtt{Id}
\end{array}
$$

Fig. 1.7 Rules for $\approx_{\alpha,\mathsf{C}}$

Finally, we now present the transformation rules used in the new algorithm: see Figure 1.6 and Figure 1.7, where $\uplus$ denotes disjoint union.

Note that $(\approx_{\alpha,\mathsf{C}} C)$ is the only rule in Figure 1.7 that generates branches. Because f^{C} is a commutative operator, either $s_0\approx_{\alpha,\mathsf{C}} t_0$ and $s_1\approx_{\alpha,\mathsf{C}} t_1$ or $s_0\approx_{\alpha,\mathsf{C}} t_1$ and $s_1\approx_{\alpha,\mathsf{C}} t_0$.

To solve a nominal C-unification problem (∇, Pr) we build the derivation tree for $\Longrightarrow$ with root labelled with $(\nabla,\mathtt{Id},Pr)$, and then apply the rules in Figures 1.6 and 1.7, as long as possible. This provides an algorithm for nominal C-unification that is terminating, sound, complete and yields a finite representation for the complete set of C-unifiers for (∇, Pr). The proofs of these results are omitted and can be found in [2].

In the following examples, we will use the notation ${}_?\approx_?$ over ${}_?\overset{\mathsf{C}}{\approx}_?$, in order to avoid cluster, since commutativity is clear in these cases.

Example 1.11. Let $+$ and $\times$ be commutative operators (infix notation) of Σ. The problem $\mathcal{P} = (\emptyset, \{(a+X) \times Y \ {}_?{\approx}_?\ Z \times (W+b)\})$ has two solutions.

$$\begin{tikzcd}
 & (\emptyset, \mathtt{Id}, \{(a+X) \times Y \ {}_?{\approx}_?\ Z \times (W+b)\}) \arrow[dl, "(\approx_{\alpha,\mathsf{C}}\ \mathsf{C})"'] \arrow[dr, "(\approx_{\alpha,\mathsf{C}}\ \mathsf{C})"] & & \\
(\emptyset, \mathtt{Id}, \{(a+X) \ {}_?{\approx}_?\ Z, Y \ {}_?{\approx}_?\ (W+b)\}) \arrow[d, "(\approx_{\alpha,\mathsf{C}}\ \text{inst})(2\times)"'] & & (\emptyset, \mathtt{Id}, \{(a+X) \ {}_?{\approx}_?\ (W+b), Y \ {}_?{\approx}_?\ Z\}) \arrow[d, "(\approx_{\alpha,\mathsf{C}}\ \text{inst})"] & \\
(\emptyset, \theta = [Y \mapsto W+b, Z \mapsto a+X], \emptyset) = \mathcal{P}_1 & & (\emptyset, \theta' = [Y \mapsto Z], \{(a+X) \ {}_?{\approx}_?\ (W+b)\}) \arrow[dl, "(\approx_{\alpha,\mathsf{C}}\ \mathsf{C})"'] \arrow[dr, "(\approx_{\alpha,\mathsf{C}}\ \mathsf{C})"] & \\
 & (\emptyset, \theta', \{a \ {}_?{\approx}_?\ W, X \ {}_?{\approx}_?\ b\}) \arrow[d, "(\approx_{\alpha,\mathsf{C}}\ \text{inst})(2\times)"'] & & (\emptyset, \theta', \{a \ {}_?{\approx}_?\ b, X \ {}_?{\approx}_?\ W\}) \arrow[d] \\
 & (\emptyset, \theta'' = \theta'[W \mapsto a, X \mapsto b], \emptyset) = \mathcal{P}_2 & & \text{no solution}
\end{tikzcd}$$

Example 1.12. Consider $*$ a commutative operator (infix notation) of the signature Σ and consider the problem $\mathcal{Q} = (\emptyset, \{[c](a\,b)\cdot X * Y \;{}_?{\approx}_?\; [d](a\,c)\cdot X * Y\})$:

$$\begin{tikzcd}
& (\emptyset, \mathtt{Id}, \{[c](a\,b)\cdot X * Y \;{}_?{\approx}_?\; [d](a\,c)\cdot X * Y\}) \arrow[d, "(\approx_{\alpha,\mathsf{C}}\ \text{[ab]})"'] & \\
& (\emptyset, \mathtt{Id}, \{(a\,b)\cdot X * Y \;{}_?{\approx}_?\; (c\,d)\circ((a\,c)\cdot X * Y),\ c\#((a\,c)\cdot X * Y)) \arrow[dl, "(\approx_{\alpha,\mathsf{C}}\ \mathsf{C})"'] \arrow[dr, "(\approx_{\alpha,\mathsf{C}}\ \mathsf{C})"] & \\
\begin{array}{c}(\emptyset, \mathtt{Id}, \{(a\,b)\cdot X \;{}_?{\approx}_?\; (c\,d)\circ(a\,c)\cdot X,\\ Y \;{}_?{\approx}_?\; (c\,d)\cdot Y,\ c\#((a\,c)\cdot X * Y)\})\end{array} \arrow[d, "(\approx_{\alpha,\mathsf{C}}\ \text{inv})(2\times)"'] & & \begin{array}{c}(\emptyset, \mathtt{Id}, \{(a\,b)\cdot X \;{}_?{\approx}_?\; (c\,d)\cdot Y,\\ Y \;{}_?{\approx}_?\; (c\,d)\circ(a\,c)\cdot X,\ c\#((a\,c)\cdot X * Y)\})\end{array} \arrow[d, "(\approx_{\alpha,\mathsf{C}}\ \text{inst})"] \\
\begin{array}{c}(\emptyset, \mathtt{Id}, \{((a\,c)\circ(c\,d))\circ(a\,b)\cdot X \;{}_?{\approx}_?\; X,\\ (c\,d)^{-1}\cdot Y \;{}_?{\approx}_?\; Y,\ c\#((a\,c)\cdot X * Y)\})\end{array} \arrow[d, "(\#\ \text{app})"'] & & \begin{array}{c}(\emptyset, \theta, \{Y \;{}_?{\approx}_?\; ((c\,d)\circ(a\,c))\circ((a\,b)\circ(c\,d))\cdot Y,\\ c\#((a\,c)\circ((a\,b)\circ(c\,d))\cdot Y * Y)\})\end{array} \arrow[d, "(\approx_{\alpha,\mathsf{C}}\ \text{inv})"] \\
\begin{array}{c}(\emptyset, \mathtt{Id}, \{((a\,c)\circ(c\,d))\circ(a\,b)\cdot X \;{}_?{\approx}_?\; X,\\ (c\,d)\cdot Y \;{}_?{\approx}_?\; Y,\ c\#(a\,c)\cdot X,\ c\#Y\})\end{array} \arrow[d, "(\#\ \text{var})(2\times)"'] & & \begin{array}{c}(\emptyset, \theta, \{((c\,d)\circ(a\,b))\circ((a\,c)\circ(c\,d))\cdot Y \;{}_?{\approx}_?\; Y,\\ c\#((a\,c)\circ((a\,b)\circ(c\,d))\cdot Y * Y)\})\end{array} \arrow[d, "(\#\ \text{app})"] \\
\begin{array}{c}(\{a\#X, c\#Y\}, \mathtt{Id},\\ \{((a\,c)\circ(c\,d))\circ(a\,b)\cdot X \;{}_?{\approx}_?\; X,\\ (c\,d)\cdot Y \;{}_?{\approx}_?\; Y\}) = \mathcal{Q}_1\end{array} & & \begin{array}{c}(\emptyset, \theta, \{((c\,d)\circ(a\,b))\circ((a\,c)\circ(c\,d))\cdot Y \;{}_?{\approx}_?\; Y,\\ c\#(a\,c)\circ((a\,b)\circ(c\,d))\cdot Y,\ c\#Y\})\end{array} \arrow[d, "(\#\ \text{var})(2\times)"] \\
& & \begin{array}{c}\mathcal{Q}_2 = (\{b\#Y, c\#Y\}, \theta,\\ \{((c\,d)\circ(a\,b))\circ((a\,c)\circ(c\,d))\cdot Y \;{}_?{\approx}_?\; Y\})\end{array}
\end{tikzcd}$$

Notice that $\mathcal{Q}_1$ and $\mathcal{Q}_2$ are the two resulting fixed point problems, and the substitution in the right branch is $\theta = [X \mapsto (a\,b)\circ(c\,d)\cdot Y]$.

Chapter 2

Nominal Narrowing

Given an equational theory E, the problem of solving term equations modulo E is called E-unification. Narrowing is a well-known technique that provides a complete unification procedure for E-unification, when the theory is represented by a convergent rewrite system [10, 15].

In this chapter we will see how narrowing behaves in the nominal syntax. In order to do that, in Section 2.1 we present the basic notions for nominal rewriting and nominal equality, together with closed nominal rewriting. In Section 2.2 we present the nominal narrowing relation and its closed definition, and in Subsection 2.2.1 we present one of the most important result of nominal narrowing: the Lifting Theorem (Theorem 2.2). Lastly, in Section 2.3 we show that the process of nominal narrowing is sound and complete for nominal unification, if we consider a closed equational theory.

2.1 Nominal Rewriting and Nominal Equality

Below we recall the definition of nominal rewriting and nominal equational reasoning. Basic notions used in this section were taken from [5, 6, 12, 13].

A *rewrite judgement* (resp. *equality judgement*) is a tuple $\Delta \vdash s \to t$ (resp. $\Delta \vdash s = t$) of a freshness context Δ and two nominal terms s and t. An *equational theory* $\mathsf{E} = (\Sigma, Ax)$ is a pair of a signature Σ and a possibly infinite set of equality judgements Ax in Σ, called *axioms*. A *rewrite theory* $\mathsf{R} = (\Sigma, Rw)$, or a nominal rewrite system (NRS), is a pair of a signature Σ and a possibly infinite set of rewrite judgements Rw in Σ, called *rewrite rules*. Σ may be omitted, and we will identify E with Ax and R with Rw when the signature is clear from the context.

Definition 2.1. (Nominal rewriting) The *one-step rewrite relation* $\Delta \vdash s \overset{\mathsf{R}}{\rightarrow}_{[\mathtt{C},R,\theta,\pi]} t$ is the least relation such that for any $R = (\nabla \vdash l \rightarrow r) \in \mathsf{R}$, position $\mathtt{C}$, term s', permutation π, and substitution θ,

$$\frac{s \equiv \mathtt{C}[s'] \qquad \Delta \vdash \left(\nabla\theta, s' \approx_\alpha \pi \cdot (l\theta), \mathtt{C}[\pi \cdot (r\theta)] \approx_\alpha t\right)}{\Delta \vdash s \overset{\mathsf{R}}{\rightarrow}_{[\mathtt{C},R,\theta,\pi]} t}$$

Subindices may be omitted if they are clear from context, writing simply $\Delta \vdash s \overset{\mathsf{R}}{\rightarrow} t$, or $\Delta \vdash s \rightarrow_{\mathsf{R}} t$. The *rewrite relation* $\Delta \vdash s \rightarrow^*_{\mathsf{R}} t$ (sometimes written $\Delta \vdash_{\mathsf{R}} s \rightarrow t$) is the reflexive transitive closure of the one-step rewrite relation, that is, the least relation that includes the one-step relation and such that:

- for all Δ, s, s' we have $\Delta \vdash s \rightarrow^*_{\mathsf{R}} s'$ if $\Delta \vdash s \approx_\alpha s'$;
- for all Δ, s, t, u we have that $\Delta \vdash s \rightarrow^*_{\mathsf{R}} t$ and $\Delta \vdash t \rightarrow^*_{\mathsf{R}} u$ implies $\Delta \vdash s \rightarrow^*_{\mathsf{R}} u$.

Example 2.1. Consider the signature $\Sigma = \{\mathtt{add}, \mathtt{mult}, \mathtt{succ}, 0\}$, with arities 2, 2, 1 and 0, respectively. Let $\mathsf{R} = (\Sigma, Rw)$ be the NRS (specifying the natural numbers with addition, multiplication, successor and zero) with the following rewrite rules:

$$Rw = \begin{cases} R_1: & \emptyset \vdash \mathtt{add}(X, 0) \rightarrow X \\ R_2: & \emptyset \vdash \mathtt{add}(X, \mathtt{succ}(Y)) \rightarrow \mathtt{succ}(\mathtt{add}(X, Y)) \\ R_3: & \emptyset \vdash \mathtt{mult}(X, 0) \rightarrow 0 \\ R_4: & \emptyset \vdash \mathtt{mult}(X, \mathtt{succ}(Y)) \rightarrow \mathtt{add}(\mathtt{mult}(X, Y), X). \end{cases}$$

By definition, and taking $\theta = [X \mapsto \mathtt{add}(X', Y'), Y \mapsto \mathtt{mult}(X', Z')]$, we have

$$\begin{aligned} &\emptyset \vdash \mathtt{succ}(\mathtt{mult}(\mathtt{add}(X', Y'), \mathtt{succ}(\mathtt{mult}(X', Z')))) \\ &\quad \overset{\mathsf{R}}{\rightarrow}_{[1, R_4, \theta, \mathtt{Id}]} \mathtt{succ}(\mathtt{add}(\mathtt{mult}(\mathtt{add}(X', Y'), \mathtt{mult}(X', Z')), \mathtt{add}(X', Y'))). \end{aligned}$$

When a term-in-context $\Delta \vdash s$ does not rewrite, that is, there is no t such that $\Delta \vdash s \rightarrow_{\mathsf{R}} t$ we call it a *normal form*. Also, a rewrite theory R is convergent if the rewrite relation is terminating and confluent.

Definition 2.2. (Confluence) A NRS is *confluent* when for all Δ, s, t and t' such that $\Delta \vdash s \rightarrow^* t$ and $\Delta \vdash s \rightarrow^* t'$, there exists u such that $\Delta \vdash t \rightarrow^* u$ and $\Delta \vdash t' \rightarrow^* u$.

2.1.1 Nominal Algebra Equality and Closed Nominal Rewriting

In general, nominal rewriting is not complete for equational reasoning. As we will see below, nominal algebra includes some extra freshness context Γ, which does not match with the rewriting reasoning. The notions and results presented here are consistent with [13]. However, the proofs are out of the scope of this work and are omitted.

Definition 2.3. (Nominal equality) A *nominal algebra equality* $\Delta \vdash_{\mathsf{E}} s = t$ is the least transitive reflexive symmetric relation such that for any $(\nabla \vdash l = r) \in \mathsf{E}$, position C, permutation π, substitution θ, and fresh context Γ (so if $a\#X \in \Gamma$ then a is not mentioned in Δ, s, t),

$$\frac{\Delta, \Gamma \vdash \left(\nabla\theta, \qquad s \approx_{\alpha} \mathsf{C}[\pi \cdot (l\theta)], \qquad \mathsf{C}[\pi \cdot (r\theta)] \approx_{\alpha} t\right)}{\Delta \vdash_{\mathsf{E}} s = t}.$$

Given an equational theory E and a rewrite theory R, we say R is a *presentation* of E if:

$$\Delta \vdash s = t \in \mathsf{E} \Leftrightarrow (\nabla \vdash s \to t \in \mathsf{R} \vee \nabla \vdash t \to s \in \mathsf{R}).$$

We write $\Delta \vdash_{\mathsf{R}} s \leftrightarrow t$ for the symmetric closure of $\Delta \vdash_{\mathsf{R}} s \to t$.

Proposition 2.1 (Soundness)**.** Suppose R is a presentation of E. Then $\Delta \vdash_{\mathsf{R}} s \leftrightarrow t$ implies $\Delta \vdash_{\mathsf{E}} s = t$.

Proof. The proof can be found in [13], at Proposition 4.2. □

Lemma 2.1. Suppose R is a presentation of E. It is not necessarily the case that $\Delta \vdash_{\mathsf{E}} s = t$ implies $\Delta \vdash_{\mathsf{R}} s \leftrightarrow t$.

Proof. Take $\mathsf{R} = \{a\#X \vdash X \to f(X)\}$. Then $\vdash_{\mathsf{E}} X = f(X)$ but $\not\vdash_{\mathsf{R}} X \leftrightarrow f(X)$. □

Theorem 2.1 (Quasi-Completeness)**.** Suppose R is a presentation of E. Then $\Delta \vdash_{\mathsf{E}} s = t$ implies that there exists some fresh Γ such that $\Delta, \Gamma \vdash_{\mathsf{R}} s \leftrightarrow t$.

Proof. The proof can be found in [13], at Theorem 4.4. □

As already shown in the work [13], *closed nominal rewriting* is complete for equational reasoning with *closed axioms*. Intuitively, in closed terms there are no occurrences of free atoms and closed axioms are identities between closed terms. Also, closed axioms do not allow abstracted atoms to become free.

Definition 2.4. (Freshened variant) If t is a term, we say that $t^{\text{\tiny N}}$ is a *freshened variant* of t when $t^{\text{\tiny N}}$ has the same structure of t, except that the atoms and unknowns have been replaced by 'fresh' ones. Similarly, if ∇ is a freshness context then $\nabla^{\text{\tiny N}}$ will denote a freshened variant of ∇, that is, if $a\#X \in \nabla$ then $a^{\text{\tiny N}}\#X^{\text{\tiny N}} \in \nabla^{\text{\tiny N}}$ where $a^{\text{\tiny N}}$ and $X^{\text{\tiny N}}$ are chosen fresh.

We may extend this to other syntax, like equality and rewrite judgements.

Example 2.2. We have that $[a^{\text{И}}][b^{\text{И}}]X^{\text{И}}$ is a freshened variant of $[a][b]X$. Also $a^{\text{И}}\#X^{\text{И}}$ is a freshened variant of $a\#X$, and $\emptyset \vdash f([a^{\text{И}}]X^{\text{И}}) \to [a^{\text{И}}]X^{\text{И}}$ is a freshened variant of $\emptyset \vdash f([a]X) \to [a]X$.

Note that neither $[a^{\text{И}}][a^{\text{И}}]X^{\text{И}}$ nor $[a^{\text{И}}][b^{\text{И}}]X$ are freshened variants of $[a][b]X$: the first one because we are identifying different atoms with the same fresh name, and the second one because we did not freshened the unknown X.

Definition 2.5. (Closed term, rule and axioms) A term-in-context $\nabla \vdash l$ is *closed* if there exists a solution for the matching problem $(\nabla^{\text{И}} \vdash l^{\text{И}})\ {}_?{\approx}\ (\nabla, A(l^{\text{И}})\#V(\nabla,l) \vdash l)$. A rule $R = (\nabla \vdash l \to r)$ and an axiom $Ax = (\nabla \vdash l = r)$ are called *closed* when $\nabla \vdash (l,r)$ is closed.

Definition 2.6. (Closed nominal rewriting) The *one-step closed nominal rewriting* $\Delta \vdash s \to_R^c t$ is the least relation such that for any $R = (\nabla \vdash l \to r) \in \mathsf{R}$ and term-in-context $\Delta \vdash s$, there exists some $R^{\text{И}}$ a freshened variant of R (that is, fresh for R, Δ, s, t), a position $\mathtt{C}$, a term s', a permutation π, and a substitution θ,

$$\frac{s \equiv \mathtt{C}[s'] \qquad \Delta, A(R^{\text{И}})\#V(\Delta,s,t) \vdash \big(\nabla^{\text{И}}\theta, s' \approx_\alpha \pi\cdot(l^{\text{И}}\theta), \mathtt{C}[\pi\cdot(r^{\text{И}}\theta)] \approx_\alpha t\big)}{\Delta \vdash s \to_R^c t}$$

The *closed-rewrite relation* $\Delta \vdash_{\mathsf{R}} s \to^c t$ is the reflexive transitive closure of the one-step relation.

Example 2.3. Consider the following rewrite theory for the λ-calculus:

$$\begin{array}{rlcll}
 & \vdash \ \mathtt{app}(\mathtt{lam}([a]X),X') & \to & \mathtt{sub}([a]X,X') & (\beta)\\
 & \vdash \ \mathtt{sub}([a]a,X) & \to & X & \\
a\#Y & \vdash \ \mathtt{sub}([a]Y,X) & \to & Y & \\
 & \vdash \ \mathtt{sub}([a]\mathtt{app}(X,X'),Y) & \to & \mathtt{app}(\mathtt{sub}([a]X,Y),\mathtt{sub}([a]X',Y)) & \\
b\#Y & \vdash \ \mathtt{sub}([a]\mathtt{lam}([b]X),Y) & \to & \mathtt{lam}([b]\mathtt{sub}([a]X,Y)) &
\end{array}$$

All the rewrite rules above are closed.

Example 2.4. Consider the rule $R \equiv \emptyset \vdash [a]f(a,X) \to a$. This rule is not closed because there is no solution to $(\emptyset \vdash [a']f(a',X'))\ {}_?{\approx}\ (a'\#X' \vdash a)$.

2.2 Nominal Narrowing

Building up on the previous sections, we can present the nominal narrowing relation, i.e., the narrowing relation for nominal terms. An interesting remark to be made is that nominal narrowing is a generalization of nominal rewriting, where instead of solving a matching problem we solve a unification problem.

This extension was proposed in [6] where it was also shown that closed nominal narrowing provides a sound and complete procedure for nominal unification modulo a theory E that can be presented by a convergent nominal rewrite system.

From now on in this work we will consider R as a convergent theory, equivalent to a set of identities E, with the aim of guaranteeing the existence of a complete set of E-unifiers of equations, which we will use the narrowing method.

Definition 2.7. (Nominal narrowing) The *one-step* $(\Delta \vdash s) \rightsquigarrow_{[\mathtt{C},R,\theta,\pi]} (\Delta' \vdash t)$ *nominal narrowing relation* is the least relation such that for any $R = (\nabla \vdash l \to r) \in \mathsf{R}$, position $\mathtt{C}$, term s', permutation π, and substitution θ,

$$\frac{s \equiv \mathtt{C}[s'] \qquad \Delta' \vdash \big(\nabla\theta,\ \Delta\theta,\ s'\theta \approx_\alpha \pi \cdot (l\theta), (\mathtt{C}[\pi \cdot r])\theta \approx_\alpha t\big)}{(\Delta \vdash s) \rightsquigarrow_{[\mathtt{C},R,\theta,\pi]} (\Delta' \vdash t)} .$$

In order to find θ and π above, we need to solve the nominal unification problem $(\Delta \vdash s')\ {}_?{\approx}_?\ (\nabla \vdash l)$. We may omit subindices if they are clear from the context.

