Universidade de Brasília

Desunificação Nominal via Restrições de Ponto Fixo

Leonardo Melo Batista

Orientadora: Dra. Daniele Nantes Sobrinho

Departamento de Matemática
Instituto de Exatas

Dissertação apresentada como requisito parcial para obtenção do grau de
Mestre em Matemática

Brasília, 28 de Setembro de 2021

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Matemática

Desunificação Nominal via Restrições de Ponto Fixo

por

Leonardo Melo Batista

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

MESTRE EM MATEMÁTICA

Brasília, 28 de setembro de 2021.

Comissão Examinadora:

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

Prof. Dr. Mauricio Ayala Rincón- MAT/UnB (Membro)

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

Resumo

Esta dissertação trata do *Problema de Desunificação Nominal*, isto é, do problema em resolver equações e desequações entre termos nominais, que são uma extensão de termos de primeira ordem com construtores para abstração de variáveis e renomeamento. Na sintaxe nominal via ponto fixo, as variáveis que podem ser abstraídas são representadas por *átomos*, e a igualdade entre dois termos nominais s e t é dada pela α-equivalência, denotada por $s \overset{\curlywedge}{\approx}_\alpha t$, que consiste na igualdade módulo renomeamento de átomos abstraídos. Por sua vez, a α-equivalência é definida usando a relação de *ponto fixo* $\pi \curlywedge t$, onde π é uma permutação e t um termo nominal, implementando o fato de que $\pi \cdot t$ (a permutação π aplicada no termo t) é α-equivalente a t, simbolicamente $\pi \cdot t \overset{\curlywedge}{\approx}_\alpha t$. Em trabalhos recentes, esta nova abordagem mostrou-se bastante promissora para o tratamento de teorias equacionais no contexto nominal.

Com o objetivo de investigar em trabalhos futuros os problemas de desunificação nominal módulo teorias equacionais, neste trabalho *propomos uma extensão do método para resolução do problema de desunificação nominal utilizando a abordagem de ponto fixo*: reformulamos os conceitos necessários para especificação do problema; provamos propriedades sintáticas; e finalmente, apresentamos o algoritmo $\texttt{desunif}_{\curlywedge}$: um algoritmo (provado correto) para computar uma representação finita e completa de soluções para o problema de desunificação nominal.

Abstract

This work is about the *Nominal Disunification Problem*, that is, the problem of solving equations and disequations between nominal terms, which are extensions of first-order terms with constructors for variable abstraction and renaming. In the nominal syntax via fixed points, variables that can be abstracted are represented by *atoms*, and equality between two nominal terms s and t is given by α-equivalence, denoted by $s \overset{\curlywedge}{\approx}_\alpha t$, and consists of equality modulo renaming of abstracted atoms. In turn, α-equivalence is defined using the fixed point relation, $\pi \curlywedge t$, where π is a permutation of atoms and t is a nominal term, and implements the fact that $\pi \cdot t$ (permutation π applied on term t) is α-equivalent to t, symbolically $\pi \cdot t \overset{\curlywedge}{\approx}_\alpha t$. In recent works, this new approach has proven to be quite promising for the treatment of equational theories in the nominal context.

With the aim of investigating nominal disunification problems modulo equational theories, in this work *we propose an extension of the method for solving the nominal disunification problems using the fixed point approach:* we reformulate the necessary concepts for the specification of the problem; we prove the necessary syntactic properties; and finally, we present the algorithm $\texttt{desunif}_{\curlywedge}$: an algorithm (proven correct) to compute a finite and complete representation of solutions for the nominal disunification problem.

Conteúdo

Capítulo 1

Introdução

O Problema de Desunificação de primeira ordem trata de resolver equações e desequações entre termos de primeira ordem e tem a seguinte forma:

$$\langle s_i =^? t_i (1 \leq i \leq n) \mid\mid p_j \neq^? q_j \ (1 \leq j \leq m) \rangle,$$

onde s_i, t_i, p_j e q_j são termos de primeira ordem.

Este problema tem sido bastante estudado nas últimas décadas. Em 1984, Colmerauer [Col84] discute pela primeira vez sobre o problema de resolver equações e desequações em programação lógica, apresentando um algoritmo para solucionar esses problemas de desunificação. Em 1986, Comon [Com86] investiga esse problema para certas aplicações na especificação algébrica, e no ano seguinte Kirchner e Lescanne [KL87] tratam de uma generalização da desunificação em uma estrutura com sistemas de equações e desequações entre termos de primeira ordem, a dizer:

$$\exists y_1, y_2, \ldots, y_n, \ \forall x_1, x_2, \ldots, x_m \ : \ P(y_1, y_2, \ldots, y_n, x_1, x_2, \ldots, x_m)$$

onde cada y_i e x_j são variáveis e P é uma disjunção de sistemas, onde um sistema é uma conjunção de equações e desequações entre termos. Em todos esses casos, a desunificação é estudada com respeito à igualdade sintática de termos, isto é, com teoria equacional $E = \emptyset$.

Extensões do problema de desunificação envolvendo um conjunto finito (e não-vazio) E de axiomas equacionais, o chamado problema de E-desunificação, começaram a ser desenvolvidas a partir da década de 90. Em 1991, Hubert Comon [Com91] fez um levantamento dos trabalhos que envolviam variações do problema de desunificação e inclui investigações sobre E-desunificação para teorias em que o problema de E-unificação é decidível (tais como comutatividade (`C`), associatividade-comutatividade (`AC`)) e também da interpretação da solu-

ção desse problema em algumas álgebras. Em 1994, Lugiez [Lug94] propõe investigações para desunificação de ordem superior, isto é, o problema de resolver equações e desequações no λ-cálculo (com tipos simples).

Em geral, a solubilidade do problema de desunificação requer a mesma propriedade do problema de unificação correspondente. Como em unificação de ordem superior o problema é indecidível, caracterizações de classes decidíveis são de grande interesse. Dentre as dificuldades encontradas na (des)unificação de ordem superior está o tratamento de variáveis ligadas, bem como a implementação de substituições que efetuam o correto renomeamento dessas variáveis quando necessário.

Técnicas nominais. Em 2002, Gabbay e Pitts [GP02], introduziram *a sintaxe nominal* como uma nova abordagem para linguagens que envolvam ligadores de variáveis, a exemplo da Lógica de Primeira Ordem (LPO), que possui quantificadores existenciais ($\exists$) e universais ($\forall$) ligando variáveis sob seu escopo em uma fórmula lógica. Por exemplo, a fórmula $\varphi := \forall x.R(x)$ é uma fórmula em LPO, com predicado unário R, cuja variável x ocorre ligada. Na abordagem nominal, as variáveis que podem ser ligadas/abstraídas são representadas por *átomos* e ligações como essa são implementadas através da *abstração* desses átomos, representada pelo uso de colchetes. Assim, utilizando a sintaxe nominal, a fórmula φ seria expressa por $\varphi := \forall[a]R(a)$.

Na linguagem nominal, os termos apresentam em sua estrutura elementos denominados *símbolos de função* $(\mathsf{f}, \mathsf{g}, \ldots)$, *átomos* $(a, b, \ldots)$ e *indeterminadas* (ou simplesmente variáveis) $(X, Y, \ldots)$. Os símbolos são construtores de termos, ao passo que os dois últimos possuem uma distinção sutil, mas fundamental, em sua função: átomos (tomados de um conjunto infinito enumerável $\mathbb{A}$) representam *variáveis a nível objeto* e, como dissemos antes, podem ser abstraídos/ligados; e indeterminadas representam *variáveis do meta-nível*, isto é, indicam um termo nominal indeterminado, ainda desconhecido, e não podem ser abstraídas/ligadas. Assim, por exemplo, na expressão de φ dada anteriormente, $\forall[a]R(a)$ é um termo nominal onde não há ocorrência de variáveis, $\forall$ e R são símbolos de função unários e a é um átomo sob abstração.

A igualdade entre termos nominais é estabelecida através da α-*equivalência* ($\approx_\alpha$), isto é, igualdade módulo renomeamento de átomos ligados, e a partir desse conceito surge, então, o problema de resolver *equações* entre termos nominais módulo α-equivalência ($s \approx_\alpha^? t$), objeto de estudo da teoria chamada de *Unificação Nominal* [UPG04]. Essa teoria pode ser vista como uma extensão da unificação de primeira ordem para termos que envolvem ligadores, como a LPO, mencionada anteriormente, o λ-cálculo, ou até mesmo a liguagem matemática corriqueira. Assim, são exemplos de problemas de unificação:

$$\forall x.P(x) \approx_\alpha^? \forall y.P(y) \qquad (1)$$
$$\cos(x) \approx_\alpha^? \cos(y) \qquad (2)$$
$$\lambda x.(x\,M) \approx_\alpha^? \lambda y.(y\,L) \qquad (3)$$

onde P é um predicado unário, M, L são λ-termos, x e y contêm apenas ocorrências ligadas. Esses problemas expressos na linguagem nominal ficam:

$$\forall [a]P(a) \approx_\alpha^? \forall [b]P(b) \qquad (1)$$
$$\cos([a]a) \approx_\alpha^? \cos([b]b) \qquad (2)$$
$$\lambda([a]\mathtt{app}(a,Y)) \approx_\alpha^? \lambda([b]\mathtt{app}(b,Z)) \qquad (3)$$

quando consideramos uma assinatura $\Sigma = \{\forall : 1, P : 1, \cos : 1, \lambda : 1, \mathtt{app} : 2\}$.

A grosso modo, uma solução para um problema de unificação nominal $s \approx_\alpha^? t$ é uma substituição σ que torna ambos os lados da equação iguais (módulo α-equivalência), isto é, $s\sigma \approx_\alpha^? t\sigma$. Enquanto que os problemas (1) e (2) têm como solução a substituição identidade (pois não contêm variáveis), o problema (3) precisa ser tratado com mais cuidado devido à presença das variáveis Y e Z.

A ocorrência desses elementos na estrutura de um termo nominal exige condições iniciais, chamadas de *contexto*, as quais dão informações prévias sobre indeterminadas e devem ser consideradas juntamente com a substituição para que, então, soluções possam ser exibidas. Esses contextos foram inicialmente tomados via uma abordagem denominada *freshness*, desenvolvida em [GP02], a qual faz uso da *relação de freshness* simbolizada pelo caractere #. Intuitivamente, $a\#t$ (lê-se: a fresco em t) significa que o átomo a não pode ocorrer livre no termo t. Assim, uma solução para (3) pode ser exibida como um par composto por um contexto de *freshness* Φ e uma substiuição σ, por exemplo $\langle \Phi, \sigma \rangle = \langle a\#Z, [Y \mapsto (a\ b) \cdot Z] \rangle$, onde $(a\ b)$ é uma permutação que troca a por b, e vice-versa.

Problemas de Desunificação Nominal. A partir desses desenvolvimentos em unificação, é natural estender também o problema de desunificação para o contexto da linguagem nominal. Recentemente, Ayala-Rincón et. al. em [AFNV20] propuseram um método para decidir o problema de desunificação nominal, que trata-se de uma extensão para termos nominais do problema de desunificação de primeira-ordem (definido por Comon e Lescanne [CL89] e Buntine e Bürckert [BB94]), e tem a forma:

$$\langle s_i \approx_\alpha^? t_i\ (1 \leq i \leq n) \mid\mid p_j \not\approx_\alpha^? q_j\ (1 \leq j \leq m) \rangle,$$

isto é, consiste de um conjunto que comporta equações nominais a serem solucionadas no lado esquerdo e desequações nominais que restringem as soluções no direito. Assim, uma substituição γ, a afim de ser uma solução, deve ser tal que $s_i\gamma \approx^?_\alpha t_i\gamma$ mas de maneira que $p_j\gamma \not\approx^?_\alpha q_j\gamma$. No entanto, o algoritmo de decisão para desunificação nominal proposto depende de que o problema de unificação nominal correspondente seja finitário, como ocorre em primeira ordem. Nesse sentido, enquanto que o problema de unificação nominal (puro) é finitário [UPG04], quando teorias equacionais são envolvidas, esta propriedade é perdida, como foi provado por Ayala-Rincón et al. [AdCSFN17], considerando, por exemplo, a teoria equacional de comutatividade.

De fato, seja $\approx_{\alpha,\mathtt{C}}$ a α-equivalência módulo comutatividade. Usando *freshness*, a equação $(a\ b)\cdot X \approx^?_{\alpha,\mathtt{C}} X$ possui um *unificador principal* $\langle\{a, b\#X\}, \mathtt{id}\rangle$ [UPG04], o qual tem como condição inicial que os átomos a e b não ocorram ligados nas *instâncias* de X. Porém, essa não é a única solução. Na verdade, como há comutatividade envolvida, existem infinitas delas, como $\langle\emptyset, [X \mapsto a+b]\rangle$, $\langle\emptyset, [X \mapsto (a+b)+(a+b)]\rangle$, $\langle\emptyset, [X \mapsto \mathtt{f}(a+b)]\rangle, \ldots$ e assim por diante, e em todas listadas, a e b ocorrem livres. Para resolver este problema, Ayala et al. [AFN20] mostram que o algoritmo de unificação nominal (módulo comutatividade) via *ponto fixo* computa um conjunto finito e completo de unificadores.

Sintaxe nominal via ponto fixo. Em 2013, Pitts apresentou uma formalização da relação de *freshness* utilizando o quantificador *new* (N), o qual, em lógica nominal, quantifica sobre *átomos novos* [Pit13]. Essa formalização é expressa pela sequinte sentença:

$$a\#x \text{ se, e somente, se } \mathsf{N}c.(a\ c)\cdot x = x, \tag{1.1}$$

isto é, a é fresco (ou, não ocorre livre) em x se, e somente se, para um átomo novo c arbitrário, a permutação $(a\ c)$, que troca a por c e vice-versa, fixa x. Aqui, x é um elemento de um conjunto nominal $\mathcal{X}$, que intuitivamente, consiste de um conjunto que tenha definida um ação do grupo $\mathtt{Perm}(\mathbb{A})$[1] e cujos elementos tem suporte finito: para $x \in \mathcal{X}$, o conjunto de átomos $A \subseteq \mathbb{A}$ tal que, se para todo $\pi \in \mathtt{Perm}(\mathbb{A})$.

$$((\forall a \in A)\pi(a) = a) \Rightarrow \pi\cdot x = x$$

é finito. Neste caso A é o suporte de x e $=$ é a igualdade de elementos em conjuntos nominais $\mathcal{X}$. Um exemplo de conjunto nominal é o próprio conjunto de termos nominais, a verificação deste fato está fora do escopo deste trabalho e pode ser encontrada em [Gab09].

[1] grupo $\mathtt{Perm}(\mathbb{A})$ das permutações finitas de $\mathbb{A}$.

Voltando à sentença, (1.1) considere novamente o termo nominal $\phi := \forall[a]Pa$. Nesse caso, a é um nome abstraído/ligado e, portanto, vale $a\#\phi$, o que é equivalente a dizer que o renomeamento de a por um nome novo c (que não ocorre em ϕ) ainda preserva ϕ, ou seja, $\mathit{И}c.(a\ c)\cdot\phi = \phi$.

Essa observação levou a uma nova axiomatização da α-equivalência entre termos nominais utilizando uma *relação de ponto fixo* ($\curlywedge$) no lugar da relação de *freshness*, trabalho feito por Ayala-Rincón et al. [AFN20], aonde também foi desenvolvida a *unificação nominal via ponto fixo* que surge como uma extensão para a unificação nominal via *freshness* devida a Urban et al. [UPG04]. Nessa nova abordagem, escreve-se $\pi \curlywedge t$ (lê-se: π fixa t) para dizer que $\pi\cdot t \overset{\curlywedge}{\approx}_\alpha t$ (o símbolo $\overset{\curlywedge}{\approx}_\alpha$ representa a relação de α-equivalência axiomatizada com a relação $\curlywedge$), ou seja, que a ação de uma permutação (finita) π em um termo nominal t é igual ao próprio t, a menos de renomeamento de átomos ligados.

Tratando-se de problemas de unificação nominal (pura, sem teoria equacional envolvida), existe uma correspondência direta entre soluções expressas usando *freshness* e aquelas usando *ponto fixo*, que é feita a partir de dois mapeamentos:

1. $[\_]_\curlywedge$, que associa cada restrição de *freshness* em um contexto Δ a uma de ponto fixo:

$$\begin{array}{rccl} [\_]_\curlywedge : & \Delta & \longrightarrow & \mathfrak{F}_\curlywedge \\ & a\#X & \mapsto & (a\ c_a) \curlywedge X \text{ onde } c_a \text{ é um nome novo;} \end{array} \tag{1.2}$$

2. e $[\_]_\#$, que associa cada restrição de ponto fixo em um contexto Υ a uma de *freshness*:

$$\begin{array}{rccl} [\_]_\# : & \Upsilon & \longrightarrow & \mathfrak{F}_\# \\ & \pi \curlywedge X & \mapsto & \mathtt{dom}(\pi)\#X. \end{array} \tag{1.3}$$

onde $\mathfrak{F}_\curlywedge$ e $\mathfrak{F}_\#$ denotam, respectivamente, as classes de contextos de ponto fixo e de freshness. Mais detalhes sobre esta correspondência podem ser encontrados em [AFNV20].

É na presença de teorias equacionais que a abordagem de ponto fixo se destaca. Quando é utilizado o símbolo comutativo $+$, por exemplo, temos que:

$$(a\ b)\cdot(a+b) = b+a \approx_{\alpha,\mathtt{C}} a+b, \tag{1.4}$$

ou seja, a permutação $(a\ b)$ *fixa* o termo $a+b$, apesar do fato de que os átomos a e b ocorrem livres em $a+b$. Reformulando o problema de unificação nominal utilizando a abordagem de ponto fixo, teríamos o seguinte problema $(a\ b)\cdot X \overset{\curlywedge\,?}{\approx}_{\alpha,\mathtt{C}} X$, e a solução deste problema seria $\langle (a\ b) \curlywedge_\mathtt{C} X, \mathtt{id}\rangle$, que incluiria a instância em (1.4) que corresponde a $[X \mapsto a+b]$, além de todas as outras que eram perdidas com a abordagem de *freshness*,

ainda com uma representação finita consistindo de um contexto e uma substituição (mais detalhes em [AFNV20]). Desta forma, extensões de problemas equacionais utilizando esta abordagem são muito promissoras.

Objetivo. Esta dissertação trata do estudo sobre *problemas de desunificação nominal* no âmbito sintático considerando a abordagem de ponto fixo, que foi desenvolvida recentemente. À luz dessa nova abordagem, apresentamos uma reformulação do método para resolução desses problemas, o qual foi estabelecido anteriormente via *freshness*. Para tanto, adaptamos as definições, propriedades, exemplos e algoritmos. Em suma, este trabalho propõe um primeiro passo para o estabelecimento de uma técnica robusta para a resolução de problemas de desunificação envolvendo teorias equacionais que incluem comutatividade.

Contribuições. As contribuições do presente trabalho consistem na apresentação detalhada de conceitos e resultados acerca do desenvolvimento de técnicas nominais utilizando abordagem de ponto fixo que foram inicialmente establecidas em [AFN20]. Além disso apresentamos algumas contribuições inéditas que são listadas abaixo:

1. Caracterização do suporte de um termo nominal com variáveis e contexto de ponto fixo. Ilustramos o conceito e damos a intuição de tal caracterização atráves de exemplos;

2. Estabelecemos uma definição para a solução de um problema de *emparelhamento sob contexto* que utiliza nomes novos (Definição 3.10), e também estendemos essa definição para um sistema particular com múltiplos problemas (4.1);

3. Definimos o problema de desunificação nominal via ponto fixo (Definição 4.1), bem como sua solução (Definição 4.2), a qual depende do conceito de *instância de um par com exceções* (Definição 4.4) que, por sua vez, leva em conta o possível surgimento de nomes novos para um problema de desunificação;

4. Reformulamos, sob a abordagem de ponto fixo, o algoritmo que decide sobre a consistência de um par com exceções (Algoritmo 1: $\texttt{consistente}_{\curlywedge}$). Este algoritmo é baseado no Lema da Inconsistência (Lema 4.1) e no Corolário 4.1.

5. Reformulamos o algoritmo que decide se um problema de desunificação nominal tem solução (Algoritmo 2: $\texttt{desunif}_{\curlywedge}$), provamos que esse algoritmo retorna uma representação completa de soluções (Teorema 4.1), e, por fim, demonstramos sua correção (Teorema 4.2).

Parte deste trabalho, principalmente sobre o Capítulo 4, foi apresentada em *35th International Workshop on Unification* com publicação informal [BFNV21] e no Segundo Workshop Brasileiro de Lógica, publicada nos anais [BN21].

Organização. Esta dissertação está organizada da seguinte maneira:

- **Capítulo 2 Sintaxe Nominal via Ponto Fixo.** Estabelecemos as principais definições e propriedades sobre a sintaxe nominal via ponto fixo que serão necessárias para o desenvolvimento deste trabalho. Na seção 2.1, apresentamos a gramática dos termos nominais, bem como as definições de permutação e substituição. Na seção 2.2, exibimos as relações de ponto fixo ($\curlywedge$) e α-equivalência ($\overset{\curlywedge}{\approx}_\alpha$), bem como o conceito de nomes novos, fundamental para a axiomatização das relações por meio de regras de derivação. Na seção 2.3, descrevemos as principais propriedades sintáticas da linguagem que serão utilizadas no restante do trabalho. Por fim, na seção 2.4, definimos o conceito de suporte para um termo nominal e enunciamos suas propriedades.

- **Capítulo 3: Unificação Nominal via Ponto Fixo.** Apresentamos as definições e resultados a respeito da unificação nominal via ponto fixo. Na seção 3.1, introduzimos as definições formais de um problema de unificação, de suas soluções e também da ordem parcial que as compara. Na seção 3.2, apresentamos as regras de simplificação para problemas de unificação, mostramos sua terminação e confluência, descrevemos o algoritmo que computa uma solução mais geral e mostramos sua correção. Por fim, na seção 3.3, exibimos a definição do problema de *emparelhamento sob contexto* e de sua solução, e descrevemos um procedimento para encontrá-la.

- **Capítulo 4: Desunificação Nominal via Ponto Fixo.** Apresentamos definições e resultados acerca da desunificação nominal via ponto fixo. Na seção 4.1, exibimos as definições de um problema de desunificação e de suas soluções, bem como a definição de pares com exceções e os conceitos de instância e consistência para essas estruturas. Na seção 4.2, descrevemos o algoritmo que verifica a consistência de pares com exceções. E finalmente, na seção 4.3, a última do trabalho, definimos o que vem a ser uma representação das soluções de um problema de desunificação, e apresentamos o algoritmo que decide sobre a solubilidade de problemas e exibe uma representação completa de soluções para eles.

- **Capítulo 5: Conclusão.** Concluímos o trabalho com as principais observações do desenvolvimento e também propomos algumas direções para trabalhos futuros.

Nota. Atentamos o leitor em nossa escolha de não exibir diretamente as demonstrações de resultados já estabelecidos e, ao invés disso, dar a sua intuição através de exemplos. Por isso e para facilitar a leitura, as provas exaustivas de lemas auxiliares que decorriam, em sua maioria, de induções e outras estratégias tradicionais foram alocadas para o apêndice do trabalho. Nossa ideia é priorizar a apresentação detalhada de provas daqueles resultados que são inéditos, ou cuja demonstração foi refeita por utilizarmos definições alternativas.

Capítulo 2

Sintaxe Nominal via Ponto Fixo

A estrutura dos termos nominais é construída a partir de *átomos* (ou *nomes atômicos*), *indeterminadas* (ou *ou simplesmente variáveis*) e *símbolos de função* (ou *formadores de termos*). Os átomos assumem um papel de *variáveis a nível objeto*, isto é, na fórmula $\forall x.\phi$ da Lógica de Primeira Ordem, por exemplo, a variável x que está sob o escopo do quantificador $\forall$ seria representada por um átomo na linguagem nominal. Já as indeterminadas assumem um papel de *variáveis no meta-nível*, ou seja, representam um termo nominal qualquer, e serão chamadas neste trabalho apenas por variáveis.

Neste capítulo, serão apresentadas a linguagem nominal utilizando a abordagem de ponto fixo, primeiramente introduzida por Ayala-Rincón et al. [AFN20], e as principais propriedades que serão úteis no decorrer do trabalho. Atenção especial ao Teorema 2.1, que trata da correção das relações de ponto fixo ($\curlywedge$) e α-equivalência ($\overset{\curlywedge}{\approx}_\alpha$), que serão definidas aqui, e também à Seção 2.4, onde estabelecemos uma caracterização ao suporte de um termo nominal (Teorema 2.3). As provas completas dos resultados auxiliares e lemas técnicos estão disponíveis no Apêndice A.

Seguimos a notação estabelecida em [AFN20] e assumimos conhecimento básico sobre permutações e álgebra de termos nominais, para mais detalhes referimos o leitor à [Pit13, Gab09].

2.1 Termos Nominais

Denotamos por $\mathbb{A} = \{a, b, c, \ldots\}$ o conjunto infinito e enumerável de átomos. Fixamos também notações para o conjunto infinito e enumerável de variáveis $\mathbb{X} = \{X, Y, Z, \ldots\}$ e para o conjunto enumerável de símbolos de função $\Sigma = \{\mathsf{f}, \mathsf{g}, \mathsf{h}, \ldots\}$, disjunto de $\mathbb{A}$ e $\mathbb{X}$, tal que cada $\mathsf{f} \in \Sigma$ é associado a um único inteiro não-negativo n que chamamos de *aridade de* f. Quando f tem aridade $n = 0$, dizemos que f é uma *constante*.

Notação: Escrevemos $\mathtt{f} : n$ para dizer que $\mathtt{f}$ é de aridade n.

Observação 2.1. Neste trabalho seguimos a *convenção permutativa* para átomos, variáveis e símbolos de função, isto é, letras distintas representam elementos distintos. Por exemplo, 'a e b' significa 'dois átomos distintos', 'X e Y' significa 'duas variáveis distintas' e '$\mathtt{f}$ e $\mathtt{g}$', por sua vez, 'dois símbolos de função distintos'.

A fim de tratar da α-conversão, precisamos lidar com o renomeamento de nomes atômicos, e para isso usamos *permutações* de átomos. Uma permutação (finita) em $\mathbb{A}$ é uma bijeção $\pi : \mathbb{A} \to \mathbb{A}$ tal que o conjunto $\{a \in \mathbb{A} \mid \pi(a) \neq a\}$, chamado *domínio de* π e denotado por $\mathtt{dom}(\pi)$, é finito.

Abaixo, apresentamos a gramática dos termos nominais.

Definição 2.1 (Gramática nominal)**.** Termos nominais são gerados pela seguinte gramática:

$$s,t ::= a \mid \pi \cdot X \mid [a]t \mid \mathtt{f}(t_1,\ldots,t_n),$$

onde a é um *átomo*, $\pi \cdot X$ é uma *variável moderada* (ou uma *suspensão*) tal que π é uma permutação de átomos, $[a]t$ denota a *abstração* do átomo a no termo t e $\mathtt{f}(t_1,\ldots,t_n)$ denota a *aplicação* de $\mathtt{f} : n$ em termos $t_1,\ldots,t_n$. O conjunto dos termos nominais será denotado por $T(\Sigma,\mathbb{X},\mathbb{A})$.

As permutações (finitas) de átomos com a operação de composição, denotada pelo símbolo '$\circ$', formam um grupo, $\mathtt{Perm}(\mathbb{A})$, onde o elemento neutro é dado por $\mathtt{Id}$ (a permutação identidade) e a inversa de uma permutação π é representada por π^{-1}. Elas serão representadas por listas finitas de *transposições*, cuja notação é dada por $(a\ b)$, que indica a troca de a por b e vice-versa. Assim, uma permutação π é gerada pela gramática:

$$\pi ::= \mathtt{Id} \mid (a\ b) \circ \pi.$$

Por exemplo, $\pi = (a\ b) \circ (b\ c)$, a composição de $(b\ c)$ com $(a\ b)$, é tal que $\pi(a) = b$, $\pi(b) = c$ e $\pi(c) = a$ e $\pi(d) = \mathtt{Id}(d) = d$, para todo $d \in \mathbb{A} \setminus \{a,b,c\}$, portanto $\mathtt{dom}(\pi) = \{a,b,c\}$. Note que sua inversa é dada por $\pi^{-1} = (c\ b) \circ (b\ a)$.

Observação 2.2. O termo $\mathtt{Id} \cdot X$ será representado simplesmente por X.

A seguir, definimos a ocorrência de átomos e variáveis na estrutura de um termo nominal.

Definição 2.2. Definimos $a \in t$ (lê-se: 'a ocorre em t') indutivamente por:

$$\frac{}{a \in a} \qquad \frac{}{a \in [a]t} \qquad \frac{a \in t}{a \in [b]t} \qquad \frac{a \in \mathtt{dom}(\pi)}{a \in \pi \cdot X} \qquad \frac{a \in t_i \ \ (1 \leq i \leq n)}{a \in \mathtt{f}(t_1,\ldots,t_n)}$$

Quando $a \in t$ não vale, escrevemos $a \notin t$ e dizemos que 'a não ocorre em t'. Além disso, definimos $\mathtt{Atm}(t) := \{a \mid a \in t\}$, o conjunto dos átomos que ocorrem em t.

Definição 2.3. Definimos $X \in t$ (lê-se: 'X ocorre em t') indutivamente por:

$$\frac{}{X \in \pi \cdot X} \qquad \frac{X \in t}{X \in [b]t} \qquad \frac{X \in t_i \ \ (1 \leq i \leq n)}{X \in \mathsf{f}(t_1, \ldots, t_n)}$$

Quando $X \in t$ não vale, escrevemos $X \notin t$ e dizemos que 'X não ocorre em t'. Além disso, definimos $\mathtt{Var}(t) := \{X \mid X \in t\}$, o conjunto das variáveis que ocorrem em t. Os termos cujo conjunto de variáveis é vazio, isto é, não há ocorrência de variáveis em suas estruturas, são chamados de *termos básicos*.