The *nominal narrowing relation* $(\Delta \vdash s) \rightsquigarrow^*_{\mathsf{R}} (\Delta' \vdash t)$ is the reflexive transitive closure of the one-step nominal narrowing relation, that is, the least relation that includes the one-step nominal narrowing relation and such that:

- for all Δ, s, s' we have $(\Delta \vdash s) \rightsquigarrow^*_{\mathsf{R}} (\Delta \vdash s')$ if $\Delta \vdash s \approx_\alpha s'$;
- for all $\Delta, \Delta', \Delta'', s, t, u$ we have that $(\Delta \vdash s) \rightsquigarrow^*_{\mathsf{R}} (\Delta' \vdash t)$ and $(\Delta' \vdash t) \rightsquigarrow^*_{\mathsf{R}} (\Delta'' \vdash u)$ implies $(\Delta \vdash s) \rightsquigarrow^*_{\mathsf{R}} (\Delta'' \vdash u)$.

Example 2.5. Consider the signature for the first-order logic $\Sigma = \{\forall, \exists, \neg, \wedge, \vee\}$ and let R be the theory over Σ consisting of the following rules:

$$\begin{array}{rl}
a\#P & \vdash P \wedge \forall[a]Q \to \forall[a](P \wedge Q) \\
a\#P & \vdash (\forall[a]Q) \wedge P \to \forall[a](Q \wedge P) \\
a\#P & \vdash P \vee \forall[a]Q \to \forall[a](P \vee Q) \\
a\#P & \vdash (\forall[a]Q) \vee P \to \forall[a](Q \vee P) \\
a\#P & \vdash P \wedge \exists[a]Q \to \exists[a](P \wedge Q) \\
a\#P & \vdash (\exists[a]Q) \wedge P \to \exists[a](Q \wedge P) \\
a\#P & \vdash P \vee \exists[a]Q \to \exists[a](P \vee Q) \\
a\#P & \vdash (\exists[a]Q) \vee P \to \exists[a](Q \vee P) \\
 & \vdash \neg(\exists[a]Q) \to \forall[a]\neg Q \\
 & \vdash \neg(\forall[a]Q) \to \exists[a]\neg Q
\end{array}$$

Let's take the rule $R = (a\#P \vdash (\forall[a]Q) \vee P \to \forall[a](Q \vee P))$ and show one nominal narrowing step from the term $s \equiv (c\#P_0 \vdash S \wedge ((\forall[b]Q_0) \vee (b\ c) \cdot P_0))$ using this rule.

- $\Delta = \{c\#P_0\}$ and $\nabla = \{a\#P\}$;
- $s = S \wedge ((\forall[b]Q_0) \vee (b\ c) \cdot P_0) \equiv \mathtt{C}[(\forall[b]Q_0) \vee (b\ c) \cdot P_0] \equiv \mathtt{C}[s']$;
- We need to find the solution to the nominal unification problem $(\Delta \vdash s')\ {}_?{\approx}_?\ (\nabla \vdash l)$. That is, a solution (Δ', θ) which is a solution to the problem
$$\{a\#P, c\#P_0, (\forall[a]Q) \vee P\ {}_?{\approx}_?\ (\forall[b]Q_0) \vee (b\ c) \cdot P_0\};$$
- A solution is $(\Delta', \theta) = (\{a\#P_0, c\#P_0, a\#Q_0\}, [P \mapsto (b\ c) \cdot P_0, Q \mapsto (a\ b) \cdot Q_0])$ (cf. Example 1.7);
- A solution to a unification problem already give us $\Delta' \vdash (\nabla\theta, \Delta\theta, s'\theta \approx_\alpha \pi \cdot (l\theta))$ if we fix $\pi = \mathtt{Id}$;
- $\mathtt{C}[\pi \cdot r] = \mathtt{C}[r] = S \wedge (\forall[a](Q \vee P))$;
- $(\mathtt{C}[\pi \cdot r])\theta = (S \wedge (\forall[a](Q \vee P)))[P \mapsto (b\ c) \cdot P_0, Q \mapsto (a\ b) \cdot Q_0] = S \wedge (\forall[a]((a\ b) \cdot Q_0 \vee (b\ c) \cdot P_0)) = t$.

Therefore, we have $(\Delta \vdash s) \rightsquigarrow_{[\mathtt{C},R,\theta,\mathtt{Id}]} (\Delta' \vdash t)$:

$$(c\#P_0 \vdash S \wedge ((\forall[b]Q_0) \vee (b\ c) \cdot P_0)) \rightsquigarrow (a\#P_0, c\#P_0, a\#Q_0 \vdash S \wedge (\forall[a]((a\ b) \cdot Q_0 \vee (b\ c) \cdot P_0))).$$

Definition 2.8. (Normalized substitution) A substitution θ is *normalized* in Δ with relation to a rewrite theory R if for every X we have that $\Delta \vdash X\theta$ is a normal formal in R.

2.2.1 The nominal Lifting Theorem

In this section we will present the nominal version of the Lifting Theorem which establishes the correspondence between nominal narrowing and nominal rewriting. This result was first presented in [6] and extends the first-order case presented by Hullot in [15]. One important difference is the use of freshness contexts both for rules and terms, since nominal terms may come with freshness conditions.

Theorem 2.2 (Lifting)**.** Let $\mathsf{R} = \{\nabla_i \vdash l_i \to r_i\}$ be a convergent rewrite theory. Let $\Delta_0 \vdash s_0$ be a nominal term-in-context and V_0 a finite set of variables containing $V = V(\Delta_0, s_0)$. Let η be

$$\begin{tikzcd}
\Delta \vdash s\eta = t_0 \arrow[r] & \dots & t_i \arrow[r] & t_{i+1} \arrow[r] & \dots \arrow[r] & t_n \\
\Delta_0 \vdash s_0 \arrow[u, dashed, two heads, "\eta_0"'] \arrow[r, rightsquigarrow] & \dots & (\Delta_i \vdash s_i) \arrow[u, dashed, two heads, "\eta_i"'] \arrow[r, rightsquigarrow] & (\Delta_{i+1} \vdash s_{i+1}) \arrow[u, dashed, two heads, "\eta_{i+1}"'] \arrow[r, rightsquigarrow] & \dots \arrow[r, rightsquigarrow] & (\Delta_n \vdash s_n) \arrow[u, dashed, two heads, "\eta_n"']
\end{tikzcd}$$

Fig. 2.1 Corresponding Rewriting and Narrowing Steps

a substitution with $\mathtt{dom}(\eta) \subseteq V_0$ and satisfying Δ_0, that is, there exists Δ such that $\Delta \vdash \Delta_0\eta$. Assume moreover that η is normalized in Δ.

Consider a rewrite derivation:

$$\Delta \vdash s_0\eta = t_0 \rightarrow_{[\mathtt{C}_0,R_0]} \cdots \rightarrow_{[\mathtt{C}_{n-1},R_{n-1}]} t_n \qquad (*)$$

There exist an associated nominal narrowing derivation:

$$(\Delta_0 \vdash s_0) \rightsquigarrow_{[\mathtt{C}'_0,R_0,\theta_0]} \cdots \rightsquigarrow_{[\mathtt{C}'_{n-1},R_{n-1},\theta_{n-1}]} (\Delta_n \vdash s_n) \qquad (**)$$

for each i, $0 \leq i \leq n$, a substitution η_i and a finite set of variables $V_i \supseteq V(s_i)$ such that:

1. $\mathtt{dom}(\eta_i) \subseteq V_i$,

2. η_i is normalized in Δ,

3. $\Delta \vdash \eta|_V \approx_\alpha \sigma_i\eta_i|_V$,

4. $\Delta \vdash s_i\eta_i \approx_\alpha t_i$,

5. $\Delta \vdash \Delta_i\eta_i$

where $\sigma_0 = Id$ and $\sigma_{i+1} = \sigma_i\theta_i$.

Conversely, to each nominal narrowing derivation of the form $(**)$ and every η such that $(\Delta_n, \sigma_n) \leq^V (\Delta, \eta)$ and $\Delta \vdash s_i\eta_i \approx_\alpha t_i$ we can associate a nominal rewriting derivation of the form $(*)$.

Proof. The proof is illustrated by Figure 2.1 and it can be found in [6], at Theorem 12. □

Example 2.6. Consider the term $s_0 = S \wedge ((\forall Q_1) \vee (b\ c) \cdot P_0)$ with the context $\Delta_0 = \{c\#P_0\}$. Using the same rules from Example 2.5, we may found the narrowing step

$$(c\#P_0 \vdash S \wedge (\forall Q_1 \vee (b\ c) \cdot P_0)) \rightsquigarrow (c\#P_0, a\#P_0 \vdash S \wedge (\forall[a](Q \vee (b\ c) \cdot P_0))),$$

where $\Delta_1 = \{c\#P_0, a\#P_0\}$ and $s_1 = S \wedge (\forall[a](Q \vee (b\ c) \cdot P_0))$ and $\theta = [P \mapsto (b\ c) \cdot P_0, Q_1 \mapsto [a]Q]$ the narrowing substitution.

$$\begin{tikzcd}
\Delta \vdash t_0 = S \wedge (\forall[a]Q \vee (b\ c) \cdot ((b\ c)^{-1} \cdot P)) \arrow[r] & t_1 = S \wedge \forall[a](Q \vee P) \\
c\#P_0 \vdash S \wedge (\forall Q_1 \vee (b\ c) \cdot P_0) \arrow[u, dashed, "\eta_0{=}[Q_1\mapsto[a]Q,\ P_0\mapsto(b\ c)^{-1}\cdot P]"'] \arrow[r, squiggly] & c\#P_0, a\#P_0 \vdash S \wedge (\forall[a](Q \vee (b\ c) \cdot P_0)) \arrow[u, dashed, "\eta_{1=[P_0\mapsto(b\ c)^{-1}\cdot P]}"']
\end{tikzcd}$$

Fig. 2.2 The “lifted” example

Let $\eta = [Q_1 \mapsto [a]Q,\ P_0 \mapsto (b\ c)^{-1} \cdot P]$ be a normalized substitution with $\mathtt{dom}(\eta) = \{Q_1, P_0\} \subseteq V_0 = \{S, P_0, Q_1\} = V(\Delta_0, s_0)$.

Observe that when $\Delta = \{c\#(b\ c) \cdot P, a\#P\}$ we have the conditions of the nominal lifting theorem:

- $(\Delta_0, \sigma_0) = (\{c\#P_0\}, \mathtt{Id}) \leq^V (\Delta, \eta)$;
- $(\Delta_1, \sigma_1) = (\{c\#P_0, a\#P_0\}, \theta) \leq^V (\Delta, \eta)$;
- $\Delta \vdash s_0\eta_0 = s_0\eta \approx_\alpha S \wedge ((\forall[a]Q) \vee (b\ c) \cdot ((b\ c)^{-1} \cdot P)) = t_0$;
- $\Delta \vdash s_1\eta_1 = S \wedge (\forall[a](Q \vee (b\ c) \cdot P_0))[P_0 \mapsto (b\ c)^{-1} \cdot P] \approx_\alpha S \wedge \forall[a](Q \vee P) = t_1$

Thus, we can associate a nominal rewriting derivation to the nominal narrowing we already have, see Figure 2.2.

2.3 Nominal Narrowing and Nominal Unification

Considering a *closed* nominal equational theory E, presented by a convergent set R of closed rules, we can state the soundness and completeness properties of the nominal narrowing procedure for nominal unification. Notice that we only have a complete procedure for nominal unification if the relation is closed.

Definition 2.9. (Closed nominal narrowing) The *one-step closed nominal narrowing* $(\Delta \vdash s) \rightsquigarrow^c_R (\Delta' \vdash t)$ is the least relation such that for any $R = (\nabla \vdash l \to r)$ and a term-in-context $\Delta \vdash s$, there exist $R^{\text{И}}$ a freshened variant of R, a position $\mathtt{C}$, a term s', a permutation π and a substitution θ,

$$\frac{s \equiv \mathtt{C}[s'] \qquad \Delta', A(R^{\text{И}})\#V(\Delta, s, t) \vdash \left(\nabla^{\text{И}}\theta, \Delta\theta, s'\theta \approx_\alpha \pi \cdot (l^{\text{И}}\theta), \mathtt{C}[\pi \cdot (r^{\text{И}}\theta)] \approx_\alpha t\right)}{(\Delta \vdash s) \rightsquigarrow^c_R (\Delta' \vdash t)}$$

The *closed narrowing relation* $(\Delta \vdash s) \rightsquigarrow^{c}_{\mathsf{R}} \cdots \rightsquigarrow^{c}_{\mathsf{R}} (\Delta' \vdash t)$ is the reflexive transitive closure of the one-step closed narrowing.

Remark 2.1. A "closed lifting" theorem can be stated by replacing nominal rewriting (narrowing) for closed rewriting (narrowing). The proof is similar.

In order to find a solution for the unification problem $(\Delta \vdash s) \;{}_{?}\overset{\mathsf{E}}{\approx}_{?}\; (\nabla \vdash t)$, the approach used in [6] was to apply closed narrowing on $\Delta \vdash s$ and $\nabla \vdash t$ in parallel, that is, they narrow a single term $u = (s,t)$ under Δ, ∇.

Lemma 2.2. (Soundness) Let $\Delta \vdash s$ and $\nabla \vdash t$ be two nominal terms-in-context and

$$\Delta, \nabla \vdash (s,t) = u_0 \rightsquigarrow^{c} \cdots \rightsquigarrow^{c} \Delta_n \vdash u_n = (s_n, t_n)$$

a closed narrowing derivation such that $\Delta_n, s_n \approx_\alpha t_n$ has a solution, say (Γ, θ).

Then $(\Gamma, \rho\theta)$ is an E-solution of the problem $\{\Delta, \nabla, s \;{}_{?}\overset{\mathsf{E}}{\approx}_{?}\; t\}$, where ρ is the composition of substitutions along the narrowing derivation.

Proof. The proof can be found in [6], at Lemma 14. □

Below, $\leq_{\mathsf{E}}$ is the restriction of $\leq$ with respect to an equational theory E.

Lemma 2.3. (Completeness) Let $\Delta \vdash s$ and $\nabla \vdash t$ be two nominal terms-in-context, such that the problem $(\Delta \vdash s) \;{}_{?}\overset{\mathsf{E}}{\approx}_{?}\; (\nabla \vdash t)$ has an E-solution, (Δ', ρ), and let V be a finite set of variables containing $V(\Delta, \nabla, s, t)$.

Then there exists a closed narrowing derivation:

$$\nabla, \Delta \vdash u = (s,t) \rightsquigarrow^{c} \cdots \rightsquigarrow^{c} \Gamma_n \vdash (s_n, t_n),$$

such that $\Gamma_n, s_n \approx_\alpha t_n$ has a solution. Let $(\Gamma, \mu) = mgu(\Gamma_n, s_n \approx_\alpha t_n)$, and θ_n the composition of narrowing substitutions. Then, $(\Gamma, \theta_n\mu) \leq^{V}_{\mathsf{E}} (\Delta', \rho)$.

Moreover, we are allowed to restrict our attention to $\rightsquigarrow^{c}$-derivations such that: $\forall i, 0 \leq i \leq n$, $\theta_i|_V$ is normalized.

Proof. The proof can be found in [6], at Lemma 15. □

Remember that our main goal in this dissertation is not to re-prove these results above, but to extend them to the case when we are dealing with a commutative theory, as we will see in the next chapter.

Chapter 3

Nominal Commutative Narrowing

In this chapter we present the main contributions of this work. We start by extending the nominal rewriting relation to take into account an equational theory E that cannot be oriented as a terminating rule. We recall some results in first-order equational rewriting to motivate further developments in nominal rewriting modulo commutativity. After that, we present our definition of nominal C-narrowing and prove the results that relate nominal C-narrowing to nominal rewriting modulo C: the proof of one direction of this correspondence is presented fully in Theorem 3.1, the other direction though, requires an extra property called C-coherence (Definition 3.6), and we present a naive version of the result in Theorem 3.2, but this has to be further investigated. We make comparisons between the nominal versions of these results and also their first-order counterpart, as presented by Jouannaud et. al. in [17].

We illustrate our results and new developments with new examples.

3.1 Basic Definitions

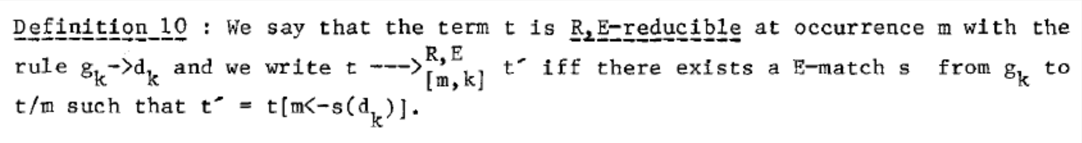
Consider a signature $\Sigma = \Sigma_{\mathsf{E}} \cup \Sigma_0$, where Σ_{E} consists of the function symbols satisfying an equational theory E and Σ_0 is a set of uninterpreted function symbols, and assuming we have a theory $\mathsf{T} = (\Sigma_{\mathsf{E}} \cup \Sigma_0, Ax)$ where Ax is a set of axioms that can be split into a set R of rules and a set E of equations, we obtain what we call an *equational term rewriting system* (ETRS), denoted $\mathsf{R}{\cup}\mathsf{E}$ (for more details see Appendix A). We will extend basic rewriting notions such as C-confluence and C-termination (following [17]) to the nominal framework, and the notations are consistent with [12, 6]. Thus, we will present novel definitions such as Equational Nominal Rewriting Systems (ENRS) and other expected related notions, such as rewriting modulo C following the definitions and notations from [17]. Formally, We start by extending the definition of Equational Term Rewriting Systems (ETRS) to the nominal terms with respect to the theory C. Below, $[t]_{\approx_{\mathsf{C}}}$, denotes the equivalence class of the nominal term t modulo C, i.e., $[t]_{\approx_{\mathsf{C}}} = \{t' \mid t' \approx_{\alpha,\mathsf{C}} t\}$.

Definition 3.1. (Equational nominal rewrite system) Let C be set of identities for commutativity and R a set of nominal rewrite rules. A nominal term-in-context $\Delta \vdash s$, reduces with respect to R/C, when its equivalence class modulo C reduces via $\to_{\mathsf{R/C}}$ as below.

$$\Delta \vdash ([s]_{\approx_{\mathsf{C}}} \to_{\mathsf{R/C}} [t]_{\approx_{\mathsf{C}}}) \text{ iff there exist } s', t' \text{ such that } \Delta \vdash (s \approx_{\alpha,\mathsf{C}} s' \to_{\mathsf{R}} t' \approx_{\alpha,\mathsf{C}} t).$$

That said, we call R/E an *equational nominal rewrite system* (ENRS). In particular, R/C is a commutative nominal rewrite system.

Remark 3.1. For a general theory E, in the first-order case, one has to consider that E-congruence classes may be infinite, and then the first-order counterpart of $\to_{\mathsf{R/E}}$ may not be decidable. To address this problem, a relation $\to_{\mathsf{R,E}}$ which deals with E-matching:



Definition 10 : We say that the term t is R,E-reducible at occurrence m with the rule g_k->d_k and we write $t \xrightarrow[{[m,k]}]{R,E} t'$ iff there exists a E-match s from g_k to t/m such that t′ = t[m<-s(d_k)].

Notice that the R, E-reducibility is decidable if the E-matching is decidable. And Jouannaud et. al. [17] assume the existence of a finite and complete E-unification algorithm, which is a sufficient condition for that decidability.

Here however, one needs to remember that we are dealing with α, E-congruence classes and they are always infinite due to availability of names for α-renaming. But the pure $\approx_\alpha$ relation is decidable (just use the rules in Figure 1.1 and Figure 1.2), but if put together with an equational theory E with infinite congruence classes, the same problem of indecidability of the nominal $\to_{\mathsf{R/E}}$ is inherited. Thus, we will follow a similar approach and define a new relation $\to_{\mathsf{R,E}}$ that deals with nominal E-matching instead of inspecting the whole α, E-congruence class of a term.

In this work, we are interested on the commutative equational theory, therefore, the next notions and results will be restricted to C. It is a fact that C-congruence classes are finite, given a term t with commutative function symbols, there exists only finite t' such that $t \approx_{\mathsf{C}} t'$. Also, when we generalize the relation $\approx_{\mathsf{C}}$ to $\approx_{\alpha,\mathsf{C}}$ the α, C-congruence classes become infinite, but the relation is still decidable [4]. Next, we will introduce the relation $\to_{\mathsf{R},\mathsf{C}}$, and use nominal C-matching [3] to follow existing strategies.

Definition 3.2. (Nominal rewriting modulo C) The *one-step rewrite modulo* C *relation* $\Delta \vdash s \to_{\mathsf{R},\mathsf{C}} t$ is the least relation such that for any $R = (\nabla \vdash l \to r) \in \mathsf{R}$, position $\mathtt{C}$, term s', permutation π, and substitution θ,

$$\frac{s \equiv \mathtt{C}[s'] \qquad \Delta \vdash \left(\nabla\theta,\ s' \approx_{\alpha,\mathsf{C}} \pi\cdot(l\theta),\ \mathtt{C}[\pi\cdot(r\theta)] \approx_{\alpha,\mathsf{C}} t\right)}{\Delta \vdash s \to_{\mathsf{R},\mathsf{C}}\ t}$$

The *rewrite modulo* C *relation* $\Delta \vdash s \to^{*}_{\mathsf{R},\mathsf{C}} t$ is the reflexive transitive closure of the one-step rewrite modulo C relation, that is, the least relation that includes the one-step rewrite modulo C relation and such that:

- for all Δ, s, s' we have $\Delta \vdash s \to^{*}_{\mathsf{R},\mathsf{C}} s'$ if $\Delta \vdash s \approx_{\alpha,\mathsf{C}} s'$;
- for all Δ, s, t, u we have that $\Delta \vdash s \to^{*}_{\mathsf{R},\mathsf{C}} t$ and $\Delta \vdash t \to^{*}_{\mathsf{R},\mathsf{C}} u$ implies $\Delta \vdash s \to^{*}_{\mathsf{R},\mathsf{C}} u$.