Exemplo 2.1.

1. Se $t_1 = \mathsf{g}(X, (c\ d) \cdot Y)$, então $\mathtt{Atm}(t_1) = \{c, d\}$ e t_1 não é um termo básico, já que $\mathtt{Var}(t_1) = \{X, Y\}$.
2. Se $t_2 = [a]\mathsf{f}\ b$, então $\mathtt{Atm}(t_2) = \{a, b\}$ e $\mathtt{Var}(t_2) = \emptyset$, e t_2 é um exemplo de termo básico.
3. Se $t_3 = [a]\mathsf{g}(c, (a\ b) \cdot Z)$, então $\mathtt{Atm}(t_3) = \{a, b, c\}$ e t_3 não é um termo básico, já que $\mathtt{Var}(t_3) = \{Z\}$.

A seguir definimos a ação de $\mathtt{Perm}(\mathbb{A})$ sobre $T(\Sigma, \mathbb{X}, \mathbb{A})$.

Definição 2.4 (Ação de permutação)**.** A ação de uma permutação π em um termo nominal t é definida por indução no número de transposições em π:

$$\mathtt{Id} \cdot t = t \quad \text{e} \quad ((a\ b) \circ \pi) \cdot t = (a\ b) \cdot (\pi \cdot t)$$

onde

$$\begin{array}{lll}
(a\ b) \cdot a = b & (a\ b) \cdot (\pi \cdot X) = ((a\ b) \circ \pi) \cdot X & (a\ b) \cdot [a]t = [b](a\ b) \cdot t \\
(a\ b) \cdot b = a & (a\ b) \cdot \mathsf{f}(t_1, \ldots, t_n) = \mathsf{f}((a\ b) \cdot t_1, \ldots, (a\ b) \cdot t_n) & (a\ b) \cdot [b]t = [a](a\ b) \cdot t \\
(a\ b) \cdot c = c & & (a\ b) \cdot [c]t = [c](a\ b) \cdot t
\end{array}$$

Definição 2.5 (Substituição)**.** Substituições são geradas pela gramática:

$$\sigma ::= \mathtt{id} \mid [X \mapsto s]\sigma.$$

Será utilizada a notação pós-fixada para a aplicação de substituições em termos e o símbolo $\circ$ para a composição: $t(\sigma \circ \sigma') = (t\sigma)\sigma'$. Substituições agem homomorficamente sobre

termos e são definidas indutivamente por:

$$t\ \mathtt{id} = t \qquad t[X \mapsto s]\sigma = (t[X \mapsto s])\sigma,$$

onde

$$\begin{array}{ll} a[X \mapsto s] = a & (\pi \cdot X)[X \mapsto s] = \pi \cdot s \\ (\mathsf{f}(t_1, \ldots, t_n))[X \mapsto s] = \mathsf{f}(t_1[X \mapsto s], \ldots, t_n[X \mapsto s]) & (\pi \cdot Y)[X \mapsto s] = \pi \cdot Y \\ ([a]t)[X \mapsto s] = [a](t[X \mapsto s]) & \end{array}$$

O conjunto (finito) de variáveis que σ não mapeia para si mesmas, ou em outras palavras, que σ não mapeia como a identidade $\mathtt{id}$, é chamado de *domínio* de σ e escrevemos $\mathtt{dom}(\sigma) := \{X \in \mathbb{X} \mid X\sigma \neq X\}$. O conjunto dos termos para os quais são mapeadas tais variáveis é chamado de *imagem* de σ, e denotado por $\mathtt{ran}(\sigma) := \{X\sigma \mid X \in \mathtt{dom}(\sigma)\}$, e o conjunto das variáveis da imagem de σ é denotado por

$$\mathtt{VRan}(\sigma) = \bigcup_{X \in \mathtt{dom}(\sigma)} \mathtt{Var}(X\sigma).$$

Terminologia:

1. Quando aplicamos uma substituição σ em um termo s, o resultado dessa aplicação é chamado de *instância de s*.
2. Dizemos que σ é uma *substituição básica* quando sua imagem é composta apenas por termos básicos, isto é, $\mathtt{VRan}(\sigma) = \emptyset$.
3. Considerando $\mathtt{Var}(s) \subseteq \mathtt{dom}(\sigma)$, o resultado da aplicação de uma substituição básica σ em um termo s é chamada de *instância básica de s*, e nesse caso $\mathtt{Var}(s\sigma) = \emptyset$.

O lema a seguir traz a propriedade de comutatividade entre substituições e permutações que justifica a notação $\pi \cdot s\sigma$, sem o uso de parênteses.

Lema 2.1. Substituições e permutações comutam: $\pi \cdot (s\sigma) = (\pi \cdot s)\sigma$.

Demonstração. A prova completa encontra-se no Apêndice A, Lema A.6 □

2.2 Ponto Fixo para α-equivalência

Nesta seção, axiomatizamos as relações de ponto fixo ($\curlywedge$) e α-equivalência ($\overset{\curlywedge}{\approx}_\alpha$) para termos nominais por meio de *regras de derivação* que serão apresentadas mais adiante com as Tabelas 2.1 e 2.2.

Iniciamos apresentando o conceito de *restrições*:

Definição 2.6. Uma *restrição de ponto fixo* é um par $\pi \curlywedge t$ de uma permutação π e um termo t. Uma *restrição de α-equivalência* é um par de termos da forma $s \overset{\curlywedge}{\approx}_\alpha t$.

Intuitivamente, considerando s e t termos básicos, a restrição $s \overset{\curlywedge}{\approx}_\alpha t$ significa que s e t são α-equivalentes, ou seja, equivalentes módulo renomeamento de átomos abstraídos/ligados. Por exemplo, os termos $[a]a$ e $[b]b$ são equivalentes ao renomear a por b, ou vice-versa, logo escrevemos $[a]a \overset{\curlywedge}{\approx}_\alpha [b]b$. Porém, é falso que $[a]\mathrm{g}(a,b) \overset{\curlywedge}{\approx}_\alpha [b]\mathrm{g}(b,b)$, uma vez que b é livre no termo à esquerda de $\overset{\curlywedge}{\approx}_\alpha$.

Já a restrição $\pi \curlywedge t$ significa que a permutação π fixa o termo t, isto é, $\pi \cdot t \overset{\curlywedge}{\approx}_\alpha t$. Em outras palavras, π não tem efeito sobre t exceto pelo renomeamento de átomos ligados, por exemplo, $(a\ b) \curlywedge [a]\mathrm{g}(a,\mathrm{g}(c,d))$ mas não $(a\ c) \curlywedge [a]\mathrm{g}(a,\mathrm{g}(c,d))$, uma vez que c é livre no termo à direita da segunda restrição de ponto fixo.

No caso de termos não-básicos, ambas as restrições precisam ser avaliadas sob um *contexto*, que fornecerá informações sobre as permutações que fixam variáveis. Um *contexto (de ponto fixo)* é um conjunto finito de *restrições primitivas* de ponto fixo, isto é, restrições da forma $\pi \curlywedge X$, e usaremos as letras maiúsculas $\Upsilon, \Psi, \Phi, \Delta$, etc. para representá-los.

Essas restrições sob contexto, formam o que chamamos de *sequentes*:

Definição 2.7 (Sequentes)**.** Um *sequente de ponto fixo*, denotada por $\Upsilon \vdash \pi \curlywedge t$, é composta por um contexto Υ e uma restrição de ponto fixo, e um *sequente de α-equivalência* (ou *de igualdade*), denotada por $\Psi \vdash s \overset{\curlywedge}{\approx}_\alpha t$, é composta por um contexto de ponto fixo e uma restrição de igualdade.

Dizemos que um sequente da forma $\Delta \vdash R$, onde R é uma restrição de ponto fixo ou de igualdade, é *válido* ou *derivável* quando existe uma prova Π a partir das regras dispostas nas Tabelas 2.1 e 2.2, chamadas de *regras de derivação* (ou *de dedução*), tal que

$$\frac{\Pi}{\Delta \vdash R}\ ,$$

ou seja, existe uma aplicação em sequência de regras de derivação de maneira que, após a última regra aplicada, se obtém $\Delta \vdash R$. Quando um sequente não é derivável, escrevemos $\Delta \nvdash R$.

Antes de irmos às regras, precisamos explicar o conceito de *nomes atômicos novos*, ou simplesmente *nomes novos*. Quando se trata de um termo nominal t, todo átomo que não está em $\mathtt{Atm}(t)$ (i.e., não ocorre em t) é novo para t, então, por exemplo, se $t = [b]\mathsf{g}(a,(c\ b)\cdot X)$, c_1 é novo para t, bem como todo nome diferente de a, b e c. Já para uma permutação π, todo aquele que não está em $\mathtt{dom}(\pi)$ é um nome novo para π, isto é, se $\pi(a) = a$ então a é novo para π. Abaixo, definimos essa noção também para restrições, contextos e sequentes.

Definição 2.8 (Nomes novos)**.** Dizemos que o átomo $c_1 \in \mathbb{A}$ é um *nome novo* com relação a

(i) um termo nominal t quando $c_1 \notin t$;

(ii) uma permutação π quando $c_1 \notin \mathtt{dom}(\pi)$;

(iii) uma restrição de ponto fixo $\pi \curlywedge t$ quando $c_1 \notin \mathtt{dom}(\pi)$ e $c_1 \notin t$;

(iv) uma restrição de igualdade $s \overset{\curlywedge}{\approx}_\alpha t$ quando $c_1 \notin s$ e $c_1 \notin t$.

Da mesma maneira é estentida a noção de *nomes novos* para um contexto de ponto fixo Δ e sequentes do tipo $\Delta \vdash R$, onde R é uma restrição de ponto fixo ou de igualdade.

Observe que a definição acima está bem posta, visto que termos são palavras finitas, as permutações foram tomadas com domínio finito e contextos são conjuntos finitos de restrições primitivas, logo sempre haverá um nome atômico que possa ser tomado como novo.

Finalmente, apresentamos a seguir tais regras de derivação que axiomatizam as relações de ponto fixo e de α-equivalência entre termos nominais.

Terminologia e notação:

1. Quando dissermos que átomos c_1 e c_2 são novos, sem especificar com relação a quem, eles estarão sendo considerados novos com respeito ao sequente a ser derivado. Por exemplo, na Tabela 2.1, regra ($\curlywedge$**abs**), c_1 e c_2 são novos para Υ, π e $[a]t$.

2. $\mathtt{perm}(\Upsilon|_X) := \{\pi \mid \pi \curlywedge X \in \Upsilon\}$

3. $\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) := \bigcup_{\pi \in \mathtt{perm}(\Upsilon|_X)} \mathtt{dom}(\pi)$

4. $\overline{\pi \curlywedge \mathtt{Var}(t)} := \{\pi \curlywedge X \mid X \in \mathtt{Var}(t)\}$

Regras de Derivação para $\curlywedge$ e $\overset{\curlywedge}{\approx}_\alpha$. As regras utilizam o conceito de *conjugado* de uma permutação. No caso de $\mathtt{Perm}(\mathbb{A})$, dados $\pi, \rho \in \mathtt{Perm}(\mathbb{A})$, o conjugado de π com respeito a ρ é a composição $\rho \circ \pi \circ \rho^{-1}$ denotada por π^ρ, e será o resultado da ação $\rho \cdot \pi$. Propriedades auxiliares sobre conjugados, que serão utilizadas no decorrer deste e dos próximos capítulos, podem ser encontradas na Seção A.1 do Apêndice A.

$$\frac{\pi(a) = a}{\Upsilon \vdash \pi \curlywedge a}\,(\curlywedge\mathbf{a}) \qquad \frac{\mathtt{dom}(\pi^{\pi'^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))}{\Upsilon \vdash \pi \curlywedge \pi' \cdot X}\,(\curlywedge\mathbf{var})$$

$$\frac{\Upsilon \vdash \pi \curlywedge t_1 \quad \ldots \quad \Upsilon \vdash \pi \curlywedge t_n}{\Upsilon \vdash \pi \curlywedge \mathtt{f}(t_1, \ldots, t_n)}\,(\curlywedge\mathtt{f}) \qquad \frac{\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge (a\ c_1) \cdot t}{\Upsilon \vdash \pi \curlywedge [a]t}\,(\curlywedge\mathbf{abs})$$

Tabela 2.1 Regras de Derivação para $\curlywedge$, onde c_1 e c_2 são nomes novos.

$$\frac{}{\Upsilon \vdash a \overset{\curlywedge}{\approx}_\alpha a}\,(\overset{\curlywedge}{\approx}_\alpha\mathbf{a}) \qquad \frac{\mathtt{dom}((\pi')^{-1} \circ \pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))}{\Upsilon \vdash \pi \cdot X \overset{\curlywedge}{\approx}_\alpha \pi' \cdot X}\,(\overset{\curlywedge}{\approx}_\alpha\mathbf{var})$$

$$\frac{\Upsilon \vdash t_1 \overset{\curlywedge}{\approx}_\alpha t_1' \quad \ldots \quad \Upsilon \vdash t_n \overset{\curlywedge}{\approx}_\alpha t_n'}{\Upsilon \vdash \mathtt{f}(t_1, \ldots, t_n) \overset{\curlywedge}{\approx}_\alpha \mathtt{f}(t_1', \ldots, t_n')}\,(\overset{\curlywedge}{\approx}_\alpha\mathtt{f}) \qquad \frac{\Upsilon \vdash t \overset{\curlywedge}{\approx}_\alpha t'}{\Upsilon \vdash [a]t \overset{\curlywedge}{\approx}_\alpha [a]t'}\,(\overset{\curlywedge}{\approx}_\alpha[\mathbf{a}])$$

$$\frac{\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha (a\ b) \cdot t \qquad \Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} \vdash (a\ c_1) \curlywedge t}{\Upsilon \vdash [a]s \overset{\curlywedge}{\approx}_\alpha [b]t}\,(\overset{\curlywedge}{\approx}_\alpha\mathbf{ab})$$

Tabela 2.2 Regras de Derivação para $\overset{\curlywedge}{\approx}_\alpha$, onde c_1 e c_2 são nomes novos.

Em cada regra, a linha horizontal que separa a parte superior da parte inferior pode ser entendida como uma implicação lógica. Então, por exemplo em ($\curlywedge\mathbf{a}$), para deduzir que $\Upsilon \vdash \pi \curlywedge a$, é necessário ter previamente que $\pi(a) = a$, isto é, que a seja um ponto fixo de π. A mesma linha de raciocínio baseia as regras sobre funções ($\curlywedge\mathtt{f}$) e ($\overset{\curlywedge}{\approx}_\alpha\mathtt{f}$) e sobre a α-igualdade entre abstrações de um mesmo átomo ($\overset{\curlywedge}{\approx}_\alpha[\mathbf{a}]$). Observe que a regra ($\overset{\curlywedge}{\approx}_\alpha\mathbf{a}$) tem premissa vazia, pois trata-se de uma trivialidade já que $a = a$.

Na regra ($\curlywedge\mathbf{var}$) a condição $\mathtt{dom}(\pi^{\pi'^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ impõe que a permutação $\pi^{\pi'^{-1}}$ tenha efeito apenas em átomos já afetados pelas permutações que fixam X. Com isso, a permutação em questão também fixará a variável. A intuição por trás de ($\overset{\curlywedge}{\approx}_\alpha$ **var**) é similar.

A regra ($\curlywedge$**abs**) define quando uma permutação π fixa uma abstração de um átomo a em um termo t, e para isso pede que π fixe a troca de a por um nome c_1 que seja novo com relação ao sequente a ser derivado. Observe que para essa derivação, é adicionado ao contexto Υ o conjunto de restrições primitivas $\overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)}$ onde c_2 também é um nome novo, assim como c_1. Isso tem a função de informar ao próprio cálculo do sequente que esses dois nomes são novos e que, consequentemente, fixam as variáveis de t.

Por fim, a regra ($\overset{\curlywedge}{\approx}_\alpha$ **ab**) estabelece que para duas abstrações $[a]t$ e $[b]s$ serem α-equivalentes deve-se obedecer duas condições: uma é que os termos s e t devem ser α-equivalentes se renomearmos em t o átomo abstraído b por a; e, segundo, a troca de a por um átomo novo c_1 deve fixar t, garantindo assim que a não ocorra livre em t. A adição de $\overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)}$ ao contexto cumpre a mesma função que em ($\curlywedge$**abs**).

Exemplo 2.2. Nos itens abaixo, explicitamos alguns cálculos usando essas regras de derivação, mas usamos algumas notações novas: a linha pontilhada na árvore de dedução serve para indicar que o sequente abaixo dela é o mesmo do que vem acima, apenas escrito de uma maneira diferente; e escrevemos $\pi_1, \ldots, \pi_n \curlywedge t$ como abreviação para $\pi_1 \curlywedge t, \ldots, \pi_n \curlywedge t$.

1. O sequente $\vdash (a\ b) \curlywedge [a]\mathtt{f}\ a$ é derivável, isto é, a restrição $(a\ b) \curlywedge [a]\mathtt{f}\ a$ é derivável sob um contexto vazio. De fato,

$$\dfrac{\dfrac{\dfrac{(a\ b)(c_1) = c_1}{\vdash (a\ b) \curlywedge c_1}\,(\curlywedge\mathbf{a})}{\vdash (a\ b) \curlywedge \mathtt{f}\ c_1}\,(\curlywedge\mathtt{f})}{\dfrac{\vdash (a\ b) \curlywedge (a\ c_1)\cdot \mathtt{f}\ a}{\vdash (a\ b) \curlywedge [a]\mathtt{f}\ a}\,(\curlywedge\mathbf{abs})}$$

   Note que $[a]\mathtt{f}\ a$ é um termo básico, logo $\overline{(c_1\ c_2) \curlywedge \mathtt{Var}([a]\mathtt{f}\ a)} = \emptyset$.

2. Como esperado, o sequente $\Upsilon \vdash (a\ b) \curlywedge \mathtt{f}\ a$ não é derivável sob qualquer contexto Υ, pois a não é ponto fixo de $(a\ b)$.

3. O sequente $\{(a\ b), (c\ d) \curlywedge X\} \vdash (a\ d) \curlywedge [a]\mathtt{h}(a, c, X)$ é derivável. De fato, seja Π_1 a derivação

$$\dfrac{(a\ d)(c_1) = c_1}{\{(a\ b), (c\ d), (c_1\ c_2) \curlywedge X\} \vdash (a\ d) \curlywedge c_1}\,(\curlywedge\mathbf{a})\ ,$$

   Π_2 a derivação

$$\frac{(a\ d)(c) = c}{\{(a\ b),(c\ d),(c_1\ c_2) \curlywedge X\} \vdash (a\ d) \curlywedge c}\ (\curlywedge\mathbf{a})\ ,$$

e Π_3 a derivação

$$\frac{\{d,c_1\} \subseteq \{a,b,c,d,c_1,c_2\}}{\{(a\ b),(c\ d),(c_1\ c_2) \curlywedge X\} \vdash (a\ d) \curlywedge (a\ c_1)\cdot X}\ (\curlywedge\mathbf{var})\ ,$$

onde $\mathtt{dom}((a\ d)^{(c_1\ a)}) = \{d,c_1\}$. Então,

$$\dfrac{\dfrac{\Pi_1 \qquad \Pi_2 \qquad \Pi_3}{\{(a\ b),(c\ d),(c_1\ c_2) \curlywedge X\} \vdash (a\ d) \curlywedge \mathsf{h}(c_1,c,(a\ c_1)\cdot X)}\ (\curlywedge\mathsf{f})}{\dfrac{\{(a\ b),(c\ d),(c_1\ c_2) \curlywedge X\} \vdash (a\ d) \curlywedge (a\ c_1)\cdot \mathsf{h}(a,c,X)}{\{(a\ b),(c\ d) \curlywedge X\} \vdash (a\ d) \curlywedge [a]\mathsf{h}(a,c,X)}\ (\curlywedge\mathbf{abs})}$$

4. O sequente $\vdash [b][a]Y \overset{\curlywedge}{\approx}_\alpha [a][b](a\ b)\cdot Y$ é derivável. De fato,

$$\dfrac{\dfrac{\dfrac{}{\vdash Y \overset{\curlywedge}{\approx}_\alpha Y}\ (\overset{\curlywedge}{\approx}_\alpha\ \mathbf{var})}{\vdash [a]Y \overset{\curlywedge}{\approx}_\alpha [a]Y}\ (\overset{\curlywedge}{\approx}_\alpha\ [\mathbf{a}]) \qquad \dfrac{\dfrac{\{c_1,c_3\} \subseteq \{c_1,c_2,c_3,c_4\}}{\{(c_1\ c_2),(c_3\ c_4) \curlywedge Y\} \vdash (b\ c_1) \curlywedge (b\ c_3)\cdot((a\ b)\cdot Y)}\ (\overset{\curlywedge}{\approx}_\alpha\ \mathbf{var})}{\{(c_1\ c_2) \curlywedge Y\} \vdash (b\ c_1) \curlywedge [b](a\ b)\cdot Y}\ (\curlywedge\mathbf{abs})}{\vdash [b][a]Y \overset{\curlywedge}{\approx}_\alpha [a][b](a\ b)\cdot Y}\ (\overset{\curlywedge}{\approx}_\alpha\ \mathbf{ab})$$

onde $\mathtt{dom}((b\ c_1)^{(b\ a)(c_3\ b)}) = \{c_1,c_3\}$. De fato, escrevendo $\rho = (b\ a) \circ (c_3\ b) = ((b\ c_3) \circ (a\ b))^{-1}$, temos

$$(b\ c_1)^\rho = (\rho(b)\ \ \rho(c_1)) = (c_3\ c_1),\ \text{pelo Lema A.1.}$$

2.3 Propriedades

Esta seção apresenta propriedades importantes das relações de ponto fixo ($\curlywedge$) e α-equivalência ($\overset{\curlywedge}{\approx}_\alpha$) que serão utilizadas durante todo o trabalho, e que foram inicialmente estabelecidas no artigo [AFN20]. Para facilitar a leitura, a demonstração das propriedades apresentadas nesta seção estão em sua maioria omitidas, e podem ser encontradas no Apêndice A. Vale ressaltar que as demonstrações feitas no trabalho original [AFN20] foram desenvolvidas com base no uso do quantificador $\mathsf{И}$, e nesta dissertação as mesmas demonstrações foram reestabelecidas sem esse uso, manipulando de maneira explícita uma escolha de nomes novos, isto é, usamos

sentenças do tipo "onde c_1, c_2 são nomes novos com relação à ...", ao invés de deixar essa informação atrelada ao quantificador И.

O resultado mais interessante é o Teorema 2.1 sobre a correção de $\curlywedge$, o qual estabelece que a relação axiomatizada pelas regras de derivação da Tabela 2.1 é de fato uma relação de ponto fixo quanto a α-equivalência entre termos nominais.

Por exemplo, afirmar que vale

$$\Upsilon \vdash (a\ b) \curlywedge [a]\mathrm{g}(a,X)$$

é equivalente a dizer que

$$\Upsilon \vdash (a\ b) \cdot [a]\mathrm{g}(a,X) \overset{\curlywedge}{\approx}_{\alpha} [a]\mathrm{g}(a,X),$$

ou, em outras palavras, se uma permutação π fixa o termo t sob um contexto Υ, então a ação de π sobre t, nessas condições, não surte efeito, a menos de renomeamento de átomos ligados em t.

Começamos com a propriedade de inversão, ou invertibilidade, das regras de derivação apresentadas na seção anterior.

Lema 2.2 (Inversão)**.** As regras de derivação presentes nas tabelas 2.1 e 2.2 são invertíveis.

Demonstração. Queremos provar que a conclusão de cada uma dessas regras implica sua respectiva premissa. Mas isso segue diretamente do fato de que cada regra corresponde a uma única classe de termos. Por exemplo, se $\Upsilon \vdash \pi \curlywedge a$, então existe uma prova Π para essa derivação, porém, a única regra aplicável como último passo é $(\curlywedge\mathbf{a})$, já que apenas essa regra lida com termos atômicos e, portanto, devemos ter $\pi(a) = a$. Esse mesmo raciocínio é feito com todas as outras regras. □

Uma outra propriedade importante é a chamada *equivariância*, a qual garante que as relações de ponto fixo e α-equivalência são fechadas para permutações, ou, em outras palavras, que essas relações são preservadas após a aplicação de permutações.

Lema 2.3 (Equivariância)**.**

(i) $\Upsilon \vdash \pi \curlywedge t$ se, e somente se, $\Upsilon \vdash \pi^{\rho} \curlywedge \rho \cdot t$, para qualquer permutação ρ.

(ii) $\Upsilon \vdash s \overset{\curlywedge}{\approx}_{\alpha} t$ se, e somente se, $\Upsilon \vdash \pi \cdot s \overset{\curlywedge}{\approx}_{\alpha} \pi \cdot t$, para qualquer permutação π.

Demonstração. A prova detalhada pode ser encontrada no Apêndice A, Lema A.7. □

Exemplo 2.3. Vimos no Exemplo 2.2 que vale $\{(a\ b),(c\ d) \curlywedge X\} \vdash (a\ d) \curlywedge [a]\mathsf{h}(a,c,X)$. Usando a equivariância com $\rho = (c\ a)$, temos que o sequente abaixo é derivável

$$\{(a\ b),(c\ d) \curlywedge X\} \vdash (c\ d) \curlywedge [c]\mathsf{h}(c,a,(c\ a)\cdot X).$$

A proposição a seguir mostra a propriedade de *fortalecimento* para as relações $\curlywedge$ e $\overset{\curlywedge}{\approx}_\alpha$, respectivamente, e estabelce quando é possível fortalecer o contexto (i.e. retirar restrições primitivas d o contexto) de um sequente que envolve uma restrição de ponto fixo ou de igualdade.

Proposição 2.1 (Fortalecimento de $\curlywedge$ e $\overset{\curlywedge}{\approx}_\alpha$)**.**

(i) Se $\Upsilon, \pi \curlywedge X \vdash \pi' \curlywedge s$ e $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ ou $X \notin \mathtt{Var}(s)$, então $\Upsilon \vdash \pi' \curlywedge s$.

(ii) Se $\Upsilon, \pi \curlywedge X \vdash s \overset{\curlywedge}{\approx}_\alpha t$ e $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ ou $X \notin \mathtt{Var}(s,t)$, então $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$.

Demonstração. A prova completa desta proposição pode ser encontrada no Apêndice A, com a Proposição A.2 e Proposição A.1. □

A propriedade de *enfraquecimento* permite deduzir que uma restrição, de ponto fixo ou igualdade, é válida sob um contexto Υ dado que a mesma já é válida sob um contexto menor, isto é, contido em Υ.

Proposição 2.2 (Enfraquecimento)**.** Suponha que $\Upsilon' \subseteq \Upsilon$.

(i) Se $\Upsilon' \vdash \pi \curlywedge s$, então $\Upsilon \vdash \pi \curlywedge s$.

(ii) Se $\Upsilon' \vdash s \overset{\curlywedge}{\approx}_\alpha t$, então $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$.

Demonstração. A prova de ambos os itens segue por indução na derivação dos respectivos sequentes, e será omitida já que a maioria dos casos segue diretamente da hipótese de que $\Upsilon' \subseteq \Upsilon$ ou da hipótese indutiva. □

O resultado abaixo mostra que a relação de ponto fixo ($\curlywedge$) é preservada pela α-equivalência ($\overset{\curlywedge}{\approx}_\alpha$).

Lema 2.4. Se $\Upsilon \vdash \pi \curlywedge s$ e $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$, então $\Upsilon \vdash \pi \curlywedge t$.

Demonstração. Prova por indução na estrutura de s, que induz uma estrutura a t. Apresentaremos apenas a prova de um caso interessante, a dizer $s = [a]s'$. A demonstração completa pode ser encontrada no Apêndice A, Lema A.9.

- $s = [a]s'$

  Nesse caso, $\Upsilon \vdash \pi \curlywedge [a]s'$, o que fornece

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s')} \vdash \pi \curlywedge (a\ c_1) \cdot s', \tag{2.1}$$

  onde c_1 e c_2 são nomes novos. Como por hipótese $\Upsilon \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha t$, duas situações são possíveis:

  1. a última regra aplicada foi $(\overset{\curlywedge}{\approx}_\alpha [\mathbf{a}])$ e força-se $t = [a]t'$;
     Com isso $\Upsilon \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha [a]t'$ e, pela invertibilidade, $\Upsilon \vdash s' \overset{\curlywedge}{\approx}_\alpha t'$.

     Observação 2.3. Se $\Upsilon' \vdash u \overset{\curlywedge}{\approx}_\alpha v$, então, por uma simples indução nessa derivação, temos que $\mathtt{Var}(u) = \mathtt{Var}(v)$, para qualqer contexto Υ' e quaisquer $u, v \in T(\Sigma, \mathbb{X}, \mathbb{A})$.

     Pela propriedade de enfraquecimento (Prop. 2.2), $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash s' \overset{\curlywedge}{\approx}_\alpha t'$, logo podemos usar o sequente em (2.1), a observação acima – para trocar $\mathtt{Var}(s')$ por $\mathtt{Var}(t')$ – e a hipótese de indução para concluir que

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1) \cdot t',$$

     e usando a regra ($\curlywedge$**abs**) segue que $\Upsilon \vdash \pi \curlywedge [a]t'$.
  2. a última regra aplicada foi $(\overset{\curlywedge}{\approx}_\alpha \mathbf{ab})$ e força-se $t = [b]t'$;
     Com isso $\Upsilon \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha [b]t'$ e, pela invertibilidade, temos

$$\Upsilon \vdash s' \overset{\curlywedge}{\approx}_\alpha (a\ b) \cdot t' \tag{2.2}$$

$$\text{e } \Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (a\ c_1) \curlywedge t'. \tag{2.3}$$

     Queremos mostrar que $\Upsilon \vdash \pi \curlywedge [b]t'$, mas para isso, precisamos primeiramente mostrar que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (b\ c_1) \cdot t'$.
     De (2.2) e pela propriedade de enfraquecimento, temos

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash s' \overset{\curlywedge}{\approx}_\alpha (a\ b) \cdot t'.$$

     Note que $\mathtt{Var}(t') = \mathtt{Var}((a\ b) \cdot t')$, pois $(a\ b) \cdot t'$ é apenas uma abreviação e não interfere no conjunto de variáveis de t'. Com isso, podemos usar o sequente em (2.1), a obs. 2.3 – para trocar $\mathtt{Var}(s')$ por $\mathtt{Var}(t')$ – e a hipótese de indução para obter $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1) \cdot ((a\ b) \cdot t')$, ou equivalentemente por

equivariância (Lema 2.3),

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi^{(b\ a)(c_1\ a)} \curlywedge t'. \tag{2.4}$$

Observe que, pelo Lema A.4, temos

$$\mathtt{dom}(\pi^{(c_1\ b)}) \subseteq \mathtt{dom}(\pi^{(b\ a)(c_1\ a)}) \cup \mathtt{dom}((a\ c_1)),$$

e com (2.3) e (2.4) podemos usar o Lema A.8 para concluir que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi^{(c_1\ b)} \curlywedge t'$ o que equivale a $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (b\ c_1) \cdot t'$, como queríamos. O resultado segue ao aplicar a regra ($\curlywedge$**abs**).

□

O resultado que segue é um caso de fortalecimento do contexto para quando se tem uma restrição primitiva cuja permutação contém apenas átomos que não ocorrem no restante do sequente.

Lema 2.5. Suponha que c_1 e c_2 sejam átomos que não ocorrem em Υ, π, s e t. Assim,

(i) $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge t$ se, e somente se, $\Upsilon \vdash \pi \curlywedge t$; e

(ii) $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s,t)} \vdash s \overset{\curlywedge}{\approx}_\alpha t$ se, e somente se, $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$.

Demonstração. A demonstração pode ser encontrada no Apêndice A, Lema A.10. □

O próximo resultado, comentado ao introduzir o capítulo, mostra a propriedade de *correção* para a relação de ponto fixo, a qual afirma que a relação $\curlywedge$ é de fato uma relação de ponto fixo para termos nominais.

Teorema 2.1 (Correção)**.** Sejam Υ, π e t um contexto de ponto fixo, uma permutação e um termo nominal, respectivamente. Então $\Upsilon \vdash \pi \curlywedge t$ se, e somente se, $\Upsilon \vdash \pi \cdot t \overset{\curlywedge}{\approx}_\alpha t$.

Demonstração. Como de costume, a prova nas duas direções é por indução na estrutura de t.
(Base da indução)

- $t = a$

  Nesse caso, por um lado, temos que $\Upsilon \vdash \pi \curlywedge a$, e a última e única regra aplicável nessa derivação é ($\curlywedge$**a**), logo $\pi(a) = a$. Com isso, podemos usar a regra ($\overset{\curlywedge}{\approx}_\alpha$ **a**) para deduzir que $\Upsilon \vdash \pi \cdot a \overset{\curlywedge}{\approx}_\alpha a$. Por outro lado, suponha que $\Upsilon \vdash \pi \cdot a \overset{\curlywedge}{\approx}_\alpha a$ seja derivável. Mas isso só acontece quando $\pi \cdot a = a$, isto é, $\pi(a) = a$. Assim, usando a regra ($\curlywedge$**a**), concluimos que $\Upsilon \vdash \pi \curlywedge a$, e o resultado segue.

- $t = \rho \cdot X$

  Nesse caso, observe que pela invertibilidade das regras,

$$\begin{array}{lll} \Upsilon \vdash \pi \curlywedge \rho \cdot X & \text{se, e somente se,} & \mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \\ & \text{se, e somente se,} & \mathtt{dom}(\rho^{-1} \circ \pi \circ \rho) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \\ & \text{se, e somente se,} & \Upsilon \vdash (\pi \circ \rho) \cdot X \overset{\curlywedge}{\approx}_\alpha \rho \cdot X \\ & \text{se, e somente se,} & \Upsilon \vdash \pi \cdot (\rho \cdot X) \overset{\curlywedge}{\approx}_\alpha \rho \cdot X \end{array}$$

  Com isso o resultado segue.

(Passo indutivo)

- $t = \mathsf{f}(t_1, \ldots, t_n)$

  Nesse caso, temos como hipótese de indução que,

$$\textit{para todo } 1 \leq i \leq n, \ \Upsilon' \vdash \pi' \curlywedge t_i \textit{ se, e somente se } \Upsilon' \vdash \pi' \cdot t_i \overset{\curlywedge}{\approx}_\alpha t_i. \tag{HI.1}$$

  Observe que usando novamente a invertibilidade das regras, temos

$$\begin{array}{lll} \Upsilon \vdash \pi \curlywedge \mathsf{f}(t_1, \ldots, t_n) & \text{se, e somente se,} & \Upsilon \vdash \pi \curlywedge t_i, \text{ para cada } 1 \leq i \leq n \\ & \overset{\text{(HI.1)}}{\text{se, e somente se,}} & \Upsilon \vdash \pi \cdot t_i \overset{\curlywedge}{\approx}_\alpha t_i, \text{ para cada } 1 \leq i \leq n \\ & \text{se, e somente se,} & \Upsilon \vdash \mathsf{f}(\pi \cdot t_1, \ldots, \pi \cdot t_n) \overset{\curlywedge}{\approx}_\alpha \mathsf{f}(t_1, \ldots, t_n) \\ & \text{se, e somente se,} & \Upsilon \vdash \pi \cdot \mathsf{f}(t_1, \ldots, t_n) \overset{\curlywedge}{\approx}_\alpha \mathsf{f}(t_1, \ldots, t_n) \end{array}$$

  E com isso o resultado segue.

- $t = [a]t'$

  Nesse caso, a hipótese de indução é que

$$\Upsilon' \vdash \pi' \curlywedge t' \textit{ se, e somente se, } \Upsilon' \vdash \pi' \cdot t' \overset{\curlywedge}{\approx}_\alpha t'. \tag{HI.2}$$

  - caso $\pi(a) = a$:

Queremos provar que $\Upsilon \vdash \pi \curlywedge [a]t'$ se, e somente se, $\Upsilon \vdash [a]\pi \cdot t' \overset{\curlywedge}{\approx}_\alpha [a]t'$. Observe que:

$$
\begin{array}{lll}
\Upsilon \vdash \pi \curlywedge [a]t' & \text{se, e somente se,} & \underbrace{\Upsilon,\ \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')}}_{\Upsilon'} \vdash \pi \curlywedge (a\ c_1) \cdot t' \\
 & \overset{\text{(HI.2)}}{\text{se, e somente se,}} & \Upsilon' \vdash \pi \cdot ((a\ c_1) \cdot t') \overset{\curlywedge}{\approx}_\alpha (a\ c_1) \cdot t' \\
 & \overset{\text{Lema A.1}}{\text{se, e somente se,}} & \Upsilon' \vdash ((\pi(a)\ c_1) \circ \pi) \cdot t' \overset{\curlywedge}{\approx}_\alpha (a\ c_1) \cdot t' \\
 & \text{se, e somente se,} & \Upsilon' \vdash ((a\ c_1) \circ \pi) \cdot t' \overset{\curlywedge}{\approx}_\alpha (a\ c_1) \cdot t' \\
 & \overset{\text{Lema 2.3}}{\text{se, e somente se,}} & \Upsilon' \vdash \pi \cdot t' \overset{\curlywedge}{\approx}_\alpha t',
\end{array}
$$

onde c_1 e c_2 são nomes novos.

Com isso, por um lado, usamos o Lema 2.5 para concluir que $\Upsilon \vdash \pi \cdot t' \overset{\curlywedge}{\approx}_\alpha t'$, o que implica $\Upsilon \vdash [a]\pi \cdot t' \overset{\curlywedge}{\approx}_\alpha [a]t'$.

Por outro lado, se $\Upsilon \vdash [a]\pi \cdot t' \overset{\curlywedge}{\approx}_\alpha [a]t'$, usamos a propriedade de enfraquecimento (Proposição 2.2) para obter $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash [a]\pi \cdot t' \overset{\curlywedge}{\approx}_\alpha [a]t'$, o que, pela inversão, implica $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \cdot t' \overset{\curlywedge}{\approx}_\alpha t'$, e pelo raciocínio acima, $\Upsilon \vdash \pi \curlywedge [a]t'$. E assim concluimos o que queríamos.

– caso $\pi(a) = b$:

Queremos provar que $\Upsilon \vdash \pi \curlywedge [a]t'$ se, e somente se, $\Upsilon \vdash [b]\pi \cdot t' \overset{\curlywedge}{\approx}_\alpha [a]t'$.

Por um lado, supondo que $\Upsilon \vdash \pi \curlywedge [a]t'$, precisamos mostrar que

$$
\begin{cases}
\Upsilon \vdash \pi \cdot t' \overset{\curlywedge}{\approx}_\alpha (b\ a) \cdot t' \text{ e} \\
\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (b\ c_1) \curlywedge t'.
\end{cases}
$$

Observe que:

$$
\begin{array}{lll}
\Upsilon \vdash \pi \curlywedge [a]t' & \text{se, e somente se,} & \underbrace{\Upsilon,\ \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')}}_{\Upsilon'} \vdash \pi \curlywedge (a\ c_1) \cdot t' \\
 & \overset{\text{(HI.2)}}{\text{se, e somente se,}} & \Upsilon' \vdash \pi \cdot ((a\ c_1) \cdot t') \overset{\curlywedge}{\approx}_\alpha (a\ c_1) \cdot t' \\
 & \overset{\text{Lema A.1}}{\text{se, e somente se,}} & \Upsilon' \vdash ((\pi(a)\ c_1) \circ \pi) \cdot t' \overset{\curlywedge}{\approx}_\alpha (a\ c_1) \cdot t' \\
 & \text{se, e somente se,} & \Upsilon' \vdash ((b\ c_1) \circ \pi) \cdot t' \overset{\curlywedge}{\approx}_\alpha (a\ c_1) \cdot t' \\
 & \overset{\text{Lema 2.3}}{\text{se, e somente se,}} & \Upsilon' \vdash ((c_1\ a) \circ (b\ c_1) \circ \pi) \cdot t' \overset{\curlywedge}{\approx}_\alpha t' \\
 & \overset{\text{(HI.2)}}{\text{se, e somente se,}} & \Upsilon' \vdash (c_1\ a) \circ (b\ c_1) \circ \pi \curlywedge t',
\end{array}
$$

onde c_1 e c_2 são nomes novos. Note que escrevendo $\gamma = (c_1\ a) \circ (b\ c_1) \circ \pi$, temos que $\{b, c_1\} \subseteq \mathtt{dom}(\gamma)$, já que $\gamma(b) \neq b$ e $\gamma(c_1) \neq c_1$. Logo, pelo Lema A.8,

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (b\ c_1) \curlywedge t'.$$

Agora, tomando $\rho = (a\ b) \circ \pi$, vemos também que $\mathtt{dom}(\rho) \subseteq \mathtt{dom}(\gamma)$, basta analisar os domínios de ρ e γ e notar que

$$\mathtt{dom}(\rho) = \mathtt{dom}(\pi) \setminus \{a\} \ \text{ e } \ \mathtt{dom}(\gamma) = (\mathtt{dom}(\pi) \cup \{c_1\}) \setminus \{a\}.$$

E, novamente pelo Lema A.8, temos que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (a\ b) \circ \pi \curlywedge t'$, o que, por (HI.2) e pela equivariância, implica

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \cdot t' \overset{\curlywedge}{\approx}_\alpha (b\ a) \cdot t' \ \text{ se, e } \overset{\text{Lema 2.5}}{\text{somente}} \text{ se, } \ \Upsilon \vdash \pi \cdot t' \overset{\curlywedge}{\approx}_\alpha (b\ a) \cdot t'.$$

Logo, podemos usar a regra ($\overset{\curlywedge}{\approx}_\alpha$ **ab**) para concluir que $\Upsilon \vdash [b]\pi \cdot t' \overset{\curlywedge}{\approx}_\alpha [a]t'$.

Por outro lado, supondo que $\Upsilon \vdash [b]\pi \cdot t' \overset{\curlywedge}{\approx}_\alpha [a]t'$, precisamos mostrar que $\Upsilon,\ \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1) \cdot t'$.
Pela invertibilidade da regra ($\overset{\curlywedge}{\approx}_\alpha$ **ab**) e pela propriedade de enfraquecimento e de equivariância, temos que

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (b\ c_1) \curlywedge t' \tag{2.5}$$

e $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash ((a\ b) \circ \pi) \cdot t' \overset{\curlywedge}{\approx}_\alpha t'$, o que implica por (HI.2) que

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (a\ b) \circ \pi \curlywedge t'. \tag{2.6}$$

Assim, usando o Lema A.5, temos que

$$\mathtt{dom}(\pi^{(c_1\ a)}) \subseteq \mathtt{dom}((a\ b) \circ \pi) \cup \mathtt{dom}((b\ c_1)),$$

e pelo Lema A.8, obtemos $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi^{(c_1\ a)} \curlywedge t'$, que por equivariância implica $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1) \cdot t'$, como queríamos. Por fim, basta aplicar a regra ($\curlywedge$**abs**) para concluir que $\Upsilon \vdash \pi \curlywedge [a]t'$, e o resultado segue.

□

Proposição 2.3 (Preservação por substituição)**.** Sejam Υ e Υ' contextos e suponha que $\Upsilon \vdash \Upsilon'\sigma$. Então,

(i) se $\Upsilon' \vdash \pi \curlywedge s$, então $\Upsilon \vdash \pi \curlywedge s\sigma$; e

(ii) se $\Upsilon' \vdash s \overset{\curlywedge}{\approx}_\alpha t$, então $\Upsilon \vdash s\sigma \overset{\curlywedge}{\approx}_\alpha t\sigma$.

Demonstração. A demonstração pode ser encontrada no Apêndice A, Proposição A.3. □

Exemplo 2.4. Mostramos no Exemplo 2.2, item 3, que $\Upsilon' \vdash (a\ d) \curlywedge [a]\mathsf{h}(a,c,X)$ é derivável, onde $\Upsilon' = \{(a\ b),(c\ d) \curlywedge X\}$. Tome $\sigma = [X \mapsto (d\ e)\cdot Z]$ e seja $\Upsilon = \{(a\ c),(b\ e) \curlywedge Z\}$. Note que:

- como $\mathtt{dom}((a\ b)^{(d\ e)}) = \{a,b\} \subset \mathtt{dom}(\mathtt{perm}(\Upsilon|_Z))$, vale $\Upsilon \vdash (a\ b) \curlywedge (d\ e)\cdot Z$; e
- como $\mathtt{dom}((c\ d)^{(d\ e)}) = \{c,e\} \subset \mathtt{dom}(\mathtt{perm}(\Upsilon|_Z))$, também vale $\Upsilon \vdash (c\ d) \curlywedge (d\ e)\cdot Z$.

Ou seja, $\Upsilon \vdash \Upsilon'\sigma$. Observe que $\big([a]\mathsf{h}(a,c,X)\big)\sigma = [a]\mathsf{h}(a,c,(d\ e)\cdot Z)$. Logo, pela propriedade de Preservação por substituição, temos que $\Upsilon \vdash (a\ d) \curlywedge [a]\mathsf{h}(a,c,(d\ e)\cdot Z)$.

A partir dos dois mapeamentos apresentados na introdução, em (1.2) e (1.3), Ayala-Rincón et al. [AFN20] mostram que as relações de α-equivalência axiomatizadas via ponto fixo e a tradicional (via *freshness* [GP02]) são equivalentes. E já que Urban et al. [UPG04] mostram que a relação nessa última abordagem é uma relação de equivalência, então $\overset{\curlywedge}{\approx}_\alpha$ também o é.

Teorema 2.2. A relação $\overset{\curlywedge}{\approx}_\alpha$ é uma relação de equivalência.

2.4 O Suporte de um Termo Nominal

Definiremos a seguir o conceito de *suporte* de um termo nominal que faz uso da noção de *termos sob contexto*[1]: um par (Υ, s), formado por um contexto de ponto fixo Υ e um termo nominal s, para o qual usamos a notação $\Upsilon \vdash s$. De maneira breve, o suporte de um termo t sob contexto Υ é o menor conjunto finito de átomos tal que se uma permutação π fixar cada um de seus elementos, o sequente $\Upsilon \vdash \pi \curlywedge t$ é derivável.

Definição 2.9. Seja $\Upsilon \vdash t$ um termo sob contexto. O *suporte* de t com relação a Υ, denotado por $\mathtt{supp}_\Upsilon(t)$, é o menor conjunto finito A de átomos tal que, para qualquer permutação π,

$$(\pi(a) = a,\ \forall a \in A) \Rightarrow \Upsilon \vdash \pi \curlywedge t.$$

Com o próximo exemplo, vamos investigar o suporte do termo em contexto $\Upsilon \vdash X$.

[1] do inglês 'terms-in-context'

Exemplo 2.5. Seja $\Upsilon = \{(d_1\ d_2) \curlywedge X, (d_1\ d_3) \curlywedge X\}$ e considere um subconjunto finito $P \subset \mathbb{A} \setminus \{d_1, d_2, d_3\}$. Observe que toda permutação π tal que $\pi(a) = a$, para todo $a \in P$, deve ter a propriedade de que $\mathtt{dom}(\pi) \subseteq \{d_1, d_2, d_3\}$, já que

$$\Upsilon \vdash \pi \curlywedge X, \text{ se } \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) = \{d_1, d_2, d_3\}.$$

Por definição, $\mathtt{supp}_\Upsilon(X)$ deve ser o menor conjunto que satisfaz tal propriedade, então $\mathtt{supp}_\Upsilon(X) \subseteq P$. Além disso, observe também que vale $\mathtt{dom}(\pi) \cap \mathtt{supp}_\Upsilon(X) = \emptyset$.

Teorema 2.3 (Caracterização do suporte de um termo)**.**

(i) $\mathtt{supp}_\Upsilon(a) = \{a\}$.

(ii) $\mathtt{supp}_\Upsilon(\rho \cdot X) \subset \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$

(iii) $\mathtt{supp}_\Upsilon(\mathtt{f}(t_1, \ldots, t_n)) = \bigcup_{i=1}^{n} \mathtt{supp}_\Upsilon(t_i)$.

(iv) $\mathtt{supp}_\Upsilon([b]t) = \mathtt{supp}_\Upsilon(t) \setminus \{b\}$.

Demonstração. A prova deste teorema é exaustiva e pode ser encontrada no Apêndice A, Seção A.3: Lema A.11, Lema A.12 e Lema A.14. □

A seguir ilustraremos a caracterização do suporte de um termo com alguns exemplos:

Exemplo 2.6. Seja $\Upsilon = \{(c_1\ c_2) \curlywedge X, (c_1\ c_3) \curlywedge X\}$. Vamos analisar $\mathtt{supp}_\Upsilon([b]X)$:
Pelo Teorema 2.3 temos que $\mathtt{supp}_\Upsilon([b]X) \subset \mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{b\}) = \mathbb{A} \setminus \{c_1, c_2, c_3, b\}$.

De fato, considere a permutação π tal que $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{b\}$. Note que $\pi(a) = a$ para todo $a \in \mathbb{A} \setminus \{c_1, c_2, c_3, b\}$. Além disso, $\Upsilon \vdash \pi \curlywedge [b]X$ pois, para $b_1, b_2 \in \mathbb{A} \setminus \{c_1, c_2, c_3, b\}$, vale $\Upsilon, (b_1\ b_2) \curlywedge X \vdash \pi \curlywedge (b\ b_1) \cdot X$. De fato,

- se $b \notin \mathtt{dom}(\pi)$ então:

$$\mathtt{dom}(\pi^{(b\ b_1)}) = \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{b_1, b_2\}$$

- se $b \in \mathtt{dom}(\pi)$ então:

$$\mathtt{dom}(\pi^{(b\ b_1)}) \overset{\text{Lema A.3}}{=} \mathtt{dom}(\pi) \setminus \{b\} \cup \{b_1\} \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{b_1, b_2\}$$

e o resultado segue.

Exemplo 2.7. Seja $\Upsilon = \{(c_1\ c_2) \curlywedge X, (c_1\ c_3) \curlywedge X\}$.

Então $\mathtt{supp}_\Upsilon(b) = \{b\}$ e $\mathtt{supp}_\Upsilon(c_1) = \{c_1\}$. De fato, para todo π, tal que $\pi(b) = b$, vale $\Upsilon \vdash \pi \curlywedge b$, e similarmente, para $\Upsilon \vdash \pi \curlywedge c_1$.

Portanto, $\mathtt{supp}_\Upsilon(\mathtt{f}(b, c_1)) = \{b, c_1\} = \mathtt{supp}_\Upsilon(c_1) \cup \mathtt{supp}_\Upsilon(b)$.

Exemplo 2.8. Seja $\Upsilon = \{(c_1\ c_2) \curlywedge X, (c_1\ c_3) \curlywedge X\}$.

Note que $\mathtt{supp}_\Upsilon(c_1) = \{c_1\}$ e pelo Teorema 2.3 temos que $\mathtt{supp}_\Upsilon(X) \subset \mathbb{A} \setminus \{c_1, c_2, c_3\}$.

De fato, para todo π tal que $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \setminus \{c_1\}$, vale que $\pi(a) = a$ para todo $a \in \mathbb{A} \setminus \{c_2, c_3\}$. Além disso, $\Upsilon \vdash \pi \curlywedge \mathtt{f}(c_1, X)$ uma vez que:

- $\Upsilon \vdash (c_2\ c_3) \curlywedge c_1$, pois $(c_2\ c_3)(c_1) = c_1$; e
- $\Upsilon \vdash (c_2\ c_3) \curlywedge X$, pois $\{c_2, c_3\} \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

Logo, $\mathtt{supp}_\Upsilon(\mathtt{f}(c_1, X)) \subset \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{c_1\}$.

Finalmente, conseguimos estabelecer uma relação entre o suporte de um termo e a derivabilidade de restrições de igualdade e ponto fixo.

Teorema 2.4.

(i) Se $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$ então $\mathtt{supp}_\Upsilon(s) = \mathtt{supp}_\Upsilon(t)$

(ii) $\Upsilon \vdash \pi \curlywedge t$ se, e somente se, $\mathtt{dom}(\pi) \cap \mathtt{supp}_\Upsilon(t) = \emptyset$.

Demonstração. A prova deste teorema encontra-se no Apêndice A, Teorema A.1. □

Capítulo 3

Unificação Nominal via Ponto Fixo

Introduzida e desenvolvida por Ayala-Rincón et al. [AFN20], a Unificação Nominal via ponto fixo, baseada nas relações $\overset{\curlywedge}{\approx}_\alpha$ e $\curlywedge$ apresentadas anteriormente, trata de resolver o problema da satisfatibilidade de restrições do tipo $s \overset{\curlywedge ?}{\approx}_\alpha t$ e $\pi \curlywedge^? t$. Mais especificamente, dados os termos $s,t \in T(\Sigma,\mathbb{X},\mathbb{A})$ e uma permutação π, o problema consiste em verificar se existem um contexto Φ e uma substituição σ para os quais valem $\Phi \vdash s\sigma \overset{\curlywedge}{\approx}_\alpha t\sigma$ e $\Phi \vdash \pi \curlywedge t\sigma$.

Neste capítulo, discorremos acerca desse tema apresentando as definições e resultados necessários para a construção do algoritmo $\texttt{unif}_\curlywedge$ (Tabela 3.1), que computa uma solução para um problema de unificação nominal via ponto fixo. Mostraremos que o algoritmo é terminante (Teorema 3.1), preserva soluções (Lema 3.4) e é correto (Teoremas 3.2 e 3.3). A utilização desse procedimento será importante para alcançar o objetivo principal deste trabalho, que será discutido no próximo capítulo.

Vale ressaltar que apesar deste capítulo ser baseado em partes do trabalho desenvolvido por Ayala-Rincón et al. em [AFN20], foram acrescentados novos exemplos, algumas definições foram ajustadas e as provas dos resultados comuns a ambos foram refeitas retirando-se o quantificador $\mathsf{И}$. Além disso, a última seção sobre *emparelhamento sob contexto* é uma reformulação inédita que foi necessária para se ter um tratamento adequado quando *nomes novos* são considerados.

3.1 Definições Básicas

Introduzimos nesta seção as definições formais de um problema de unificação nominal e suas soluções, bem como a ordem parcial que as compara e que possibilita definir uma solução principal ou mais geral para um problema.

Definição 3.1. Um *problema de unificação nominal* $\mathtt{Pr}$ é um conjunto finito de restrições de igualdade e de ponto fixo da forma $s \overset{\curlywedge ?}{\approx}_\alpha t$ e $\pi \curlywedge^? t$, respectivamente.

Na sequência de todo o capítulo, a menos que seja dito o contrário, um "problema $\mathtt{Pr}$" se refere a um "problema de unificação nominal $\mathtt{Pr}$" como definido acima.

Definição 3.2. Uma *solução* para um problema $\mathtt{Pr}$ é um par $\langle \Phi, \sigma \rangle$ onde são satisfeitas as seguintes condições:

(i) $\Phi \vdash \pi \curlywedge t\sigma$, se $\pi \curlywedge^? t \in \mathtt{Pr}$,

(ii) $\Phi \vdash s\sigma \overset{\curlywedge}{\approx}_\alpha t\sigma$, se $s \overset{\curlywedge ?}{\approx}_\alpha t \in \mathtt{Pr}$,

e o conjunto de soluções para $\mathtt{Pr}$ é denotado por $\mathcal{U}(\mathtt{Pr})$.

Notação:

1. Quando for conveniente, escreveremos $\Phi \vdash \mathtt{Pr}\sigma$ para dizer que $\Phi \vdash \pi \curlywedge t\sigma$ e $\Phi \vdash s\sigma \overset{\curlywedge}{\approx}_\alpha t\sigma$, para todo $\pi \curlywedge^? t$ e $s \overset{\curlywedge ?}{\approx}_\alpha t$ em $\mathtt{Pr}$.

2. Sejam σ e σ' duas substituições. Escrevemos

$$\Phi \vdash \sigma \overset{\curlywedge}{\approx}_\alpha \sigma'$$

para dizer que $\Phi \vdash X\sigma \overset{\curlywedge}{\approx}_\alpha X\sigma'$, para todo $X \in \mathbb{X}$.

As soluções de um problema $\mathtt{Pr}$, elementos do conjunto $\mathcal{U}(\mathtt{Pr})$, podem ser comparadas utilizando a *ordem de instanciação* definida a seguir:

Definição 3.3 (Ordem de instanciação)**.** Sejam Φ_1, Φ_2 contextos e σ_1, σ_2 substituições. Dizemos que $\langle \Phi_2, \sigma_2 \rangle$ é uma *instância* de $\langle \Phi_1, \sigma_1 \rangle$, e denotamos $\langle \Phi_1, \sigma_1 \rangle \lesssim \langle \Phi_2, \sigma_2 \rangle$, quando existe alguma substituição δ tal que

$$\Phi_2 \vdash \sigma_2 \overset{\curlywedge}{\approx}_\alpha \sigma_1\delta \;\text{ e }\; \Phi_2 \vdash \Phi_1\delta.$$

Se quisermos explicitar a substituição δ, escrevemos $\langle \Phi_1, \sigma_1 \rangle \lesssim_\delta \langle \Phi_2, \sigma_2 \rangle$. Além disso, no caso em que $\langle \Phi_1, \sigma_1 \rangle \lesssim \langle \Phi_2, \sigma_2 \rangle$ e $\langle \Phi_2, \sigma_2 \rangle \lesssim \langle \Phi_1, \sigma_1 \rangle$, dizemos que são pares *equivalentes* e escrevemos $\langle \Phi_1, \sigma_1 \rangle \sim \langle \Phi_2, \sigma_2 \rangle$.

A ordem de instanciação é de fato uma ordem parcial (i.e, é reflexiva, transitiva) no conjunto de soluções de um problema $\mathtt{Pr}$.

Lema 3.1. A relação $\lesssim$ define uma ordem parcial entre os elementos de $\mathcal{U}(\mathtt{Pr})$.

Demonstração. De fato, $\lesssim$ satisfaz as propriedades:

1. Reflexividade:

   Basta notar que para todo par $\langle \Phi, \sigma \rangle$ vale: $\langle \Phi, \sigma \rangle \lesssim_{\mathtt{id}} \langle \Phi, \sigma \rangle$.

2. Transitividade:

   Suponha que $\langle \Phi_1, \sigma_1 \rangle \lesssim_{\delta_1} \langle \Phi_2, \sigma_2 \rangle$ e $\langle \Phi_2, \sigma_2 \rangle \lesssim_{\delta_2} \langle \Phi_3, \sigma_3 \rangle$. Então

   - $\Phi_2 \vdash \sigma_2 \overset{\curlywedge}{\approx}_\alpha \sigma_1 \delta_1$ e $\Phi_2 \vdash \Phi_1 \delta_1$; e
   - $\Phi_3 \vdash \sigma_3 \overset{\curlywedge}{\approx}_\alpha \sigma_2 \delta_2$ e $\Phi_3 \vdash \Phi_2 \delta_2$.