Example 3.1. Notice that the rules in Example 2.5 have two copies for dealing with the commutativity of disjunction and conjunction. If we work now with the set of identities $\mathsf{C} = \{\vdash P \vee Q \approx Q \vee P, \vdash P \wedge Q \approx Q \wedge P\}$, we can write the NRS as

$$\begin{array}{rl}
a\#P & \vdash\ P \wedge \forall[a]Q \to \forall[a](P \wedge Q) \\
a\#P & \vdash\ P \vee \forall[a]Q \to \forall[a](P \vee Q) \\
a\#P & \vdash\ P \wedge \exists[a]Q \to \exists[a](P \wedge Q) \\
a\#P & \vdash\ P \vee \exists[a]Q \to \exists[a](P \vee Q) \\
 & \vdash\ \neg(\exists[a]Q) \to \forall[a]\neg Q \\
 & \vdash\ \neg(\forall[a]Q) \to \exists[a]\neg Q
\end{array}$$

We will show that we have the one-step rewrite modulo C: $a\#P' \vdash S' \vee (P' \vee \exists[a]Q') \to_{\mathsf{R},\mathsf{C}} S' \vee (\exists[a](Q' \vee P'))$ with the rule $a\#P \vdash P \vee \exists[a]Q \to \exists[a](P \vee Q)$. Indeed,

- $\Delta = \{a\#P'\}$ and $\nabla = \{a\#P\}$;
- $s = S' \vee (P' \vee \exists[a]Q') \equiv \mathtt{C}[P' \vee \exists[a]Q'] \equiv \mathtt{C}[s']$;

If we fix $\pi = \mathtt{Id}$ and $\theta = [P \mapsto P', Q \mapsto Q']$ we have:

- $\Delta = a\#P' \vdash a\#P' = (a\#P)[P \mapsto P', Q \mapsto Q'] = \nabla\theta$;
- $s' = P' \vee \exists[a]Q' \approx_{\alpha,\mathsf{C}} (P \vee \exists[a]Q)[P \mapsto P', Q \mapsto Q'] = l\theta = \pi \cdot (l\theta)$;

- $\mathtt{C}[\pi \cdot (r\theta)] = \mathtt{C}[r\theta] = \mathtt{C}[(\exists[a](P \vee Q))[P \mapsto P', Q \mapsto Q']] = \mathtt{C}[\exists[a](P' \vee Q')] = S' \vee (\exists[a](P' \vee Q')) \approx_{\alpha,\mathsf{C}} t$

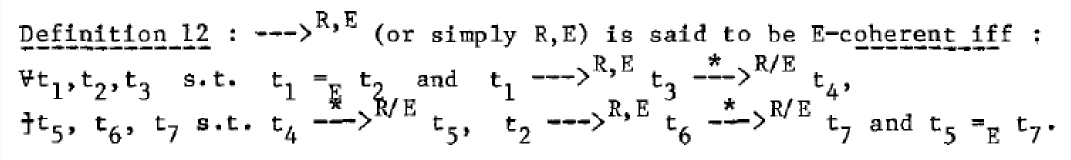
Thus, $a\#P' \vdash S' \vee (P' \vee \exists[a]Q') \to_{\mathsf{R},\mathsf{C}} S' \vee (\exists[a](Q' \vee P'))$.

Observe that since $\vee$ is a commutative symbol, we could reduce the initial term to other three possible terms, because we have two occurrences of the disjunction and thus we can "permute" the subterms inside the rewriting modulo C.

Remark 3.2. It is important to notice that nominal C-unification is not finitary when one uses freshness constraints and substitutions for representing solutions [2], but what causes infinite set of C-unifiers does not appear in nominal C-matching [3]. We will come back to this issue later on.

Definition 3.3. (C-confluence and C-termination) If $\Delta \vdash s \to^*_{\mathsf{R},\mathsf{C}} t$ and $\Delta \vdash s \to^*_{\mathsf{R},\mathsf{C}} t'$, then we say a nominal rewrite system R is C-*confluent* when there exists a term u such that $\Delta \vdash t \to^*_{\mathsf{R},\mathsf{C}} u$ and $\Delta \vdash t' \to^*_{\mathsf{R},\mathsf{C}} u$. Also, R is said to be C-*terminating* if there is no infinite rewrite modulo C sequence. A NRS R is called C-*convergent* if it is C-confluent and C-terminating.

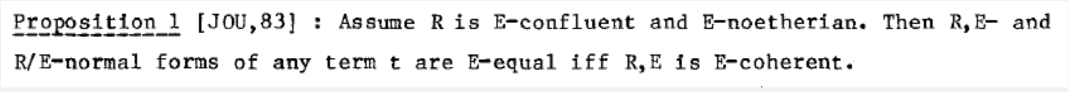
Remark 3.3. Following the approach by Jouannaud et. al. [17], E-confluence is a consequence of relating $\to_{\mathsf{R/E}}$ and $\to_{\mathsf{R,E}}$, which relies on a property called E-*coherence*:



Definition 12 : $\dashrightarrow^{R,E}$ (or simply R,E) is said to be E-coherent iff :
$\forall t_1, t_2, t_3$ s.t. $t_1 =_E t_2$ and $t_1 \dashrightarrow^{R,E} t_3 \overset{*}{\dashrightarrow}{}^{R/E} t_4$,
$\exists t_5, t_6, t_7$ s.t. $t_4 \overset{*}{\dashrightarrow}{}^{R/E} t_5$, $t_2 \dashrightarrow^{R,E} t_6 \overset{*}{\dashrightarrow}{}^{R/E} t_7$ and $t_5 =_E t_7$.

$$\begin{tikzcd}
t_1 \arrow[r, "\mathsf{R,E}"'] \arrow[d, squiggly, "\approx_{\mathsf{E}}"] & t_3 \arrow[r, "*", "\mathsf{R/E}"'] & t_4 \arrow[r, dashed, "*", "\mathsf{R/E}"'] & t_5 \arrow[d, squiggly, "\approx_{\mathsf{E}}"] \\
t_2 \arrow[rr, dashed, "\mathsf{R,E}"'] & & t_6 \arrow[r, dashed, "*", "\mathsf{R/E}"'] & t_7
\end{tikzcd}$$

We would like to stress the abuse of notation above, since $t_3 \to^*_{\mathsf{R/E}} t_4$, for instance, should be written $[t_3]_{\approx_{\mathsf{E}}} \to^*_{\mathsf{R/E}} [t_4]_{\approx_{\mathsf{E}}}$, and is acting as an abbreviation for $t_3 \approx_{\mathsf{E}} t_3' \to_{\mathsf{R}} t_4' \approx_{\mathsf{E}} t_4$.



Proposition 1 [JOU,83] : Assume R is E-confluent and E-noetherian. Then R,E- and R/E-normal forms of any term t are E-equal iff R,E is E-coherent.

For more details, see Jouannaud et. al. [17].

Definition 3.4. (Normalized substitution w.r.t $\to_{\mathsf{R},\mathsf{C}}$) A substitution θ is *normalized in* Δ *with relation to* $\to_{\mathsf{R},\mathsf{C}}$ if $\Delta \vdash X\theta$ is a R,C-normal form in R for every X. A substitution θ satisfies the freshness context Δ if there exists a freshness context ∇ such that $\nabla \vdash a\#X\theta$ for each $a\#X \in \Delta$. The minimal such ∇ is $\langle \Delta\theta \rangle_{nf}$.

Now we define the nominal narrowing relation modulo C, extending previous works, and illustrate it with two examples.

Definition 3.5. (Nominal narrowing modulo C) The *one-step narrowing modulo* C *relation* $(\Delta \vdash s) \rightsquigarrow_{\mathsf{R},\mathsf{C}} (\Delta' \vdash t)$ is the least relation such that for any $R = (\nabla \vdash l \to r) \in \mathsf{R}$, position $\mathtt{C}$, term s', permutation π, and substitution θ,

$$\frac{s \equiv \mathtt{C}[s'] \qquad \Delta' \vdash \big(\nabla\theta,\ \Delta\theta,\ s'\theta \approx_{\alpha,\mathsf{C}} \pi \cdot (l\theta),\ (\mathtt{C}[\pi \cdot r])\theta \approx_{\alpha,\mathsf{C}} t\big)}{(\Delta \vdash s) \rightsquigarrow_{\mathsf{R},\mathsf{C}} (\Delta' \vdash t)}.$$

Notice that the permutation and substitution above are found by solving the C-unification problem $(\nabla \vdash l)\ {}_?\overset{\mathsf{C}}{\approx}_?\ (\Delta \vdash s')$.

The *nominal narrowing modulo* C *relation* $(\Delta \vdash s) \rightsquigarrow^*_{\mathsf{R},\mathsf{C}} (\Delta' \vdash t)$ is the reflexive transitive closure of the one-step nominal narrowing modulo C relation, that is, the least relation that includes the one-step nominal narrowing modulo C relation and such that:

- for all Δ, s, s' we have $(\Delta \vdash s) \rightsquigarrow^*_{\mathsf{R},\mathsf{C}} (\Delta \vdash s')$ if $\Delta \vdash s \approx_{\alpha,\mathsf{C}} s'$;
- for all $\Delta, \Delta', \Delta'', s, t, u$ we have that $(\Delta \vdash s) \rightsquigarrow^*_{\mathsf{R},\mathsf{C}} (\Delta' \vdash t)$ and $(\Delta' \vdash t) \rightsquigarrow^*_{\mathsf{R},\mathsf{C}} (\Delta'' \vdash u)$ implies $(\Delta \vdash s) \rightsquigarrow^*_{\mathsf{R},\mathsf{C}} (\Delta'' \vdash u)$.

We now return to Remark 3.2: nominal C-narrowing is defined on nominal C-unification, which is not finitary when we use pairs (Δ', θ) of freshness contexts and substitutions for representing solutions. This implies that our nominal C-narrowing trees are infinitely branching.

Example 3.2. In Example 2.5 we made a narrowing step:

$$(c\#P_0 \vdash S \wedge ((\forall[b]Q_0) \vee (b\ c) \cdot P_0)) \rightsquigarrow_{\mathsf{R}} (a\#P_0, c\#P_0, a\#Q_0 \vdash S \wedge (\forall[a]((a\ b) \cdot Q_0 \vee (b\ c) \cdot P_0))).$$

Now considering the commutativity of $\vee$, we have

$$(\mathtt{C}[\pi \cdot r])\theta = S \wedge (\forall[a]((a\ b) \cdot Q_0 \vee (b\ c) \cdot P_0)) \approx_{\alpha,\mathsf{C}} S \wedge (\forall[a]((b\ c) \cdot P_0) \vee (a\ b) \cdot Q_0).$$

In that way we can make now a narrowing modulo C step:

$$(c\#P_0 \vdash S \wedge ((\forall[b]Q_0) \vee (b\ c) \cdot P_0)) \rightsquigarrow_{\mathsf{R},\mathsf{C}} (a\#P_0, c\#P_0, a\#Q_0 \vdash S \wedge (\forall[a]((b\ c) \cdot P_0) \vee (a\ b) \cdot Q_0)).$$

3.2 Nominal Lifting Theorem modulo C

Through this section we assume R to be a nominal rewriting system that is C-convergent. Similarly to Section 2.2.1 we want to establish correspondence between nominal C-narrowing and nominal C-rewriting. We will do that via an extension of the Nominal Lifting Theorem (cf. Theorem 2.2) for the extended relations $\rightsquigarrow_{\mathsf{R},\mathsf{C}}$ and $\rightarrow_{\mathsf{R},\mathsf{C}}$.

From nominal C-narrowing to C-rewriting

We start proving a correctness result: one step of narrowing $\rightsquigarrow_{\mathsf{R},\mathsf{C}}$ maps to one step of $\rightarrow_{\mathsf{R},\mathsf{C}}$.

Lemma 3.1. ($\rightsquigarrow_{\mathsf{R},\mathsf{C}}$ to $\rightarrow_{\mathsf{R},\mathsf{C}}$) Let $(\Delta_0 \vdash s_0) \rightsquigarrow_{\mathsf{R},\mathsf{C}} (\Delta_1 \vdash s_1)$. Then, for any substitution ρ that satisfies Δ_0, i.e., there exists Δ such that $\Delta \vdash \Delta_0\rho$, the following holds

$$\Delta \vdash (s_0\theta)\rho \rightarrow_{\mathsf{R},\mathsf{C}} s_1\rho$$

where θ is the substitution applied in the narrowing step. In particular, Δ will be $\langle \Delta_0\rho \rangle_{nf}$.

Proof. First, we start by extending the remark made by [17]: $s_0 \rightsquigarrow^{\theta}_{\mathsf{R},\mathsf{E}} s_1$ implies $s_0\theta \rightarrow_{\mathsf{R},\mathsf{E}} s_1$ to the nominal syntax and the commutative case.

- $(\Delta_0 \vdash s_0) \rightsquigarrow^{\theta}_{\mathsf{R},\mathsf{C}} (\Delta_1 \vdash s_1)$ implies $\Delta_0 \vdash (s_0\theta) \rightarrow_{\mathsf{R},\mathsf{C}} s_1$.

  Indeed, suppose we have $(\Delta_0 \vdash s_0) \rightsquigarrow^{\theta}_{\mathsf{R},\mathsf{C}} (\Delta_1 \vdash s_1)$. The narrowing step guarantees that for any substitution θ

  - $s_0 \equiv \mathtt{C}[s_0']$;
  - $\Delta_1 \vdash \big(\nabla\theta,\ \Delta_0\theta,\ s_0'\theta \approx_{\alpha,\mathsf{C}} \pi\cdot(l\theta),\ (\mathtt{C}[\pi\cdot r])\theta \approx_{\alpha,\mathsf{C}} s_1\big)$.

  By the definition of rewrite modulo C, if $s_0\theta \equiv \mathtt{C}\theta[s_0'\theta]$, and if $\Delta_0 \vdash (\nabla\theta', s_0'\theta \approx_{\alpha,\mathsf{C}} \pi\cdot(l\theta'), \mathtt{C}\theta[\pi\cdot(r\theta')] \approx_{\alpha,\mathsf{C}} s_1)$, then $\Delta_0 \vdash s_0\theta \rightarrow_{\mathsf{R},\mathsf{C}} s_1$. We just need to fix the substitution θ used in the narrowing step as θ', and the result follows.

Since $\Delta_0 \vdash s_0\theta \rightarrow_{\mathsf{R},\mathsf{C}} s_1$, by Lemma 1.4 we have

- $(s_0\theta)\rho \equiv \mathtt{C}\theta[s_0'\theta]\rho = \mathtt{C}\theta[(s_0'\theta)\rho]$
- $\Delta_0 \vdash \nabla\theta$ implies $\langle \Delta_0\rho \rangle_{nf} \vdash \nabla\theta\rho$
- $\Delta_0 \vdash s_0'\theta \approx_{\alpha,\mathsf{C}} \pi\cdot(l\theta)$ implies $\langle \Delta_0\rho \rangle_{nf} \vdash s_0'\theta\rho \approx_{\alpha,\mathsf{C}} (\pi\cdot(l\theta))\rho = \pi\cdot(l\theta\rho)$
- $\Delta_0 \vdash \mathtt{C}\theta[\pi\cdot(r\theta)] \approx_{\alpha,\mathsf{C}} s_1'$ implies $\langle \Delta_0\rho \rangle_{nf} \vdash \mathtt{C}\theta[\pi\cdot(r\theta\rho)] = (\mathtt{C}\theta[\pi\cdot(r\theta)]\rho) \approx_{\alpha,\mathsf{C}} s_1\rho$

which implies that $\langle \Delta_0 \rho \rangle_{nf} \vdash (s_0\theta)\rho \rightarrow_{\mathsf{R,C}} s_1\rho$.

□

Now we can prove that this correctness result is preserved for finite sequences of narrowing steps.

Theorem 3.1. ($\rightsquigarrow^*_{\mathsf{R,C}}$ to $\rightarrow^*_{\mathsf{R,C}}$) Let $(\Delta_0 \vdash s_0) \rightsquigarrow^*_{\mathsf{R,C}} (\Delta_n \vdash s_n)$ a nominal narrowing derivation. Let ρ be a substitution satisfying Δ_0, i.e., there exists Δ such that $\Delta \vdash \Delta_0\rho$.

Then, there exists a rewriting derivation

$$\Delta \vdash s_0\rho_0 \rightarrow^*_{\mathsf{R,C}} s_n\rho$$

such that $\Delta \vdash \Delta_i\rho_{i+1}$ and $\rho_i = \theta_i \dots \theta_{n-1}\rho$, for all $0 \le i < n$. In other words,

$$\Delta \vdash (s_0\theta)\rho \rightarrow^*_{\mathsf{R,C}} s_n\rho$$

where θ is the composition of the successive $\mathsf{R,C}$-narrowing substitutions.

Proof. By induction on the length n of the narrowing derivation $(\Delta_0 \vdash s_0) \rightsquigarrow^{n-1}_{\mathsf{R,C}} (\Delta_n \vdash s_n)$, using the previous Lemma.

- **Base Case:** For $n = 1$, we have by assumption that $\rho_0 = \theta_0\rho$ and $\Delta \vdash \Delta_0\rho_0$. The result follows directly from Lemma 3.1:

$$\Delta \vdash s_0\rho_0 = (s_0\theta_0)\rho \rightarrow_{\mathsf{R,C}} s_1\rho.$$

- **Induction Step:** Assume that the result holds for i, that is $(\Delta_0 \vdash s_0) \rightsquigarrow^{i+1}_{\mathsf{R,C}} (\Delta_i \vdash s_i)$ implies that there exists a rewriting derivation $\Delta \vdash s_0\rho_0 \rightarrow^{i+1}_{\mathsf{R,C}} s_i\rho_i$. Figure 3.1 illustrates this setting.

  We want to show that $\Delta \vdash s_0\rho_0 \rightarrow^{i+2}_{\mathsf{R,C}} s_{i+1}\rho_{i+1}$, in other words,

$$\Delta \vdash s_0\theta_0 \dots \theta_{i-1}\theta_i\rho_{i+1} \rightarrow^{i+2}_{\mathsf{R,C}} s_{i+1}\rho_{i+1}.$$

  Consider the narrowing step $(\Delta_i \vdash s_i) \rightsquigarrow^{\theta_i} (\Delta_{i+1} \vdash s_{i+1})$. By the Lemma 3.1, for any substitution, let's name it ρ_{i+1}, that satisfies Δ_i there exists Δ that satisfies $\Delta_i\rho_{i+1}$, and we have

$$\Delta \vdash (s_i\theta_i)\rho_{i+1} \rightarrow_{\mathsf{R,C}} s_{i+1}\rho_{i+1}.$$

  By the induction hypothesis, we already have

$$\Delta \vdash s_0\theta_0 \dots \theta_{i-1}\rho_i \rightarrow^{i+1}_{\mathsf{R,C}} s_i\rho_i$$

$$\begin{tikzcd}
(\Delta_0 \vdash s_0) \arrow[r, rightsquigarrow, "\theta_0"] \arrow[d, dashed, "\rho_0"] & (\Delta_1 \vdash s_1) \arrow[r, rightsquigarrow, "*"] \arrow[d, dashed, "\rho_1"] & (\Delta_i \vdash s_i) \arrow[r, rightsquigarrow, "\theta_i"] \arrow[d, dashed, "\rho_i"] & (\Delta_{i+1} \vdash s_{i+1}) \arrow[r, rightsquigarrow, "*"] \arrow[d, dashed, "\rho_{i+1}"] & \ldots \arrow[r, rightsquigarrow, "\theta_n"] & (\Delta_n \vdash s_n) \arrow[d, dashed, "\rho_n"] \\
\Delta \vdash s_0\rho_0 \arrow[r] & s_1\rho_1 \arrow[r, "*"] & s_i\rho_i \arrow[r] & s_{i+1}\rho_{i+1} \arrow[r, "*"] & \ldots \arrow[r] & s_n\rho_n
\end{tikzcd}$$

Fig. 3.1 Corresponding Narrowing to Rewriting Derivations

and if we fix $\rho_i = \theta_i\rho_{i+1}$, we get:

$$\Delta \vdash s_0\theta_0 \ldots \theta_{i-1}\theta_i\rho_{i+1} \rightarrow^{i+1}_{\mathsf{R,C}} s_i\theta_i\rho_{i+1} \rightarrow_{\mathsf{R,C}} s_{i+1}\rho_{i+1},$$

and the theorem is proved.

Notice that ρ_i is $\mathsf{R,C}$-normalized if ρ_0 is $\mathsf{R,C}$-normalized. □

With Lemma 3.1 and Theorem 3.1 we have extended the converse part of the Nominal Lifting Theorem (Section 2.2.1, Theorem 2.2), from the previous chapter, to the nominal rewriting/narrowing modulo C.

From nominal C-rewriting to C-narrowing

As expected, the completeness proof is based on the converse construction of the correctness proof, i.e., we want to prove that to each finite sequence of $\rightarrow_{\mathsf{R,C}}$ steps corresponds a sequence of $\rightsquigarrow_{\mathsf{R,C}}$ steps.

Lemma 3.2. ($\rightarrow_{\mathsf{R,C}}$ to $\rightsquigarrow_{\mathsf{R,C}}$) Let $\mathsf{R} = \{\nabla_i \vdash l_i \rightarrow r_i\}$ be a C-convergent NRS. Let $\Delta_0 \vdash s_0$ be a nominal term in context and V_0 a finite set of variables containing $V = V(\Delta_0, s_0)$. Let ρ_0 be a $\mathsf{R,C}$-normalized substitution with $\mathtt{dom}(\rho_0) \subseteq V$ that satisfies Δ_0 with Δ and

$$\Delta \vdash s_0\rho_0 = t_0 \rightarrow_{\mathsf{R,C}} t_1.$$

Then, there exist a nominal commutative narrowing step

$$(\Delta_0 \vdash s_0) \rightsquigarrow_{\mathsf{R,C}} (\Delta_1 \vdash s_1)$$

a substitution θ, a finite set of variables $V_1 \supseteq V(s_0)$ and a $\mathsf{R,C}$-normalized substitution ρ_1 such that

- $\Delta \vdash s_1\rho_1 \approx_{\alpha,\mathsf{C}} t_1$;

- $\mathtt{dom}(\rho_1) \subseteq V_1$;

- $\Delta \vdash \rho_0|_V \approx_{\alpha,\mathsf{C}} \theta\rho_1|_V$.

Proof. Consider that $\Delta \vdash t_0 \to_{[\mathtt{C}_0,R_0],\mathsf{C}} t_1$, where $R_0 = \nabla_0 \vdash l_0 \to r_0 \in \mathsf{R}$. That means that for some position $\mathtt{C}_0[\_]$ we have $t_0 \equiv \mathtt{C}_0[t'_0]$ and $\Delta \vdash \nabla_0\theta,\ \pi \cdot (l_0\theta) \approx_{\alpha,\mathsf{C}} t'_0,\ \mathtt{C}_0[\pi \cdot (r_0\theta)] \approx_{\alpha,\mathsf{C}} t_1$ (*). Also, $\mathtt{dom}(\theta) \cap V_0 = \emptyset$, since $V(R_0) \cap V(\Delta, t_0) = \emptyset$, because R_0 is a variable renamed rule with respect to $t_0 = s_0\rho_0$.

It follows from the fact that ρ_0 is normalized in Δ, that $\Delta \vdash s_0\rho_0 \approx_{\alpha,\mathsf{C}} t_0$ and $\Delta \vdash \Delta_0\rho_0$ that there exists a non-variable position $\mathtt{C}'_0$ such that $s_0 \equiv \mathtt{C}'_0[s'_0]$ and $\Delta \vdash s'_0\rho_0 \approx_{\alpha,\mathsf{C}} t'_0 \approx_{\alpha,\mathsf{C}} \pi \cdot (l_0\theta)$.

Now consider $\eta = \rho_0 \cup \theta$ (**). Then we may write $\Delta \vdash s'_0\eta \approx_{\alpha,\mathsf{C}} \pi \cdot (l_0\eta)$. Note that (Δ, η) is a solution for the unification problem $(\Delta_0 \vdash s'_0)\ {}_?{\approx}_?\ (\nabla_0 \vdash \pi \cdot l_0)$:

- $\Delta \vdash \Delta_0\eta$: from the hypothesis we already have $\Delta \vdash \Delta_0\rho_0$ and θ does not affect Δ_0 since $\mathtt{dom}(\theta) = V(R_0)$.

- $\Delta \vdash \nabla_0\eta$.

- $\Delta \vdash s'_0\eta \approx_{\alpha,\mathsf{C}} \pi \cdot (l_0\eta)$.

Let's take one solution (Δ_1, θ) from the complete set of solutions of $(\Delta_0 \vdash s'_0)\ {}_?{\approx}_?\ (\nabla_0 \vdash \pi \cdot l_0)$, i.e., from $\mathcal{U}_{\mathsf{C}}(\Delta_0 \vdash s'_0, \nabla_0 \vdash \pi \cdot l_0)$. Then, $\Delta_1 \vdash \Delta_0\theta, \nabla_0\theta, s'_0\theta \approx_{\alpha,\mathsf{C}} \pi \cdot (l_0\theta)$. Let s_1 be a nominal term such that $\Delta_1 \vdash \mathtt{C}'_0[\pi \cdot r_0]\theta \approx_{\alpha,\mathsf{C}} s_1$. With these conditions, we get $(\Delta_0 \vdash s_0) \rightsquigarrow_{[\mathtt{C}_0,R_0,\theta],\mathsf{C}} (\Delta_1 \vdash s_1)$.

Since $(\Delta_1, \theta) \in \mathcal{U}_{\mathsf{C}}(\Delta_0 \vdash s'_0, \nabla_0 \vdash \pi \cdot l_0)$ is one least unifier of $(\Delta_0 \vdash s'_0)\ {}_?{\approx}_?\ (\nabla_0 \vdash \pi \cdot l_0)$, in other words $(\Delta_1, \theta) \leq (\Delta, \eta)$, thus there exists a substitution ρ' such that for all X, $\Delta \vdash X\theta\rho' \approx_{\alpha,\mathsf{C}} X\eta$ and $\Delta \vdash \Delta_1\rho'$.

It remains to show the side conditions:

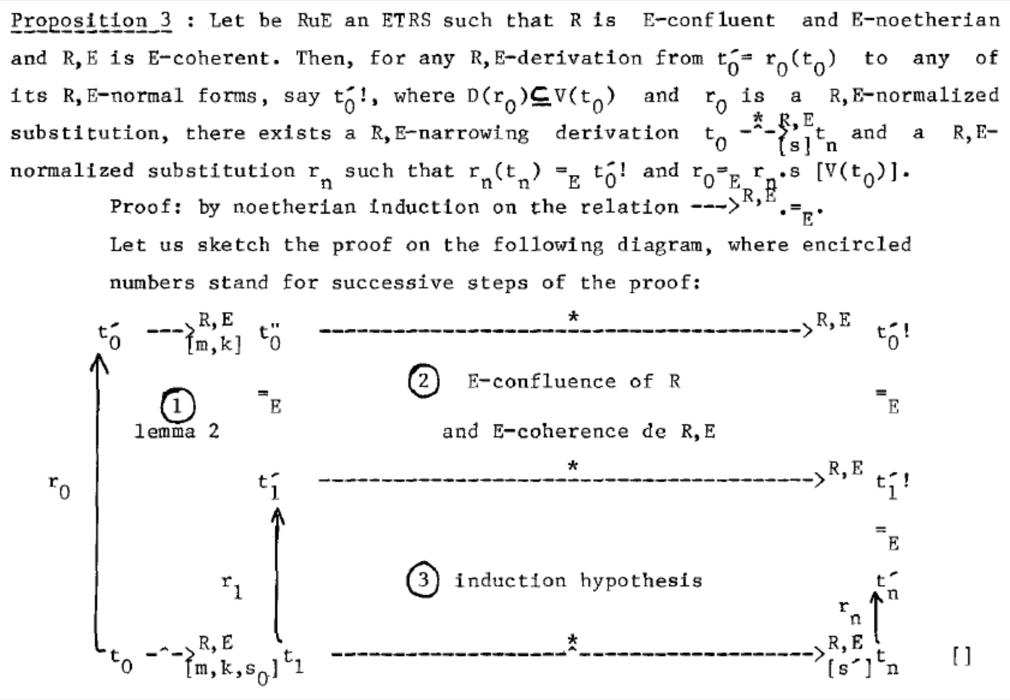
- $\Delta \vdash s_1\rho_1 \approx_{\alpha,\mathsf{C}} t_1$: On the one hand we have $\Delta \vdash s_1\rho_1 \approx_{\alpha,\mathsf{C}} (\mathtt{C}'_0[\pi \cdot r_0]\theta)\rho_1 \approx_{\alpha,\mathsf{C}} (\mathtt{C}'_0[\pi \cdot r_0]\theta)\rho' \approx_{\alpha,\mathsf{C}} \mathtt{C}'_0\theta\rho'[\pi \cdot r_0\theta\rho'] \approx_{\alpha,\mathsf{C}} \mathtt{C}'_0\eta[\pi \cdot r_0\eta]$. Since (**), we have $\mathtt{C}'_0\eta[\pi \cdot r_0\eta] \approx_{\alpha,\mathsf{C}} \mathtt{C}'_0\rho_0[\pi \cdot r_0\rho_0] \approx_{\alpha,\mathsf{C}} \mathtt{C}_0[\pi \cdot r_0\rho_0]$. On the other hand, $\Delta \vdash t_1 \overset{(*)}{\approx}_{\alpha,\mathsf{C}} \mathtt{C}_0[\pi \cdot r_0\theta] \approx_{\alpha,\mathsf{C}} \mathtt{C}_0[\pi \cdot r_0\eta] \approx_{\alpha,\mathsf{C}} \mathtt{C}_0[\pi \cdot r_0\rho_0]$.

- $\mathtt{dom}(\rho_1) \subseteq V_1$: Let $V_1 = (V_0 \cup \mathtt{Im}(\rho_0)) - \mathtt{dom}(\rho_0)$ and let ρ_1 be such that $\Delta \vdash \rho_1 \approx_{\alpha,\mathsf{C}} \rho'|_{V_1}$, with this α,C-equivalence, the result follows.

- $\Delta \vdash \rho_0|_V \approx_{\alpha,\mathsf{C}} \theta\rho_1|_V$: We already have that $\Delta \vdash \rho_0 \approx_{\alpha,\mathsf{C}} (\theta\rho_1)|_{V_0}$. We just need to apply the restriction to V in both sides and the result follows.

□

Remark 3.4. The generalization of the previous lemma to finite sequences of nominal rewrite steps is a bit more problematic. It was proven, for a general theory E, in the first-order case by Jouannaud et. al. [17], following an inductive proof, and with the conditions E-coherence for $\rightarrow_{\mathsf{R,E}}$ and E-confluence, to guarantee that relations $\rightarrow_{\mathsf{R/E}}$ and $\rightarrow_{\mathsf{R,E}}$ coincide, see below.



Proposition 3 : Let be RuE an ETRS such that R is E-confluent and E-noetherian and R,E is E-coherent. Then, for any R,E-derivation from $t'_0 = r_0(t_0)$ to any of its R,E-normal forms, say $t'_0!$, where $D(r_0) \subseteq V(t_0)$ and r_0 is a R,E-normalized substitution, there exists a R,E-narrowing derivation $t_0 \overset{*}{\dashrightarrow}{}^{R,E}_{[s]} t_n$ and a R,E-normalized substitution r_n such that $r_n(t_n) =_E t'_0!$ and $r_0 =_E r_n.s\ [V(t_0)]$.

Proof: by noetherian induction on the relation $\dashrightarrow^{R,E} . =_E$.

Let us sketch the proof on the following diagram, where encircled numbers stand for successive steps of the proof:

$$\begin{tikzcd}
t'_0 \arrow[r, dashed, "{R,E}", "{[m,k]}"'] & t''_0 \arrow[rr, dashed, "{*\ R,E}"] \arrow[dd, phantom, "=_E"] & & t'_0! \arrow[dd, phantom, "=_E"] \\
\textcircled{1}\ \text{lemma 2} & & \textcircled{2}\ \text{E-confluence of R and E-coherence de R,E} & \\
& t'_1 \arrow[rr, dashed, "{*\ R,E}"] & & t'_1! \arrow[d, phantom, "=_E"] \\
& & \textcircled{3}\ \text{induction hypothesis} & t'_n \\
t_0 \arrow[uuuu, "r_0"] \arrow[r, dashed, "{R,E}", "{[m,k,s_0]}"'] & t_1 \arrow[uu, "r_1"] \arrow[rr, dashed, "{*\ R,E}", "{[s']}"'] & & t_n \arrow[u, "r_n"]
\end{tikzcd}$$

[]

More details about the relations between $\rightarrow_{\mathsf{R/E}}$ and $\rightarrow_{\mathsf{R,E}}$ and the property of E-coherence can be found in Appendix A.

With that in mind, we propose a new definition for C-coherence in the nominal framework, that "extends" the one for first-order, but only considers the commutative theory.

Definition 3.6 (Nominal C-Coherence)**.** The relation $\Delta \vdash \_ \rightarrow_{\mathsf{R,C}} \_$ is called C-*coherent* iff for all t_1, t_2, t_3, t_4 such that $\Delta \vdash t_1 \approx_{\alpha,\mathsf{C}} t_2$ and $\Delta \vdash t_1 \rightarrow_{\mathsf{R,C}} t_3 \rightarrow^*_{\mathsf{R/C}} t_4$, there exist t_5, t_6, t_7 such that $\Delta \vdash t_4 \rightarrow^*_{\mathsf{R/C}} t_5$, $t_2 \rightarrow_{\mathsf{R,C}} t_6 \rightarrow^*_{\mathsf{R/C}} t_7$ and $\Delta \vdash t_5 \approx_{\mathsf{C}} t_7$, for some Δ. The idea of nominal C-coherence is similar to the one in first-order (cf. Remark 3.1) and illustrated below: dashed lines represent existentially quantified reductions.

$$\begin{tikzcd}
\Delta \vdash t_1 \arrow[r, "\mathsf{R,C}"'] \arrow[d, phantom, "\wr \approx_{\alpha,\mathsf{C}}"] & t_3 \arrow[r, "*", "\mathsf{R/C}"'] & t_4 \arrow[r, dashed, "*", "\mathsf{R/C}"'] & t_5 \arrow[d, phantom, "\wr \approx_{\alpha,\mathsf{C}}"] \\
\Delta \vdash t_2 \arrow[r, dashed, "\mathsf{R,C}"'] & t_6 \arrow[rr, dashed, "*", "\mathsf{R/C}"'] & & t_7
\end{tikzcd}$$

Below we present a naive version of Proposition 3 in Remark 3.4 and a draft of the proof: the exact conditions on the freshness contexts have to be further investigated due to our naive version of nominal C-coherence.

Theorem 3.2 (Naive version of Proposition 3 in [17])**.** Let R∪C be an ENRS such that R is C-confluent and C-terminating and $\rightarrow_{\mathsf{R,C}}$ is C-coherent. Let V_0 be a finite set of variables containing $V = V(\Delta_0, s_0)$. Then, for any R, C-derivation

$$\Delta \vdash t_0 = s_0\rho_0 \rightarrow^{*}_{\mathsf{R,C}} t_0{\downarrow}$$

to any of its R, C-normal forms, say $t_0{\downarrow}$, where $\mathtt{dom}(\rho_0) \subseteq V(s_0) \subseteq V_0$ and ρ_0 is a R, C-normalized substitution that satisfies Δ_0 with Δ, there exist a R, C-narrowing derivation

$$(\Delta_0 \vdash s_0) \rightsquigarrow^{*}_{\mathsf{R,C}} (\Delta_n \vdash s_n),$$

for each i, $0 \leq i < n$, with the composition of substitutions θ, and a R, C-normalized substitution ρ_n such that $\Delta \vdash s_n\rho_n \approx_{\alpha,\mathsf{C}} t_0{\downarrow}$ and $\Delta \vdash \rho_0|_V \approx_{\alpha,\mathsf{C}} \theta\rho_n|_V$.

Proof Sketch. By induction on the number of steps k applied in the derivation $\Delta \vdash t_0 = s_0\rho_0 \rightarrow^{*}_{\mathsf{R,C}} t_0{\downarrow}$.

- **Base Case:** For $k = 1$, by hypothesis we have $\Delta \vdash s_0\rho_0 = t_0 \rightarrow_{\mathsf{R,C}} t_1 = t_0{\downarrow}$. Since ρ_0 is normalized in Δ, the result follows directly from Lemma 3.2: $(\Delta_0 \vdash s_0) \rightsquigarrow^{\theta_0}_{\mathsf{R,C}} (\Delta_1 \vdash s_1)$, with $\Delta \vdash s_1\rho_1 \approx_{\alpha,\mathsf{C}} t_1 = t_0{\downarrow}$ and $\Delta \vdash \rho_0|_V \approx_{\alpha,\mathsf{C}} \theta_0\rho_1|_V$.

- **Induction Step:** Assume that the result holds for $k-1$. Then, we have

$$\Delta \vdash t_0 = s_0\rho_0 \rightarrow^{\{k-1\}}_{\mathsf{R,C}} t_{k-1} \rightarrow_{\mathsf{R,C}} t_0{\downarrow},$$

  for some t_{k-1}. By the induction hypothesis, there exist a R, C-narrowing derivation

$$(\Delta_0 \vdash s_0) \rightsquigarrow^{\{k-1\}}_{\mathsf{R,C}} (\Delta_{k-1} \vdash s_{k-1}),$$

with the composition of substitutions $\theta_0\theta_1\dots\theta_{k-2}$ and a R,C-normalized substitution ρ_{k-1} such that $\Delta \vdash s_{k-1}\rho_{k-1} \approx_{\alpha,\mathsf{C}} t_{k-1}$ and $\Delta \vdash \rho_0|_V \approx_{\alpha,\mathsf{C}} \theta_0\theta_1\dots\theta_{k-2}\rho_{k-1}|_V$.

Now using Lemma 3.2 over the rewrite step $\Delta \vdash t_{k-1} \to_{\mathsf{R},\mathsf{C}} t_0{\downarrow}$ we get that

$$(\Delta_{k-1} \vdash s_{k-1}) \rightsquigarrow^{\theta_{k-1}}_{\mathsf{R},\mathsf{C}} (\Delta_k \vdash s_k),$$

with $\Delta \vdash s_k\rho_k \approx_{\alpha,\mathsf{C}} t_k = t_0{\downarrow}$, where ρ_k is a R,C-normalized substitution, and $\Delta \vdash \rho_{k-1}|_V \approx_{\alpha,\mathsf{C}} \theta_{k-1}\rho_k|_V$.

Therefore, we obtained the R,C-narrowing derivation of k steps

$$(\Delta_0 \vdash s_0) \rightsquigarrow^{\theta_0\theta_1\dots\theta_{k-2}}_{\mathsf{R},\mathsf{C}} (\Delta_{k-1} \vdash s_{k-1}) \rightsquigarrow^{\theta_{k-1}}_{\mathsf{R},\mathsf{C}} (\Delta_k \vdash s_k),$$

with $\Delta \vdash s_k\rho_k \approx_{\alpha,\mathsf{C}} t_0{\downarrow}$ and

$$\Delta \vdash \rho_0|_V \approx_{\alpha,\mathsf{C}} \theta_0\theta_1\dots\theta_{k-2}\rho_{k-1}|_V \approx_{\alpha,\mathsf{C}} \theta_0\theta_1\dots\theta_{k-2}\theta_{k-1}\rho_k|_V,$$

thus, the result follows.

$$\begin{tikzcd}
\Delta \vdash t_0 \arrow[r, "\{k-1\}", "\mathsf{R,C}"'] & t_{k-1} \arrow[r, "\mathsf{R,C}"'] & t_0{\downarrow} \\
(\Delta_0 \vdash s_0) \arrow[u, dashed, "\rho_0"'] \arrow[r, rightsquigarrow, "k-1", "\mathsf{R,C}"'] & (\Delta_{k-1} \vdash s_{k-1}) \arrow[u, dashed, "\rho_{k-1}"'] \arrow[r, rightsquigarrow, "\theta_{k-1}", "\mathsf{R,C}"'] & (\Delta_k \vdash s_k) \arrow[u, dashed, "\rho_k"']
\end{tikzcd}$$

Fig. 3.2 Draft of the inductive step of the proof

□

Corollary 3.1. (C-Lifting Theorem) Nominal lifting modulo C is a consequence of Theorem 3.1 and Theorem 3.2.

The contributions of this dissertation end here, and next we present some new directions for future work.

Chapter 4

Conclusion and Future Work

In this dissertation we have extended definitions and notations of the works [6, 12] to the nominal framework with commutativity. Initially, we presented the syntax of nominal terms and their properties, in order to, in a second moment, deal with nominal unification modulo commutativity. Once we defined α-equivalence modulo C, we were able to extend Lemma 22 from [12] that guarantees that the derivability of judgements are preserved under substitutions, taking into account the commutativity theory. We have also shown the approach for nominal C-unification made by Ayala-Rincón et.al. [1], using triples of the form (Δ, θ, Pr) for solutions of C-unification problems, which is an alternative finitary representation of solutions, that can be taken into account in further developments of this work.

With basic results defined, we introduced the concepts and results that we would like to extend to the commutative equational theory, such as nominal rewriting, nominal narrowing and the nominal Lifting Theorem (Theorem 2.2).

Finally, taking advantage of everything that has been done so far, we started extending the results by Jouannaud et.al. [17], such as first-order rewriting and narrowing modulo C to the nominal framework, and also standard results that are necessary for proving the Lifting Theorem modulo commutativity. We proposed definitions for nominal R,C-rewriting and R,C-narrowing, proved some properties and illustrated with examples. We observed that due to the fact that nominal C-unification based on freshness constraints only is not finitary, then our nominal C-narrowing tree is not finitary.

What we presented here is just the beginning of a long hard work. Below we present our work-in-progress and what remains as future work.

4.1 Work in progress

4.1.1 Nominal C-narrowing for Nominal R∪C-unification

In order to use nominal C-narrowing as a procedure for nominal R∪C-unification, thus extending the results in Section 2.3, we will "apply" the narrowing relation in two terms in parallel, and for that we will follow the same strategy as before, and use a "new" function symbol h, that is $h \notin \Sigma$, and s and t represent the initial two terms that are to be unified modulo C.

Theorem 4.1 (Soundness without closedness)**.** Let $\Delta \vdash s$ and $\nabla \vdash t$ be two nominal terms-in-context and $\Delta, \nabla \vdash h(s,t) = u_0 \rightsquigarrow^*_{\mathsf{R,C}} \Delta_n \vdash u_n = h(s_n,t_n)$ a R, C-narrowing derivation such that $\{\Delta_n, s_n \,{}_?\overset{\mathsf{C}}{\approx}_?\, t_n\}$ has a solution, say (Γ,ρ). Then (Γ,ρ_0) is an C-solution of the problem $\Delta,\nabla, s \,{}_?\overset{\mathsf{C}}{\approx}_?\, t$, where $\rho_0 = \theta\rho$ and θ is the composition of substitutions along the narrowing derivation.

Proof. From the previous Theorem 3.1 using $\rho = \rho_n$, we can associate the R, C-narrowing derivation with the following R, C-rewriting derivation:

$$\Gamma \vdash u_0\rho_0 = v_0 \to_{\mathsf{R,C}} v_1 \to_{\mathsf{R,C}} \cdots \to_{\mathsf{R,C}} v_n = h(v_n^s, v_n^t),$$

see Figure 4.1. Remember that h makes a role of cartesian product, and since $u_0\rho_0 = h(s,t)\rho_0 = h(s\rho_0, t\rho_0)$ it follows that we can write two R, C-rewriting derivations starting on $s\rho_0$ and $t\rho_0$, that is, $\Gamma \vdash s\rho_0 \to^*_{\mathsf{R,C}} v_n^s$ and $\Gamma \vdash t\rho_0 \to^*_{\mathsf{R,C}} v_n^t$.