   Tome $\delta = \delta_1 \delta_2$. Como $\Phi_3 \vdash \Phi_2 \delta_2$ e $\Phi_2 \vdash \Phi_1 \delta_1$, segue da preservabilidade por substituições (Proposição 2.3) que $\Phi_3 \vdash \Phi_1 \delta_1 \delta_2$, isto é, $\Phi_3 \vdash \Phi_1 \delta$.

   Além disso, como $\Phi_2 \vdash \sigma_2 \overset{\curlywedge}{\approx}_\alpha \sigma_1 \delta_1$, ainda pela Proposição 2.3, podemos concluir que $\Phi_3 \vdash \sigma_2 \delta_2 \overset{\curlywedge}{\approx}_\alpha \sigma_1 \delta_1 \delta_2$. Como $\overset{\curlywedge}{\approx}_\alpha$ é relação de equivalência, segue da transitividade, que $\Phi_3 \vdash \sigma_3 \overset{\curlywedge}{\approx}_\alpha \sigma_1 \delta_1 \delta_2$, isto é,

   $$\Phi_3 \vdash \sigma_3 \overset{\curlywedge}{\approx}_\alpha \sigma_1 \delta.$$

   Logo $\langle \Phi_1, \sigma_1 \rangle \lesssim_\delta \langle \Phi_3, \sigma_3 \rangle$.

□

Exemplo 3.1. Considere o problema de unificação $\mathtt{Pr} = \{\mathtt{g}(\mathtt{f}\ Y) \overset{\curlywedge ?}{\approx}_\alpha \mathtt{g}\ X\}$.

Se $\sigma_1 = [X \mapsto \mathtt{f}\ Y]$, é fácil ver que o par $\langle \emptyset, \sigma_1 \rangle$ é uma solução para $\mathtt{Pr}$, já que

$$\emptyset \vdash (\mathtt{g}(\mathtt{f}\ Y))\sigma_1 \overset{\curlywedge}{\approx}_\alpha (\mathtt{g}\ \mathsf{X})\sigma_1.$$

Observe que se tomarmos $\sigma_2 = [X \mapsto \mathtt{f}\ X,\ Y \mapsto X]$, então o par $\langle \emptyset, \sigma_2 \rangle$ é instância de $\langle \emptyset, \sigma_1 \rangle$ por $\delta = [Y \mapsto X]$, isto é, $\langle \emptyset, \sigma_1 \rangle \lesssim_\delta \langle \emptyset, \sigma_2 \rangle$.

Além disso, $\langle \emptyset, \sigma_2 \rangle$ é solução para $\mathtt{Pr}$.

Definição 3.4. Um par $\langle \Phi, \sigma \rangle$ é dito ser uma *solução mais geral* (*smg*) para um problema $\mathtt{Pr}$ quando é um menor elemento de $\mathcal{U}(\mathtt{Pr})$ com respeito a $\lesssim$, isto é, para todo $\langle \Phi', \sigma' \rangle \in \mathcal{U}(\mathtt{Pr})$, $\langle \Phi, \sigma \rangle \lesssim \langle \Phi', \sigma' \rangle$.

Problemas de unificação são *fechados por instanciação*, isto é, toda instância de uma solução também é uma solução. O próximo lema retrata examente essa propriedade.

Lema 3.2 (Fechamento por Instanciação)**.** Seja $\mathtt{Pr}$ um problema de unificação nominal e tome $\langle\Phi,\sigma\rangle \in \mathcal{U}(\mathtt{Pr})$. Se $\langle\Phi,\sigma\rangle \lesssim \langle\Delta,\lambda\rangle$, então $\langle\Delta,\lambda\rangle \in \mathcal{U}(\mathtt{Pr})$.

Demonstração. De fato, como $\langle\Phi,\sigma\rangle \in \mathcal{U}(\mathtt{Pr})$, então $\Phi \vdash \mathtt{Pr}\sigma$. Agora suponha que $\langle\Phi,\sigma\rangle \lesssim_\delta \langle\Delta,\lambda\rangle$, ou seja,

$$\Delta \vdash \lambda \overset{\curlywedge}{\approx}_\alpha \sigma\delta \;\text{ e }\; \Delta \vdash \Phi\delta.$$

Com isso e pela Preservabilidade por substituições (Proposição 2.3), segue que $\Delta \vdash \mathtt{Pr}\lambda$, logo, $\langle\Delta,\lambda\rangle \in \mathcal{U}(\mathtt{Pr})$. □

3.2 O Algoritmo $\mathtt{unif}_{\curlywedge}$

A Tabela 3.1 a seguir apresenta as *Regras de Simplificação* para problemas de unificação nominal, as quais são uma extensão das tradicionais regras de Martelli e Montanari para unificação de primeira ordem [BN98] com adições de regras para o tratamento de restrições de ponto fixo e construtores de termos nominais, próprios da abordagem usada no presente trabalho. Elas permitem o desenvolvimento do algoritmo que chamamos de $\mathtt{unif}_{\curlywedge}$, fornecendo, cada uma, um passo do procedimento.

Essas regras agem em problemas de unificação transformando-os em outros "mais simples", ora retirando restrições ora adicionando restrições cujos termos envolvidos possuem uma estrutura simplificada, característica que é usufruída pela prova de terminação (veja Lema 3.1), pois diminui o "tamanho de restrições". Os conceitos formais de problemas "mais simples" e "tamanho de restrições" serão estabelecidos mais adiante.

Notação:
Para abreviar tuplas $(t_1,\ldots,t_n)$ escrevemos $\widetilde{(t)}_n$, e para um termo t, usamos a notação $\overline{\pi \curlywedge^? \mathtt{Var}(t)} := \{\pi \curlywedge^? X \mid X \in \mathtt{Var}(t)\}$. Além disso, usamos o símbolo $\uplus$ para indicar uma união disjunta e para um problema $\mathtt{Pr}$ e uma substituição σ, escrevemos $\mathtt{Pr}\sigma$ para representar o conjunto $\{\pi \curlywedge^? t\sigma,\ s\sigma \overset{\curlywedge^?}{\approx}_\alpha t\sigma \mid \pi \curlywedge^? t, s \overset{\curlywedge^?}{\approx}_\alpha t \in \mathtt{Pr}\}$.

$(\curlywedge a)$	$\mathtt{Pr} \uplus \{\pi \curlywedge^{?} a\}$	$\Longrightarrow \mathtt{Pr}$, se $\pi(a) = a$
$(\curlywedge f)$	$\mathtt{Pr} \uplus \{\pi \curlywedge^{?} \mathtt{f}(\widetilde{t})_n\}$	$\Longrightarrow \mathtt{Pr} \cup \{\pi \curlywedge^{?} t_1, \ldots, \pi \curlywedge^{?} t_n\}$
$(\curlywedge abs)$	$\mathtt{Pr} \uplus \{\pi \curlywedge^{?} [a]t\}$	$\Longrightarrow \mathtt{Pr} \cup \{\pi \curlywedge^{?} (a\ c_1) \cdot t, \overline{(c_1\ c_2) \curlywedge^{?} \mathtt{Var}(t)}\}$
$(\curlywedge var)$	$\mathtt{Pr} \uplus \{\pi \curlywedge^{?} \pi' \cdot X\}$	$\Longrightarrow \mathtt{Pr} \cup \{\pi^{(\pi')^{-1}} \curlywedge^{?} X\}$, se $\pi' \neq \mathtt{Id}$
$(\overset{\curlywedge}{\approx}_\alpha del)$	$\mathtt{Pr} \uplus \{t \overset{\curlywedge^{?}}{\approx}_\alpha t\}$	$\Longrightarrow \mathtt{Pr}$
$(\overset{\curlywedge}{\approx}_\alpha f)$	$\mathtt{Pr} \uplus \{\mathtt{f}(\widetilde{t})_n \overset{\curlywedge^{?}}{\approx}_\alpha \mathtt{f}(\widetilde{t'})_n\}$	$\Longrightarrow \mathtt{Pr} \cup \{t_1 \overset{\curlywedge^{?}}{\approx}_\alpha t'_1, \ldots, t_n \overset{\curlywedge^{?}}{\approx}_\alpha t'_n\}$
$(\overset{\curlywedge}{\approx}_\alpha abs1)$	$\mathtt{Pr} \uplus \{[a]t \overset{\curlywedge^{?}}{\approx}_\alpha [a]t'\}$	$\Longrightarrow \mathtt{Pr} \cup \{t \overset{\curlywedge^{?}}{\approx}_\alpha t'\}$
$(\overset{\curlywedge}{\approx}_\alpha abs2)$	$\mathtt{Pr} \uplus \{[a]s \overset{\curlywedge^{?}}{\approx}_\alpha [b]t\}$	$\Longrightarrow \mathtt{Pr} \cup \{s \overset{\curlywedge^{?}}{\approx}_\alpha (a\ b) \cdot t, (a\ c_1) \curlywedge^{?} t, \overline{(c_1\ c_2) \curlywedge^{?} \mathtt{Var}(t)}\}$
$(\overset{\curlywedge}{\approx}_\alpha var)$	$\mathtt{Pr} \uplus \{\pi \cdot X \overset{\curlywedge^{?}}{\approx}_\alpha \pi' \cdot X\}$	$\Longrightarrow \mathtt{Pr} \cup \{(\pi')^{-1} \circ \pi \curlywedge^{?} X\}$, se $\pi \neq \pi'$
$(\overset{\curlywedge}{\approx}_\alpha inst1)$	$\mathtt{Pr} \uplus \{\pi \cdot X \overset{\curlywedge^{?}}{\approx}_\alpha t\}$	$\overset{\sigma}{\Longrightarrow} \mathtt{Pr}\sigma$, se $X \notin \mathtt{Var}(t)$, onde $\sigma = [X \mapsto \pi^{-1} \cdot t]$
$(\overset{\curlywedge}{\approx}_\alpha inst2)$	$\mathtt{Pr} \uplus \{t \overset{\curlywedge^{?}}{\approx}_\alpha \pi \cdot X\}$	$\overset{\sigma}{\Longrightarrow} \mathtt{Pr}\sigma$, se $X \notin \mathtt{Var}(t)$, onde $\sigma = [X \mapsto \pi^{-1} \cdot t]$

Tabela 3.1 Regras de Simplificação de Problemas. Em $(\curlywedge abs)$ e $(\overset{\curlywedge}{\approx}_\alpha abs2)$, c_1 e c_2 são nomes atômicos novos quaisquer com relação a $\mathtt{Pr}$.

Observe que, com exceção de $(\overset{\curlywedge}{\approx}_\alpha inst1)$ e $(\overset{\curlywedge}{\approx}_\alpha inst2)$ chamadas de *regras de instanciação*, a simplificação consiste basicamente na aplicação de baixo para cima das regras de derivação presentes nas Tabelas 2.1 e 2.2. Abaixo descrevemos com detalhes as regras mais interessantes:

- Quando identificamos uma restrição de ponto fixo do tipo $\pi \curlywedge^{?} [a]t$, aplicamos a regra $(\curlywedge abs)$ que consiste em removê-la do problema, porém, conservando as outras restrições que lá haviam e as unindo com $\{\pi \curlywedge^{?} (a\ c_1) \cdot t,\ \overline{(c_1\ c_2) \curlywedge^{?} \mathtt{Var}(t)}\}$, um conjunto de restrições cujos termos são estruturalmente mais simples que a abstração presente anteriormente e que pede uma verificação de que o átomo a não ocorra livre em t, fazendo uso de átomos c_1 e c_2 que são novos com relação ao problema;

- A regra $(\curlywedge var)$ é aplicada quando se tem uma restrição de ponto fixo $\pi \curlywedge^{?} \pi' \cdot X$ e $\pi' \neq \mathtt{Id}$, e como consequência da aplicação, ela é removida do problema e acrescenta-se a restrição de ponto fixo primitiva $\pi^{(\pi')^{-1}} \curlywedge^{?} X$ no lugar. Observe que a condição $\pi' \neq \mathtt{Id}$ garante que a regra não ocasione um *loop* no procedimento, já que evita que possa ser aplicada novamente a mesma regra em restrições primitivas;

- Caso seja identificada uma restrição de igualdade envolvendo dois termos com a abstração de átomos distintos como $[a]s \overset{\curlywedge^{?}}{\approx}_\alpha [b]t$, aplicamos a regra $(\overset{\curlywedge}{\approx}_\alpha abs2)$, que

a remove do problema e une as restrições restantes com $\{s \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot t,\ (a\ c_1) \curlywedge^? t,\ \overline{(c_1\ c_2) \curlywedge^? \mathtt{Var}(t)}\}$, um conjunto semelhante àquele visto na primeira regra descrita acima, mas nesse caso pede a verificação de uma α-equivalência entre s e a troca de b por a (e vice-versa) em t, e também de que a não ocorra livre em t;

- Por fim, as regras $(\overset{\curlywedge}{\approx}_\alpha\ inst1)$ e $(\overset{\curlywedge}{\approx}_\alpha\ inst2)$ são aplicadas ao ser identificada uma restrição de igualdade na qual um dos lados seja uma variável suspensa. Em seguida, como efeito da aplicação, essa restrição é retirada do problema bem como a variável em questão, que é instanciada por σ nas restrições restantes. Porém, observe que há uma condição prévia para se aplicar essas duas regras: que $X \notin \mathtt{Var}(t)$. Isso evita uma simplicação de uma restrição do tipo $X \overset{\curlywedge ?}{\approx}_\alpha \mathtt{f}\ X$ que pode induzir uma não terminação.

Observação 3.1. Originalmente em [AFN20], uma outra regra chamada $(\overset{\curlywedge}{\approx}_\alpha\ a)$, a qual retira do problema restrições do tipo $a \overset{\curlywedge ?}{\approx}_\alpha a$, aparece no lugar de $(\overset{\curlywedge}{\approx}_\alpha\ del)$, e a condição $\pi \neq \pi'$ em $(\overset{\curlywedge}{\approx}_\alpha\ var)$ não é considerada. Porém, escolhemos substituí-la no nosso procedimento e também adicionar tal condição na regra sobre variáveis, afim de cobrir de maneira mais eficiente a simplificação do problema

$$\{X_1 \overset{\curlywedge ?}{\approx}_\alpha X_2,\ X_2 \overset{\curlywedge ?}{\approx}_\alpha X_1\},$$

que apesar de simples pode ser tratado de duas maneiras diferentes, como discutimos a seguir:

1. Sem $(\overset{\curlywedge}{\approx}_\alpha\ del)$ e sem a condição em $(\overset{\curlywedge}{\approx}_\alpha\ var)$, teríamos que

$$\{X_1 \overset{\curlywedge ?}{\approx}_\alpha X_2,\ X_2 \overset{\curlywedge ?}{\approx}_\alpha X_1\} \overset{[X_1 \mapsto X_2]}{\Longrightarrow} \{X_2 \overset{\curlywedge ?}{\approx}_\alpha X_2\} \Longrightarrow_{(\overset{\curlywedge}{\approx}_\alpha var)} \{\mathtt{Id} \curlywedge^? X_2\},$$

ou

$$\{X_1 \overset{\curlywedge ?}{\approx}_\alpha X_2,\ X_2 \overset{\curlywedge ?}{\approx}_\alpha X_1\} \overset{[X_2 \mapsto X_1]}{\Longrightarrow} \{X_1 \overset{\curlywedge ?}{\approx}_\alpha X_1\} \Longrightarrow_{(\overset{\curlywedge}{\approx}_\alpha var)} \{\mathtt{Id} \curlywedge^? X_1\},$$

de maneira que nenhuma outra regra é aplicável, e uma tautologia seria deixada como resto;

2. Considerando $(\overset{\curlywedge}{\approx}_\alpha \mathit{del})$ e a condição em $(\overset{\curlywedge}{\approx}_\alpha \mathit{var})$, isso não acontece, pois nesse caso

$$\{X_1 \overset{\curlywedge ?}{\approx}_\alpha X_2,\ X_2 \overset{\curlywedge ?}{\approx}_\alpha X_1\} \overset{[X_1 \mapsto X_2]}{\Longrightarrow} \{X_2 \overset{\curlywedge ?}{\approx}_\alpha X_2\} \Longrightarrow_{(\overset{\curlywedge}{\approx}_\alpha del)} \{\},$$

ou

$$\{X_1 \overset{\curlywedge ?}{\approx}_\alpha X_2,\ X_2 \overset{\curlywedge ?}{\approx}_\alpha X_1\} \overset{[X_2 \mapsto X_1]}{\Longrightarrow} \{X_1 \overset{\curlywedge ?}{\approx}_\alpha X_1\} \Longrightarrow_{(\overset{\curlywedge}{\approx}_\alpha del)} \{\}.$$

Com a nossa escolha, também se evita a decomposição desnecessária de tautologias do tipo $t \overset{\curlywedge ?}{\approx}_\alpha t$, como em $\mathsf{f}\ t' \overset{\curlywedge ?}{\approx}_\alpha \mathsf{f}\ t'$ ou $[a]t' \overset{\curlywedge ?}{\approx}_\alpha [a]t'$.

Escrevemos $\mathtt{Pr} \Longrightarrow \mathtt{Pr}'$ quando $\mathtt{Pr}'$ é obtido a partir de $\mathtt{Pr}$ aplicando uma regra de simplificação, e $\overset{*}{\Longrightarrow}$ denota o fecho reflexivo-transitivo da relação $\Longrightarrow$. Se $\mathtt{Pr} \overset{*}{\Longrightarrow} \mathtt{Pr}'$ e $\mathtt{Pr}'$ é irredutível, isto é, nenhuma regra pode ser aplicada, dizemos que $\mathtt{Pr}'$ é *uma forma normal* de $\mathtt{Pr}$ e escrevemos $\mathtt{Pr}' = \langle \mathtt{Pr} \rangle_{\mathtt{nf}}$. A seguir mostramos como a aplicação das regras da Tabela 3.1 operam em um dado problema de unificação.

Exemplo 3.2. Considere o problema $\mathtt{Pr} = \{(a\ c) \curlywedge^? X,\ (b\ c) \curlywedge^? Y,\ [a]\mathsf{g}(a,X) \overset{\curlywedge ?}{\approx}_\alpha [b]\mathsf{g}(b,Y)\}$. Então,

$$\begin{array}{lll}
\mathtt{Pr} & \Longrightarrow_{(\overset{\curlywedge}{\approx}_\alpha abs2)} & \{(a\ c) \curlywedge^? X,\ (b\ c) \curlywedge^? Y,\ \mathsf{g}(a,X) \overset{\curlywedge ?}{\approx}_\alpha \mathsf{g}(a,(a\ b)\cdot Y),\ (a\ c_1) \curlywedge^? \mathsf{g}(b,Y),\ (c_1\ c_2) \curlywedge^? Y\} & (\mathtt{Pr}_1) \\
 & \Longrightarrow_{(\overset{\curlywedge}{\approx}_\alpha f)} & \{(a\ c) \curlywedge^? X,\ (b\ c) \curlywedge^? Y,\ a \overset{\curlywedge ?}{\approx}_\alpha a,\ X \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot Y,\ (a\ c_1) \curlywedge^? \mathsf{g}(b,Y),\ (c_1\ c_2) \curlywedge^? Y\} & (\mathtt{Pr}_2) \\
 & \Longrightarrow_{(\overset{\curlywedge}{\approx}_\alpha del)} & \{(a\ c) \curlywedge^? X,\ (b\ c) \curlywedge^? Y,\ X \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot Y,\ (a\ c_1) \curlywedge^? \mathsf{g}(b,Y),\ (c_1\ c_2) \curlywedge^? Y\} & (\mathtt{Pr}_3) \\
 & \Longrightarrow_{(\curlywedge f)} & \{(a\ c) \curlywedge^? X,\ (b\ c) \curlywedge^? Y,\ X \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot Y,\ (a\ c_1) \curlywedge^? b,\ (a\ c_1) \curlywedge^? Y,\ (c_1\ c_2) \curlywedge^? Y\} & (\mathtt{Pr}_4) \\
 & \Longrightarrow_{(\curlywedge a)} & \{(a\ c) \curlywedge^? X,\ (b\ c) \curlywedge^? Y,\ X \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot Y,\ (a\ c_1) \curlywedge^? Y,\ (c_1\ c_2) \curlywedge^? Y\} & (\mathtt{Pr}_5) \\
 & \overset{\sigma}{\Longrightarrow} & \{(a\ c) \curlywedge^? (a\ b)\cdot Y,\ (b\ c) \curlywedge^? Y,\ (a\ c_1) \curlywedge^? Y,\ (c_1\ c_2) \curlywedge^? Y\}, \text{onde } \sigma = [X \mapsto (a\ b)\cdot Y] & (\mathtt{Pr}_6) \\
 & \Longrightarrow_{(\curlywedge var)} & \{(b\ c) \curlywedge^? Y,\ (a\ c_1) \curlywedge^? Y,\ (c_1\ c_2) \curlywedge^? Y\} & (\mathtt{Pr}_7)
\end{array}$$

No último passo a restrição $(a\ c) \curlywedge^? (a\ b)\cdot Y$ foi reduzida para $(b\ c) \curlywedge^? Y$, onde $(b\ c)$ é o resultado da conjugação $(a\ c)^{(b\ a)}$. Como não há mais regras aplicáveis após esse passo, $\mathtt{Pr}' = \{(b\ c) \curlywedge^? Y,\ (a\ c_1) \curlywedge^? Y,\ (c_1\ c_2) \curlywedge^? Y\}$ é uma forma normal de $\mathtt{Pr}$.

3.2.1 Terminação e Confluência de $\Longrightarrow$

As regras de simplificação apresentam duas características que são essenciais para sua terminação: elas podem diminuir o número de variáveis distintas do problema ou mantê-lo e, nesse caso, reduzir o tamanho de restrições não-primitivas. A seguir, definimos o tamanho de um termo nominal e de uma restrição qualquer.

Definição 3.5.

(i) O *tamanho de um termo* $t \in T(\Sigma, \mathbb{X}, \mathbb{A})$, denotado por $|t|$, é definido indutivamente por:

$$|a| = 1 \qquad |\pi \cdot X| = 1 \qquad |\pi \cdot t| = |t|$$
$$|[a]t| = 1 + |t| \qquad |\mathsf{f}(t_1, \ldots t_n)| = 1 + |t_1| + \cdots + |t_n|$$

(ii) O *tamanho de uma restrição* é dado por:

$$|\pi \curlywedge^{?} t| = |t| \quad \text{e} \quad |s \overset{\curlywedge^{?}}{\approx}_{\alpha} t| = |s| + |t|$$

Para provar a terminação de $\Longrightarrow$, definimos uma medida para o tamanho de um problema $\mathtt{Pr}$ como sendo $[\mathtt{Pr}] = (n, M)$, onde n é o número de variáveis distintas usadas em $\mathtt{Pr}$ e M é o multiconjunto dos tamanhos das restrições de igualdade e restrições não-primitivas de ponto fixo ocorrendo em $\mathtt{Pr}$. Além disso, comparamos a medida de dois problemas usando a ordem denotada por $>_{lex}$, que consiste na combinação lexicográfica da ordem usual dos naturais ($>$) e sua extensão para multiconjuntos ($>_{mul}$).

Exemplo 3.3. Para ilustrar a medida definida acima, vamos voltar ao Exemplo 3.2 e analisar a medida dos problemas em cada passo da simplificação: $\mathtt{Pr} \Longrightarrow \mathtt{Pr}_1 \Longrightarrow \ldots \Longrightarrow \mathtt{Pr}_7$, onde $\mathtt{Pr} = \{(a\ c) \curlywedge^{?} X,\ (b\ c) \curlywedge^{?} Y,\ [a]\mathsf{g}(a,X) \overset{\curlywedge^{?}}{\approx}_{\alpha} [b]\mathsf{g}(b,Y)\}$ e os problemas indexados correspondem àqueles ali expostos respeitando a ordem em que aparecem. Assim, temos

$$[\mathtt{Pr}] = (2, \{8\}) \qquad [\mathtt{Pr}_4] = (2, \{2,1\})$$
$$[\mathtt{Pr}_1] = (2, \{6,3\}) \qquad [\mathtt{Pr}_5] = (2, \{2\})$$
$$[\mathtt{Pr}_2] = (2, \{2,2,3\}) \qquad [\mathtt{Pr}_6] = (1, \{1\})$$
$$[\mathtt{Pr}_3] = (2, \{2,3\}) \qquad [\mathtt{Pr}_7] = (1, \{\})$$

e portanto $[\mathtt{Pr}] >_{lex} [\mathtt{Pr}_1] >_{lex} \cdots >_{lex} [\mathtt{Pr}_7]$.

Agora estamos prontos para mostrar a terminação de $\Longrightarrow$ e garantir que sempre é possível se chegar a uma forma normal de um problema.

Teorema 3.1 (Terminação)**.** A relação $\Longrightarrow$ é terminante, isto é, não existe cadeia infinita de reduções $\Longrightarrow$ partindo de um problema $\mathtt{Pr}$.

Demonstração. A terminação da relação segue do fato de que a medida definida acima é estritamente decrescente com respeito à ordem $>_{lex}$. Com efeito, quando $\mathtt{Pr} \Longrightarrow \mathtt{Pr}'$, ou o número de variáveis distintas diminui caso tenha sido aplicada uma regra de instanciação, ou

ele se mantém o mesmo enquanto que restrições de $\mathtt{Pr}$ são substiuídas em $\mathtt{Pr}'$ por outras de menor tamanho, no caso em que as regras restantes são usadas. Assim, $[\mathtt{Pr}] >_{lex} [\mathtt{Pr}']$ e com isso não é possível haver uma cadeia infinita de reduções descendentes.

Apresentamos alguns casos abaixo, onde consideramos M_0 como o multiconjunto dos tamanhos das restrições de igualdade e restrições não-primitivas de ponto fixo ocorrendo em $\mathtt{Pr}_0$. Os casos restantes são análogos.

- Se $\mathtt{Pr} \Longrightarrow_{(\curlywedge var)} \mathtt{Pr}'$, então

$$[\mathtt{Pr}] = \big[\mathtt{Pr}_0 \uplus \{\pi \curlywedge^? \pi' \cdot X\}\big] = \big(|\mathtt{Var}(\mathtt{Pr})|,\ M_0 \cup \{1\}\big) \text{ e}$$
$$[\mathtt{Pr}'] = \big[\mathtt{Pr}_0 \cup \{\pi^{(\pi')^{-1}} \curlywedge^? X\}\big] = \big(|\mathtt{Var}(\mathtt{Pr}')|,\ M_0\big),$$

  onde $\pi' \neq \mathtt{Id}$. Como essa regra não tem efeito sobre o número de variáveis distintas em $\mathtt{Pr}$, segue que $|\mathtt{Var}(\mathtt{Pr})| = |\mathtt{Var}(\mathtt{Pr}')|$. Porém, $M_0 \cup \{1\} >_{mul} M_0$, logo $[\mathtt{Pr}] >_{lex} [\mathtt{Pr}']$.

- Se $\mathtt{Pr} \Longrightarrow_{(\curlywedge\approx_\alpha abs2)} \mathtt{Pr}'$, então

$$[\mathtt{Pr}] = \big[\mathtt{Pr}_0 \uplus \{[a]s \overset{\curlywedge ?}{\approx}_\alpha [b]t\}\big] = \big(|\mathtt{Var}(\mathtt{Pr})|,\ M_0 \cup \{2 + |s| + |t|\}\big) \text{ e}$$
$$[\mathtt{Pr}'] = \big[\mathtt{Pr}_0 \cup \{s \overset{\curlywedge ?}{\approx}_\alpha (a\ b) \cdot t, (a\ c_1) \curlywedge^? t,\ \overline{(c_1\ c_2) \curlywedge^? \mathtt{Var}(t)}\}\big]$$
$$= \begin{cases} \big(|\mathtt{Var}(\mathtt{Pr}')|,\ M_0 \cup \{|s| + |t|,\ |t|\}\big) & \text{, se } \mathtt{Pr}' \text{ for uma união disjunta} \\ \big(|\mathtt{Var}(\mathtt{Pr}')|,\ M_0 \cup \{|s| + |t|\}\big) & \text{, se } (a\ c_1) \curlywedge^? t \in \mathtt{Pr}_0 \\ \big(|\mathtt{Var}(\mathtt{Pr}')|,\ M_0 \cup \{|t|\}\big) & \text{, se } s \overset{\curlywedge ?}{\approx}_\alpha (a\ b) \cdot t \in \mathtt{Pr}_0 \\ \big(|\mathtt{Var}(\mathtt{Pr}')|,\ M_0\big) & \text{, se } s \overset{\curlywedge ?}{\approx}_\alpha (a\ b) \cdot t, (a\ c_1) \curlywedge^? t \in \mathtt{Pr}_0 \end{cases}$$

  Essa regra também não afeta o número de variáveis distintas em $\mathtt{Pr}$, então $|\mathtt{Var}(\mathtt{Pr})| = |\mathtt{Var}(\mathtt{Pr}')|$. Como $M_0 \cup \{2 + |s| + |t|\} >_{mul} M_0 \cup \{|s| + |t|,\ |t|\}$, segue que $[\mathtt{Pr}] >_{lex} [\mathtt{Pr}']$.