Moreover, from Theorem 3.1 we get that $\Gamma \vdash v_n^s \approx_{\alpha,\mathsf{C}} s_n\rho_n$ and $\Gamma \vdash v_n^t \approx_{\alpha,\mathsf{C}} t_n\rho_n$. Also, by hypothesis, (Γ,ρ_n) is a solution for $\{\Delta_n, s_n \,{}_?\overset{\mathsf{C}}{\approx}_?\, t_n\}$, thus $\Gamma \vdash_{\mathsf{C}} s_n\rho_n = t_n\rho_n$ and therefore $\Gamma \vdash_{\mathsf{C}} v_n^s = v_n^t$.

Thus, we get the following result:

$$\Gamma \vdash_{\mathsf{C}} s\rho_0 = s\theta\rho = s\theta_0 \dots \theta_{n-1}\rho = v_n^s = v_n^t = t\theta_0 \dots \theta_{n-1}\rho = t\theta\rho = t\rho_0.$$

Therefore, $\Gamma \vdash_{\mathsf{C}} s\theta\rho = t\theta\rho$ and hence $(\Gamma,\theta\rho)$ is a C-solution for $\Delta,\nabla, s \,{}_?\overset{\mathsf{C}}{\approx}_?\, t$.

□

For soundness we do not need narrowing derivations to be closed, but we can still add this condition to the hypothesis without loss of generality. The only need for closedness is to construct a complete set of R∪E-unifiers. Thus, we state the theorem with closedness, and the proof is analogous to the proof of the previous theorem.

$$\begin{tikzcd}
\Delta,\nabla \vdash h(s,t) = u_0 \arrow[r, rightsquigarrow] \arrow[d, dashed, "\rho_0"] & \Delta_1 \vdash u_1 \arrow[r, rightsquigarrow, "*"] \arrow[d, dashed, "\rho_1"] & \Delta_n \vdash u_n = h(s_n,t_n) \arrow[d, dashed, "\rho_n"] \\
\Gamma \vdash u_0\rho_0 = v_0 \arrow[r] & v_1 \arrow[r, "*"] & v_n = h(v_n^s, v_n^t)
\end{tikzcd}$$

Fig. 4.1 Schema of the proof of Soundness

Remark 4.1 (Soundness with closedness). Let $\Delta \vdash s$ and $\nabla \vdash t$ be two nominal terms-in-context and $\Delta,\nabla \vdash h(s,t) = u_0 \rightsquigarrow^{c}_{\mathsf{R},\mathsf{C}} \cdots \rightsquigarrow^{c}_{\mathsf{R},\mathsf{C}} \Delta_n \vdash u_n = h(s_n,t_n)$ a closed R,C-narrowing derivation such that $\{\Delta_n, s_n \,{}_?\overset{\mathsf{C}}{\approx}_?\, t_n\}$ has a solution, say (Γ,ρ). Then (Γ,ρ_0) is a C-solution of the problem $\Delta,\nabla, s \,{}_?\overset{\mathsf{C}}{\approx}_?\, t$, where $\rho_0 = \theta\rho$ and θ is the composition of substitutions along the narrowing derivation.

Proof. Future work. □

The following result is still not stable, since it depends on Theorem 3.2. Assuming this last one is correct, we propose a draft of the proof of the following lemma.

Remark 4.2 (Naive Completeness). Let $\Delta \vdash s$ and $\nabla \vdash t$ be two nominal terms-in-context such that the problem $(\Delta \vdash s) \,{}_?\overset{\mathsf{C}}{\approx}_?\, (\nabla \vdash t)$ has a C-solution, (Δ',ρ), and let V be a finite set of variables containing $V(\Delta,\nabla,s,t)$. Then, there exists a closed narrowing derivation: $\nabla,\Delta \vdash u = (s,t) \rightsquigarrow^{c}_{\mathsf{R},\mathsf{C}} \cdots \rightsquigarrow^{c}_{\mathsf{R},\mathsf{C}} \Gamma_n \vdash (s_n,t_n)$, such that $\{\Gamma_n, s_n \,{}_?\overset{\mathsf{C}}{\approx}_?\, t_n\}$ has a solution. Let $(\Gamma,\mu) = mgu(\Gamma_n, s_n \approx_{\alpha,\mathsf{C}} t_n)$, and θ the composition of the narrowing substitutions. Then, $(\Gamma,\theta\mu) \leq^{V}_{\mathsf{C}} (\Delta',\rho)$. Moreover, we are allowed to restrict our attention to $\rightsquigarrow^{c}_{\mathsf{R},\mathsf{C}}$-derivations such that: $\forall i$, $0 \leq i < n$, $\theta_i|_V$ and θ are normalized.

Proof Sketch. By definition of C-solution, we have $\Delta' \vdash_{\mathsf{C}} \Delta\rho, \nabla\rho, s\rho \approx_{\alpha,\mathsf{C}} t\rho$. Let's take $\eta \approx_{\alpha,\mathsf{C}} \rho\!\downarrow$, that is, the normal form substitution of ρ in Δ': $\Delta' \vdash X\eta \approx_{\alpha,\mathsf{C}} (X\rho)\!\downarrow$. Thus, we can write $\Delta' \vdash_{\mathsf{C}} \Delta\eta, \nabla\eta, s\eta \approx_{\alpha,\mathsf{C}} t\eta$, because the rules are closed. Since C is a closed nominal theory presented by a convergent rewrite system R, and since closed rewriting is complete for equational reasoning in this case, $s\eta$ and $t\eta$ have the same normal form in Δ', and we will call it r. Thus,

$$\Delta' \vdash u\eta = h(s\eta, t\eta) = t'_0 \rightarrow^{c}_{\mathsf{R},\mathsf{C}} \cdots \rightarrow^{c}_{\mathsf{R},\mathsf{C}} t'_n = h(r,r).$$

By Theorem 3.2 there exists a corresponding narrowing derivation ending with $\Gamma_n \vdash h(s_n,t_n)$ such that: $\Delta' \vdash h(s_n\eta_n, t_n\eta_n) \approx_{\alpha,\mathsf{C}} t'_n = h(r,r)$ and $\Delta' \vdash \Gamma_n\eta_n$. Thus, (Δ',η_n) is a solution of $\{\Gamma_n, s_n \,{}_?\overset{\mathsf{C}}{\approx}_?\, t_n\}$.

The hypothesis gives us that (Γ,μ) is the least unifier, so it follows that $(\Gamma,\mu) \leq (\Delta',\eta_n)$ and there exists δ such that $\forall X$, $\Delta' \vdash X\mu\delta \approx_{\alpha,\mathsf{C}} X\eta_n$ and $\Delta' \vdash \Gamma\delta$. Therefore, by Theorem 3.2, $\Delta' \vdash (\theta\mu\delta|_V \approx_{\alpha,\mathsf{C}} \theta\eta_n|_V \approx_{\alpha,\mathsf{C}} \eta|_V)$ and $\Delta' \vdash_{\mathsf{C}} \eta|_V = \rho|_V$ that is, $(\Gamma,\theta\mu) \leq_{\mathsf{C}}^{V} (\Delta',\rho)$.

□

4.2 Future Work

We observe that the approach taken in [1] where solutions for C-unification problems are represented by triples containing fixed-point equations could give a finite number of narrowing branches. Another approach could be the use of fixed-point constraints [8] that also gives a complete set of C-unifiers for a nominal C-unification problem. These two approaches need to be taken into account in nominal versions of our definitions of nominal C-narrowing and rewriting, and in related results. Both approaches will be investigated later.

We still need to check the lifting theorem modulo C for closed rewriting to deal with nominal rewrite systems modulo C that are complete presentations of equational theories. Then, we need to investigate if our statements and proof sketches for completeness results are accurate.

Another future work would be to extend the proposition that relates $\rightarrow_{\mathsf{R},\mathsf{C}}$ with $\rightarrow_{\mathsf{R}/\mathsf{C}}$ to the nominal framework. In first-order this result is as Proposition 1 in Remark 3.3. It would also be interesting to prove a version of the Critical Pair Lemma for nominal rewriting modulo C. The first order version can be found in Appendix A, Theorem A.1.

References

[1] Ayala-Rincón, M., de Carvalho Segundo, W., Fernández, M., and Nantes-Sobrinho, D. (2017a). Nominal C-unification. *CoRR*, abs/1709.05384.

[2] Ayala-Rincón, M., de Carvalho Segundo, W., Fernández, M., and Nantes-Sobrinho, D. (2017b). On solving nominal fixpoint equations. In Dixon, C. and Finger, M., editors, *Frontiers of Combining Systems - 11th International Symposium, FroCoS 2017, Brasília, Brazil, September 27-29, 2017, Proceedings*, volume 10483 of *Lecture Notes in Computer Science*, pages 209–226. Springer.

[3] Ayala-Rincón, M., de Carvalho Segundo, W., Fernández, M., and Nantes-Sobrinho, D. (2018a). A formalisation of nominal C-matching through unification with protected variables. In Accattoli, B. and Olarte, C., editors, *Proceedings of the 13th Workshop on Logical and Semantic Frameworks with Applications, LSFA 2018, Fortaleza, Brazil, September 26-28, 2018*, volume 344 of *Electronic Notes in Theoretical Computer Science*, pages 47–65. Elsevier.

[4] Ayala-Rincón, M., de Carvalho Segundo, W., Fernández, M., Nantes-Sobrinho, D., and Oliveira, A. C. R. (2019). A formalisation of nominal α-equivalence with A, C, and AC function symbols. *Theor. Comput. Sci.*, 781:3–23.

[5] Ayala-Rincón, M., Fernández, M., Gabbay, M. J., and Oliveira, A. C. R. (2015). Checking overlaps of nominal rewriting rules. In Benevides, M. R. F. and Thiemann, R., editors, *Proceedings of the Tenth Workshop on Logical and Semantic Frameworks, with Applications, LSFA 2015, Natal, Brazil, August 31 - September 1, 2015*, volume 323 of *Electronic Notes in Theoretical Computer Science*, pages 39–56. Elsevier.

[6] Ayala-Rincón, M., Fernández, M., and Nantes-Sobrinho, D. (2016). Nominal narrowing. In Kesner, D. and Pientka, B., editors, *1st International Conference on Formal Structures for Computation and Deduction, FSCD 2016, June 22-26, 2016, Porto, Portugal*, volume 52 of *LIPIcs*, pages 11:1–11:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

[7] Ayala-Rincón, M., Fernández, M., and Nantes-Sobrinho, D. (2018b). Fixed-point constraints for nominal equational unification. In Kirchner, H., editor, *3rd International Conference on Formal Structures for Computation and Deduction, FSCD 2018, July 9-12, 2018, Oxford, UK*, volume 108 of *LIPIcs*, pages 7:1–7:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

[8] Ayala-Rincón, M., Fernández, M., and Nantes-Sobrinho, D. (2020). On nominal syntax and permutation fixed points. *Log. Methods Comput. Sci.*, 16(1).

[9] Baader, F. and Nipkow, T. (1998). *Term rewriting and all that.* Cambridge University Press.

[10] Escobar, S., Meseguer, J., and Sasse, R. (2008). Variant narrowing and equational unification. In Rosu, G., editor, *Proceedings of the Seventh International Workshop on Rewriting Logic and its Applications, WRLA 2008, Budapest, Hungary, March 29-30, 2008*, volume 238 of *Electronic Notes in Theoretical Computer Science*, pages 103–119. Elsevier.

[11] Fay, M. (1978). *First-order Unification in an Equational Theory.* University of California, Santa Cruz.

[12] Fernández, M. and Gabbay, M. (2007). Nominal rewriting. *Inf. Comput.*, 205(6):917–965.

[13] Fernández, M. and Gabbay, M. J. (2010). Closed nominal rewriting and efficiently computable nominal algebra equality. In Crary, K. and Miculan, M., editors, *Proceedings 5th International Workshop on Logical Frameworks and Meta-languages: Theory and Practice, LFMTP 2010, Edinburgh, UK, 14th July 2010*, volume 34 of *EPTCS*, pages 37–51.

[14] Gabbay, M. and Pitts, A. M. (2002). A new approach to abstract syntax with variable binding. *Formal Aspects Comput.*, 13(3-5):341–363.

[15] Hullot, J. (1980). Canonical forms and unification. In Bibel, W. and Kowalski, R. A., editors, *5th Conference on Automated Deduction, Les Arcs, France, July 8-11, 1980, Proceedings*, volume 87 of *Lecture Notes in Computer Science*, pages 318–334. Springer.

[16] Jouannaud, J. (1983). Confluent and coherent equational term rewriting systems: Application to proofs in abstract data types. In Ausiello, G. and Protasi, M., editors, *CAAP'83, Trees in Algebra and Programming, 8th Colloquium, L'Aquila, Italy, March 9-11, 1983, Proceedings*, volume 159 of *Lecture Notes in Computer Science*, pages 269–283. Springer.

[17] Jouannaud, J., Kirchner, C., and Kirchner, H. (1983). Incremental construction of unification algorithms in equational theories. In Díaz, J., editor, *Automata, Languages and Programming, 10th Colloquium, Barcelona, Spain, July 18-22, 1983, Proceedings*, volume 154 of *Lecture Notes in Computer Science*, pages 361–373. Springer.

[18] Lankford, D. S. (1975). Canonical inference. Report ATP-32, Departments of Mathematics and Computer Sciences, University of Texas at Austin.

[19] Newman, M. H. A. (1942). On theories with a combinatorial definition of "equivalence". *Annals of Mathematics*, 43(2):223–243.

[20] Peterson, G. E. and Stickel, M. E. (1981). Complete sets of reductions for some equational theories. *J. ACM*, 28(2):233–264.

[21] Slagle, J. R. (1974). Automated theorem-proving for theories with simplifiers commutativity, and associativity. *J. ACM*, 21(4):622–642.

[22] Suzuki, T., Kikuchi, K., Aoto, T., and Toyama, Y. (2015). Confluence of orthogonal nominal rewriting systems revisited. In Fernández, M., editor, *26th International Conference on Rewriting Techniques and Applications, RTA 2015, June 29 to July 1, 2015, Warsaw, Poland*, volume 36 of *LIPIcs*, pages 301–317. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

[23] Terese (2003). *Term rewriting systems*, volume 55 of *Cambridge tracts in theoretical computer science*. Cambridge University Press.

[24] Urban, C., Pitts, A. M., and Gabbay, M. (2004). Nominal unification. *Theor. Comput. Sci.*, 323(1-3):473–497.

[25] Viola, E. (2001). E-unifiability via narrowing. In Restivo, A., Rocca, S. R. D., and Roversi, L., editors, *Theoretical Computer Science, 7th Italian Conference, ICTCS 2001, Torino, Italy, October 4-6, 2001, Proceedings*, volume 2202 of *Lecture Notes in Computer Science*, pages 426–438. Springer.

Appendix A

First Order Results

A *rewrite rule* (or *reduction rule*) for a signature Σ is defined by [23] as a pair (l,r) of terms, written as $l \to r$, which the left-hand side l is not a variable and $V(r) \subseteq V(l)$. A *term rewrite system (TRS)* is a pair $\mathsf{R} = (\Sigma, R)$ of a signature Σ and a set of rewrite rules R for Σ. The *one-step rewrite relation* $s \to_R t$ is defined whenever we have a substitution θ such that $\mathtt{C}[l\theta] \to_R \mathtt{C}[r\theta]$ for some $R \in \mathsf{R}$.

Intuitively, narrowing a term consists of applying a minimal substitution to this term and then make a rewrite step [15].

Definition A.1. The *one-step narrowing relation* $s \rightsquigarrow_{[\mathtt{C},R,\theta]} t$ is the least relation such that for any $R = \{l \to r\} \in \mathsf{R}$, position $\mathtt{C}$, term s' and substitution θ,

$$\frac{s \equiv \mathtt{C}[s'] \qquad s'\theta = l\theta \quad (\mathtt{C}[r])\theta = t,}{s \rightsquigarrow_{[\mathtt{C},R,\theta]} t} \text{ where } \theta = mgu(l,s').$$

We may omit subindices if they are clear from the context. If we want to explicit the substitution used in the narrowing step, we may write $s \rightsquigarrow^{\theta} t$.

Example A.1. A simple example of a narrowing step can be seen as follows. Let $\Sigma = \{f : 2, g : 1\}$ be the signature over the sets $\mathcal{X}$ and $\mathcal{A}$, and consider the rewrite system $\mathsf{R} = \{R : g(Z) \to a\}$.

Taking $\theta = \{Z \to Y\}$, we say that $f(X, g(Y)) \rightsquigarrow_{[2,R,\theta]} f(X,a)$.

Considering R a convergent term rewrite system, i.e., confluent and terminating, that is equivalent to the set of identities E, the narrowing process can be applied iteratively to an equation until find another equation whose terms are syntactically unifiable. The composition of this generated *mgu* with all substitutions used in the narrowing sequence yields an E-unifier of the initial equation [9].

One of the main problems of commutativity is that it cannot be oriented into a terminating rewrite rule, and therefore we cannot have a terminating system. An useful way out is to build the commutativity into the rewrite process [9]. In other words, we take into account the commutativity when applying a rewrite rule.

Considering a signature $\Sigma = \Sigma_{\mathsf{E}} \cup \Sigma_0$, where Σ_{E} consists of the function symbols satisfying an equational theory E and Σ_0 is a set of uninterpreted function symbols, and assuming we have a theory $\mathsf{T} = (\Sigma_{\mathsf{E}} \cup \Sigma_0, Ax)$ where Ax is a set of axioms that can be split into a set R of rules and a set E of equations, following the definitions and notations from [17], we obtain what we call an *equational term rewriting system* (ETRS), denoted $\mathsf{R} \cup \mathsf{E}$. Formally,

Definition A.2. (Relation $\rightarrow_{\mathsf{R/E}}$) Given a theory $\mathsf{T} = (\Sigma, Ax) = (\Sigma_{\mathsf{E}} \cup \Sigma_0, \mathsf{R} \cup \mathsf{E})$, we denote as $\rightarrow_{\mathsf{R/E}}$ the relation $\approx_{\mathsf{E}} \cdot \rightarrow_{\mathsf{R}}$, which is defined on E-equivalence classes of terms, i.e.,

$$[s]_{\approx_{\mathsf{E}}} \rightarrow_{\mathsf{R/E}} [t]_{\approx_{\mathsf{E}}} \text{ iff there exist } s', t' \text{ such that } s \approx_{\mathsf{E}} s' \rightarrow_{\mathsf{R}} t' \approx_{\mathsf{E}} t.$$

To make a clear distinction between atoms $(a, b, c, \ldots)$ and constants $(\mathtt{a}, \mathtt{b}, \mathtt{c}, \ldots)$ we will use a different font for the later.

Example A.2. Consider the theory $\mathsf{T} = \{f(h(\mathtt{a}), \mathtt{b}) \approx \mathtt{a}, f(\mathtt{b}, X) \approx X, h(X) \approx X, f(X,Y) \approx f(Y,X)\}$, with $\mathtt{a}, \mathtt{b}$ constants. We will decompose T in $\mathsf{R} \cup \mathsf{C}$ where

- $\mathsf{R} = \{f(h(\mathtt{a}), \mathtt{b}) \rightarrow \mathtt{a}, f(\mathtt{b}, X) \rightarrow X, h(X) \rightarrow X\}$; and
- $\mathsf{C} = \{f(X,Y) \approx f(Y,X)\}$.

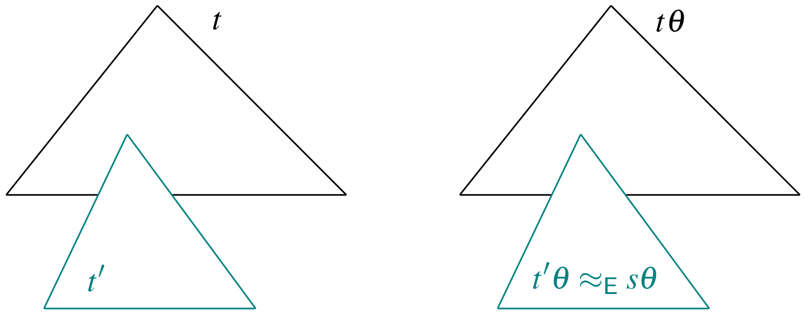
The following is a $\mathsf{R/C}$-reduction step example:

$$f(\mathtt{b}, h(\mathtt{a})) \rightarrow_{\mathsf{R/C}} \mathtt{a} \text{ because } f(\mathtt{b}, h(\mathtt{a})) \approx_{\mathsf{C}} f(h(\mathtt{a}), \mathtt{b}) \rightarrow_{\mathsf{R}} \mathtt{a} \approx_{\mathsf{C}} \mathtt{a}.$$

As expected, the notions of overlappings and critical pairs have to be extended to deal with an equational theory E.

Definition A.3. (E-overlap of terms and rules)

- We say a term s E-*overlaps* a term t at position $\mathtt{C}$, say $t \equiv \mathtt{C}[t']$, with a complete set S of E-overlappings iff S is a complete set of E-unifiers of s and t'.



- Given two rules $l_1 \to r_1$ and $l_2 \to r_2$ such that $V(l_1) \cap V(l_2) = \emptyset$ and l_1 E-overlaps l_2 at position $\mathtt{C}$, say $l_2 \equiv \mathtt{C}[l_2']$, with a complete set S of E-overlappings, then the set

$$\{\langle u_1, u_2\rangle \mid u_1 = r_2\theta,\ u_2 = \mathtt{C}\theta[r_1\theta],\ \forall\theta \in S\}$$

  is called a *complete set of* E*-critical pairs* of the rule $l_1 \to r_1$ on the rule $l_2 \to r_2$ at position $\mathtt{C}$.

  - Let $CS\mathsf{E}CP(\mathsf{R})$ be the complete set of non trivial E-critical pairs[1] for all $l_1 \to r_1$ and $l_2 \to r_2$ belonging both to R;
  - Let $CS\mathsf{E}CP(\mathsf{R}/\mathsf{E})$ be the complete set of non trivial E-critical pairs for all $l_1 \to r_1$ in R together with all $l_2 \to r_2$ such that $l_2 \approx r_2$ or $r_2 \approx l_2$ belongs to E.

Since in this work we are interested in the commutative theory C, we will mostly be analysing the sets $CS\mathsf{C}CP(\mathsf{R})$ and $CS\mathsf{C}CP(\mathsf{R}/\mathsf{C})$.

Recall that we use disjunctions of reduced problems when applying $(\approx_{\alpha,\mathsf{C}} C)$, because since f is a commutative function symbol, we need to consider both of the possible pairs generated by its arguments, that is:

$$\{f(s_0,s_1)\ {}_?\overset{\mathsf{C}}{\approx}_?\ f(t_0,t_1)\} \Longrightarrow \{s_0\ {}_?\overset{\mathsf{C}}{\approx}_?\ t_0, s_1\ {}_?\overset{\mathsf{C}}{\approx}_?\ t_1\} \vee \{s_0\ {}_?\overset{\mathsf{C}}{\approx}_?\ t_1, s_1\ {}_?\overset{\mathsf{C}}{\approx}_?\ t_0\}$$

Example A.3 (Cont. Example A.2)**.** Let's analyze the complete set of C-critical pairs of R, $CS\mathsf{C}CP(\mathsf{R})$.

$$\mathsf{R} = \begin{cases} l_1 \to r_1: & f(h(\mathtt{a}),\mathtt{b}) \to \mathtt{a} \\ l_2 \to r_2: & f(\mathtt{b},X) \to X \\ l_3 \to r_3: & h(X) \to X \end{cases}$$

Since we are in a restricted nominal version of the nominal C-unification, we will use in this example, and in following ones, the same notation from Definition 1.13.

[1] that means E-critical pairs such that $u_1 \neq u_2$.

1) The following table summarizes where l_i C-overlaps l_1, for $i = 1,2,3$:

$\mathtt{C}$	$[\_]$	renamed l_1	l_2	l_3
$[\_]$	$f(h(\mathtt{a}),\mathtt{b})$	$\oslash$	$\langle \mathtt{a}, h(\mathtt{a})\rangle$	$\perp$
$f([\_],\mathtt{b})$	$h(\mathtt{a})$	$\perp$	$\perp$	$\langle \mathtt{a}, f(\mathtt{a},\mathtt{b})\rangle$
$f(h(\mathtt{a}),[\_])$	$\mathtt{b}$	$\perp$	$\perp$	$\perp$
$f(h([\_]),\mathtt{b})$	$\mathtt{a}$	$\perp$	$\perp$	$\perp$

The fourth column gives the complete set of C-critical pairs of the rule $l_2 \to r_2$ and $l_1 \to r_1$ in position of l_1 given by column 1.

For instance,

a. $l_2 \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; l_1$:

$$\begin{aligned}
\{f(\mathtt{b},X) \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; f(h(\mathtt{a}),\mathtt{b})\} &\implies \{\mathtt{b} \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; h(\mathtt{a}), X \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; \mathtt{b}\} \vee \{X \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; h(\mathtt{a}), \mathtt{b} \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; \mathtt{b}\} \\
&\implies \perp \vee \{X \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; h(\mathtt{a})\} \\
&\;\therefore \quad \theta = [X \mapsto h(\mathtt{a})]
\end{aligned}$$

Thus, l_2 C-overlaps with l_1 in position $\mathtt{C} = [\_]$ with set $S = \{\theta\}$, where θ is the only C-unifier of l_2 and l_1 in this case.

The C-critical pair of the rule $l_2 \to r_2$ with the rule $l_1 \to r_1$ at position $\mathtt{C} = [\_]$ is $\langle r_1\theta, \mathtt{C}\theta[r_2\theta]\rangle = \langle \mathtt{a}, h(\mathtt{a})\rangle$.

$$\begin{tikzcd}
 & l_1\theta \arrow[dl, "l_1 \to r_1"'] \arrow[dr, "l_2 \to r_2"] & \\
\mathtt{a} = r_1\theta & & \mathtt{C}\theta[r_2\theta] = h(\mathtt{a})
\end{tikzcd}$$

Notice they are joinable since $h(\mathtt{a}) \to \mathtt{a}$ via rule $l_3 \to r_3$.

b. $l_3 \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; l_1' = h(\mathtt{a})$:

$$\begin{aligned}
\{h(X) \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; h(\mathtt{a})\} &\implies \{X \;{}_{?}\overset{\mathtt{C}}{\approx}_{?}\; \mathtt{a}\} \\
&\;\therefore \quad \theta = [X \mapsto \mathtt{a}]
\end{aligned}$$

Thus, l_3 C-overlaps with $l_1' = h(\mathtt{a})$ at position $\mathtt{C} = f([\_],\mathtt{b})$ with set $S = \{\theta\}$, where θ is the only C-unifier of l_3 and $h(\mathtt{a})$ in this case.

The C-critical pair of the rule $l_3 \rightarrow r_3$ with the rule $l_1 \rightarrow r_1$ at position $\mathtt{C} = f([\_], \mathtt{b})$ is $\langle r_1\theta, \mathtt{C}\theta[r_3\theta]\rangle = \langle \mathtt{a}, f(\mathtt{a}, \mathtt{b})\rangle$.

$$\begin{tikzcd}
 & l_1\theta \arrow[dl, "l_1 \rightarrow r_1"'] \arrow[dr, "l_3 \rightarrow r_3"] & \\
\mathtt{a} = r_1\theta & & \mathtt{C}\theta[r_3\theta] = f(\mathtt{a}, \mathtt{b})
\end{tikzcd}$$

Notice that $\mathtt{a}$ and $f(\mathtt{a}, \mathtt{b})$ are joinable, since f is a commutative function symbol, that is $f(\mathtt{a}, \mathtt{b}) \approx_{\mathtt{C}} f(\mathtt{b}, \mathtt{a})$, and this last term reduces via rule $l_2 \rightarrow r_2$ to $\mathtt{a}$.

For the other cases we get a clash $\bot$, because we can not unify terms with two different function symbols. Also, $\oslash$ means that we do not need to unify a term with a copy of itself.

2) l_i C-overlaps l_2, for $i = 1, 2, 3$:

$\mathtt{C}$	$[\_]$	l_1	renamed l_2	l_3
$[\_]$	$f(\mathtt{b}, X)$	$\langle h(\mathtt{a}), \mathtt{a}\rangle$	$\oslash$	$\bot$
$f([\_], X)$	$\mathtt{b}$	$\bot$	$\bot$	$\bot$

The reasoning is the same:

a. $l_1 \; {}_?\overset{\mathtt{C}}{\approx}_? \; l_2$:

$$\begin{aligned}
\{f(h(\mathtt{a}), \mathtt{b}) \; {}_?\overset{\mathtt{C}}{\approx}_? \; f(\mathtt{b}, X)\} &\Longrightarrow \{h(\mathtt{a}) \; {}_?\overset{\mathtt{C}}{\approx}_? \; \mathtt{b}, \mathtt{b} \; {}_?\overset{\mathtt{C}}{\approx}_? \; X\} \vee \{h(\mathtt{a}) \; {}_?\overset{\mathtt{C}}{\approx}_? \; X, \mathtt{b} \; {}_?\overset{\mathtt{C}}{\approx}_? \; \mathtt{b}\} \\
&\Longrightarrow \bot \vee \{X \; {}_?\overset{\mathtt{C}}{\approx}_? \; h(\mathtt{a})\} \\
&\;\;\therefore \quad \theta = [X \mapsto h(\mathtt{a})]
\end{aligned}$$

Thus, l_1 C-overlaps with l_2 at position $\mathtt{C} = [\_]$ with set $S = \{\theta\}$, where θ is the only C-unifier of l_1 and l_2 in this case.

The C-critical pair of the rule $l_1 \rightarrow r_1$ with the rule $l_2 \rightarrow r_2$ at position $\mathtt{C} = [\_]$ is $\langle r_2\theta, \mathtt{C}\theta[r_1\theta]\rangle = \langle h(\mathtt{a}), \mathtt{a}\rangle$.

$$\begin{tikzcd}
 & l_2\theta \arrow[dl, "l_2 \rightarrow r_2"'] \arrow[dr, "l_1 \rightarrow r_1"] & \\
h(\mathtt{a}) = r_2\theta & & \mathtt{C}\theta[r_1\theta] = \mathtt{a}
\end{tikzcd}$$

3) l_i C-overlaps l_3, for $i = 1,2,3$:

$\mathtt{C}$	$[\_]$	l_1	l_2	renamed l_3
$[\_]$	$h(X)$	$\perp$	$\perp$	$\oslash$

Thus, $CS\mathsf{C}CP(\mathsf{R}) = \{\langle \mathtt{a}, h(\mathtt{a})\rangle, \langle \mathtt{a}, f(\mathtt{a},\mathtt{b})\rangle, \langle h(\mathtt{a}), \mathtt{a}\rangle\}$.

Example A.4 (Cont. Example A.2)**.** Now let's analyze the complete set $CS\mathsf{C}CP(\mathsf{R}/\mathsf{C})$ of C-critical pairs of R/C.

Since $f(X,Y) \approx f(Y,X)$ is in C, we consider the new rule $l_4 \to r_4 : f(X,Y) \to f(Y,X)$, and now we will look for non-trivial C-critical pairs of each of the other three rules with $l_4 \to r_4$.

1) l_i C-overlaps l_4, for $i = 1,2,3$:

$\mathtt{C}$	$[\_]$	l_1	l_2	l_3
$[\_]$	$f(X,Y)$	$\langle f(\mathtt{b},h(\mathtt{a})),\mathtt{a}\rangle$	$\langle f(X,\mathtt{b}),X\rangle$	$\perp$

a. $l_1 \ {}_?\overset{\mathsf{C}}{\approx}_? \ l_4$:

$$\begin{aligned}
\{f(h(\mathtt{a}),\mathtt{b}) \ {}_?\overset{\mathsf{C}}{\approx}_? \ f(X,Y)\} \quad &\Longrightarrow \quad \{h(\mathtt{a}) \ {}_?\overset{\mathsf{C}}{\approx}_? \ X, \mathtt{b} \ {}_?\overset{\mathsf{C}}{\approx}_? \ Y\} \vee \\
&\qquad\quad \vee \{h(\mathtt{a}) \ {}_?\overset{\mathsf{C}}{\approx}_? \ Y, \mathtt{b} \ {}_?\overset{\mathsf{C}}{\approx}_? \ X\} \\
&\therefore \quad \theta = [X \mapsto h(\mathtt{a}), Y \mapsto \mathtt{b}] \cup \\
&\qquad\quad \cup \theta' = [Y \mapsto h(\mathtt{a}), X \mapsto \mathtt{b}]
\end{aligned}$$

Thus, l_1 C-overlaps with l_4 at position $\mathtt{C} = [\_]$ with set $S = \{\theta, \theta'\}$, where θ and θ' are both C-unifiers of l_1 and l_4 in this case.

The C-critical pairs of the rule $l_1 \to r_1$ with the rule $l_4 \to r_4$ at position $\mathtt{C} = [\_]$ are $\langle r_4\theta, \mathtt{C}\theta[r_1\theta]\rangle = \langle f(\mathtt{b},h(\mathtt{a})),\mathtt{a}\rangle$ and $\langle r_4\theta', \mathtt{C}\theta'[r_1\theta']\rangle = \langle f(h(\mathtt{a}),\mathtt{b}),\mathtt{a}\rangle$

$$\begin{tikzcd}
& l_4\theta \arrow[dl, "l_4 \to r_4"'] \arrow[dr, "l_1 \to r_1"] & & & l_4\theta' \arrow[dl, "l_4 \to r_4"'] \arrow[dr, "l_1 \to r_1"] & \\
f(\mathtt{b},h(\mathtt{a})) = r_4\theta & & \mathtt{C}\theta[r_1\theta] = \mathtt{a} & f(h(\mathtt{a}),\mathtt{b}) = r_4\theta' & & \mathtt{C}\theta'[r_1\theta'] = \mathtt{a}
\end{tikzcd}$$

b. $l_2 \; {}_?\overset{\mathsf{C}}{\approx}_? \; l_4$:

$$
\begin{array}{rl}
\{f(\mathtt{b},X) \; {}_?\overset{\mathsf{C}}{\approx}_? \; f(X',Y')\} \implies & \{\mathtt{b} \; {}_?\overset{\mathsf{C}}{\approx}_? \; X', X \; {}_?\overset{\mathsf{C}}{\approx}_? \; Y'\} \vee \{\mathtt{b} \; {}_?\overset{\mathsf{C}}{\approx}_? \; Y', X \; {}_?\overset{\mathsf{C}}{\approx}_? \; X'\} \\
\therefore \quad & \theta = [X' \mapsto \mathtt{b}, Y' \mapsto X] \cup \\
& \quad \cup \; \theta' = [Y' \mapsto \mathtt{b}, X' \mapsto X]
\end{array}
$$

Thus, l_2 C-overlaps with l_4 at position $\mathtt{C} = [\_]$ with set $S = \{\theta, \theta'\}$, where θ and θ' are both C-unifiers of l_2 and l_4 in this case.

The C-critical pairs of the rule $l_2 \to r_2$ with the rule $l_4 \to r_4$ at position $\mathtt{C} = [\_]$ are $\langle r_4\theta, \mathtt{C}\theta[r_2\theta] \rangle = \langle f(X,\mathtt{b}), X \rangle$ and $\langle r_4\theta', \mathtt{C}\theta'[r_2\theta'] \rangle = \langle f(\mathtt{b},X), X \rangle$.

$$\begin{tikzcd}
& l_4\theta \arrow[dl, "l_4 \to r_4"'] \arrow[dr, "l_2 \to r_2"] & & & l_4\theta' \arrow[dl, "l_4 \to r_4"'] \arrow[dr, "l_2 \to r_2"] & \\
f(X,\mathtt{b}) = r_4\theta & & \mathtt{C}\theta[r_2\theta] = X & f(\mathtt{b},X) = r_4\theta' & & \mathtt{C}\theta'[r_2\theta'] = X
\end{tikzcd}$$

Thus, $\mathit{CS}\mathsf{C}\mathit{CP}(\mathsf{R}/\mathsf{C}) = \{\langle f(\mathtt{b},h(\mathtt{a})),\mathtt{a}\rangle, \langle f(h(\mathtt{a}),\mathtt{b}),\mathtt{a}\rangle, \langle f(X,\mathtt{b}),X\rangle, \langle f(\mathtt{b},X),X\rangle\}$.

Now we can continue with the extension of other standard notations such as termination and confluence modulo an equational theory E:

We say that R is E-*terminating* or E-*noetherian* iff $\to_{\mathsf{R}/\mathsf{E}}$ is terminating. Also, R is said to be E-*confluent* iff for all terms t, t_1, t_2 such that $t \to^*_{\mathsf{R}/\mathsf{E}} t_1$ and $t \to^*_{\mathsf{R}/\mathsf{E}} t_2$, there exist t'_1, t'_2 such that $t_1 \to^*_{\mathsf{R}/\mathsf{E}} t'_1$, $t_2 \to^*_{\mathsf{R}/\mathsf{E}} t'_2$ and $t'_1 \approx_{\mathsf{E}} t'_2$. In the following we will consider such notions to the commutative theory, thus we will be interested on the cases of C-termination and C-confluence.

Example A.5 (Cont. Example A.3)**.** We now analyze the property of termination of $\mathsf{R} = \{f(h(\mathtt{a}),\mathtt{b}) \to \mathtt{a}, f(\mathtt{b},X) \to X, h(X) \to X\}$ and $\mathsf{C} = \{f(X,Y) \approx f(Y,X)\}$.

- R is C-terminating: It is easy to see the termination using an embedding into $(\mathbb{N}, >)$, which is known to terminate, checking that the length of the terms decrease in each R/C-reduction.

  More details about termination can be found in [9].

The problem with R/E is that, in general, R/E-reducibility is not decidable, because E-congruence classes may be infinite [17]. This is not the case for C, but for instance if we consider the theory $\mathsf{E} = \{X + 0 \approx 0\}$, the E-congruence class of an arbitrary term t would be $[t]_{\mathsf{E}} = \{t,\ t+0,\ t+0+0,\ t+0+0+0,\ \ldots\}$.

With that in mind, we see a need to refine this relation. A solution was given by [20]:

Definition A.4. (Relation $\rightarrow_{\mathsf{R,E}}$) A new relation $\rightarrow_{\mathsf{R,E}}$ is defined as following

$$s \rightarrow_{\mathsf{R,E}} t :\Leftrightarrow \exists (l \rightarrow r) \in \mathsf{R}, s \equiv \mathtt{C}[s'], \theta.\ s' \approx_{\mathsf{E}} l\theta \ \wedge\ t = \mathtt{C}[r\theta].$$

That way we can involve matching modulo $\approx_{\mathsf{E}}$ in each rewrite step, as noted in [23]. Whenever we cannot reduce a term t w.r.t $\rightarrow_{\mathsf{R,E}}$ we say that t in its R,E-*normal form*, $t{\downarrow}$. We call θ a R,E-*normalized substitution* iff for all $X \in \mathtt{dom}(\theta)$, $X\theta$ is R,E-irreducible.

With the aim of proving the completeness of $\rightarrow_{\mathsf{R,E}}$ with relation to $\rightarrow_{\mathsf{R/E}}$ we need the relation $\rightarrow_{\mathsf{R,E}}$ to be E-*coherent*:

The following result establishes necessary conditions for E-confluence and E-coherence of an ETRS:

Theorem A.1. (Theorem 2 in [16]) Let R be an ETRS such that:

1. R is E-noetherian;
2. $\approx_{\mathsf{E}}$ is decidable and for all $(l \approx r) \in \mathsf{E}$, $V(l) = V(r)$;
3. A complete and finite unification algorithm exists for the theory E.

Then R is E-*confluent* and $\rightarrow_{\mathsf{R,E}}$ is E-*coherent* if:

- any E-critical pair $\langle u_1, u_2 \rangle$ of $CS\mathsf{E}CP(\mathsf{R})$ satisfies $u_1{\downarrow} \approx_{\mathsf{E}} u_2{\downarrow}$ (we will say that u_1 and u_2 are E-joinable);
- any E-critical pair $\langle u_1 = r'\theta, u_2 \rangle$ of $CS\mathsf{E}CP(\mathsf{R/E})$ satisfies $u_1 \rightarrow_{\mathsf{R,E}} u_1'$ at some occurrence of r' and $u_1'{\downarrow} \approx_{\mathsf{E}} u_2{\downarrow}$.

The following example illustrates the Theorem A.1 at work:

Example A.6 (Cont. Example A.3)**.** We now analyze the property of C-confluence of $\mathsf{R} = \{f(h(\mathtt{a}),\mathtt{b}) \rightarrow \mathtt{a}, f(\mathtt{b},X) \rightarrow X, h(X) \rightarrow X\}$ and $\mathsf{C} = \{f(X,Y) \approx f(Y,X)\}$, and C-coherence of R,C. Since R is C-terminating, $\approx_{\mathsf{C}}$ is decidable, $V(l) = V(r)$ for all $l \rightarrow r \in \mathsf{R}$, we just need to check the last two conditions

- All C-critical pairs of $CS\mathsf{C}CP(\mathsf{R})$ are C-joinable:

  Since $CS\mathsf{C}CP(\mathsf{R}) = \{\langle \mathtt{a}, h(\mathtt{a})\rangle, \langle \mathtt{a}, f(\mathtt{a},\mathtt{b})\rangle, \langle h(\mathtt{a}), \mathtt{a}\rangle\}$, let's check that the terms of each of these pairs are C-joinable:

  - $\langle \mathtt{a}, h(\mathtt{a})\rangle$: We have $h(\mathtt{a}) \rightarrow_{\mathsf{R/C}} \mathtt{a}$ using the third rule, and the terms are C-joinable;
  - $\langle h(\mathtt{a}), \mathtt{a}\rangle$: Analogue to the item above;

- $\langle \mathtt{a}, f(\mathtt{a},\mathtt{b})\rangle$: We have $f(\mathtt{a},\mathtt{b}) \to_{\mathsf{R/C}} \mathtt{a}$ because $f(\mathtt{a},\mathtt{b}) \approx_{\mathsf{C}} f(\mathtt{b},\mathtt{a}) \to_{\mathsf{R}} \mathtt{a}$ using the second rule, and the terms are C-joinable.

- All C-critical pairs $\langle u_1 = r'\theta, u_2\rangle$ of $CSCCP(\mathsf{R/C})$ satisfy that $u_1 \to_{\mathsf{R,C}} u_1'$ at some occurrence of r' and $u_1'{\downarrow}$ and that $u_2{\downarrow}$ are C-joinable:

  We found $CSCCP(\mathsf{R/C}) = \{\langle f(\mathtt{b},h(\mathtt{a})),\mathtt{a}\rangle, \langle f(h(\mathtt{a}),\mathtt{b}),\mathtt{a}\rangle, \langle f(X,\mathtt{b}),X\rangle, \langle f(\mathtt{b},X),X\rangle\}$, from Example A.4. Let's check that $u_1'{\downarrow}$ and $u_2{\downarrow}$ are C-joinable, for each C-critical pair:

  - $\langle f(\mathtt{b},h(\mathtt{a})),\mathtt{a}\rangle$: First, we have $u_1 = f(\mathtt{b},h(\mathtt{a})) = r_4\theta \to_{\mathsf{R,C}} f(\mathtt{b},h(\mathtt{a})) = u_1'$, with $r_4 = f(Y,X)$ and $\theta = [Y \mapsto \mathtt{b}, X \mapsto h(\mathtt{a})]$, because

    $$f(\mathtt{b},h(\mathtt{a})) \approx_{\mathsf{C}} l_4\theta = f(X,Y)\theta = f(h(\mathtt{a}),\mathtt{b}).$$

    Since $f(\mathtt{b},h(\mathtt{a})) \to_{R_2} h(\mathtt{a}) \to_{R_3} \mathtt{a}$, we obtain $u_1'{\downarrow} = \mathtt{a}$, and as $u_2{\downarrow} = \mathtt{a}{\downarrow} = \mathtt{a}$, the result follows;

  - $\langle f(X,\mathtt{b}),X\rangle$: Initially, we have $u_1 = f(X,\mathtt{b}) = r_4\theta \to_{\mathsf{R,C}} f(X,\mathtt{b}) = u_1'$, where $r_4 = f(Y',X')$ and $\theta = [Y' \mapsto X, X' \mapsto \mathtt{b}]$, because

    $$f(X,\mathtt{b}) \approx_{\mathsf{C}} l_4\theta = f(X',Y')\theta = f(\mathtt{b},X).$$

    Since $f(X,\mathtt{b}) \approx_{\mathsf{C}} f(\mathtt{b},X) \to_{R_2} X$, we obtain that $u_1'{\downarrow} = X{\downarrow}$, and clearly $u_2{\downarrow} = X{\downarrow}$. Thus, the result follows;

  - For $\langle f(h(\mathtt{a}),\mathtt{b}),\mathtt{a}\rangle$ and $\langle f(\mathtt{b},X),\mathtt{a}\rangle$ the C-joinability is direct because both $f(\mathtt{b},X)$ and $f(h(\mathtt{a}),\mathtt{b})$ reduces to $\mathtt{a}$.