- Se $\mathtt{Pr} \Longrightarrow_{(\curlywedge\approx_\alpha inst1)} \mathtt{Pr}'$, então

$$[\mathtt{Pr}] = \big[\mathtt{Pr}_0 \uplus \{\pi \cdot X \overset{\curlywedge ?}{\approx}_\alpha t\}\big] = \big(|\mathtt{Var}(\mathtt{Pr})|,\ M_0 \cup \{1 + |t|\}\big) \text{ e}$$
$$[\mathtt{Pr}'] = [\mathtt{Pr}_0\sigma] = \big(|\mathtt{Var}(\mathtt{Pr}')|,\ \overline{M_0}\big),$$

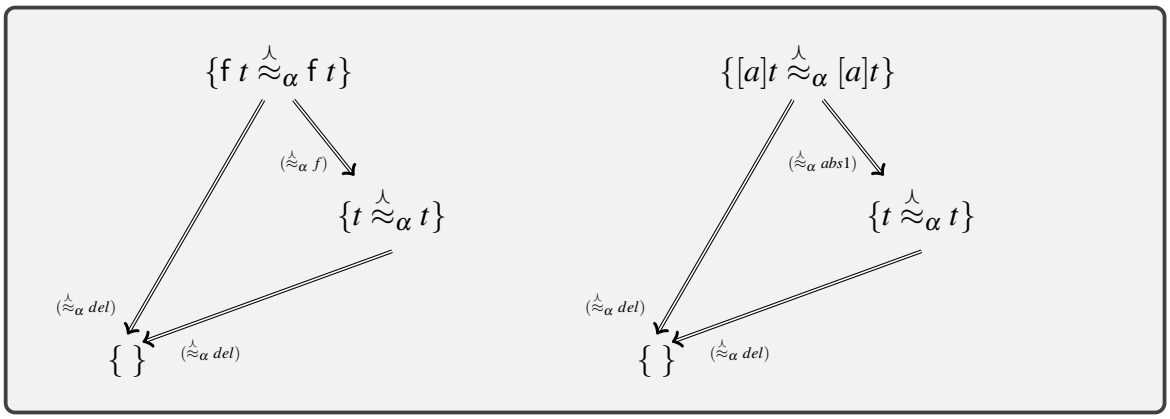
  onde $X \notin \mathtt{Var}(t)$, $\sigma = [X \mapsto \pi^{-1} \cdot t]$ e $\overline{M_0}$ é o multiconjunto correspondente a $\mathtt{Pr}_0\sigma$. Nesse caso, a variável X é instanciada por σ e não volta a ocorrer mais em $\mathtt{Pr}'$. Logo, $|\mathtt{Var}(\mathtt{Pr}')| = |\mathtt{Var}(\mathtt{Pr})| - 1$ e, com isso, segue que $[\mathtt{Pr}] >_{lex} [\mathtt{Pr}']$.

□

No Exemplo 3.2, poderíamos ter escolhido aplicar $(\overset{\curlywedge}{\approx}_\alpha \mathit{inst}2)$ ao invés de $(\overset{\curlywedge}{\approx}_\alpha \mathit{inst}1)$, usando a substituição $[Y \mapsto (b\ a) \cdot X]$, o que ao final resultaria em uma forma normal distinta: $\{(a\ c) \curlywedge^? X,\ (b\ c_1) \curlywedge^? X,\ (c_1\ c_2) \curlywedge^? X\}$. Isso mostra que a relação $\Longrightarrow$ não é confluente. Entretanto, sem o uso das regras de instanciação, todo problema possui uma única forma normal.

Corolário 3.1 (Confluência)**.** A relação $\Longrightarrow$ definida pelas regras da Tabela 3.1 sem as regras $(\overset{\curlywedge}{\approx}_\alpha \mathit{inst}1)$ e $(\overset{\curlywedge}{\approx}_\alpha \mathit{inst}2)$ é confluente.

Demonstração. Suponha que $\mathtt{Pr}_1 \Longleftarrow \mathtt{Pr} \Longrightarrow \mathtt{Pr}_2$ sem o uso das regras de instanciação. Basta observar a definição das regras para concluir que, no caso em que foram usadas regras que tratam de restrições de ponto fixo, não é possível haver sobreposição entre elas pois cada uma corresponde a uma classe de termos distinta, portanto, há confluência local. Agora, com respeito àquelas que tratam de restrições de igualdade, vemos que existe sobreposição apenas entre $(\overset{\curlywedge}{\approx}_\alpha \mathit{del})$ e as regras $(\overset{\curlywedge}{\approx}_\alpha f)$ e $(\overset{\curlywedge}{\approx}_\alpha \mathit{abs}1)$, porém, a confluência local ainda é mantida pois trata-se de uma sobreposição trivial.



Logo, como a relação $\Longrightarrow$ é terminante, sua confluência segue diretamente do Lema de Newman [BN98]: "se uma relação terminante é localmente confluente, então ela é confluente". □

Apesar de conseguirmos a confluência apenas quando desconsideramos as regras de instanciação, todo problema possui uma forma irredutível e as soluções computadas por $\mathtt{unif}_\curlywedge$ (Definição 3.8), quando existem, são equivalentes com relação à ordem de instanciação $\lesssim$ como veremos mais adiante (Teorema 3.2).

A seguir definimos o conceito de *restrições reduzidas* que será usado para caracterizar as formas normais de um problema de unificação nominal.

Definição 3.6 (Restrições reduzidas)**.**

(i) Dizemos que uma restrição de igualdade $s \overset{\curlywedge ?}{\approx}_{\alpha} t$ está *reduzida* quando acontece um dos seguintes casos:

- s e t são átomos distintos (e.g., $a \overset{\curlywedge ?}{\approx}_{\alpha} b$);
- s e t são encabeçados por símbolos de função diferentes (e.g., $\mathsf{f}\ s' \overset{\curlywedge ?}{\approx}_{\alpha} \mathsf{g}\ t'$);
- s e t não são variáveis suspensas e têm construtores de termos distintos na raiz (e.g., $[a]s' \overset{\curlywedge ?}{\approx}_{\alpha} \mathsf{f}\ t'$ ou $a \overset{\curlywedge ?}{\approx}_{\alpha} \mathsf{g}\ t'$);
- s e t são tais que $s = \pi \cdot X$ e $X \in \mathtt{Var}(t)$, ou vice-versa (e.g., $X \overset{\curlywedge ?}{\approx}_{\alpha} \mathsf{f}\ X$).

(ii) Uma restrição de ponto-fixo $\pi \curlywedge^{?} s$ está *reduzida* quando é da forma $\pi \curlywedge^{?} a$ e $\pi(a) \neq a$, ou $\pi \curlywedge^{?} X$. A primeira forma é dita *inconsistente* enquanto que a última é dita *consistente*.

Definição 3.7 (Caracterização de formas normais)**.** Seja $\mathtt{Pr}$ um problema de unificação tal que $\langle \mathtt{Pr} \rangle_{\mathtt{nf}} = \mathtt{Pr}'$. Dizemos que $\mathtt{Pr}'$ está *reduzido* quando é composto apenas por restrições reduzidas, e $\langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ é *bem-sucedido* quando $\mathtt{Pr}' = \emptyset$ ou contém apenas restrições de ponto fixo reduzidas consistentes; caso contrário, $\langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ *falha*.

Definição 3.8 (Soluções computadas)**.** Se $\langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ *falha* ou contém restrições equacionais reduzidas, dizemos que $\mathtt{Pr}$ é *insolúvel*; caso contrário, $\mathtt{Pr}$ é *solúvel* e a solução de $\langle \mathtt{Pr} \rangle_{\mathtt{nf}}$, denotada por $\langle \mathtt{Pr} \rangle_{sol}$, é definida como sendo o par $\langle \Phi, \sigma \rangle$ composto pelo contexto de ponto fixo $\Phi = \{\pi \curlywedge X \mid \pi \curlywedge^{?} X \in \langle \mathtt{Pr} \rangle_{\mathtt{nf}}\}$ e pela substituição $\sigma = \mathtt{Id}$, caso não tenham sido aplicadas regras de instanciação nos passos de simplificação, ou $\sigma = \sigma_1 \circ \cdots \circ \sigma_n$, caso tenham sido aplicadas n dessas regras, com as substituições $\sigma_1, \ldots, \sigma_n$.

Em suma, dado um problema $\mathtt{Pr}$ como entrada, o algoritmo $\mathtt{unif}_{\curlywedge}$ consiste em aplicar as regras de simplicaficação até encontrar uma forma normal $\langle \mathtt{Pr} \rangle_{\mathtt{nf}}$; se $\langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ falha, o algortimo retorna $\mathtt{falha}$, e se $\langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ é bem-sucedido, o mesmo retorna o par $\langle \mathtt{Pr} \rangle_{sol}$. Lembrando que, neste trabalho, substituições são aplicadas à direita, (e.g., $X[X \mapsto t] = t$), assim a substituição da solução é o resultado da composição das substituições, oriundas das regras ($\overset{\curlywedge}{\approx}_{\alpha}$ *inst*1) ou ($\overset{\curlywedge}{\approx}_{\alpha}$ *inst*2), na ordem em que são aplicadas (e.g., $\sigma = \sigma_1 \circ \cdots \circ \sigma_n$, se foram aplicadas regras $\overset{\sigma_1}{\Longrightarrow}, \ldots, \overset{\sigma_n}{\Longrightarrow}$, nessa ordem, em meio aos passos de simplificação usados para atingir uma forma normal). Abaixo, apresentamos uma descrição em pseudo-algoritmo para $\mathtt{unif}_{\curlywedge}$:

$\texttt{unif}_{\curlywedge}(\texttt{Pr})$ = **enquanto** existir $\texttt{Pr}'$ tal que $\texttt{Pr} \Longrightarrow \texttt{Pr}'$ **faça** $\texttt{Pr} := \texttt{Pr}'$;
se $\texttt{Pr} = \langle \texttt{Pr} \rangle_{\texttt{nf}}$ **e** $\langle \texttt{Pr} \rangle_{\texttt{nf}}$ é bem-sucedido **então**
retorna $\langle \texttt{Pr} \rangle_{sol}$;
se não retorna `falha`.

Observação 3.2. Vimos que a relação $\Longrightarrow$ não é confluente, portanto não se pode falar em uma *única* forma normal de um problema. Assim, quando $\texttt{Pr} \overset{*}{\Longrightarrow} \texttt{Pr}'$ e escrevemos $\texttt{Pr}' = \langle \texttt{Pr} \rangle_{\texttt{nf}}$, queremos dizer que $\texttt{Pr}'$ é uma forma irredutível de $\texttt{Pr}$, isto é, não há mais regras de simplificação aplicáveis.

Exemplo 3.4. 1. $\texttt{Pr}_1 = \{[a][b]\mathtt{g}(X,b) \overset{\curlywedge ?}{\approx}_\alpha [b][a]\mathtt{g}(a,X)\}$

$$\begin{array}{rcl}
\texttt{Pr}_1 & \Longrightarrow & \{[b]\mathtt{g}(X,b) \overset{\curlywedge ?}{\approx}_\alpha [b]\mathtt{g}(b,(a\ b)\cdot X),\ (a\ c_1) \curlywedge^? [a]\mathtt{g}(a,X),\ (c_1\ c_2) \curlywedge^? X\} \\
 & \overset{*}{\Longrightarrow} & \{X \overset{\curlywedge ?}{\approx}_\alpha b,\ b \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot X,\ (a\ c_1) \curlywedge^? [a]\mathtt{g}(a,X),\ (c_1\ c_2) \curlywedge^? X\} \\
 & \overset{[X\mapsto b]}{\Longrightarrow} & \{b \overset{\curlywedge ?}{\approx}_\alpha a,\ (a\ c_1) \curlywedge^? [a]\mathtt{g}(a,b),\ (c_1\ c_2) \curlywedge^? b\} \\
 & \Longrightarrow & \{b \overset{\curlywedge ?}{\approx}_\alpha a,\ (a\ c_1) \curlywedge^? \mathtt{g}(c_1',b),\ (c_1\ c_2) \curlywedge^? b\} \\
 & \overset{*}{\Longrightarrow} & \{b \overset{\curlywedge ?}{\approx}_\alpha a\} = \langle \texttt{Pr}_1 \rangle_{\texttt{nf}}
\end{array}$$

onde c_1, c_2 e c_1' são átomos novos.

Solução: nenhuma.

2. $\texttt{Pr}_2 = \{[a][b]\mathtt{g}(X,b) \overset{\curlywedge ?}{\approx}_\alpha [b][a]\mathtt{g}(a,Y)\}$

$$\begin{array}{rcl}
\texttt{Pr}_2 & \overset{*}{\Longrightarrow} & \{X \overset{\curlywedge ?}{\approx}_\alpha b,\ b \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot Y,\ (a\ c_1) \curlywedge^? [a]\mathtt{g}(a,Y),\ (c_1\ c_2) \curlywedge^? Y\} \\
 & \overset{[X\mapsto b]}{\Longrightarrow} & \{b \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot Y,\ (a\ c_1) \curlywedge^? [a]\mathtt{g}(a,Y),\ (c_1\ c_2) \curlywedge^? Y\} \\
 & \overset{[Y\mapsto a]}{\Longrightarrow} & \{(a\ c_1) \curlywedge^? [a]\mathtt{g}(a,a),\ (c_1\ c_2) \curlywedge^? a\} \\
 & \overset{*}{\Longrightarrow} & \{\} = \langle \texttt{Pr}_2 \rangle_{\texttt{nf}}
\end{array}$$

onde c_1 e c_2 são átomos novos.

Solução: $\langle \emptyset, [X \mapsto b,\ Y \mapsto a] \rangle$.

3. $\mathtt{Pr}_3 = \left\{[a][a]\mathsf{g}(a,\mathsf{f}\ X) \overset{\curlywedge ?}{\approx}_\alpha [a][b]\mathsf{g}(b,Y),\ Y \overset{\curlywedge ?}{\approx}_\alpha \mathsf{f}\ Z\right\}$

$$
\begin{array}{lcl}
\mathtt{Pr}_3 & \overset{[Y\mapsto \mathsf{f}\ Z]}{\Longrightarrow} & \left\{[a][a]\mathsf{g}(a,\mathsf{f}\ X) \overset{\curlywedge ?}{\approx}_\alpha [a][b]\mathsf{g}(b,\mathsf{f}\ Z)\right\} \\
 & \Longrightarrow & \left\{[a]\mathsf{g}(a,\mathsf{f}\ X) \overset{\curlywedge ?}{\approx}_\alpha [b]\mathsf{g}(b,\mathsf{f}\ Z)\right\} \\
 & \Longrightarrow & \left\{\mathsf{g}(a,\mathsf{f}\ X) \overset{\curlywedge ?}{\approx}_\alpha \mathsf{g}(a,\mathsf{f}\ (a\ b)\cdot Z),\ (a\ c_1) \curlywedge^? \mathsf{g}(b,\mathsf{f}\ Z),\ (c_1\ c_2) \curlywedge^? Z\right\} \\
 & \overset{*}{\Longrightarrow} & \left\{X \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot Z,\ (a\ c_1) \curlywedge^? Z,\ (c_1\ c_2) \curlywedge^? Z\right\} \\
 & \overset{[Z\mapsto (b\ a)\cdot X]}{\Longrightarrow} & \left\{(a\ c_1) \curlywedge^? (b\ a)\cdot X,\ (c_1\ c_2) \curlywedge^? (b\ a)\cdot X\right\} \\
 & \Longrightarrow & \left\{(b\ c_1) \curlywedge^? X,\ (c_1\ c_2) \curlywedge^? X\right\} = \langle \mathtt{Pr}_3\rangle_{\mathtt{nf}}
\end{array}
$$

onde c_1 e c_2 são átomos novos.

Solução: $\left\langle \left\{(b\ c_1) \curlywedge Z,\ (c_1\ c_2) \curlywedge Z\right\}, [X \mapsto \mathsf{f}\ (b\ a)\cdot Z,\ Y \mapsto \mathsf{f}\ Z]\right\rangle$.

Lema 3.3. Se $\langle \mathtt{Pr}\rangle_{sol} = \langle \Phi, \sigma\rangle$ então $\mathtt{dom}(\sigma) \cap \mathtt{VRan}(\sigma) = \emptyset$.

Demonstração. Por construção temos que $\sigma = \sigma_1 \circ \cdots \circ \sigma_n$, onde supomos que foram aplicadas n regras de instanciação e, para cada $1 \leq i \leq n$, $\sigma_i = [X_i \mapsto t_i]$ e $X_i \in \mathtt{Var}(\mathtt{Pr})$. Se $\mathtt{Pr} \overset{*}{\Longrightarrow} \mathtt{Pr}' \overset{\sigma_i}{\Longrightarrow} \mathtt{Pr}'\sigma_i$, então X_i é adicionada a $\mathtt{dom}(\sigma)$ de maneira que $X_i \notin \mathtt{Var}(t_i)$ e todas as suas ocorrências em $\mathtt{Pr}'$ são instanciadas por σ_i, logo X_i não volta a aparecer em nenhuma simplificação futura do problema. Com isso, temos que $\mathtt{dom}(\sigma) = \{X_1, \ldots, X_n\}$ e

$$X_j \notin \bigcup_{i=j}^{n} \mathtt{Var}(t_j), \text{ para cada } 1 \leq j \leq n. \tag{3.1}$$

Além disso, por definição $\mathtt{VRan}(\sigma) = \bigcup_{X_i \in \mathtt{dom}(\sigma)} \mathtt{Var}(X_i\sigma)$. Analisando $X_i\sigma$, vemos que

$$
\begin{aligned}
X_i\sigma &= X_i\sigma_1 \circ \cdots \circ \sigma_n \\
&= (X_i\sigma_i)\sigma_{i+1} \circ \cdots \circ \sigma_n \\
&= t_i\sigma_{i+1} \circ \cdots \circ \sigma_n.
\end{aligned}
$$

Assim, para caracterizar $\mathtt{Var}(X_i\sigma)$, precisamos entender mais sobre $\mathtt{Var}(t_i)$. Sabemos por (3.1) que, para $1 \leq j \leq i$, $X_j \notin \mathtt{Var}(t_i)$, e com isso vemos que $X_j \notin \mathtt{Var}(X_i\sigma)$. Então suponha que $X_j \in \mathtt{Var}(t_i)$ para algum $j \in \{i+1, \ldots, n\}$. Nesse caso, escrevemos $t_i = t_i[X_j]$ para expressar esse fato e temos

$$
\begin{aligned}
t_i[X_j]\sigma_{i+1} \circ \cdots \circ \sigma_n &= \underbrace{(t_i\sigma_{i+1} \circ \cdots \circ \sigma_{j-1})[X_j]\sigma_j}_{\overline{t_0}} \circ \cdots \circ \sigma_n \\
&= \overline{t_0}[\not{X_j}]\sigma_{j+1} \circ \cdots \circ \sigma_n \\
&= \overline{t_i},
\end{aligned}
$$

onde usamos a notação $\overline{t_0}[\not{X_j}]$ para dizer que $X_j \notin \mathtt{Var}(\overline{t_0})$, de modo que $X_j \notin \mathtt{Var}(\overline{t_i})$, isto é, $X_j \notin \mathtt{Var}(X_i\sigma)$. Portanto, com todo esse raciocínio, vemos que se $Y \in \mathtt{dom}(\sigma) \cap \mathtt{Var}(t_i)$, então $Y \notin \mathtt{Var}(X_i\sigma)$, para todo $i \in \{1, \ldots, n\}$. Como, pela própria construção de σ,

$$
\mathtt{VRan}(\sigma) \subseteq \bigcup_{i=1}^{n} \mathtt{Var}(t_i),
$$

concluímos, enfim, que $\mathtt{dom}(\sigma) \cap \mathtt{VRan}(\sigma) = \emptyset$.

□

Definição 3.9. Dizemos que $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr})$ é uma solução *idempotente* para o problema $\mathtt{Pr}$ quando vale $\Phi \vdash \sigma \overset{\curlywedge}{\approx}_\alpha \sigma\sigma$, isto é, $\Phi \vdash X\sigma \overset{\curlywedge}{\approx}_\alpha X\sigma\sigma$, para todo $X \in \mathtt{Var}(\mathtt{Pr})$.

Observação 3.3. No exemplo 3.1, o par $\langle \emptyset, \sigma_1 \rangle$ é uma solução idempotente, enquanto que sua instância $\langle \emptyset, \sigma_2 \rangle$ não. Isso motiva o fato de não ser pedido idempotência à solução de um problema na Definição 3.2, apesar do algoritmo $\mathtt{unif}_{\curlywedge}$ sempre computar uma solução que é uma *smg* e idempotente quando bem-sucedido (Teorema 3.2).

3.2.2 Correção do Algoritmo

A palavra *correção* é um tradução do termo em inglês *correctness*, que indica a qualidade de ser correto, sem falhas, e usada para englobar os conceitos de *solidez/robustez* (do inglês *soundess*), e *completude* (do inglês *completeness*). Portanto, dizemos que um algoritmo é correto quando ele é sólido e completo. Nesta seção, tratamos sobre a solidez e a completude do algoritmo $\mathtt{unif}_{\curlywedge}$, isto é:

1. $\mathtt{unif}_{\curlywedge}$ **é sólido/robusto**: se o algoritmo for bem-sucedido a partir de um problema $\mathtt{Pr}$ dado como entrada, então o que ele retorna como saída é, de fato, uma solução para $\mathtt{Pr}$;

2. $\mathtt{unif}_{\curlywedge}$ **é completo**: se for dado como entrada um problema $\mathtt{Pr}$ que possua solução, então o algoritmo não falha (ou equivalentemente, se o algoritmo falha, então o problema não possui solução).

O lema a seguir mostra que $\texttt{unif}_{\curlywedge}$ preserva as soluções durante todo o processo de simplificação de problemas quando regras de instanciação não são usadas.

Lema 3.4 (Preservabilidade)**.** Seja $\texttt{Pr}$ um problema de unificação tal que $\texttt{Pr} \overset{*}{\Longrightarrow} \texttt{Pr}'$ sem o uso das regras $(\overset{\curlywedge}{\approx}_\alpha \mathit{inst}1)$ e $(\overset{\curlywedge}{\approx}_\alpha \mathit{inst}2)$. Então $\mathcal{U}(\texttt{Pr}) = \mathcal{U}(\texttt{Pr}')$.

Demonstração. A prova será feita por indução sobre o número de passos na simplificação $\texttt{Pr} \overset{*}{\Longrightarrow} \texttt{Pr}'$.

(Base da indução)

A base da indução corresponde ao caso em que $\texttt{Pr} \overset{0}{\Longrightarrow} \texttt{Pr}'$, o que implica que $\texttt{Pr}' = \texttt{Pr}$ e o resultado segue trivialmente.

(Passo indutivo)

Agora vamos provar que o resultado vale para reduções de comprimento n, isto é, $\texttt{Pr} \overset{n}{\Longrightarrow} \texttt{Pr}'$, tomando como hipótese de indução que:

$$\textit{se } \texttt{Pr}' \overset{n-1}{\Longrightarrow} \texttt{Pr}'', \textit{ então } \mathcal{U}(\texttt{Pr}') = \mathcal{U}(\texttt{Pr}''). \tag{HI}$$

Observe que podemos escrever $\texttt{Pr} \overset{n-1}{\Longrightarrow} \overline{\texttt{Pr}} \Longrightarrow \texttt{Pr}'$. Por HI, segue que $\mathcal{U}(\texttt{Pr}) = \mathcal{U}(\overline{\texttt{Pr}})$ e agora basta analisar a última regra aplicada.

1. A última regra aplicada foi $(\curlywedge a)$.

   Nesse caso
   $$\overline{\texttt{Pr}} = \overline{\texttt{Pr}}_0 \uplus \{\pi \curlywedge^? a\} \Longrightarrow \texttt{Pr}' = \overline{\texttt{Pr}}_0,$$
   onde $\pi(a) = a$. Se tomarmos $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\texttt{Pr}})$, então $\langle \Phi, \sigma \rangle$ é solução para todas as restrições de igualdade e de ponto fixo em $\overline{\texttt{Pr}}$ e, em particular, para todas aquelas em $\overline{\texttt{Pr}}_0 = \texttt{Pr}'$. Logo, $\langle \Phi, \sigma \rangle \in \mathcal{U}(\texttt{Pr}')$. A outra inclusão segue do fato de que, como $\pi(a) = a$, para qualquer par $\langle \Phi, \sigma \rangle$ temos $\Phi \vdash \pi \curlywedge a\sigma$. Com isso, se $\langle \Phi, \sigma \rangle \in \mathcal{U}(\texttt{Pr}')$, então $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\texttt{Pr}})$, e portanto $\mathcal{U}(\overline{\texttt{Pr}}) = \mathcal{U}(\texttt{Pr}')$.

2. A última regra aplicada foi $(\curlywedge f)$.

   Nesse caso
   $$\overline{\texttt{Pr}} = \overline{\texttt{Pr}}_0 \uplus \{\pi \curlywedge^? \mathsf{f}(t_1, \cdots, t_n)\} \Longrightarrow \texttt{Pr}' = \overline{\texttt{Pr}}_0 \cup \{\pi \curlywedge^? t_1, \ldots, \pi \curlywedge^? t_n\}.$$

   Note que pelo Lema da inversão (Lema 2.2):
   $$\Phi \vdash \pi \curlywedge \mathsf{f}(t_1\sigma, \cdots, t_n\sigma) \quad \text{se, e somente se,} \quad \Phi \vdash \pi \curlywedge t_1\sigma, \ldots, \pi \curlywedge t_n\sigma$$

Com isso, se $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\mathtt{Pr}})$, então $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr}')$ e vice-versa.

Logo $\mathcal{U}(\overline{\mathtt{Pr}}) = \mathcal{U}(\mathtt{Pr}')$, e o resultado segue.

3. A última regra aplicada foi $(\curlywedge abs)$. Nesse caso

$$\overline{\mathtt{Pr}} = \overline{\mathtt{Pr}}_0 \uplus \{\pi \curlywedge^? [a]t\} \Longrightarrow \mathtt{Pr}' = \overline{\mathtt{Pr}}_0 \cup \{\pi \curlywedge^? (a\ c_1) \cdot t, \overline{(c_1\ c_2) \curlywedge^? \mathtt{Var}(t)}\},$$

onde c_1 e c_2 são nomes novos.

Primeiro, vamos provar que $\mathcal{U}(\overline{\mathtt{Pr}}) \subseteq \mathcal{U}(\mathtt{Pr}')$. Para isso, tome $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\mathtt{Pr}})$. Então $\langle \Phi, \sigma \rangle$ é solução para $\overline{\mathtt{Pr}}_0$ e $\Phi \vdash \pi \curlywedge [a]t\sigma$. Pelo Lema da inversão (Lema 2.2), temos que

$$\Phi, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t\sigma)} \vdash \pi \curlywedge (a\ c_1) \cdot t\sigma. \tag{3.2}$$

Como c_1 e c_2 são novos, $\mathtt{dom}((c_1\ c_2)) \cap \mathtt{supp}_\Phi(t\sigma) = \emptyset$, logo, pelo Teorema 2.4, segue que $\Phi \vdash (c_1\ c_2) \curlywedge t\sigma$ e, em particular,

$$\Phi \vdash \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t\sigma)}.$$

Então, denotando $\Phi' := \Phi \cup \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t\sigma)}$, temos que $\Phi \vdash \Phi'$.

Como $\Phi' \vdash \pi \curlywedge (a\ c_1) \cdot t\sigma$ por (3.2), segue da Proposição 2.3 que $\Phi \vdash \pi \curlywedge (a\ c_1) \cdot t\sigma$. Assim, $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr}')$.

Segundo, para a inclusão oposta, tome $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr}')$. Então, além de ser solução para $\overline{\mathtt{Pr}}_0$, $\langle \Phi, \sigma \rangle$ é tal que $\Phi \vdash \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t\sigma)}$ e $\Phi \vdash \pi \curlywedge (a\ c_1) \cdot t\sigma$. Como $\Phi \subseteq \Phi'$, segue da Proposição do enfraquecimento (Proposição 2.2) que $\Phi' \vdash \pi \curlywedge (a\ c_1) \cdot t\sigma$, e usando a regra de derivação $(\curlywedge \mathbf{abs})$ da Tabela 2.1 temos que

$$\Phi \vdash \pi \curlywedge [a]t\sigma.$$

Assim, $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\mathtt{Pr}})$ e, finalmente, podemos concluir que $\mathcal{U}(\overline{\mathtt{Pr}}) = \mathcal{U}(\mathtt{Pr}')$.

4. A última regra aplicada foi $(\curlywedge var)$.

Nesse caso

$$\overline{\mathtt{Pr}} = \overline{\mathtt{Pr}}_0 \uplus \{\pi \curlywedge^? \pi' \cdot X\} \Longrightarrow \mathtt{Pr}' = \overline{\mathtt{Pr}}_0 \cup \{\pi^{(\pi')^{-1}} \curlywedge^? X\},$$

onde $\pi' \neq \mathtt{Id}$. Observe que, por equivariância (Lema 2.3),

$$\Phi \vdash \pi \curlywedge \pi' \cdot X\sigma \ \text{ sse } \ \Phi \vdash \pi^{(\pi')^{-1}} \curlywedge X\sigma.$$

Logo, se $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\mathtt{Pr}})$, então $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr}')$ e vice-versa. Portanto, $\mathcal{U}(\overline{\mathtt{Pr}}) = \mathcal{U}(\mathtt{Pr}')$.

5. A última regra aplicada foi $(\overset{\curlywedge}{\approx}_\alpha \, del)$.

   Nesse caso
$$\overline{\mathtt{Pr}} = \overline{\mathtt{Pr}}_0 \uplus \{t \overset{\curlywedge ?}{\approx}_\alpha t\} \Longrightarrow \mathtt{Pr}' = \overline{\mathtt{Pr}}_0.$$

   Note que com uma simples indução na estrutura de t, é possível derivar $\Phi \vdash t\sigma \overset{\curlywedge}{\approx}_\alpha t\sigma$, para qualquer par $\langle \Phi, \sigma \rangle$. Assim, se $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\mathtt{Pr}})$, então $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr}')$ e vice-versa. Logo, $\mathcal{U}(\overline{\mathtt{Pr}}) = \mathcal{U}(\mathtt{Pr}')$.

6. A última regra aplicada foi $(\overset{\curlywedge}{\approx}_\alpha \, var)$.

   Nesse caso
$$\overline{\mathtt{Pr}} = \overline{\mathtt{Pr}}_0 \uplus \{\pi \cdot X \overset{\curlywedge ?}{\approx}_\alpha \pi' \cdot X\} \Longrightarrow \mathtt{Pr}' = \overline{\mathtt{Pr}}_0 \cup \{(\pi')^{-1} \circ \pi \curlywedge^? X\}.$$

   Seja $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\mathtt{Pr}})$. Então $\langle \Phi, \sigma \rangle$ é solução para $\overline{\mathtt{Pr}}_0$ e $\Phi \vdash \pi \cdot X\sigma \overset{\curlywedge}{\approx}_\alpha \pi' \cdot X\sigma$. Por equivariância (Lema 2.3), $\Phi \vdash (\pi')^{-1} \circ \pi \cdot X\sigma \overset{\curlywedge}{\approx}_\alpha X\sigma$, e pelo Teorema 2.1,
$$\Phi \vdash (\pi')^{-1} \circ \pi \curlywedge X\sigma.$$

   Logo $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr}')$.

   Por outro lado, tome $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr}')$. Então, além de ser solução para $\overline{\mathtt{Pr}}_0$, o par $\langle \Phi, \sigma \rangle$ é tal que $\Phi \vdash (\pi')^{-1} \circ \pi \curlywedge X\sigma$. Pelo Teorema 2.1, $\Phi \vdash (\pi')^{-1} \circ \pi \cdot X\sigma \overset{\curlywedge}{\approx}_\alpha X\sigma$ e por equivariância,
$$\Phi \vdash \pi \cdot X\sigma \overset{\curlywedge}{\approx}_\alpha \pi' \cdot X\sigma.$$

   Assim, $\langle \Phi, \sigma \rangle \in \mathcal{U}(\overline{\mathtt{Pr}})$ e, portanto, $\mathcal{U}(\overline{\mathtt{Pr}}) = \mathcal{U}(\mathtt{Pr}')$.

As provas para os casos omitidos são análogas. □

A seguir, mostramos que as soluções computadas por $\mathtt{unif}_{\curlywedge}$ são soluções do problema de unificação dado como entrada, isto é, o algoritmo é sólido.

Teorema 3.2 (Solidez)**.** Seja $\mathtt{Pr}$ um problema de unificação, e suponha que $\langle \mathtt{Pr} \rangle_{sol} = \langle \Phi, \sigma \rangle$. Então:

(i) $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr})$, e

(ii) $\langle \Phi, \sigma \rangle$ é uma *smg* idempotente de $\mathtt{Pr}$.

Demonstração. (i) Primeiro, observamos que $\langle \mathtt{Pr} \rangle_{sol}$ denota a solução de $\langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ (veja Definição 3.8), isto é, $\langle \Phi, \sigma \rangle \in \mathcal{U}(\langle \mathtt{Pr} \rangle_{\mathtt{nf}})$. Para mostrar que $\langle \mathtt{Pr} \rangle_{sol} \in \mathcal{U}(\mathtt{Pr})$, considere a cadeia de reduções $\mathtt{Pr} \stackrel{*}{\Longrightarrow} \langle \mathtt{Pr} \rangle_{\mathtt{nf}}$. Supondo que foram usadas n regras de instanciação nessa cadeia, a prova será feita por indução sobre n.

(Base da indução)

Nesse caso, $n = 0$, ou seja, não foram usadas regras de instanciação na cadeia de reduções mencionada. Pelo Lema da Preservabilidade (Lema 3.4), $\mathcal{U}(\mathtt{Pr}) = \mathcal{U}(\langle \mathtt{Pr} \rangle_{\mathtt{nf}})$, logo $\langle \mathtt{Pr} \rangle_{sol} \in \mathcal{U}(\mathtt{Pr})$.

(Passo indutivo)

Tomamos como hipótese de indução que:

$$\begin{array}{c} \textit{se } \mathtt{Pr}' \stackrel{*}{\Longrightarrow} \langle \mathtt{Pr}' \rangle_{\mathtt{nf}} \textit{ usando } n-1 \textit{ regras de instanciação,} \\ \textit{então } \langle \mathtt{Pr}' \rangle_{sol} \in \mathcal{U}(\mathtt{Pr}'), \end{array} \tag{HI.1}$$

e vamos mostrar que o resultado vale para n.

Separamos a cadeia de reduções do início da seguinte forma:

$$\mathtt{Pr} \stackrel{*}{\Longrightarrow} \overline{\mathtt{Pr}} \stackrel{\sigma_1}{\Longrightarrow} \overline{\mathtt{Pr}}_1 \stackrel{*}{\Longrightarrow} \langle \mathtt{Pr} \rangle_{\mathtt{nf}},$$

onde na redução $\mathtt{Pr} \stackrel{*}{\Longrightarrow} \overline{\mathtt{Pr}}$ não são usadas regras de instanciação, $\stackrel{\sigma_1}{\Longrightarrow}$ é o primeiro uso, e em $\overline{\mathtt{Pr}}_1 \stackrel{*}{\Longrightarrow} \langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ são usadas as $n-1$ outras. Note que com essa construção, temos que $\langle \overline{\mathtt{Pr}}_1 \rangle_{\mathtt{nf}} = \langle \mathtt{Pr} \rangle_{\mathtt{nf}}$, logo podemos escrever $\overline{\mathtt{Pr}}_1 \stackrel{*}{\Longrightarrow} \langle \overline{\mathtt{Pr}}_1 \rangle_{\mathtt{nf}}$ e, por (HI.1), segue que $\langle \overline{\mathtt{Pr}}_1 \rangle_{sol} \in \mathcal{U}(\overline{\mathtt{Pr}}_1)$. Se $\overline{\mathtt{Pr}} \stackrel{\sigma_1}{\Longrightarrow} \overline{\mathtt{Pr}}_1$ usando $(\overset{\curlywedge}{\approx}_\alpha \textit{inst}1)$ (com a outra regra é análogo), temos que

$$\overline{\mathtt{Pr}} = \overline{\mathtt{Pr}}_0 \uplus \{\pi_1 \cdot X_1 \overset{\curlywedge ?}{\approx}_\alpha t_1\} \text{ e } \overline{\mathtt{Pr}}_1 = \overline{\mathtt{Pr}}_0 \sigma_1,$$

onde $\sigma_1 = [X_1 \mapsto {\pi_1}^{-1} \cdot t_1]$ e $X_1 \notin \mathtt{Var}(t_1)$. Assim, $\langle \overline{\mathtt{Pr}}_1 \rangle_{sol} = \langle \Phi, \sigma'' \rangle$, onde $\sigma'' = \sigma_2 \circ \cdots \circ \sigma_n$.

Agora, basta mostrar que $\langle \mathtt{Pr} \rangle_{sol} = \langle \Phi, \sigma \rangle$ é uma solução para $\overline{\mathtt{Pr}}$, a qual por construção é tal que $\sigma = \sigma_1 \circ \cdots \circ \sigma_n$. Aplicando σ em $\overline{\mathtt{Pr}}$, temos

$$\begin{aligned}
\overline{\mathtt{Pr}}\sigma &= \overline{\mathtt{Pr}}_0\sigma \cup \{\pi_1 \cdot X_1\sigma \overset{\curlywedge ?}{\approx}_\alpha t_1\sigma\} \\
&= \overline{\mathtt{Pr}}_0\sigma_1\sigma'' \cup \{\pi_1 \cdot (X_1\sigma_1)\sigma'' \overset{\curlywedge ?}{\approx}_\alpha \pi_1 \cdot (X_1\sigma_1)\sigma''\} \\
&= \overline{\mathtt{Pr}}_1\sigma'' \cup \{\pi_1 \cdot X_1\sigma \overset{\curlywedge ?}{\approx}_\alpha \pi_1 \cdot X_1\sigma\}.
\end{aligned}$$

Como $\langle \Phi, \sigma'' \rangle$ é solução para $\overline{\mathtt{Pr}}_1$, temos que $\Phi \vdash \overline{\mathtt{Pr}}_1\sigma''$, o que implica que $\Phi \vdash \overline{\mathtt{Pr}}_0\sigma$. Logo $\langle \mathtt{Pr} \rangle_{sol} \in \mathcal{U}(\overline{\mathtt{Pr}})$, como queríamos. Como $\mathcal{U}(\overline{\mathtt{Pr}}) = \mathcal{U}(\mathtt{Pr})$, pois na redução $\mathtt{Pr} \overset{*}{\Longrightarrow} \overline{\mathtt{Pr}}$ consideramos que nenhuma regra de instanciação foi usada, temos que $\langle \mathtt{Pr} \rangle_{sol} \in \mathcal{U}(\mathtt{Pr})$ e o resultado do item (i) é, enfim, provado.

(ii) Como no item anterior, suponha que na cadeia de reduções $\mathtt{Pr} \overset{*}{\Longrightarrow} \langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ tenham sido usadas n regras de instanciação. A prova de que $\langle \mathtt{Pr} \rangle_{sol} = \langle \Phi, \sigma \rangle$ é uma *smg* idempotente de $\mathtt{Pr}$ será por indução sobre n.

(Base da indução)

Se $n = 0$, então não foram usadas regras de instanciação na cadeia, logo $\sigma = \mathtt{Id}$. Além disso, $\mathcal{U}(\mathtt{Pr}) = \mathcal{U}(\langle \mathtt{Pr} \rangle_{\mathtt{nf}})$. Assim, para qualquer $\langle \Phi', \sigma' \rangle \in \mathcal{U}(\mathtt{Pr})$ temos que $\Phi' \vdash \Phi\sigma'$, já que $\Phi = \{\pi \curlywedge X \mid \pi \curlywedge^? X \in \langle \mathtt{Pr} \rangle_{\mathtt{nf}}\}$. Como $\Phi' \vdash X\sigma' \overset{\curlywedge}{\approx}_\alpha X\mathtt{Id}\sigma'$ para todo X, segue que $\langle \Phi, \sigma \rangle \lesssim_{\sigma'} \langle \Phi', \sigma' \rangle$ e $\langle \mathtt{Pr} \rangle_{sol}$ é uma *smg* de $\mathtt{Pr}$. A sua idempotência nesse caso segue trivialmente.

(Passo indutivo)

Vamos tomar como hipótese de indução que:

$$\begin{gathered}
\textit{se } \mathtt{Pr}' \overset{*}{\Longrightarrow} \langle \mathtt{Pr}' \rangle_{\mathtt{nf}} \textit{ usando } n-1 \textit{ regras de instanciação,} \\
\textit{então } \langle \mathtt{Pr}' \rangle_{sol} \textit{ é uma smg idempotente de } \mathtt{Pr}',
\end{gathered} \tag{HI.2}$$

e vamos mostrar que o resultado vale para n. Novamente, separamos a cadeia de reduções como no item (i):

$$\mathtt{Pr} \overset{*}{\Longrightarrow} \overline{\mathtt{Pr}} \overset{\sigma_1}{\Longrightarrow} \overline{\mathtt{Pr}}_1 \overset{*}{\Longrightarrow} \langle \mathtt{Pr} \rangle_{\mathtt{nf}}.$$

Por um raciocínio análogo ao feito ali, usamos a (HI.2) para obter que $\langle \Phi, \sigma'' \rangle$ é uma *smg* idempotente de $\overline{\mathtt{Pr}}_1$, lembrando que $\sigma'' = \sigma_2 \circ \cdots \circ \sigma_n$.

Agora, seja $\langle \Phi', \sigma' \rangle \in \mathcal{U}(\mathtt{Pr}) = \mathcal{U}(\overline{\mathtt{Pr}})$. Então, como $\pi_1 \cdot X_1 \overset{\curlywedge ?}{\approx}_\alpha t_1 \in \overline{\mathtt{Pr}}$, temos que $\Phi' \vdash \pi_1 \cdot X_1\sigma' \overset{\curlywedge}{\approx}_\alpha t_1\sigma'$ e por equivariância (Lema 2.3), $\Phi' \vdash X_1\sigma' \overset{\curlywedge}{\approx}_\alpha (X_1\sigma_1)\sigma'$. Como $X_k\sigma_1 = X_k$, para todo $2 \leq k \leq n$, segue que:

$$\Phi' \vdash \sigma' \overset{\curlywedge}{\approx}_\alpha \sigma_1\sigma'. \tag{3.3}$$

Com isso, e sabendo que $\overline{\mathtt{Pr}}_0 \subseteq \overline{\mathtt{Pr}}$ e $\overline{\mathtt{Pr}}_1 = \overline{\mathtt{Pr}}_0\sigma_1$, temos que

$$\begin{array}{lll} \Phi' \vdash \overline{\mathtt{Pr}}_0\sigma' & \text{implica} & \Phi' \vdash \overline{\mathtt{Pr}}_0\sigma_1\sigma' \qquad \text{(por (3.3))} \\ & \text{implica} & \Phi' \vdash \overline{\mathtt{Pr}}_1\sigma' \end{array}$$

Portanto, $\langle \Phi', \sigma' \rangle \in \mathcal{U}(\overline{\mathtt{Pr}}_1)$, e assim, $\langle \Phi, \sigma'' \rangle \lesssim \langle \Phi', \sigma' \rangle$, isto é, existe δ tal que

$$\Phi' \vdash \Phi\delta \ \text{ e} \tag{3.4}$$

$$\Phi' \vdash \sigma' \overset{\curlywedge}{\approx}_\alpha \sigma''\delta. \tag{3.5}$$

Note que, para todo $i \in \{1, \ldots, n\}$, temos que

$$\begin{array}{lll} \Phi' \vdash X_i\sigma_1 \overset{\curlywedge}{\approx}_\alpha X_i\sigma_1 & \text{implica} & \Phi' \vdash X_i\sigma_1\sigma' \overset{\curlywedge}{\approx}_\alpha X_i\sigma_1\sigma''\delta \qquad \text{(por (3.5))} \\ & \text{implica} & \Phi' \vdash X_i\sigma' \overset{\curlywedge}{\approx}_\alpha X_i\sigma_1\sigma''\delta \qquad \text{(por (3.3))} \\ & \text{implica} & \Phi' \vdash X_i\sigma' \overset{\curlywedge}{\approx}_\alpha X_i\sigma\delta. \end{array}$$

Logo, $\langle \Phi, \sigma \rangle \lesssim \langle \Phi', \sigma' \rangle$ e $\langle \mathtt{Pr} \rangle_{sol}$ é uma *smg* de $\mathtt{Pr}$. A sua idempotência, dessa vez, segue diretamente do Lema 3.3, o qual fornece que $\mathtt{dom}(\sigma) \cap \mathtt{VRan}(\sigma) = \emptyset$ e com isso é fácil ver que $(X\sigma)\sigma = X\sigma$, para todo $X \in \mathtt{Var}(\mathtt{Pr})$ e, portanto, $\Phi \vdash \sigma \overset{\curlywedge}{\approx}_\alpha \sigma\sigma$.

□

Exemplo 3.5 (Continuação do Exemplo 3.2.)**.** Vimos que o problema

$$\mathtt{Pr} = \{(a\ c) \curlywedge^? X, (b\ c) \curlywedge^? Y, [a]\mathtt{g}(a,X) \overset{\curlywedge ?}{\approx}_\alpha [b]\mathtt{g}(b,Y)\}$$

possui duas formas normais distintas a depender da regra de instanciação usada:

- $\langle \mathtt{Pr} \rangle^1_{\mathtt{nf}} = \{(b\ c) \curlywedge^? Y,\ (a\ c_1) \curlywedge^? Y,\ (c_1\ c_2) \curlywedge^? Y\}$ e,
- $\langle \mathtt{Pr} \rangle^2_{\mathtt{nf}} = \{(a\ c) \curlywedge^? X,\ (b\ c_1) \curlywedge^? X,\ (c_1\ c_2) \curlywedge^? X\}$

e com isso obtemos duas soluções computadas correspondentes:

- $\langle \mathtt{Pr} \rangle^1_{sol} = \Big\langle \underbrace{\{(b\ c) \curlywedge Y,\ (a\ c_1) \curlywedge Y,\ (c_1\ c_2) \curlywedge Y\}}_{\Phi_1}, \underbrace{[X \mapsto (a\ b) \cdot Y]}_{\sigma_1} \Big\rangle$
- $\langle \mathtt{Pr} \rangle^2_{sol} = \Big\langle \underbrace{\{(a\ c) \curlywedge X,\ (b\ c_1) \curlywedge X,\ (c_1\ c_2) \curlywedge X\}}_{\Phi_2}, \underbrace{[Y \mapsto (b\ a) \cdot X]}_{\sigma_2} \Big\rangle.$

O item (ii) do Teorema 3.2 nos permite concluir que $\langle \mathtt{Pr} \rangle^1_{sol} \sim \langle \mathtt{Pr} \rangle^2_{sol}$, isto é, essas soluções são equivalentes. E, de fato, observe que $\Phi_1 = \Phi_2\sigma_1$ e $\Phi_2 = \Phi_1\sigma_2$. Com isso, é fácil ver que:

1. $\Phi_2 \vdash \Phi_1\sigma_2$, e como

$$\Phi_2 \vdash \underbrace{X\sigma_2}_{X} \overset{\curlywedge}{\approx}_\alpha \underbrace{X\sigma_1\sigma_2}_{X} \ \text{ e } \ \Phi_2 \vdash \underbrace{Y\sigma_2}_{(b\ a)\cdot X} \overset{\curlywedge}{\approx}_\alpha \underbrace{Y\sigma_1\sigma_2}_{(b\ a)\cdot X},$$

   segue que $\langle \Phi_1, \sigma_1 \rangle \lesssim_{\sigma_2} \langle \Phi_2, \sigma_2 \rangle$;

2. $\Phi_1 \vdash \Phi_2\sigma_1$, e como

$$\Phi_1 \vdash \underbrace{X\sigma_1}_{(a\ b)\cdot Y} \overset{\curlywedge}{\approx}_\alpha \underbrace{X\sigma_2\sigma_1}_{(a\ b)\cdot Y} \ \text{ e } \ \Phi_1 \vdash \underbrace{Y\sigma_1}_{Y} \overset{\curlywedge}{\approx}_\alpha \underbrace{Y\sigma_2\sigma_1}_{Y},$$

   segue que $\langle \Phi_2, \sigma_2 \rangle \lesssim_{\sigma_1} \langle \Phi_1, \sigma_1 \rangle$.

Logo, $\langle \mathtt{Pr} \rangle^1_{sol} \sim \langle \mathtt{Pr} \rangle^2_{sol}$.

Enfim, apresentamos o Teorema da Completude, que vem mostrar que $\mathtt{unif}_{\curlywedge}$ é completo, isto é, se o algoritmo falha, então o problema não possui solução.

Teorema 3.3 (Completude)**.** Seja $\mathtt{Pr}$ um problema de unificação tal que $\mathtt{Pr} \overset{*}{\Longrightarrow} \mathtt{Pr}'$. Se $\mathtt{Pr}'$ contém restrições reduzidas de igualdade ou de ponto fixo inconsistentes, então $\mathcal{U}(\mathtt{Pr}) = \emptyset$.

Demonstração. Suponha, sem perda de generalidade, que $\mathtt{Pr}' = \langle \mathtt{Pr} \rangle_{\mathtt{nf}}$. Nesse caso, não há mais regras que possam ser aplicadas, já que as restrições em $\mathtt{Pr}'$ estão reduzidas (Definição 3.6). Com isso, $\mathtt{Pr}' = \langle \mathtt{Pr} \rangle_{\mathtt{nf}}$ e falha, logo $\mathtt{Pr}$ é insolúvel e $\mathcal{U}(\mathtt{Pr}) = \emptyset$. □

3.3 O Problema de Emparelhamento sob Contexto

A definição de *emparelhamento* (ou *casamento*) *sob contexto*, que é uma tradução livre[1] da expressão inglesa *matching-in-context*, faz uso de termos sob contexto do tipo $\Psi \vdash s$, que apresentamos no capítulo anterior.

[1]Escolhemos a palavra "emparelhamento" pelo seu uso mais comum em traduções para o português de materiais sobre teoria de grafos.

Intuitivamente, um problema de *emparelhamento sob contexto* trata de uma versão do problema de unificação nominal, no qual apenas um lado (o direito, no nosso caso) das restrições de igualdade pode ser instanciado. A definição formal é dada a seguir.

Notação:
Dado um contexto de ponto fixo Υ, escrevemos $\Upsilon^?$ para indicar o conjunto de restrições $\{\pi \curlywedge^? X \mid \pi \curlywedge X \in \Upsilon\}$.

Definição 3.10. Um *problema de emparelhamento sob contexto* $\mathtt{MPr}$ é um par

$$(\Psi \vdash s) \,{}_?{\approx}\, (\Upsilon \vdash t),$$

onde s e t são termos sob contextos Ψ e Υ, respectivamente. Uma *solução* para esse problema de emparelhamento, caso exista, é uma substituição δ tal que

(i) $\langle \Upsilon^?, s \overset{\curlywedge^?}{\approx}_\alpha t \rangle_{sol} = \langle \Psi', \delta \rangle$;

(ii) $\Psi^{\overline{c},\overline{X}} \vdash \Psi'$, onde

- $\overline{c} := \{c_1, \ldots, c_n\}$ é o conjunto finito de átomos novos (possivelmente vazio) gerados em (i);
- $\overline{X} := \mathtt{Var}(\Psi, \Psi') = \{X_1, \ldots, X_m\}$;
- $$\Psi^{\overline{c},\overline{X}} := \Psi \biguplus_{k=1}^{m} \left(\bigcup_{i=1}^{n-1} \bigcup_{j=1}^{n-i} \left\{ (c_i\ c_{i+j}) \curlywedge X_k \right\} \right);$$

(iii) $\mathtt{dom}(\delta) \cap \mathtt{Var}(\Psi, s) = \emptyset$.

Note que, para encontrar a solução para um problema de emparelhamento sob contexto, usamos o algoritmo $\mathtt{unif}_{\curlywedge}$ no problema de unificação $\mathtt{Pr} = \Upsilon^? \cup \{s \overset{\curlywedge^?}{\approx}_\alpha t\}$. Com isso, é possível que sejam gerados átomos novos com relação a $\mathtt{Pr}$ no processo de computar a solução $\langle \Psi', \delta \rangle$, de maneira que permutações em Ψ' possam mencioná-los. Por esse motivo, a fim de garantir que problemas de emparelhamento sob contexto solúveis na abordagem via *freshness* [FG07] assim também o sejam na abordagem via ponto fixo, adotada neste trabalho, consideramos $\Psi^{\overline{c},\overline{X}}$ no segundo item, uma extensão do contexto Ψ.

Apesar de intimidadora, a união de restrições que adicionamos é apenas um jeito formal de dizer que unimos a Ψ um conjunto de restrições primitivas $(c\ d) \curlywedge X$, para cada par $c, d \in \overline{c}$

e para cada $X \in \overline{X}$. Por exemplo, se $\overline{c} = \{c_1, c_2, c_3\}$ e $\overline{X} = \{X_1, X_2\}$, então

$$\Psi^{\overline{c},\overline{X}} = \Psi \uplus \big\{(c_1\ c_2) \curlywedge X_1, (c_1\ c_3) \curlywedge X_1, (c_2\ c_3) \curlywedge X_1,\ (c_1\ c_2) \curlywedge X_2, (c_1\ c_3) \curlywedge X_2, (c_2\ c_3) \curlywedge X_2\big\}.$$

Observe que esse conjunto adicionado reúne uma série de tautologias, já que c_i foi escolhido de tal forma que ele é novo para qualquer instância de X_k, e só é considerado para manter a coerência de soluções com a abordagem que nos propomos a estender com este trabalho.

Já o item (iii) é a condição que garante que a aplicação da substituição δ afete apenas as variáveis do lado direito das restrições de igualdade. Ademais, quando existe uma solução, as três condições da definição acima garantem que

$$\Psi \vdash \Upsilon\delta \ \text{ e } \ \Psi \vdash s \overset{\curlywedge}{\approx}_\alpha t\delta,$$

e, portanto, $\langle \Psi, \delta \rangle \in \mathcal{U}(\mathtt{Pr})$. Segundo Fernández e Gabbay [FG07], podemos entender a solução para $(\Psi \vdash s)\ {}_?{\approx}(\Upsilon \vdash t)$ como sendo uma substituição mais geral δ tal que o par $\langle \Psi, \delta \rangle$ soluciona $\Upsilon^?, s \overset{\curlywedge^?}{\approx}_\alpha t$ sem instanciar o termo s.

Exemplo 3.6. 1. $\mathtt{MPr}_1 = (\vdash \mathsf{f}\ b)\ {}_?{\approx}(\vdash \mathsf{f}\ a)$ é insolúvel.

De fato, seja $\mathtt{Pr}_1 = \{\mathsf{f}\ b \overset{\curlywedge^?}{\approx}_\alpha \mathsf{f}\ a\}$. Então

$$\mathtt{Pr}_1 \Longrightarrow \big\{b \overset{\curlywedge^?}{\approx}_\alpha a\big\} = \langle \mathtt{Pr}_1 \rangle_{\mathtt{nf}}.$$

Logo, $\langle \mathtt{Pr}_1 \rangle_{\mathtt{nf}}$ falha e $\mathtt{MPr}_1$ não possui solução.

2. $\mathtt{MPr}_2 = \big((a\ c) \curlywedge Y \vdash [a]Y\big)\ {}_?{\approx}\big((a\ c) \curlywedge X \vdash [a]X\big)$ tem solução $\delta_2 = [X \mapsto Y]$.

De fato, seja $\mathtt{Pr}_2 = \big\{(a\ c) \curlywedge^? X,\ [a]Y \overset{\curlywedge^?}{\approx}_\alpha [a]X\big\}$. Então

$$\begin{array}{rcl} \mathtt{Pr}_2 & \Longrightarrow & \big\{(a\ c) \curlywedge^? X,\ Y \overset{\curlywedge^?}{\approx}_\alpha X\big\} \\ & \overset{[X \mapsto Y]}{\Longrightarrow} & \big\{(a\ c) \curlywedge^? Y\big\} = \langle \mathtt{Pr}_2 \rangle_{\mathtt{nf}}. \end{array}$$

Com isso, temos que:

(i) $\langle \mathtt{Pr}_2 \rangle_{sol} = \langle \Psi'_2, \delta_2 \rangle$, onde $\delta_2 = [X \mapsto Y]$, $\Psi'_2 = \{(a\ c) \curlywedge^? Y\}$ e obtemos $\overline{c} = \emptyset$ e $\overline{X} = \{Y\}$;

(ii) como $\overline{c} = \emptyset$, a extensão $\{(a\ c) \curlywedge Y\}^{\overline{c},\overline{X}} = \{(a\ c) \curlywedge Y\}$, e claramente segue que $\{(a\ c) \curlywedge Y\}^{\overline{c},\overline{X}} \vdash \Psi'_2$;

(iii) e $\mathtt{dom}(\delta_2) \cap \mathtt{Var}((a\ c) \curlywedge Y \vdash [a]Y) = \emptyset$.

Logo, δ_2 é solução para $\mathtt{MPr}_2$.

3. $\mathtt{MPr}_3 = \big((c\ d) \curlywedge X \vdash \mathtt{g}(X,a)\big) \,{}_?{\approx} \big((c\ d) \curlywedge X \vdash \mathtt{g}(X,X)\big)$ é insolúvel.

   De fato, seja $\mathtt{Pr}_3 = \big\{(c\ d) \curlywedge^? X,\ \mathtt{g}(X,a) \overset{\curlywedge ?}{\approx}_\alpha \mathtt{g}(X,X)\big\}$. Então

$$\begin{array}{rcl}
\mathtt{Pr}_2 & \Longrightarrow & \big\{(c\ d) \curlywedge^? X,\ X \overset{\curlywedge ?}{\approx}_\alpha X,\ a \overset{\curlywedge ?}{\approx}_\alpha X\big\} \\
 & \Longrightarrow & \big\{(c\ d) \curlywedge^? X,\ a \overset{\curlywedge ?}{\approx}_\alpha X\big\} \\
 & \overset{[X \mapsto a]}{\Longrightarrow} & \big\{(c\ d) \curlywedge^? a\big\} \\
 & \Longrightarrow & \emptyset = \langle \mathtt{Pr}_3 \rangle_{\mathtt{nf}}.
\end{array}$$

   Com isso, temos que $\langle \mathtt{Pr}_3 \rangle_{sol} = \langle \emptyset, [X \mapsto a] \rangle$. Porém, $\mathtt{dom}([X \mapsto a]) \cap \mathtt{Var}((c\ d) \curlywedge X \vdash \mathtt{g}(X,a)) = \{X\}$ e não há outra substituição $\delta \neq [X \mapsto a]$ que acompanhe um par de solução para $\mathtt{Pr}_3$. Logo, o problema $\mathtt{MPr}_3$ não possui solução.

4. $\mathtt{MPr}_4 = (\vdash [a][b]Y) \,{}_?{\approx} (\vdash [b][a]X)$ tem solução $\delta_4 = [X \mapsto (b\ a) \cdot Y]$.