An interesting property of E-coherence is that R/E-reducibility coincides with R, E-reducibility, that is, we can compute R/E-normal forms using R, E-reductions. This result is due to Jouannaud et. al. with the following proposition:

Proposition A.1. (Proposition 1 in [17]) Assume R is E-confluent and E-noetherian. Then R, E- and R/E- normal forms of any term t are E-equal if and only if $\to_{\mathsf{R,E}}$ is E-coherent.

Proof. The proof of this result is out of the scope of this dissertation, and a proof can be found in [16]. □

Appendix B

Nominal Results and Examples

In this appendix we present nominal known results as confluence, overlaps and critical pairs, in order to understand some examples.

B.1 Nominal Confluence

Notice that atoms a and b stand for object-variables and their names should not matter when specifying a rewrite rule, in the same way that the choice of the names X and Y for the meta-level variables in Example 2.1 were not relevant. Since atoms are not affected by substitution action, we consider renaming of atoms in rewrite rules.

Call $R^{(a\ b)}$ the rule obtained by swapping a and b throughout R. A set of rewrite rules is called *equivariant* when it is closed under $(-)^{(a\ b)}$ for all atoms a and b. We use this technicality because atoms are not affected by substitution actions, but they can be swapped.

Definition B.1. (Equivariance closure) The *equivariant closure* of a set *Rw* of rewrite rules is the closure of *Rw* by the meta-action of permutations, that is, it is the set of all permutative variants of rules in *Rw*. We denote *eq-closure(Rw)* for the equivariant closure of *Rw*.

Notice that the equivariant closure of the NRS in Example 2.1 is the set of rules itself, i.e., *eq-closure(Rw)* $=$ *Rw*, since there are no atoms in these rules.

Example B.1. Consider the NRS with the single rule $R \equiv \emptyset \vdash f(b) \to a$. In order to find the *eq-closure(Rw)*, we need to analyze all the permutative variants of $R \in Rw$, they are $R^{(a\ b)}$, $R^{(a\ c)}$ and $R^{(b\ c)}$, where c is an arbitrary new atom.

$$R_1 = R^{(a\ b)} = \emptyset \vdash f(a) \to b$$

$$R_2 = R^{(a\ c)} = \emptyset \vdash f(b) \to c$$

$$R_3 = R^{(b\ c)} = \emptyset \vdash f(c) \to a$$

Therefore, *eq-closure(Rw)* $= \{R, R_1, R_2, R_3\}$.

A weaker property is known as *local confluence*, which is defined as “joinability of peaks”, as we will see in the next definition.

Definition B.2. (Peak and local confluence) Let R be an equivariant rewrite system, and let Δ, s, t_1 and t_2 such that $\Delta \vdash s \to t_1$ and $\Delta \vdash s \to t_2$. This pair will be denoted as $\Delta \vdash s \to t_1, t_2$ and called a *peak*. If there is such a peak, then we call a NRS *locally confluent* when there exists a term u such that $\Delta \vdash t_1 \to^* u$ and $\Delta \vdash t_2 \to^* u$. We say such a peak is *joinable*.

Here is an important remark made by [5, 12] to clarify resolve any doubt between equivariance and permuted variants of a set of rewrite rules.

Remark B.1. Since the definition of the rewriting relation generated by a rewrite theory $\mathsf{R} = (\Sigma, Rw)$ takes into account permuted variant of rules (via the use of the permutation π in the one-step rewrite relation, see Definition 2.1), it is not necessary to include permuted variant of rules in Rw. For convenience, in the rest of this work we assume that for any $R \in Rw$, if R and R^π are both in Rw then $\pi = \mathtt{Id}$; in other words, Rw does not contain permuted variants of the same rule.

With nominal rules the nominal rewrite relation is generated by the equivariant closure of a set of rules. Thus, in order to find overlaps and critical pairs, we must consider permuted variant of rules, and use nominal unification instead of first-order unification [5, 12].

Definition B.3. (Overlaps and critical pairs) Suppose

1. $R_1 = \nabla_1 \vdash l_1 \to r_1$ and $R_2 = \nabla_2 \vdash l_2 \to r_2$ are copies of two rules in *eq-closure*(Rw) such that $V(R_1) \cap V(R_2) = \emptyset$[1];

2. $l_1 \equiv \mathtt{C}[l_1']$ such that $\{\nabla_1, \nabla_2, l_1' \ {}_?{\approx}_? \ l_2\}$ has a principal solution (Γ, θ), so that $\Gamma \vdash l_1'\theta \approx_\alpha l_2\theta$ and $\Gamma \vdash \nabla_i\theta$ for $i = 1, 2$.

We say R_1 *overlaps* with R_2, and we call then the pair of terms-in-context $\Gamma \vdash \langle r_1\theta, \mathtt{C}\theta[r_2\theta] \rangle$ a *critical pair*. We say the critical pair is *trivial* if $\mathtt{C} = [\_]$ and R_1, R_2 are copies of the same rule, or if l_1' is a variable.

$$\begin{tikzcd}
 & \Gamma \vdash l_1\theta \arrow[dl, "R_1"'] \arrow[dr, "R_2"] & \\
\Gamma \vdash r_1\theta & & \Gamma \vdash \mathtt{C}\theta[r_2\theta]
\end{tikzcd}$$

[1] R_1 and R_2 could be copies of the same rule.

Example B.2. Consider the set of rules

$$Rw = \begin{cases} \nabla_1 \vdash l_1 \to r_1 : & \emptyset \vdash f(h(a),b) \to a \\ \nabla_2 \vdash l_2 \to r_2 : & \emptyset \vdash f(b,X) \to X \\ \nabla_3 \vdash l_3 \to r_3 : & a\#X \vdash h(X) \to X \end{cases}$$

where a,b are atoms and f,h are function symbols.

1. $R_1 = \emptyset \vdash f(h(a),b) \to a$ and $R_3^\pi = a'\#X' \vdash h(X') \to X'$, with $\pi = (a\ a')$ and a' a new atom, are copies of two rules in *eq-closure*(Rw) such that $V(R_1) \cap V(R_3^\pi) = \emptyset$;

2. If $f(h(a),b) \equiv \mathtt{C}[h(a)]$, let's check if $\{\emptyset, a'\#X', h(a)\ {}_?{\approx}_?\ h(X')\}$ has a principal solution (Γ,θ):

$$\begin{array}{rcl} \{\emptyset, a'\#X', h(a)\ {}_?{\approx}_?\ h(X')\} & \Longrightarrow & \{\emptyset, a'\#X', a\ {}_?{\approx}_?\ X'\} \\ & \overset{X'\mapsto a}{\Longrightarrow} & \{\emptyset, a'\#a\} = \{\ \} \\ & \therefore & (\Gamma,\theta) = (\emptyset, [X' \mapsto a]) \end{array}$$

   Since $\emptyset \vdash (a'\#X')\theta$ holds, the pair (Γ,θ) is indeed a solution for the unification problem $\{\emptyset, a'\#X', h(a)\ {}_?{\approx}_?\ h(X')\}$.

   Thus, l_1 overlaps with l_3^π at position $\mathtt{C} = f([\_],b)$ with $\theta = [X' \mapsto a]$.

   The critical pair is $\vdash \langle r_1\theta, \mathtt{C}\theta[r_3^\pi\theta]\rangle = \langle a, f(a,b)\rangle$.

$$\begin{tikzcd} & \emptyset \vdash l_1\theta \arrow[dl, "l_1 \to r_1"'] \arrow[dr, "l_3^\pi \to r_3^\pi"] & \\ \emptyset \vdash a = r_1\theta & & \emptyset \vdash \mathtt{C}\theta[r_3^\pi\theta] = f(a,b) \end{tikzcd}$$

Definition B.4. (Permutative overlaps and critical pairs) Let $R_1 = \nabla_1 \vdash l_1 \to r_1$ and $R_2 = \nabla_2 \vdash l_2 \to r_2$ be copies of two rewrite rules in *eq-closure(Rw)* such that there is an overlap. If R_2 is a copy of R_1^π, we say that the overlap is *permutative*. A permutative overlap at the root position is called *root-permutative*. We call an overlap that is not trivial and not root-permutative *proper*. The same terminology is used to classify critical pairs.

A permutative overlap is an indication that there exists a critical pair generated by a rule an one of its permuted variants.

Example B.3. (Cont. Example B.2) Consider the same set of rules from the previous example.

1. $R_3 = a\#X' \vdash h(X') \to X'$ and $R_3^\pi = a'\#Y \vdash h(Y) \to Y$ are copies of two rewrite rules in *eq-closure(Rw)*, with $\pi = (a\ a')$ and a' a new atom, such that $V(R_3) \cap V(R_3^\pi) = \emptyset$.

2. If we consider the root position of $h(X')$, that is $h(X') \equiv \mathtt{C}[h(X')]$, let's check if $\{a\#X', a'\#Y, h(X')\ {}_?\!\approx_?\ h(Y)\}$ has a principal solution (Γ, θ):

$$\begin{array}{rcl} \{a\#X', a'\#Y, h(X')\ {}_?\!\approx_?\ h(Y)\} & \Longrightarrow & \{a\#X', a'\#Y, X'\ {}_?\!\approx_?\ Y\} \\ & \overset{X' \mapsto Y}{\Longrightarrow} & \{a\#Y, a'\#Y\} \\ & \therefore & (\Gamma, \theta) = (\{a\#Y, a'\#Y\}, [X' \mapsto Y]) \end{array}$$

   Since $a\#Y, a'\#Y \vdash (a\#X')\theta$ and $a\#Y, a'\#Y \vdash (a'\#Y)\theta$, the pair (Γ, θ) is indeed a solution for $\{a\#X', a'\#Y, h(X')\ {}_?\!\approx_?\ h(Y)\}$.

   Thus, l_3 overlaps with l_3^π at position $\mathtt{C} = [\_]$ with $\theta = [X' \mapsto Y]$.

   The critical pair is $a\#Y, a'\#Y \vdash \langle r_3\theta, \mathtt{C}\theta[r_3^\pi\theta]\rangle = \langle Y, Y\rangle$.

$$\begin{tikzcd} & \Gamma \vdash l_3\theta \arrow[dl, "l_3 \to r_3"'] \arrow[dr, "l_3^\pi \to r_3^\pi"] & \\ \Gamma \vdash Y = r_3\theta & & \Gamma \vdash \mathtt{C}\theta[r_3^\pi\theta] = Y \end{tikzcd}$$

Observe that R_3^π is a copy of permuted R_3. According to Definition B.4, the overlap is permutative. Even more, since the overlap occurs at the root position, the overlap is root-permutative. Therefore, $a\#Y, a'\#Y \vdash \langle Y, Y\rangle$ is a root-permutative critical pair.

Example B.4. (Cont. Example B.3) Let's see an example of a proper critical pair. Consider the same set of rules from Example B.2.

1. $R_1^\pi = f(h(c), d) \to c$ and $R_3 = h(X') \to X'$, with $\pi = (a\ c)(b\ d)$ and c, d new atoms, are copies of two rules in *eq-closure(Rw)* such that $V(R_1^\pi) \cap V(R_3) = \emptyset$.

2. If $f(h(c), d) \equiv \mathtt{C}[h(c)]$, let's check if $\{\emptyset, a\#X', h(c)\ {}_?\!\approx_?\ h(X')\}$ has a principal solution (Γ, θ):

$$\begin{array}{rcl} \{\emptyset, a\#X', h(c)\ {}_?\!\approx_?\ h(X')\} & \Longrightarrow & \{\emptyset, a\#X', c\ {}_?\!\approx_?\ X'\} \\ & \overset{X' \mapsto c}{\Longrightarrow} & \{\emptyset, a\#c\} = \{\ \} \\ & \therefore & (\Gamma, \theta) = (\emptyset, [X' \mapsto c]) \end{array}$$

   Since we have $\emptyset \vdash \emptyset\theta$ and $\emptyset \vdash (a\#X')\theta = a\#c$, we get that (Γ, θ) is indeed a solution for $\{\emptyset, a\#X', h(c)\ {}_?\!\approx_?\ h(X')\}$.

Thus, l_1^π overlaps with l_3 at position $\mathtt{C} = f([\_],d)$ with $\theta = [X' \mapsto c]$.

The critical pair is $\vdash \langle r_1^\pi\theta, \mathtt{C}\theta[r_3\theta]\rangle = \langle c, f(c,d)\rangle$.

$$\begin{tikzcd}
& \emptyset \vdash l_1^\pi\theta \arrow[dl, "l_1^\pi \to r_1^\pi"'] \arrow[dr, "l_3 \to r_3"] & \\
\emptyset \vdash c = r_1^\pi\theta & & \emptyset \vdash \mathtt{C}\theta[r_3\theta] = f(c,d)
\end{tikzcd}$$

Since this critical pair is neither trivial, nor root-permutative, Definition B.4 gives us that $\emptyset \vdash \langle c, f(c,d)\rangle$ is a proper critical pair.

Lemma B.1. It is not necessarily the case that trivial critical pairs are joinable.

Proof. We give a counterexample. Consider the rules

$$R_1 \equiv \emptyset \vdash f(b) \to a \qquad \text{and} \qquad R_2 \equiv a\#X \vdash X \to [a]X.$$

They have a trivial critical pair $\vdash \langle a, [a]f(b)\rangle$.

$$\begin{tikzcd}
& \vdash f(b) \arrow[dl, "R_1"'] \arrow[dr, "R_2"] & \\
\vdash a & & \vdash [a]f(b)
\end{tikzcd}$$

It is clear that these terms are not joinable. □

Observe that overlaps at the root between variable-renamed versions of first-order rules can be discarded, because they generate equal terms. Meanwhile, in nominal rewriting we must also consider overlaps at the root between permuted variants of rules. The following example shows that they do not necessarily produce the same result:

Example B.5. Consider $R = (\vdash f(X) \to f([a]X))$ a rule of R. There is an overlap at the root between this rule and its variant $R^{(a\ b)} = (\vdash f(X) \to f([b]X))$. It generates the critical pair $\vdash \langle f([a]X), f([b]X)\rangle$. As we can see, the terms $f([a]X)$ and $f([b]X)$ are not necessarily α-equivalent. Therefore, R is not confluent.

$$\begin{tikzcd}
& \vdash f(a) \arrow[dl, "{[\_],R,[X\mapsto a],\mathtt{Id}}"'] \arrow[dr, "{[\_],R,[X\mapsto a],(a\ b)}"] & \\
\vdash f([a]a) & \not\approx_\alpha & \vdash f([b]a)
\end{tikzcd}$$

Another property that will be important is called ‘uniformity’ which means than an atom that does not occur free in a term cannot become free after the application of an ‘uniform’ rule. Formally,

Definition B.5. (Uniformity) A rule R is called *uniform* when for all Δ, s and t such that $\Delta \vdash s \rightarrow_R t$ we have $\Delta, \langle a\#s\rangle_{nf} \vdash a\#t$ for any a such that $\langle a\#s\rangle_{nf}$ is consistent.

To check uniformity of $R = l \rightarrow r$ it is enough to check that l and r satisfy the condition of the definition above. Intuitively, if a is not free in l then a is not free in r.

Example B.6. Consider the NRS

$$\mathsf{R} = \{\emptyset \vdash f(h(a),b) \rightarrow a,\ \emptyset \vdash f(b,X) \rightarrow X,\ a\#X \vdash h(X) \rightarrow X\},$$

where a, b are atoms and f, h are function symbols.

By definition, R is uniform, since all of its rules are uniform. In fact:

- For the first two rules, the result follows trivially, since they have an empty context.
- For the last rule, it holds that $a\#X, \langle c\#h(X)\rangle_{nf} \vdash c\#X$ for any c such that $\langle c\#h(X)\rangle_{nf}$ is consistent.

Remark B.2. In [12], at Theorem 62, a version of the critical pair lemma was considered: "If all non-trivial critical pairs of a uniform nominal rewrite system are joinable, then it is locally confluent". However, it was observed in another work [5], as we defined a different kind of critical pair, the permutative one, joinability of proper critical pairs is insufficient for local confluence, even for a uniform theory, see for example the rule in Example B.5, which is uniform.

Fortunately, an additional condition allows us to prove that uniform theories with joinable proper critical pairs are locally confluent. Let’s state the definition of α-stability from [22].

Definition B.6. (α-stability) A rewrite rule $R = \nabla \vdash l \rightarrow r$ is *α-stable* when, for all $\Delta, \pi, \theta, \theta'$,

$$\Delta \vdash (\nabla\theta,\ \nabla^{\pi}\theta',\ l\theta \approx_\alpha l^{\pi}\theta') \ \text{ implies } \ \Delta \vdash r\theta \approx_\alpha r^{\pi}\theta'.$$

We say a rewrite theory $\mathsf{R} = (\Sigma, Rw)$ is α-stable if every rule in Rw is α-stable.

Example B.7. (Cont. Example B.6) Let's see that all of the rules in the NRS of Example B.6 are α-stable.

- $\emptyset \vdash f(h(a),b) \to a$ is α-stable: for all Δ, π, θ, θ' such that $\Delta \vdash (f(h(a),b))\theta \approx_\alpha (f(h(a),b))^\pi\theta'$, we will have $\Delta \vdash a\theta \approx_\alpha a^\pi\theta'$ since substitutions do not act over atoms and in this case $\pi = \mathtt{Id}$ in order to get $a \approx_\alpha a^\pi$.
- $\emptyset \vdash f(b,X) \to X$ is α-stable: for all Δ, π, θ, θ' such that $\Delta \vdash f(b,X\theta) = (f(b,X))\theta \approx_\alpha (f(b,X))^\pi\theta' = f(b^\pi,X^\pi\theta')$, we will have $\Delta \vdash X\theta \approx_\alpha X^\pi\theta'$ since in this case we have $\pi = \mathtt{Id}$ because we need b to be α-equivalent to b^π and therefore $X\theta \approx_\alpha X\theta'$ holds directly from hypothesis.
- $a\#X \vdash h(X) \to X$ is α-stable: for all Δ, π, θ, θ' such that $\Delta \vdash h(X\theta) = (h(X))\theta \approx_\alpha (h(X))^\pi\theta' = h(X^\pi\theta')$, we will have $\Delta \vdash X\theta \approx_\alpha X^\pi\theta'$ directly since permutations actions do not act over variables.

Now we can give a new version for the Critical Pair Lemma:

Theorem B.1. (Critical Pair Lemma for uniform α-stable theories) Let $\mathsf{R} = (\Sigma, Rw)$ be an uniform rewrite theory where all the rewrite rules in Rw are α-stable. If every proper critical pair is joinable, then R is locally confluent.

Proof. The proof of this result is out of the scope of this work, but it can be found in [5], Theorem 4.6. □

Definition B.7. (Termination) A NRS is *terminating* if all the rewrite sequences are finite.

Now, with the previous definitions, and using Newman's Lemma [19] we get the following result of confluence:

Corollary B.1. Consider a NRS R.

1. If R is terminating, uniform, α-stable and proper critical pairs are joinable, then it is confluent.
2. Under the same assumptions, normal forms are unique modulo $\approx_\alpha$.

The analysis of critical pairs in the next example, and in the following ones in this work, will be made by means of tables. The first column shows the position by means of its context $\mathtt{C}$ of a term l_i and the second column shows the subterm l_i' given by the respective position to the first column. The remaining columns will represent all the terms l_j, renamed, of a set of rules that we want to unify with l_i (and its non-variable subterms).

- l_i^π stands for l_i with the meta-action of $\pi \neq \mathtt{Id}$, see Definition 1.3. Here we rename all atoms in l_i.
- We write $\oslash$ to illustrate that the unification between l_i and a copy of itself does not need to be made. In the first case, we do not need to unify l_1 and a renamed l_1.
- The symbol $\perp$ means that we couldn't unify the respective terms, or even if we could the solution substitution θ did not respect $\Gamma \vdash \nabla_i\theta$, according to Definition B.3.

Example B.8. Consider the NRS

$$\mathsf{R} = \begin{cases} \nabla_1 \vdash l_1 \to r_1 : & \emptyset \vdash f(h(a),b) \to a \\ \nabla_2 \vdash l_2 \to r_2 : & \emptyset \vdash f(b,X) \to X \\ \nabla_3 \vdash l_3 \to r_3 : & a\#X \vdash h(X) \to X \end{cases}$$

where a,b are atoms and f,h are function symbols. Let's analyze the critical pairs of R and check if it is confluent.

1) R_1 overlaps with R_i, for $i = 1,2,3$:

$\mathtt{C}$	$[\_]$	l_1	l_1^π	l_2	l_2^π	l_3	l_3^π
$[\_]$	$f(h(a),b)$	$\oslash$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$f([\_],b)$	$h(a)$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\vdash \langle a, f(a,b)\rangle$
$f(h(a),[\_])$	b	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$f(h([\_]),b)$	a	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

Below we present only the interesting cases.

a. $l_1 \ {}_?{\approx}_? \ l_1^\pi$: Remember $l_1 = f(h(a),b)$.

Notice that we need to analyze all possible meta-actions of π: $(a\ b)$, $(a\ a')$. $(b\ b')$ and $(a\ a')(b\ b')$, where a',b' are new atoms.