   De fato, seja $\mathtt{Pr}_4 = \big\{[a][b]Y \overset{\curlywedge ?}{\approx}_\alpha [b][a]X\big\}$. Então

$$\begin{array}{rcl}
\mathtt{Pr}_4 & \Longrightarrow & \big\{[b]Y \overset{\curlywedge ?}{\approx}_\alpha [b](a\ b) \cdot X,\ (a\ c_1) \curlywedge^? [a]X,\ (c_1\ c_2) \curlywedge^? X\big\} \\
 & \Longrightarrow & \big\{Y \overset{\curlywedge ?}{\approx}_\alpha (a\ b) \cdot X,\ (a\ c_1) \curlywedge^? [a]X,\ (c_1\ c_2) \curlywedge^? X\big\} \\
 & \Longrightarrow & \big\{Y \overset{\curlywedge ?}{\approx}_\alpha (a\ b) \cdot X,\ (a\ c_1) \curlywedge^? (a\ c_3) \cdot X,\ (c_1\ c_2) \curlywedge^? X,\ (c_3\ c_4) \curlywedge^? X\big\} \\
 & \overset{[X \mapsto (b\ a) \cdot Y]}{\Longrightarrow} & \big\{(a\ c_1) \curlywedge^? (a\ c_3)(b\ a) \cdot Y,\ (c_1\ c_2) \curlywedge^? (b\ a) \cdot Y,\ (c_3\ c_4) \curlywedge^? (b\ a) \cdot Y\big\} \\
 & \overset{*}{\Longrightarrow} & \big\{\ \underbrace{(c_3\ c_1)}_{(a\ c_1)^{(a\ b)(c_3\ a)}} \curlywedge^? Y,\ \underbrace{(c_1\ c_2)}_{(c_1\ c_2)^{(a\ b)}} \curlywedge^? Y,\ \underbrace{(c_3\ c_4)}_{(c_3\ c_4)^{(a\ b)}} \curlywedge^? Y\big\} = \langle \mathtt{Pr}_4 \rangle_{\mathtt{nf}}.
\end{array}$$

   Com isso, temos que:

   (i) $\langle \mathtt{Pr}_4 \rangle_{sol} = \langle \Psi'_4, \delta_4 \rangle$, onde $\delta_4 = [X \mapsto (b\ a) \cdot Y]$,

$$\Psi'_4 = \big\{(c_3\ c_1) \curlywedge Y,\ (c_1\ c_2) \curlywedge Y,\ (c_3\ c_4) \curlywedge Y\big\}$$

   e obtemos $\overline{c} = \{c_1, c_2, c_3, c_4\}$ e $\overline{X} = \{Y\}$;

   (ii) como o termo $(\vdash [a][b]Y)$ tem contexto $\Psi = \emptyset$, temos que

$$\Psi^{\overline{c},\overline{X}} = \big\{(c_1\ c_2) \curlywedge Y,\ (c_2\ c_3) \curlywedge Y,\ (c_3\ c_4) \curlywedge Y\big\}.$$

Note que $\mathtt{dom}(\mathtt{perm}(\Psi'_4|_Y)) \subseteq \mathtt{dom}(\mathtt{perm}(\Psi^{\overline{c},\overline{X}}|_Y))$ e, portanto, $\Psi^{\overline{c},\overline{X}} \vdash \Psi'_4$;

(iii) e $\mathtt{dom}(\delta_4) \cap \mathtt{Var}(\vdash [a][b]Y) = \emptyset$.

Logo, δ_4 é solução para $\mathtt{MPr}_4$.

Como esperado, é possível obter um algoritmo para computar uma solução de um problema de emparelhamento sob contexto $\mathtt{MPr}$ a partir de $\mathtt{unif}_{\curlywedge}$. Denotamos por $\mathtt{matching}_{\curlywedge}$ o algoritmo de emparelhamento sob contexto, e o descrevemos com o pseudo-algoritmo abaixo, aplicando em um problema de emparelhamento $\mathtt{MPr} = (\Psi \vdash s)\ {}_?{\approx}(\Upsilon \vdash t)$:

$\mathtt{matching}_{\curlywedge}(\mathtt{MPr})$ = **faça** $\mathtt{unif}_{\curlywedge}(\{\Upsilon^?, s \overset{\curlywedge\,?}{\approx}_\alpha t\})$;
  se $\mathtt{unif}_{\curlywedge}(\{\Upsilon^?, s \overset{\curlywedge\,?}{\approx}_\alpha t\}) = \mathtt{falha}$ **então**
    retorna $\mathtt{falha}$;
  se $\mathtt{unif}_{\curlywedge}(\{\Upsilon^?, s \overset{\curlywedge\,?}{\approx}_\alpha t\}) = \langle\{\Upsilon^?, s \overset{\curlywedge\,?}{\approx}_\alpha t\}\rangle_{sol}$ **então**
    seja $\langle\Psi', \delta\rangle := \langle\{\Upsilon^?, s \overset{\curlywedge\,?}{\approx}_\alpha t\}\rangle_{sol}$;
    se $\Psi^{\overline{c},\overline{X}} \vdash \Psi'$ **e** $\mathtt{dom}(\delta) \cap \mathtt{Var}(\Psi, s) = \emptyset$ **então**
      retorna δ;
    se não retorna $\mathtt{falha}$.

Emparelhamento sob contexto foi introduzido em [FG07] para definir *Reescrita Nominal* usando a abordagem de *freshness*. No presente trabalho, ele será fundamental para a construção do Algoritmo 1 no Capítulo 4, que por sua vez será usado para determinar soluções para *problemas de Desunificação Nominal.*

Capítulo 4

Desunificação Nominal via Ponto Fixo

Como visto no capítulo anterior, o processo de resolver equações entre termos nominais, chamado de unificação nominal, consiste em encontrar um contexto adequado e uma substituição que instancie igualmente os termos de ambos os lados das equações de um dado problema. Neste capítulo, vamos desenvolver um procedimento para resolver o problema de *desunificação nominal* que consiste em resolver equações *e* diferenças entre termos nominais.

Nesse cenário, uma diferença entre termos será chamada de *desequação* e denotada por $s \overset{\curlywedge}{\not\approx}_\alpha t$, e um problema de desunificação nominal, expresso por

$$\langle \mathtt{Pr} \,||\, D_{\curlywedge} \rangle,$$

será composto por um problema de unificação nominal $\mathtt{Pr}$, que conterá equações $s_i \overset{\curlywedge}{\approx}_\alpha t_i$, $1 \leq i \leq n$ e possíveis restrições de ponto fixo, e uma parte desequacional $D_{\curlywedge}$, formada por desequações $p_j \overset{\curlywedge}{\not\approx}_\alpha q_j$, $1 \leq j \leq m$.

Dado um problema de desunificação nominal, nosso objetivo é definir um procedimento (Algoritmo 2: $\mathtt{desunif}_{\curlywedge}$) correto e terminante que encontre uma representação completa de soluções, o qual, intuitivamente, utiliza o algoritmo $\mathtt{unif}_{\curlywedge}$ para resolver a parte contendo $\mathtt{Pr}$ e entende a parte desequacional como limitações ao unificador encontrado, e após feita uma análise de consistência (Algoritmo 1: $\mathtt{consistente}_{\curlywedge}$) no sentido que definiremos aqui, retorna, finalmente, a desejada representação completa de soluções (Teorema 4.1). Alguns resultados e algoritmos do presente capítulo foram apresentados nos eventos *35th International Workshop on Unification* e *II Workshop Brasileiro de Lógica*, e publicados informalmente em [BFNV21, BN21].

4.1 Definições Principais

A seguir definimos um problema de desunificação nominal via ponto fixo.

Definição 4.1. Um problema de desunificação nominal $\mathcal{P}_{\curlywedge}$ é um par $\langle \mathtt{Pr} \mid\mid D_{\curlywedge} \rangle$ da forma

$$\mathcal{P}_{\curlywedge} = \langle\, \mathtt{Pr} \mid\mid p_1 \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q_1, \ldots, p_m \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q_m \rangle,$$

onde $\mathtt{Pr}$ é um problema de unificação nominal e $D_{\curlywedge}$ consiste em um conjunto finito (possivelmente vazio) de desequações nominais.

Para o próximo exemplo, uma *instância básica de um par* $\langle \Phi, \sigma \rangle$, é um par $\langle \Delta, \lambda \rangle$ tal que $\langle \Phi, \sigma \rangle \lesssim \langle \Delta, \lambda \rangle$ e λ é uma substituição básica, isto é mapeia cada $X \in \mathcal{D}om(\lambda)$ para um termo nominal básico. O exemplo ilustra como interpretar uma solução para um problema de desunificação nominal.

Exemplo 4.1. Uma solução para o problema $\mathcal{P}^1_{\curlywedge} = \langle X \overset{\curlywedge}{\approx}^{?}_{\alpha} \mathsf{f}\, Y \mid\mid Y \overset{\curlywedge}{\not\approx}^{?}_{\alpha} a, Y \overset{\curlywedge}{\not\approx}^{?}_{\alpha} b \rangle$, enquanto resolve a equação $X \overset{\curlywedge}{\approx}^{?}_{\alpha} \mathsf{f}\, Y$, deve também respeitar a limitação imposta pela parte desequacional: que não se é permitido mapear X para $\mathsf{f}\, a$ ou para $\mathsf{f}\, b$. Assim, o par $\langle \emptyset, [X \mapsto \mathsf{f}\, Y] \rangle$, que é uma *smg* para $\{X \overset{\curlywedge}{\approx}^{?}_{\alpha} \mathsf{f}\, Y\}$, também é uma solução $\mathcal{P}^1_{\curlywedge}$, bem como sua instância básica $\langle \emptyset, [X \mapsto \mathsf{f}\, c, Y \mapsto c] \rangle$. O mesmo não acontece com a instância básica $\langle \emptyset, [X \mapsto \mathsf{f}\, a, Y \mapsto a] \rangle$, que apesar de resolver a equação, não respeita a restrição imposta por $Y \overset{\curlywedge}{\not\approx}^{?}_{\alpha} a$.

Quando dizemos que um par *não respeita* (ou *não satisfaz*) as restrições impostas pela parte desequacional significa dizer, precisamente, que o par em questão é solução para uma ou mais *equações associadas* às desequações. No exemplo anterior, há duas equações associadas a $Y \overset{\curlywedge}{\not\approx}^{?}_{\alpha} a$ e $Y \overset{\curlywedge}{\not\approx}^{?}_{\alpha} b$, a dizer, $Y \overset{\curlywedge}{\approx}^{?}_{\alpha} a$ e $Y \overset{\curlywedge}{\approx}^{?}_{\alpha} b$, respectivamente. Note que a primeira instância básica apresentada ali não soluciona essas igualdades, enquanto que a última é solução para $Y \overset{\curlywedge}{\approx}^{?}_{\alpha} a$, e por isso é descartada.

Terminologia:
Dada uma desequação $p \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q \in D_{\curlywedge}$, chamaremos $p \overset{\curlywedge}{\approx}^{?}_{\alpha} q$ de *equação associada a* $p \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q$.

Exemplo 4.2. Seja $\mathcal{P}^2_{\curlywedge} = \langle \mathsf{g}(X,Y) \overset{\curlywedge}{\approx}^{?}_{\alpha} \mathsf{g}(\mathsf{k}, a) \mid\mid X \overset{\curlywedge}{\not\approx}^{?}_{\alpha} \mathsf{k} \rangle$ um problema de desunificação nominal, onde $\mathsf{k} : 0$ é uma constante.

Observe que o par $\langle \emptyset, [X \mapsto \mathsf{k}, Y \mapsto a] \rangle$ é uma *smg* para $\{\mathsf{g}(X,Y) \overset{\curlywedge ?}{\approx}_\alpha \mathsf{g}(\mathsf{k},a)\}$, porém, ao mesmo tempo, não satisfaz $X \overset{\curlywedge ?}{\not\approx}_\alpha \mathsf{k}$, pois $\emptyset \vdash \mathsf{k} \overset{\curlywedge}{\approx}_\alpha \mathsf{k}$. O mesmo acontece com qualquer instância desse par, e portanto, $\mathcal{P}^2_{\curlywedge}$ não possui solução.

Feita essa breve explanação sobre a noção de uma solução para um problema de desunificação nominal, estamos prontos para apresentar sua definição formal.

Definição 4.2. Seja $\mathcal{P}_{\curlywedge} = \langle\, \mathtt{Pr} \,||\, D_{\curlywedge} \rangle$ um problema de desunificação nominal, onde $D_{\curlywedge} = \{p_1 \overset{\curlywedge ?}{\not\approx}_\alpha q_1, \ldots, p_n \overset{\curlywedge ?}{\not\approx}_\alpha q_n\}$. Uma *solução* para $\mathcal{P}_{\curlywedge}$ é um par $\langle \Delta, \lambda \rangle$ formado por um contexto Δ e uma substituição λ satisfazendo as seguintes condições:

(i) $\langle \Delta, \lambda \rangle$ é uma solução para $\mathtt{Pr}$; e

(ii) $\langle \Delta, \lambda \rangle$ satisfaz a parte desequacional de $\mathcal{P}_{\curlywedge}$, isto é:

$$\Delta \nvdash p\lambda \overset{\curlywedge}{\approx}_\alpha q\lambda,$$

para toda $p \overset{\curlywedge ?}{\not\approx}_\alpha q \in D_{\curlywedge}$.

Em geral, instanciação possui um papel importante na teoria de unificação nominal, pois os problemas de unificação possuem a propriedade de serem fechados por instanciação (Lema 3.2) e, por esse motivo, é possível obter uma representação finita de todas as suas soluções por meio da solução mais geral (*smg*) produzida pelo algoritmo $\mathtt{unif}_{\curlywedge}$.

Entretanto, como vimos no Exemplo 4.1, o par $\langle \emptyset, [X \mapsto \mathsf{f}\ Y] \rangle$ é solução para $\mathcal{P}^1_{\curlywedge}$, porém, instanciando-a por $\delta = [Y \mapsto a]$ obtemos o par $\langle \emptyset, [X \mapsto \mathsf{f}\ a, Y \mapsto a] \rangle$, e vimos que esse não pode ser solução. Portanto, infelizmente, perdemos a propriedade de fechamento por instanciação na desunificação nominal.

Apesar disso, existe uma maneira de obter uma representação finitária para as soluções de um problema de desunificação nominal por meio de uma estrutura denominada *par com exceções*, a qual definimos a seguir.

Definição 4.3 (Par com exceções)**.** Seja $\mathcal{P}_{\curlywedge}$ um problema de desunificação. Um *par com exceções para* $\mathcal{P}_{\curlywedge}$, denotado por $\langle \Phi, \sigma \rangle - \Theta$, consiste em um par $\langle \Phi, \sigma \rangle$, e uma família indexada de pares $\Theta = \{\langle \nabla^1_l \uplus \nabla^2_l, \theta_l \rangle \mid l \in I\}$, onde ∇^2_l é um contexto de ponto fixo (possivelmente vazio) cujas restrições primitivas envolvem nomes novos relativos a $\mathcal{P}_{\curlywedge}$.

Na prática, como veremos no Algoritmo 2, se $\langle \Phi, \sigma \rangle - \Theta$ é um par com exceções para um problema $\mathcal{P}_{\curlywedge} = \langle \mathtt{Pr} \,||\, D_{\curlywedge} \rangle$, então $\langle \Phi, \sigma \rangle$ será uma *smg* para $\mathtt{Pr}$ enquanto que a família Θ irá colecionar *smg*'s correspondentes a cada equação associada às desequações de $D_{\curlywedge}$, ou seja, cada par $\langle \nabla_l, \theta_l \rangle \in \Theta$ será uma *smg* de uma equação associada.

Mas como essa estrutura representará de fato uma solução? Bem, mais adiante, dado um problema de desunificação $\mathcal{P}_{\curlywedge}$, construiremos um conjunto formado por pares com exceções vinculados ao problema (Definição 4.6), de modo que toda *instância* de algum desses pares com exceções será uma solução para $\mathcal{P}_{\curlywedge}$.

Essa nova noção de instância é definida a seguir, onde utilizamos como base a noção de *instância* da Definição 3.3 e a extensão de um contexto por nomes novos, dada no final do capítulo anterior:

$$\Phi^{\overline{c},\overline{X}} := \Phi \uplus \{(c\ d) \curlywedge X \mid c,d \in \overline{c} \ \text{ e } \ X \in \overline{X}\}$$

onde Φ é um contexto de ponto fixo, $\overline{X} := \{X_1,\ldots,X_m\}$ é o conjunto de variáveis que ocorrem em $\mathcal{P}_{\curlywedge}$ e $\overline{c} := \{c_1,\ldots,c_n\}$ é um conjunto finito (possivelmente vazio) de átomos novos para $\mathcal{P}_{\curlywedge}$.

Definição 4.4 (Instância de um par com exceções)**.** Sejam $\mathcal{P}_{\curlywedge}$ um problema de desunificação nominal, $\overline{c}$ um conjunto finito (possivelmente vazio) de nomes novos para $\mathcal{P}_{\curlywedge}$ e $\overline{X} = \mathtt{Var}(\mathcal{P}_{\curlywedge})$. Dizemos que:

(i) um par $\langle\Delta,\lambda\rangle$ é uma *instância de uma família* $\Theta = \{\langle\nabla_l^1 \uplus \nabla_l^2, \theta_l\rangle \mid l \in I\}$ se, e somente se, toda instância de $\langle\Delta^{\overline{c},\overline{X}},\lambda\rangle$ é uma instância de algum $\langle\nabla_l^1 \uplus \nabla_l^2, \theta_l\rangle \in \Theta$, onde ∇_l^2 é um contexto de ponto fixo cujas restrições primitivas envolvem átomos em $\overline{c}$.

Notação: $\Theta \lesssim \langle\Delta,\lambda\rangle$;

(ii) um par $\langle\Delta,\lambda\rangle$ é uma *instância de um par com exceções* $\langle\Phi,\sigma\rangle - \Theta$ se, e somente se, $\langle\Delta,\lambda\rangle$ é uma instância de $\langle\Phi,\sigma\rangle$ mas não de Θ.

Notação: $\langle\Phi,\sigma\rangle - \Theta \lesssim \langle\Delta,\lambda\rangle$.

Durante todo o trabalho, mostramos que a ideia de átomos novos aparece quando precisamos lidar com a α-equivalência entre termos nominais que possuem abstração de átomos distintos (e.g., $\Upsilon \vdash [a]s \overset{\curlywedge}{\approx}_\alpha [b]t$). A necessidade de uma extensão do contexto no item (i) da definição acima surge como uma resposta ao caso em que a parte desequacional de problemas de desunificação contém desequações entre esses tipos de termos.

Nesse cenário, restrições de ponto fixo envolvendo permutações cujo domínio contém átomos novos (para $\mathcal{P}_{\curlywedge}$) aparecem indiretamente como uma limitação imposta pela parte desequacional, conforme mostra o exemplo a seguir.

Exemplo 4.3. Considere o problema $\mathcal{P}_{\curlywedge} = \big\langle \underbrace{X \overset{\curlywedge}{\approx}{}^{?}_\alpha (a\ b)\cdot Y}_{\mathtt{Pr}} \mid\mid [a]X \overset{\curlywedge}{\not\approx}{}^{?}_\alpha [b]Y \big\rangle$. Então o par $\langle\Phi,\sigma\rangle = \big\langle\emptyset, [X \mapsto (a\ b)\cdot Y]\big\rangle$ é solução para $\mathtt{Pr}$ ao mesmo tempo em que satisfaz a parte

desequacional, isto é, $\Phi \not\vdash [a]X\sigma \overset{\curlywedge}{\approx}_\alpha [b]Y\sigma$ e, portanto, é uma solução para $\mathcal{P}_\curlywedge$. Por outro lado, aplicando a regra ($\overset{\curlywedge}{\approx}_\alpha$ *abs*2) da Tabela 3.1 à equação associada segue que

$$\begin{array}{rcl} \{[a]X \overset{\curlywedge ?}{\approx}_\alpha [b]Y\} & \Longrightarrow & \{X \overset{\curlywedge ?}{\approx}_\alpha (a\ b)\cdot Y,\ (a\ c_1) \curlywedge^? Y,\ (c_1\ c_2) \curlywedge^? Y\} \\ & \overset{[X\mapsto (a\ b)\cdot Y]}{\Longrightarrow} & \{(a\ c_1) \curlywedge^? Y,\ (c_1\ c_2) \curlywedge^? Y\} \end{array}$$

onde c_1, c_2 são nomes novos para $\mathcal{P}_\curlywedge$ e nesse caso temos $\overline{c} = \{c_1, c_2\}$ e $\overline{X} = \{X, Y\}$.

Ou seja, existem restrições de ponto fixo como uma limitação implícita dada pela desequação $[a]X \overset{\curlywedge ?}{\not\approx}_\alpha [b]Y$ e é justamente pelo fato de $a \notin \mathtt{perm}(\mathtt{dom}(\Phi|_Y))$ que a parte desequacional é satisfeita.

Além disso, observe que o par $\langle \nabla, \theta \rangle = \big\langle \{(a\ c_1) \curlywedge Y,\ (c_1\ c_2) \curlywedge Y\}, [X \mapsto (a\ b)\cdot Y] \big\rangle$ resolve a equação associada. Tomando $\Theta = \{\langle \nabla, \theta \rangle\}$ e como

$$\Phi^{\overline{c},\overline{X}} = \{(c_1\ c_2) \curlywedge X,\ (c_1\ c_2) \curlywedge Y\},$$

temos que $\langle \nabla, \theta \rangle \not\lesssim \langle \Phi^{\overline{c},\overline{X}}, \sigma \rangle$ o que, pela Definição 4.4, implica que $\Theta \not\lesssim \langle \Phi, \sigma \rangle$ e, portanto, $\langle \Phi, \sigma \rangle - \Theta \lesssim \langle \Phi, \sigma \rangle$.

Antes do próximo e último conceito desta seção, vamos a um outro exemplo sobre pares com exceções.

Exemplo 4.4. Apresentamos dois problemas de desunificação nos Exemplos 4.1 e 4.2: $\mathcal{P}^1_\curlywedge = \langle X \overset{\curlywedge ?}{\approx}_\alpha \mathsf{f}\ Y \mid\mid Y \overset{\curlywedge ?}{\not\approx}_\alpha a, Y \overset{\curlywedge ?}{\not\approx}_\alpha b \rangle$ e $\mathcal{P}^2_\curlywedge = \langle \mathsf{g}(X,Y) \overset{\curlywedge ?}{\approx}_\alpha \mathsf{g}(\mathsf{k}, a) \mid\mid X \overset{\curlywedge ?}{\not\approx}_\alpha \mathsf{k} \rangle$, respectivamente, com o primeiro contendo soluções e o segundo, insolúvel. Observe que podemos vincular a $\mathcal{P}^1_\curlywedge$ o par com exceções

$$\langle \Phi_1, \sigma_1 \rangle - \Theta = \big\langle \emptyset, [X \mapsto \mathsf{f}\ Y] \big\rangle - \big\{ \langle \emptyset, [Y \mapsto a] \rangle,\ \langle \emptyset, [Y \mapsto b] \rangle \big\}$$

e a $\mathcal{P}^2_\curlywedge$,

$$\langle \Phi_2, \sigma_2 \rangle - \Omega = \big\langle \emptyset, [X \mapsto \mathsf{k}, Y \mapsto a] \big\rangle - \big\{ \langle \emptyset, [X \mapsto \mathsf{k}] \rangle \big\}.$$

Vimos que o par $\big\langle \emptyset, [X \mapsto \mathsf{f}\ c, Y \mapsto c] \big\rangle$ é solução para $\mathcal{P}^1_\curlywedge$ e é fácil ver que também é instância do par com exceções $\langle \Phi_1, \sigma_1 \rangle - \Theta$, já que

$$\langle \Phi_1, \sigma_1 \rangle \lesssim_{[Y \mapsto c]} \big\langle \emptyset, [X \mapsto \mathsf{f}\ c, Y \mapsto c] \big\rangle \text{ e } \Theta \not\lesssim \big\langle \emptyset, [X \mapsto \mathsf{f}\ c, Y \mapsto c] \big\rangle,$$

enquanto que nada disso acontece com $\big\langle \emptyset, [X \mapsto \mathsf{f}\ a, Y \mapsto a] \big\rangle$. Já para o segundo problema, o par $\big\langle \emptyset, [X \mapsto \mathsf{k}, Y \mapsto a] \big\rangle$ não é solução para $\mathcal{P}^2_\curlywedge$e toda instância dele, inclusive ele prórprio, é

instância de $\langle \Phi_2, \sigma_2 \rangle$ mas também de Ω, logo o par com exceções $\langle \Phi_2, \sigma_2 \rangle - \Omega$ não possui instâncias.

O conceito de *consistência* de um par com exceções apresentada pela definição a seguir terá fundamental importância para a representação de soluções de um problema de desunificação.

Definição 4.5. Dado um problema de desunificação nominal $\mathcal{P}_{\curlywedge}$, um par com exceções $\langle \Phi, \sigma \rangle - \Theta$ para $\mathcal{P}_{\curlywedge}$ é *consistente* se, e somente se, possui pelo menos uma instância.

No exemplo anterior, o par $\langle \Phi_1, \sigma_1 \rangle - \Theta$ é consistente enquanto que $\langle \Phi_2, \sigma_2 \rangle - \Omega$ exemplifica um par com exceções inconsistente.

4.2 O Algoritmo $\texttt{consistente}_{\curlywedge}$

Esta seção apresenta os resultados necessários para construir o algortimo $\texttt{consistente}_{\curlywedge}$ (Algoritmo 1) o qual, dado um problema de desunificação, testa a consistência de pares com exceções vinculados a ele.

A seguir, apresentamos uma caracterização importante sobre a consistência de pares com exceções.

Lema 4.1 (Lema da Inconsistência)**.** Seja $\mathcal{P}_{\curlywedge}$ um problema de desunificação nominal. Um par com exceções $\langle \Phi, \sigma \rangle - \Theta$ para $\mathcal{P}_{\curlywedge}$ é inconsistente se, e somente se, $\Theta \lesssim \langle \Phi, \sigma \rangle$.

Demonstração. Por um lado, suponha que $\langle \Phi, \sigma \rangle - \Theta$ seja inconsistente. Então ele não possui qualquer instância, ou seja, para todo par $\langle \Delta, \lambda \rangle$, temos que $\langle \Phi, \sigma \rangle \not\lesssim \langle \Delta, \lambda \rangle$ ou é uma instância de Θ, $\Theta \lesssim \langle \Delta, \lambda \rangle$. Em particular, como $\langle \Phi, \sigma \rangle \lesssim \langle \Phi, \sigma \rangle$, segue que $\Theta \lesssim \langle \Phi, \sigma \rangle$.

Por outro lado, suponha que $\Theta \lesssim \langle \Phi, \sigma \rangle$, i.e., $\langle \Phi, \sigma \rangle$ é uma instância da família Θ. Pela Definição 4.4, para todo par $\langle \Delta, \lambda \rangle$ tal que $\langle \Phi^{\overline{c},\overline{X}}, \sigma \rangle \lesssim \langle \Delta, \lambda \rangle$, então $\langle \nabla_l, \theta_l \rangle \lesssim \langle \Delta, \lambda \rangle$, para algum $\langle \nabla_l, \theta_l \rangle \in \Theta$, onde $\overline{c}$ é um conjunto de nomes novos para $\mathcal{P}_{\curlywedge}$ e $\overline{X} = \texttt{Var}(\mathcal{P}_{\curlywedge})$. Suponha, por contradição, que $\langle \Phi, \sigma \rangle - \Theta$ seja consistente, isto é, que exista um par $\langle \Psi, \gamma \rangle$ tal que $\langle \Phi, \sigma \rangle \lesssim \langle \Psi, \gamma \rangle$ e $\Theta \not\lesssim \langle \Psi, \gamma \rangle$.

Afirmação: Se $\langle \Phi, \sigma \rangle \lesssim_\delta \langle \Psi, \gamma \rangle$, então $\langle \Phi^{\overline{c},\overline{X}}, \sigma \rangle \lesssim_\delta \langle \Psi^{\overline{c},\overline{X}}, \gamma \rangle$.

De fato, se $\langle \Phi, \sigma \rangle \lesssim_\delta \langle \Psi, \gamma \rangle$, então $\Psi \vdash \Phi\delta$ e $\Psi \vdash \gamma \overset{\curlywedge}{\approx}_\alpha \sigma\delta$. Pela propriedade de Enfraquecimento (Proposição 2.2), temos que

$$\Psi^{\overline{c},\overline{X}} \vdash \Phi\delta \ \text{ e } \ \Psi^{\overline{c},\overline{X}} \vdash \gamma \overset{\curlywedge}{\approx}_\alpha \sigma\delta.$$

Assim, para provar a afirmação, falta mostrar que $\Psi^{\overline{c},\overline{X}} \vdash \underbrace{\{(c\ d) \curlywedge X\delta \mid c,d \in \overline{c} \text{ e } X \in \overline{X}\}}_{\Phi^{\overline{c},\overline{X}} \setminus \Phi}$, lembrando que $\Psi^{\overline{c},\overline{X}} = \Psi \uplus \{(c\ d) \curlywedge X \mid c,d \in \overline{c} \text{ e } X \in \overline{X}\}$. Mas, como $\langle \Phi, \sigma \rangle - \Theta$ é um par com exceções vinculado a $\mathcal{P}_{\curlywedge}$ e os átomos em $\overline{c}$ são novos para $\mathcal{P}_{\curlywedge}$, podemos considerar, sem perda de generalidade, que $\texttt{Var}(\Phi, \Psi) \subseteq \texttt{Var}(\mathcal{P}_{\curlywedge})$ e $\texttt{VRan}(\sigma, \gamma, \delta) \subseteq \texttt{Var}(\mathcal{P}_{\curlywedge})$ e que $\overline{c}$ também é novo para os contextos Φ, Ψ e substituições σ, γ e δ. Com isso, segue por indução na derivação $\Psi^{\overline{c},\overline{X}} \vdash (c\ d) \curlywedge Y\delta$, para cada $Y \in \overline{X}$, que $\Psi^{\overline{c},\overline{X}} \vdash \{(c\ d) \curlywedge X\delta \mid c,d \in \overline{c} \text{ e } X \in \overline{X}\}$. Portanto, $\langle \Phi^{\overline{c},\overline{X}}, \sigma \rangle \lesssim_{\delta} \langle \Psi^{\overline{c},\overline{X}}, \gamma \rangle$, como queríamos.