Below we only show the case for $\pi = (a\ a')(b\ b')$. But the other cases will also result in $\perp$.

$$\begin{aligned} \{\emptyset, \emptyset, f(h(a),b) \ {}_?{\approx}_? \ f(h(a'),b')\} &\implies \{\emptyset, \emptyset, h(a) \ {}_?{\approx}_? \ h(a'), b \ {}_?{\approx}_? \ b'\} \\ &\implies \perp \end{aligned}$$

By definition of α-equivalence, all atoms are different, that is, $b \not\approx_\alpha b'$.

b. $h(a) = l'_1 \ {}_?{\approx}_? \ l_3^\pi$: Remember $l_1 = \mathtt{C}[l'_1]$, here $\mathtt{C} \equiv f([\_],b)$, and $l_3 = h(X')$. Here $\pi = (a\ a')$, with a' a new name. Observe that $l_3^\pi = l_3$, but $\nabla_3^\pi = a'\#X'$. This case was already made in Example B.2, and its critical pair is $\emptyset \vdash \langle a, f(a,b) \rangle$.

c. $l_1 =^? l_2$: Remember $l_1 = f(h(a),b)$ and $l_2 = f(b,X')$.

$$\begin{array}{rcl} \{\emptyset,\emptyset, f(h(a),b) \ {}_?{\approx}_? \ f(b,X')\} & \Longrightarrow & \{\emptyset,\emptyset, h(a) \ {}_?{\approx}_? \ b, b \ {}_?{\approx}_? \ X'\} \\ & \Longrightarrow & \perp \end{array}$$

We get a $\perp$, because we cannot unify an atom with a function symbol.

d. $l_1 =^? l_2^\pi$: Remember $l_1 = f(h(a),b)$ and $l_2 = f(b,X')$. Here $\pi = (b\ b')$ or $\pi = (b\ a)$, with b' a new atom. Either way, the result is the same. We present the case $\pi = (b\ b')$.

$$\begin{array}{rcl} \{\emptyset,\emptyset, f(h(a),b) \ {}_?{\approx}_? \ f(b',X')\} & \Longrightarrow & \{\emptyset,\emptyset, h(a) \ {}_?{\approx}_? \ b, b' \ {}_?{\approx}_? \ X'\} \\ & \Longrightarrow & \perp \end{array}$$

We get a $\perp$, because we cannot unify an atom with a function symbol.

e. $h(a) = l'_1 =^? l_3$: Remember $l_1 = \mathtt{C}[l'_1]$, $\mathtt{C} \equiv f([\_],b)$, $l_3 = h(X')$.

$$\begin{array}{rcl} \{\emptyset, a\#X', h(a) \ {}_?{\approx}_? \ h(X')\} & \Longrightarrow & \{\emptyset, a\#X', a \ {}_?{\approx}_? \ X'\} \\ & \overset{X' \mapsto a}{\Longrightarrow} & \{\emptyset, a\#a\} \\ & \Longrightarrow & \perp \end{array}$$

Applying the possible solution substitution in the constraint $a\#X'$ we get an inconsistency, thus we get a $\perp$.

All the remaining cases result in $\perp$ because we try to unify two different function symbols or a function symbol with an atom.

2) R_2 overlaps with R_i, for $i = 1,2,3$:

The table obtained is the following. All the checks result in $\perp$ or $\oslash$. The analysis is similar to the previous.

$\mathtt{C}$	$[\_]$	l_1	l_1^π	l_2	l_2^π	l_3	l_3^π
$[\_]$	$f(b,X')$	$\perp$	$\perp$	$\oslash$	$\perp$	$\perp$	$\perp$
$f([\_],X')$	b	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

a. $l_2 =^? l_1$: Remember $l_2 = f(b,X')$ and $l_1 = f(h(a),b)$.

$$\begin{aligned} \{\emptyset,\emptyset, f(b,X') \,{}_?{\approx}_?\, f(h(a),b)\} &\Longrightarrow \{\emptyset,\emptyset, b \,{}_?{\approx}_?\, h(a), X' \,{}_?{\approx}_?\, b\} \\ &\Longrightarrow \perp \end{aligned}$$

An atom does not unify with a function symbol.

b. $l_2 =^? l_1^\pi$: Remember $l_2 = f(b,X')$ and $l_1 = f(h(a),b)$.

Again, we need to check all the possibilities for π: $(a\ b)$, $(a\ a')$, $(b\ b')$ and $(a\ a')(b\ b')$, with a', b' new names. We will only show $\pi = (a\ b)$, but the result is the same for all cases.

$$\begin{aligned} \{\emptyset,\emptyset, f(b,X') \,{}_?{\approx}_?\, f(h(b),a)\} &\Longrightarrow \{\emptyset,\emptyset, b \,{}_?{\approx}_?\, h(b), X' \,{}_?{\approx}_?\, a\} \\ &\Longrightarrow \perp \end{aligned}$$

An atom does not unify with a function symbol.

c. $l_2 =^? l_2^\pi$: Remember $l_2 = f(b,X')$.

The only π we need to take into account is $(b\ b')$, where b' is a new atom. Also note that $V(l_2) \cap V(l_2^\pi) = \emptyset$ is necessary, so we change the name of the unknown in l_2^π to X''.

$$\begin{aligned} \{\emptyset,\emptyset, f(b,X') \,{}_?{\approx}_?\, f(b',X'')\} &\Longrightarrow \{\emptyset,\emptyset, b \,{}_?{\approx}_?\, b', X' \,{}_?{\approx}_?\, X''\} \\ &\Longrightarrow \perp \end{aligned}$$

By definition of α-equivalence, all atoms are different, that is, $b \not\approx_\alpha b'$.

All the remaining cases result in $\perp$ because we try to unify two different function symbols or a function symbol with an atom.

3) R_3 overlaps with R_i, for $i = 1,2,3$:

C	$[\_]$	l_1	l_1^π	l_2	l_2^π	l_3	l_3^π
$[\_]$	$h(X')$	$\perp$	$\perp$	$\perp$	$\perp$	$\oslash$	$a\#Y, a'\#Y \vdash \langle Y,Y \rangle$

a. $l_3 \,{}_?{\approx}_?\, l_3^\pi$: Remember $l_3 = h(X')$.

We just need to check when $\pi = (a\ a')$, with a' a new atom. Notice that the condition $V(l_3) \cap V(l_3^\pi) = \emptyset$ is necessary, so we change the name of the unknown in l_3^π to Y. Also note that we have $\nabla_3^\pi = \{a'\#Y\}$.

This case was already made in Example B.3, and its critical pair is $a\#Y, a'\#Y \vdash \langle Y,Y \rangle$.

All the remaining cases result in $\perp$ because we try to unify two different function symbols.

4) R_1^π overlaps with R_i, for $i = 1,2,3$:
As seen earlier, π may be $(a\ b)$, $(a\ c)$, $(b\ d)$ or $(a\ c)(b\ d)$, with c,d new names. Of course, with these conditions, we would need to build four different tables. As this work is exhaustive, we will show only the most general case, $\pi = (a\ c)(b\ d)$.

$\mathtt{C}$	$[\_]$	l_1	$l_1^{\pi'}$	l_2	$l_2^{\pi'}$	l_3	$l_3^{\pi'}$
$[\_]$	$f(h(c),d)$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$f([\_],d)$	$h(c)$	$\perp$	$\perp$	$\perp$	$\perp$	$\vdash \langle c, f(c,d)\rangle$	$\vdash \langle c, f(c,d)\rangle$
$f(h(c),[\_])$	d	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$f(h([\_]),d)$	c	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

a. $h(c) = l_1'^{\pi}\ {}_?{\approx}_?\ l_3$: Remember $l_1^\pi = \mathtt{C}[l_1'^{\pi}]$, here $\mathtt{C} \equiv f([\_],d)$, and $l_3 = h(X')$.
Observe that $\nabla_1^\pi = \emptyset$ and $\nabla_3 = \{a\#X'\}$.
This case was already made in Example B.4, and its critical pair is $\emptyset \vdash \langle c, f(c,d)\rangle$.

b. $h(c) = l_1'^{\pi}\ {}_?{\approx}_?\ l_3^{\pi'}$: Remember $l_1^\pi = \mathtt{C}[l_1'^{\pi}]$, here $\mathtt{C} \equiv f([\_],d)$, and $l_3 = h(X')$.
Here $\pi' = (a\ c)$ or $\pi' = (a\ a')$, with a' a new name.

$*$ If $\pi' = (a\ c)$:

$$\begin{aligned} \{\emptyset, c\#X', h(c)\ {}_?{\approx}_?\ h(X')\} &\Longrightarrow \{\emptyset, c\#X', c\ {}_?{\approx}_?\ X'\} \\ &\overset{X'\mapsto c}{\Longrightarrow} \{\emptyset, c\#c\} \\ &\Longrightarrow \perp \end{aligned}$$

$*$ If $\pi' = (a\ a')$:

$$\begin{aligned} \{\emptyset, a'\#X', h(c)\ {}_?{\approx}_?\ h(X')\} &\Longrightarrow \{\emptyset, a'\#X', c\ {}_?{\approx}_?\ X'\} \\ &\overset{X'\mapsto c}{\Longrightarrow} \{\emptyset, a'\#c\} = \{\ \} \\ &\therefore\quad (\Gamma, \theta) = (\emptyset, [X' \mapsto c]) \end{aligned}$$

Since we have $\emptyset \vdash \emptyset\theta$ and $\emptyset \vdash (a\#X')\theta = a'\#c$, the pair (Γ, θ) is indeed a solution for $\{\emptyset, a'\#X', h(c)\ {}_?{\approx}_?\ h(X')\}$.
Thus, l_1^π overlaps with $l_3^{\pi'}$ at position $\mathtt{C} = f([\_],d)$ with $\theta = [X' \mapsto c]$.
The critical pair is $\vdash \langle r_1^\pi\theta, \mathtt{C}\theta[r_3^{\pi'}\theta]\rangle = \langle c, f(c,d)\rangle$.

$$\begin{tikzcd}
 & \vdash l_1^{\pi}\theta \arrow[dl, "l_1^{\pi} \rightarrow r_1^{\pi}"'] \arrow[dr, "l_3^{\pi'} \rightarrow r_3^{\pi'}"] & \\
\vdash c = r_1^{\pi}\theta & & \vdash \mathtt{C}\theta[r_3\theta] = f(c,d)
\end{tikzcd}$$

5) R_2^{π} overlaps with R_i, for $i = 1,2,3$: Here π might be $(b\ a)$ or $(b\ d)$, where d is a new atom, consequently generating two tables. The table below shows the case where $\pi = (b\ d)$.

$\mathtt{C}$	$[\_]$	l_1	$l_1^{\pi'}$	l_2	$l_2^{\pi'}$	l_3	$l_3^{\pi'}$
$[\_]$	$f(d,X)$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$f([\_],X)$	d	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

6) R_3^{π} overlaps with R_i, for $i = 1,2,3$: Here we have $(a\ b)$ or $(a\ c)$, with c a new name, as options for π. Again, we need to build two tables, but we are only going to show one, when $\pi = (a\ c)$.

$\mathtt{C}$	$[\_]$	l_1	$l_1^{\pi'}$	l_2	$l_2^{\pi'}$	l_3	$l_3^{\pi'}$
$[\_]$	$h(X')$	$\perp$	$\perp$	$\perp$	$\perp$	$c\#Y, a\#Y \vdash \langle Y,Y \rangle$	$c\#Y, a'\#Y \vdash \langle Y,Y \rangle$

a. $l_3^{\pi}\ {}_?{\approx}_?\ l_3$: Remember $l_3 = h(X')$.

Notice that the condition $V(l_3^{\pi}) \cap V(l_3) = \emptyset$ is necessary, so we change the name of the unknown in l_3 to Y. Also note that we have $\nabla_3^{\pi} = \{c\#X'\}$ and $\nabla_3 = \{a\#Y\}$.

$$\begin{array}{rcl}
\{c\#X', a\#Y, h(X')\ {}_?{\approx}_?\ h(Y)\} & \Longrightarrow & \{c\#X', a\#Y, X'\ {}_?{\approx}_?\ Y\} \\
 & \overset{X' \mapsto Y}{\Longrightarrow} & \{c\#Y, a\#Y\} \\
 & \therefore & (\Gamma, \theta) = (\{c\#Y, a\#Y\}, [X' \mapsto Y])
\end{array}$$

Since $c\#Y, a\#Y \vdash (c\#X')\theta$ and $c\#Y, a\#Y \vdash (a\#Y)\theta$, the pair (Γ, θ) is indeed a solution for $\{c\#X', a\#Y, h(X')\ {}_?{\approx}_?\ h(Y)\}$.

Thus, l_3^{π} overlaps with l_3 at position $\mathtt{C} = [\_]$ with $\theta = [X' \mapsto Y]$.

The critical pair is $c\#Y, a\#Y \vdash \langle r_3^{\pi}\theta, \mathtt{C}\theta[r_3\theta] \rangle = \langle Y, Y \rangle$.

$$\begin{tikzcd}
& \Gamma \vdash l_3^{\pi}\theta \arrow[dl, "l_3^{\pi} \to r_3^{\pi}"'] \arrow[dr, "l_3 \to r_3"] & \\
\Gamma \vdash Y = r_3^{\pi}\theta & & \Gamma \vdash \mathtt{C}\theta[r_3\theta] = Y
\end{tikzcd}$$

b. $l_3^{\pi} \;{}_?{\approx}_?\; l_3^{\pi'}$: Remember $l_3 = h(X')$.

Observe we do not need to do the case $\pi' = (a\ c)$, because we do not want to unify a term with a copy of itself. Thus, $\pi' = (a\ a')$, where a' is a new atom. Again $V(l_3^{\pi}) \cap V(l_3^{\pi'}) = \emptyset$ is necessary, so we change the name of the unknown in $l_3^{\pi'}$ to Y. Note that $\nabla_3^{\pi} = c\#X'$ and $\nabla_3^{\pi'} = a'\#Y$.

$$\begin{array}{rcl}
\{c\#X', a'\#Y, h(X') \;{}_?{\approx}_?\; h(Y)\} & \Longrightarrow & \{c\#X', a'\#Y, X' \;{}_?{\approx}_?\; Y\} \\
& \overset{X'\mapsto Y}{\Longrightarrow} & \{c\#Y, a'\#Y\} \\
& \therefore & (\Gamma,\theta) = (\{c\#Y, a'\#Y\}, [X' \mapsto Y])
\end{array}$$

Since $c\#Y, a'\#Y \vdash (c\#X')\theta$ and $c\#Y, a'\#Y \vdash (a'\#Y)\theta$, the pair (Γ,θ) is indeed a solution for $\{c\#X', a'\#Y, h(X') \;{}_?{\approx}_?\; h(Y)\}$.

Thus, l_3^{π} overlaps with $l_3^{\pi'}$ at position $\mathtt{C} = [\_]$ with $\theta = [X' \mapsto Y]$.

The critical pair is $c\#Y, a'\#Y \vdash \langle r_3^{\pi}\theta, \mathtt{C}\theta[r_3^{\pi'}\theta]\rangle = \langle Y, Y\rangle$.

$$\begin{tikzcd}
& \Gamma \vdash l_3^{\pi}\theta \arrow[dl, "l_3^{\pi} \to r_3^{\pi}"'] \arrow[dr, "l_3^{\pi'} \to r_3^{\pi'}"] & \\
\Gamma \vdash Y = r_3^{\pi}\theta & & \Gamma \vdash \mathtt{C}\theta[r_3^{\pi'}\theta] = Y
\end{tikzcd}$$

Therefore, the presented critical pairs of R are:

$$\{\vdash \langle a, f(a,b)\rangle;\ a\#Y, a'\#Y \vdash \langle Y,Y\rangle;\ \vdash \langle c, f(c,d)\rangle;\ c\#Y, a\#Y \vdash \langle Y,Y\rangle;\ c\#Y, a'\#Y \vdash \langle Y,Y\rangle\}.$$

It is easy to see that R is terminating using an embedding into $(\mathbb{N},>)$, which is known to terminate, checking that the length of the terms decrease in each R-reduction. More details about termination can be found in [9].

From Example B.6, R is uniform and, in Example B.7, we showed that the rules are α-stable.

Corollary B.1 gives us that R would be confluent if all the proper critical pairs are joinable. But picking out the first critical pair found $\vdash \langle a, f(a,b)\rangle$, we see it is proper and it is not joinable. Indeed, we cannot reduce neither a nor $f(a,b)$ using the three rules in R, hence $a \not\Downarrow f(a,b)$.

Lastly, we conclude that R is not confluent.

B.2 Commutative

This example was used before we defined nominal rewriting modulo an equational theory, as an attempt to identify the notions and results that needed to be extended.

Example B.9. Consider the nominal theory

$$\mathsf{T} = \{\emptyset \vdash f(h(a),b) \approx_\alpha a,\ \emptyset \vdash f(b,X) \approx_\alpha X,\ a\#X \vdash h(X) \approx_\alpha X,\ \emptyset \vdash f(X,Y) \approx_\alpha f(Y,X)\},$$

where a,b are atoms and f,h are function symbols. Similarly to the previous appendix, we will decompose T in $\mathsf{R}\cup\mathsf{C}$ where

- $\mathsf{R} = \{\vdash f(h(a),b) \to a,\ \vdash f(b,X) \to X,\ a\#X \vdash h(X) \to X\}$;
- and $\mathsf{C} = \{\vdash f(X,Y) \approx_\alpha f(Y,X)\}$.

The idea is to analyze the equivariance, closedness and C-confluence of such R.

Equivariance We saw in Example B.8 that all rules were in *eq-closure*(Rw). Just to state them all: let c,d be arbitrary atoms, we have

$$\mathsf{R}^{(a\ b)} = \{\vdash f(h(b),a) \to b,\ \vdash f(a,X) \to X,\ b\#X \vdash h(X) \to X\}$$

$$\mathsf{R}^{(a\ c)} = \{\vdash f(h(c),b) \to c,\ \vdash f(b,X) \to X,\ c\#X \vdash h(X) \to X\}$$

$$\mathsf{R}^{(b\ d)} = \{\vdash f(h(a),d) \to a,\ \vdash f(d,X) \to X,\ a\#X \vdash h(X) \to X\}$$

$$\mathsf{R}^{(a\ c)(b\ d)} = \{\vdash f(h(c),d) \to c,\ \vdash f(d,X) \to X,\ c\#X \vdash h(X) \to X\}$$

Closedness Notice that we have $\vdash_{\mathsf{T}} h(X) \approx X$ and there is no rewriting path from $h(X)$ to X, i.e., $\not\vdash_{\mathsf{R}} h(X) \leftrightarrow X$. Indeed, we only have $h(X) \to X$ if we add the context $a\#X$.

Observe that we need to be careful because we are in $\approx_{\alpha,\mathsf{C}}$ and we need the formal definition of $\leftrightarrow_{\mathsf{R},\mathsf{C}}$. Notice, however, that the problem is not with the equational theory C but with the freshenss constraint.

- **Task 1:** we need to extend the definition of $\to$ to $\to_{\mathsf{R},\mathsf{C}}$.
- **Task 2:** also the definition of complete presentation needs to be extended for $\mathsf{T} = \mathsf{R} \cup \mathsf{C}$.

These two extensions will be done in a naive way, in the next section.

Confluence Observe that R is the same NRS as from Example B.8, where it was not confluent because the proper critical pairs were not joinable. But now, with commutativity we check again the joinability of these proper critical pairs found.

- $\emptyset \vdash \langle a, f(a,b) \rangle$: they are C-joinable.

$$\begin{tikzcd}
 & l_2\theta \arrow[dl] \arrow[dr] & \\
\vdash a & & \vdash f(a,b) \arrow[d, "\approx_{\mathsf{C}}"] \\
 & & \vdash f(b,a) \arrow[ull, "l_2 \to r_2"]
\end{tikzcd}$$

- $a\#Y, a'\#Y \vdash \langle Y, Y \rangle$: trivially joinable.
- $\emptyset \vdash \langle c, f(c,d) \rangle$: analogous to $\vdash \langle a, f(a,b) \rangle$.
- $c\#Y, a\#Y \vdash \langle Y, Y \rangle$: trivially joinable.
- $c\#Y, a'\#Y \vdash \langle Y, Y \rangle$: trivially joinable.

- **Task 3:** define C-confluence, C-critical pairs and C-coherence of $\to_{\mathsf{R},\mathsf{C}}$ in the nominal framework.

B.2.1 Nominal C-overlappings and complete sets of C-critical pairs

Next we extend the definitions of C-overlappings and C-critical pairs in the nominal framework.

Definition B.8. (C-overlappings and C-critical pairs) We say a term-in-context $\Delta \vdash s$ C-*overlaps* a term-in-context $\Delta' \vdash t$ at position $\mathtt{C}$, say $t \equiv \mathtt{C}[t']$, with a complete set S of C-overlappings iff S is a complete set of C-unifiers of $\Delta \vdash s$ and $\Delta' \vdash t'$.

Given two rules $\nabla_1 \vdash l_1 \to r_1$ and $\nabla_2 \vdash l_2 \to r_2$ in *eq-closure(Rw)* such that we have $V(l_1) \cap V(l_2) = \emptyset$ and $\nabla_1 \vdash l_1$ C-overlaps $\nabla_2 \vdash l_2$ at position $\mathtt{C}$, say $l_2 \equiv \mathtt{C}[l_2']$, with a complete set S of C-overlappings, then the set

$$\{\Gamma \vdash \langle u_1, u_2 \rangle \mid u_1 = r_2\theta,\ u_2 = \mathtt{C}\theta[r_1\theta],\ \forall \theta \in S, \Gamma \vdash l_1\theta \approx_{\alpha,\mathsf{C}} l_2'\theta, \Gamma \vdash \nabla_1\theta, \Gamma \vdash \nabla_2\theta\}$$

is called a *complete set of* C*-critical pairs* of the rule $\nabla_1 \vdash l_1 \to r_1$ on the rule $\nabla_2 \vdash l_2 \to r_2$ at position $\mathtt{C}$.

- Let $CS\mathsf{C}CP(\mathsf{R})$ be the complete set of non trivial C-critical pairs[2] for all $\nabla_1 \vdash l_1 \to r_1$ and $\nabla_2 \vdash l_2 \to r_2$ belonging both to R;
- Let $CS\mathsf{C}CP(\mathsf{R}/\mathsf{C})$ be the complete set of non trivial C-critical pairs for all $\nabla_1 \vdash l_1 \to r_1$ in R together with all $\nabla_2 \vdash l_2 \to r_2$ such that $\nabla_2 \vdash l_2 \approx_\alpha r_2$ or $\nabla_2 \vdash r_2 \approx_\alpha l_2$ belongs to C.

[2] that means C-critical pairs such that $\Delta_1 \vdash u_1 \neq \Delta_2 \vdash u_2$.