Agora, como $\Theta \not\lesssim \langle \Psi, \gamma \rangle$ e usando a afirmação anterior e a transitividade de $\lesssim$, temos que existe um par $\langle \Delta', \lambda' \rangle$ tal que $\langle \Phi^{\overline{c},\overline{X}}, \sigma \rangle \lesssim \langle \Psi^{\overline{c},\overline{X}}, \gamma \rangle \lesssim \langle \Delta', \lambda' \rangle$ e $\langle \nabla_l, \theta_l \rangle \not\lesssim \langle \Delta', \lambda' \rangle$, para todo $\langle \nabla_l, \theta_l \rangle \in \Theta$, o que é uma contradição com a hipótese de que $\Theta \lesssim \langle \Phi, \sigma \rangle$. Logo, $\langle \Phi, \sigma \rangle - \Theta$ é inconsistente.

□

Apesar desse lema apresentar uma maneira simples de verificar a consistência, é o próximo corolário que nos permite construir o algoritmo $\texttt{consistente}_{\curlywedge}$ (Algoritmo 1) que testa a consistência de pares com exceções.

Dado um par com exceção $\langle \Phi, \sigma \rangle - \Theta$ para um problema de desunificação $\mathcal{P}_{\curlywedge}$, para determinar se $\Theta \lesssim \langle \Phi, \sigma \rangle$, o corolário propõe que resolvamos um problema de emparelhamento sob contexto com múltiplas equações, uma para cada variável $X \in \texttt{Var}(\mathcal{P}_{\curlywedge}) = \{X_1, \ldots, X_m\}$:

$$\begin{cases} (\Phi \vdash X_1\sigma) \ {}_{?}{\approx} (\nabla_l \vdash X_1\theta_l) \\ \qquad\qquad \vdots \\ (\Phi \vdash X_m\sigma) \ {}_{?}{\approx} (\nabla_l \vdash X_m\theta_l) \end{cases} \tag{4.1}$$

onde $\langle \nabla_l, \theta_l \rangle$ é algum par na família Θ.

Observe que na definição de emparelhamento sob contexto dada ao final do capítulo anterior (Definição 3.10), o problema possui uma única equação e a solução também é apresentada em função disso. Agora, para que uma substituição δ seja solução para (4.1), estendemos aquela definição requerindo que:

(i) $\langle \nabla_l{}^{?},\ X_1\sigma \overset{\curlywedge\ ?}{\approx}_{\alpha} X_1\theta_l,\ \ldots,\ X_m\sigma \overset{\curlywedge\ ?}{\approx}_{\alpha} X_m\theta_l \rangle_{sol} = \langle \Psi', \delta \rangle$;

(ii) $\Phi^{\overline{c},\overline{X}} \vdash \Psi'$;

(iii) $\texttt{dom}(\delta) \cap \texttt{Var}(\Phi, X_1\sigma, \ldots, X_m\sigma) = \emptyset$.

Além disso, usaremos um método proposto por Ayala-Rincón et al. [ARdCSF$^+$21] o qual utiliza um conjunto de *variáveis protegidas* para resolver problemas de emparelhamento.

Intuitivamente o conjunto das variáveis protegidas $\mathcal{X}$ de um problema de emparelhamento consistirá das variáveis que ocorrem do lado esquerdo do problema, bem como do seu contexto. Basicamente o método consiste em aplicar as regras de simplificação para unificaçao nominal, com o cuidado de que as regras ($\overset{\curlywedge}{\approx}_\alpha$ *inst*1) e ($\overset{\curlywedge}{\approx}_\alpha$ *inst*2) não instanciem variáveis protegidas. Este procedimento foi provado ser terminante e correto, mas a descrição detalhada do algoritmo, que chamaremos de $\texttt{matching}^+_{\curlywedge}$, bem como suas propriedades estão fora do escopo deste trabalho, e podem ser verificadas em [ARdCSF$^+$21].

Corolário 4.1. Seja $\langle\Phi,\sigma\rangle - \Theta$ um par com exceções para um problema de desunificação $\mathcal{P}_{\curlywedge}$ e denote por $\overline{X} := \texttt{Var}(\mathcal{P}_{\curlywedge}) = \{X_1,\ldots,X_m\}$. Se para algum par $\langle\nabla_l,\theta_l\rangle \in \Theta$ existe uma substituição δ que seja solução para o sistema de emparelhamento sob contexto

$$\begin{cases} (\Phi \vdash X_1\sigma) \,{}_?{\approx}\, (\nabla_l \vdash X_1\theta_l) \\ \qquad\qquad \vdots \\ (\Phi \vdash X_m\sigma) \,{}_?{\approx}\, (\nabla_l \vdash X_m\theta_l) \end{cases} \tag{4.2}$$

então o par $\langle\Phi,\sigma\rangle - \Theta$ é inconsistente.

Demonstração. Suponha que δ seja solução para (4.2). Então

(i) $\langle \nabla_l{}^?,\ X_1\sigma \overset{\curlywedge}{\approx}{}^?_\alpha X_1\theta_l,\ \ldots,\ X_m\sigma \overset{\curlywedge}{\approx}{}^?_\alpha X_m\theta_l\rangle_{sol} = \langle\Psi',\delta\rangle$;

(ii) $\Phi^{\overline{c},\overline{X}} \vdash \Psi'$;

(iii) $\texttt{dom}(\delta) \cap \texttt{Var}(\Phi, X_1\sigma,\ldots,X_m\sigma) = \emptyset$.

De (i) segue que $\Psi' \vdash \nabla_l\delta$ e $\Psi' \vdash X_i\sigma\delta \overset{\curlywedge}{\approx}_\alpha X_i\theta_l\delta$, para todo $X_i \in \overline{X}$. Como $\Phi^{\overline{c},\overline{X}} \vdash \Psi'$ e usando a Preservabilidade por Substiuições (Proposição 2.3), temos que

$$\Phi^{\overline{c},\overline{X}} \vdash \nabla_l\delta$$

e $\Phi^{\overline{c},\overline{X}} \vdash X_i\sigma\delta \overset{\curlywedge}{\approx}_\alpha X_i\theta_l\delta$, para todo $X_i \in \overline{X}$. Porém, por (iii), $X_i\sigma\delta = X_i\sigma$, para todo $X_i \in \overline{X}$, logo

$$\Phi^{\overline{c},\overline{X}} \vdash X_i\sigma \overset{\curlywedge}{\approx}_\alpha X_i\theta_l\delta,$$

para todo $X_i \in \overline{X}$. Portanto, $\langle\nabla_l,\theta_l\rangle \lesssim \langle\Phi^{\overline{c},\overline{X}},\sigma\rangle$, e pela definição de instância de um par com exceções (Definição 4.4) segue que $\Theta \lesssim \langle\Phi,\sigma\rangle$. Assim, pelo Lema da Inconsistência (Lema 4.1), temos que o par com exceções $\langle\Phi,\sigma\rangle - \Theta$ é inconsistente. □

Agora, estamos prontos para apresentar o algoritmo do teste de consistência de um par com exceções, o qual chamaremos por $\texttt{consistente}_{\curlywedge}$. Ele começa a partir de um par com

exceções finito, isto é, cuja família Θ é finita, e decide sobre sua consistência usando como apoio o algoritmo $\texttt{matching}^{+}_{\curlywedge}$ da seguinte maneira: para cada par $\langle \nabla_l, \theta_l \rangle \in \Theta$ é verificado se o sistema de emparelhamento sob contexto (4.2) possui uma solução; em caso afirmativo, o algoritmo retorna `falha` e é finalizado, o que significa que o par com exceções dado como entrada é inconsistente; caso não haja solução após ter analisado todos os pares em Θ, o procedimento retorna `sucesso`, o que significa que a entrada é consistente.

Algoritmo 1 $\texttt{consistente}_{\curlywedge}$

entrada: $\langle \Phi, \sigma \rangle - \Theta$, um par com exceções finito para um problema de desunificação $\mathcal{P}_{\curlywedge}$
saída: `sucesso`, se a entrada é consistente; `falha`, caso contrário
para $\langle \nabla_l, \theta_l \rangle \in \Theta$ **faça**
  se $\texttt{matching}^{+}_{\curlywedge}\big((\Phi \vdash X_1\sigma) \,{}_{?}{\approx} (\nabla_l \vdash X_1\theta_l), \ldots, (\Phi \vdash X_m\sigma) \,{}_{?}{\approx} (\nabla_l \vdash X_m\theta_l)\big) = \delta$ **então**
    retorne `falha` e **pare**
  fim se
fim para
  retorne `sucesso`

Observe que, de fato, $\texttt{consistente}_{\curlywedge}$ trata-se de um algoritmo terminante, pois $\texttt{matching}^{+}_{\curlywedge}$ o é, e correto: quando é retornado `falha`, pelo Corolário 4.1, o par com exceções dado como entrada é inconsistente; caso seja retornado `sucesso`, significa que o problema de emparelhamento não possui solução, logo $\Theta \not\lesssim \langle \Phi, \sigma \rangle$ e pelo Lema da Inconsistência (Lema 4.1) a entrada é consistente.

4.3 Representação Completa de Soluções

Vimos que em Desunificação Nominal as soluções para os problemas podem não ser preservadas por instanciações, o que dificulta representá-las de maneira finitária. Entretanto, dado um problema de desunificação $\mathcal{P}_{\curlywedge}$, com o auxílio dos pares com exceções e suas propriedades sobre instanciação e consistência, é possível definir um conjunto S contendo essas estruturas de forma que toda instância de seus elementos é solução para $\mathcal{P}_{\curlywedge}$. Esse conjunto é o que chamaremos de *Representação Completa de Soluções*.

Definição 4.6 (Representação Completa de Soluções)**.** Dizemos que um conjunto S de pares com exceções é uma *representação completa de soluções* para um problema de desunificação $\mathcal{P}_{\curlywedge}$ se, e somente se, S satisfaz as seguintes condições:

(i) se $\langle \Phi, \sigma \rangle - \Theta \lesssim \langle \Delta, \lambda \rangle$ para algum $\langle \Phi, \sigma \rangle - \Theta$ em S, então $\langle \Delta, \lambda \rangle$ é solução para $\mathcal{P}_{\curlywedge}$;

(ii) se $\langle \Delta, \lambda \rangle$ é solução para $\mathcal{P}_{\curlywedge}$, então ele é uma instância de algum $\langle \Phi, \sigma \rangle - \Theta$ em S;

(iii) $\langle \Phi, \sigma \rangle - \Theta$ é consistente, para todo $\langle \Phi, \sigma \rangle - \Theta \in S$.

A seguir apresentamos o principal resultado deste trabalho.

Teorema 4.1 (Teorema da Representação)**.** Seja $\mathcal{P}_{\curlywedge} = \langle \mathtt{Pr} \mid\mid D_{\curlywedge} \rangle$ um problema de desunificação nominal, onde $D_{\curlywedge} = \{p_1 \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q_1, \ldots, p_n \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q_n\}$. Defina a família

$$\Theta := \bigcup_{p_i \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q_i \in D_{\curlywedge}} \mathcal{U}(p_i \overset{\curlywedge}{\approx}^{?}_{\alpha} q_i).$$

Então o conjunto $S := \{\langle \Phi, \sigma \rangle - \Theta \mid \langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr}) \text{ e } \Theta \not\lesssim \langle \Phi, \sigma \rangle\}$ é uma representação completa de soluções para o problema $\mathcal{P}_{\curlywedge}$.

Demonstração. Mostraremos que S satisfaz os itens da Definição 4.6:

(i) Tome $\langle \Delta, \lambda \rangle$ uma instância de algum par com exceções $\langle \Phi, \sigma \rangle - \Theta \in S$. Por definição, $\langle \Phi, \sigma \rangle \lesssim \langle \Delta, \lambda \rangle$ mas $\Theta \not\lesssim \langle \Delta, \lambda \rangle$. Logo, existe uma substituição δ tal que

$$\Delta \vdash \Phi\delta \;\text{ e }\; \Delta \vdash \sigma\delta \overset{\curlywedge}{\approx}_{\alpha} \lambda. \tag{4.3}$$

Como $\langle \Phi, \sigma \rangle \in \mathcal{U}(\mathtt{Pr})$, pela propriedade de Fechamento por Instanciação (Lema 3.2) segue que $\langle \Delta, \lambda \rangle \in \mathcal{U}(\mathtt{Pr})$.

Falta mostrar que o par satisfaz a parte desequacional $D_{\curlywedge}$. Para isso, suponha por contradição que $\Delta \vdash p_j\lambda \overset{\curlywedge}{\approx}_{\alpha} q_j\lambda$, para algum $p_j \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q_j \in D_{\curlywedge}$. Logo, pela definição de Θ, temos que $\langle \Delta, \lambda \rangle \in \Theta$. Note que, pela propriedade de Enfraquecimento (Proposição 2.2), $\langle \Delta, \lambda \rangle \lesssim_{\mathtt{id}} \langle \Delta^{\overline{c},\overline{X}}, \lambda \rangle$, onde $\overline{c}$ é um conjunto finito de nomes novos para $\mathcal{P}_{\curlywedge}$ e $\overline{X} = \mathtt{Var}(\mathcal{P}_{\curlywedge})$. Com isso, toda instância de $\langle \Delta^{\overline{c},\overline{X}}, \lambda \rangle$ é instância de $\langle \Delta, \lambda \rangle$ que está em Θ, ou seja, $\Theta \lesssim \langle \Delta, \lambda \rangle$, o que contradiz com a hipótese inicial. Portanto, o par $\langle \Delta, \lambda \rangle$ é solução para $\mathcal{P}_{\curlywedge}$;

(ii) Suponha que $\langle \Delta, \lambda \rangle$ seja solução para $\mathcal{P}_{\curlywedge}$. Então, $\langle \Delta, \lambda \rangle \in \mathcal{U}(\mathtt{Pr})$ e $\Delta \not\vdash p_i\lambda \overset{\curlywedge}{\approx}_{\alpha} q_i\lambda$, para toda $p_i \overset{\curlywedge}{\not\approx}^{?}_{\alpha} q_i \in D_{\curlywedge}$. Vamos mostrar que o par com exceções que procuramos é $\langle \Delta, \lambda \rangle - \Theta$, ou seja, que $\langle \Delta, \lambda \rangle - \Theta \in S$ e $\langle \Delta, \lambda \rangle - \Theta \lesssim \langle \Delta, \lambda \rangle$.

Afirmação: $\langle \Delta^{\overline{c},\overline{X}}, \lambda \rangle \lesssim_{\mathtt{id}} \langle \Delta, \lambda \rangle$.

De fato, como os átomos de $\overline{c}$ são novos para $\mathcal{P}_{\curlywedge}$, então $\mathtt{dom}((c\ d)) \cap \mathtt{supp}_{\Delta}(X) = \emptyset$, para todo $c, d \in \overline{c}$ e todo $X \in \overline{X}$. Logo, pelo Teorema 2.4, segue que $\Delta \vdash (c\ d) \curlywedge X$, para todo $c, d \in \overline{c}$ e todo $X \in \overline{X}$, isto é, $\Delta \vdash \Delta^{\overline{c},\overline{X}}$. Como $\Delta \vdash \lambda \overset{\curlywedge}{\approx}_{\alpha} \lambda$, a afirmação segue.

Note também que $\langle \nabla_l, \theta_l \rangle \not\lesssim \langle \Delta, \lambda \rangle$, para todo $\langle \nabla_l, \theta_l \rangle \in \Theta$, pois caso contrário, pela definição de Θ, o par $\langle \Delta, \lambda \rangle$ seria instância da solução de alguma desequação $p_j \overset{\curlywedge ?}{\not\approx}_\alpha q_j \in D_\curlywedge$, o que implicaria em $\Delta \vdash p_j\lambda \overset{\curlywedge}{\approx}_\alpha q_j\lambda$, uma contradição. Com isso, provamos que $\langle \Delta, \lambda \rangle$ é uma instância de $\langle \Delta^{\overline{c},\overline{X}}, \lambda \rangle$ tal que $\langle \nabla_l, \theta_l \rangle \not\lesssim \langle \Delta, \lambda \rangle$, para todo $\langle \nabla_l, \theta_l \rangle \in \Theta$, ou seja, $\Theta \not\lesssim \langle \Delta, \lambda \rangle$.

Logo, $\langle \Delta, \lambda \rangle - \Theta \in S$ e $\langle \Delta, \lambda \rangle - \Theta \lesssim \langle \Delta, \lambda \rangle$.

(iii.) Se $\langle \Phi, \sigma \rangle - \Theta \in S$, sua consistência segue diretamente da construção de S, pois $\Theta \not\lesssim \langle \Phi, \sigma \rangle$.

□

Finalmente, apresentamos o algoritmo que computa uma representação das soluções para um problema de desunificação nominal, o qual chamaremos por $\texttt{desunif}_\curlywedge$. Em suma, o algoritmo tem início a partir de um problema de desunificação $\mathcal{P}_\curlywedge$ dado como entrada, e utiliza como apoio o algoritmo $\texttt{unif}_\curlywedge$ para encontrar a *smg* para $\texttt{Pr}$ e construir a família Θ contendo as *smg*'s correspondentes a cada desequação presente em $D_\curlywedge$; com isso, é formado um par com exceções cuja consistência é testada pelo algoritmo $\texttt{consistente}_\curlywedge$, e, em caso positivo, é retornado tal par com exceções que será uma representação completa de soluções para $\mathcal{P}_\curlywedge$; caso contrário, retorna um conjunto vazio, o que representa que $\mathcal{P}_\curlywedge$ não possui solução.

Algoritmo 2 $\texttt{desunif}_\curlywedge$

entrada: um problema de desunificação $\mathcal{P}_\curlywedge = \langle \texttt{Pr} \mid\mid D_\curlywedge \rangle$.
saída: um conjunto unitário S de um par com exceções para $\mathcal{P}_\curlywedge$, se $\mathcal{P}_\curlywedge$ for solúvel; um conjunto vazio, caso contrário.
seja $\langle \Phi, \sigma \rangle := \texttt{unif}_\curlywedge(\texttt{Pr})$
seja $\Theta := \bigcup_{p_l \overset{\curlywedge ?}{\not\approx}_\alpha q_l \in D_\curlywedge} \{\langle \nabla_l, \theta_l \rangle := \texttt{unif}_\curlywedge(p_l \overset{\curlywedge ?}{\approx}_\alpha q_l)\}$
se $\texttt{consistente}_\curlywedge(\langle \Phi, \sigma \rangle - \Theta)$ **então**
  retorne $\{\langle \Phi, \sigma \rangle - \Theta\}$
se não
  retorne $\emptyset$
fim se

A terminação de $\texttt{desunif}_\curlywedge$ segue do fato de que os algoritmos $\texttt{unif}_\curlywedge$ e $\texttt{consistente}_\curlywedge$ são terminantes e o seguinte resultado mostra que ele também é correto, isto é:

1. $\texttt{desunif}_{\curlywedge}$ é **sólido/robusto**: se o algoritmo for bem-sucedido a partir de um problema de desunificação $\mathcal{P}_{\curlywedge}$ dado como entrada, então o que ele retorna como saída é, de fato, uma *representação completa de soluções* para $\mathcal{P}_{\curlywedge}$;

2. $\texttt{desunif}_{\curlywedge}$ é **completo**: se o algoritmo falha (ou, no nosso caso, retorna $\emptyset$), então o problema de desunificação não possui solução.

Teorema 4.2 (Correção)**.** Seja $\mathcal{P}_{\curlywedge} = \langle \texttt{Pr} \mid\mid D_{\curlywedge} \rangle$ um problema de desunificação nominal, onde $D_{\curlywedge} = \{p_1 \overset{\curlywedge ?}{\not\approx}_{\alpha} q_1, \ldots, p_n \overset{\curlywedge ?}{\not\approx}_{\alpha} q_n\}$, considere $\langle \Phi, \sigma \rangle := \langle \texttt{Pr} \rangle_{sol}$ e defina a família

$$\Theta := \bigcup_{p_l \overset{\curlywedge ?}{\not\approx}_{\alpha} q_l \in D_{\curlywedge}} \left\{ \langle \nabla_l, \theta_l \rangle := \langle p_l \overset{\curlywedge ?}{\approx}_{\alpha} q_l \rangle_{sol} \right\}.$$

Se $\langle \Phi, \sigma \rangle - \Theta$ é consistente, então o conjunto unitário $\mathcal{S} := \left\{ \langle \Phi, \sigma \rangle - \Theta \right\}$ é uma representação completa para $\mathcal{P}_{\curlywedge}$. Caso contrário, $\mathcal{P}_{\curlywedge}$ não possui solução.

Demonstração. A prova de que $\mathcal{S}$ é uma representação completa de soluções para $\mathcal{P}_{\curlywedge}$ segue de maneira análoga ao demonstrado no Teorema da Representação, basta observar que $\langle \Phi, \sigma \rangle$ é uma *smg* para $\texttt{Pr}$ e que todo par $\langle \nabla_l, \theta_l \rangle \in \Theta$ é uma *smg* para alguma equação associada a $D_{\curlywedge}$, e portanto $\langle \Phi, \sigma \rangle \in \mathcal{U}(\texttt{Pr})$ e $\langle \nabla_l, \theta_l \rangle \in \mathcal{U}(p_l \overset{\curlywedge ?}{\approx}_{\alpha} q_l)$, para alguma desequação $p_l \overset{\curlywedge ?}{\not\approx}_{\alpha} q_l \in D_{\curlywedge}$.

Agora, suponha que $\langle \Phi, \sigma \rangle - \Theta$ seja inconsistente. Então, pelo Lema da Inconsistência (Lema 4.1), segue que $\Theta \lesssim \langle \Phi, \sigma \rangle$, ou seja, toda instância de $\langle \Phi^{\overline{c}, \overline{X}}, \sigma \rangle$ é instância de algum par $\langle \nabla_l, \theta_l \rangle \in \Theta$, onde $\overline{c}$ é um conjunto finito de nomes novos para $\mathcal{P}_{\curlywedge}$ e $\overline{X} = \texttt{Var}(\mathcal{P}_{\curlywedge})$.

Seja $\langle \Delta, \lambda \rangle \in \mathcal{U}(\texttt{Pr})$. Como $\langle \Phi, \sigma \rangle$ é uma *smg* para $\texttt{Pr}$, temos que $\langle \Phi, \sigma \rangle \lesssim \langle \Delta, \lambda \rangle$. Pela afirmação provada na demonstração do Teorema da Representação, vimos que $\langle \Phi^{\overline{c}, \overline{X}}, \sigma \rangle \lesssim \langle \Phi, \sigma \rangle$. Logo, por hipótese, $\langle \nabla_l, \theta_l \rangle \lesssim \langle \Delta, \lambda \rangle$, para algum $\langle \nabla_l, \theta_l \rangle \in \Theta$. Como $\langle \nabla_l, \theta_l \rangle$ é uma *smg* para uma equação associada $p_l \overset{\curlywedge ?}{\approx}_{\alpha} q_l$, por definição, então $\Delta \vdash p_l \lambda \overset{\curlywedge}{\approx}_{\alpha} q_l \lambda$, ou seja, não satisfaz a parte desequacional $D_{\curlywedge}$.

Assim, provamos que toda solução para $\texttt{Pr}$ não satisfaz $D_{\curlywedge}$, portanto, $\mathcal{P}_{\curlywedge}$ não possui solução nesse caso. $\square$

Por fim, finalizaremos os trabalhos aplicando o algoritmo $\texttt{desunif}_{\curlywedge}$ em um problema de desunificação para ilustrar seu funcionamento.

Exemplo 4.5. Considere o problema

$$\mathcal{P}'_{\curlywedge} = \Big\langle \underbrace{(a\ d) \curlywedge Z,\ g(X,Y) \overset{\curlywedge ?}{\approx}_{\alpha} g((a\ b) \cdot Y, \mathsf{f}\ Z)}_{\texttt{Pr}} \mid\mid [a]X \overset{\curlywedge ?}{\not\approx}_{\alpha} [b]\mathsf{f}\ Z \Big\rangle.$$

Vamos aplicar $\texttt{desunif}_{\curlywedge}$ para encontrar uma representação de sua solução.

1. Unificar $\texttt{Pr}$ utilizando $\texttt{unif}_{\curlywedge}$:

   Aplicando $\texttt{unif}_{\curlywedge}$ em $\texttt{Pr}$, temos:

$$\begin{array}{rcl} \texttt{Pr} & \Longrightarrow & \{(a\ d) \curlywedge Z,\ X \overset{\curlywedge ?}{\approx}_{\alpha} (a\ b)\cdot Y,\ Y \overset{\curlywedge ?}{\approx}_{\alpha} \mathsf{f}\ Z\} \\ & \overset{[X\mapsto (a\ b)\cdot Y]}{\Longrightarrow} & \{(a\ d) \curlywedge Z,\ Y \overset{\curlywedge ?}{\approx}_{\alpha} \mathsf{f}\ Z\} \\ & \overset{[Y\mapsto \mathsf{f}\ Z]}{\Longrightarrow} & \{(a\ d) \curlywedge Z\} \end{array}$$

   Portanto, o procedimento retorna $\langle \Phi, \sigma \rangle = \big\langle \{(a\ d) \curlywedge Z\}, [X \mapsto \mathsf{f}((a\ b)\cdot Z),\ Y \mapsto \mathsf{f}\ Z] \big\rangle$.

2. Unificar as equações associadas às desequações de $D_{\curlywedge}$ utilizando $\texttt{unif}_{\curlywedge}$ e colecionar as soluções no conjunto Θ:

   A parte desequacional $D_{\curlywedge}$ possui apenas uma equação associada, a saber $[a]X \overset{\curlywedge ?}{\approx}_{\alpha} [b]\mathsf{f}\ Z$. Aplicando $\texttt{unif}_{\curlywedge}$, temos:

$$\begin{array}{rcl} \{[a]X \overset{\curlywedge ?}{\approx}_{\alpha} [b]\mathsf{f}\ Z\} & \Longrightarrow & \{X \overset{\curlywedge ?}{\approx}_{\alpha} \mathsf{f}((a\ b)\cdot Z),\ (a\ c_1) \curlywedge^{?} \mathsf{f}\ Z,\ (c_1\ c_2) \curlywedge^{?} Z\} \\ & \Longrightarrow & \{X \overset{\curlywedge ?}{\approx}_{\alpha} \mathsf{f}((a\ b)\cdot Z),\ (a\ c_1) \curlywedge^{?} Z,\ (c_1\ c_2) \curlywedge^{?} Z\} \\ & \overset{[X\mapsto \mathsf{f}((a\ b)\cdot Z)]}{\Longrightarrow} & \{(a\ c_1) \curlywedge^{?} Z,\ (c_1\ c_2) \curlywedge^{?} Z\}, \end{array}$$

   onde c_1 e c_2 são nomes novos. Portanto, obtemos $\Theta = \{\langle \nabla_1, \theta_1 \rangle\}$, onde $\nabla_1 = \{(a\ c_1) \curlywedge Z,\ (c_1\ c_2) \curlywedge Z\}$ e $\theta_1 = [X \mapsto \mathsf{f}((a\ b)\cdot Z)]$. Nesse passo, é formado o conjunto de nomes novos para $\mathcal{P}_{\curlywedge}$, $\overline{c} = \{c_1, c_2\}$.

3. Testar a consistência do par com exceções gerado pelos passos 1 e 2, utilizando $\texttt{consistente}_{\curlywedge}$:

   Como Θ possui um único par e $\overline{X} = \texttt{Var}(\mathcal{P}_{\curlywedge}) = \{X, Y, Z\}$, consideramos o sistema de emparelhamento sob contexto:

$$\begin{cases} (\Phi \vdash X\sigma)\ {}_{?}{\approx}(\nabla_1 \vdash X\theta_1) \\ (\Phi \vdash Y\sigma)\ {}_{?}{\approx}(\nabla_1 \vdash Y\theta_1) \\ (\Phi \vdash Z\sigma)\ {}_{?}{\approx}(\nabla_1 \vdash Z\theta_1) \end{cases}$$

   que equivale a

$$\begin{cases} (\Phi \vdash \mathsf{f}((a\ b)\cdot Z))\ {}_{?}{\approx}(\nabla_1 \vdash \mathsf{f}((a\ b)\cdot Z)) \\ (\Phi \vdash \mathsf{f}\ Z)\ {}_{?}{\approx}(\nabla_1 \vdash Y) \\ (\Phi \vdash Z)\ {}_{?}{\approx}(\nabla_1 \vdash Z) \end{cases} \tag{4.4}$$

Note que $\delta = [Y \mapsto \mathtt{f}\ Z]$ é solução para (4.4), já que:

(i) $\langle \nabla_1{}^?,\ \mathtt{f}((a\ b)\cdot Z) \overset{\curlywedge ?}{\approx}_\alpha \mathtt{f}((a\ b)\cdot Z),\ \mathtt{f}\ Z \overset{\curlywedge ?}{\approx}_\alpha Y,\ Z \overset{\curlywedge ?}{\approx}_\alpha Z\rangle_{sol} = \langle \nabla_1, \delta\rangle$;

(ii) como $\Phi^{\overline{c},\overline{X}} = \{(a\ d) \curlywedge Z,\ (c_1\ c_2) \curlywedge Z\}$ e $\nabla_1 = \{(a\ c_1) \curlywedge Z,\ (c_1\ c_2) \curlywedge Z\}$, temos que $\mathtt{dom}(\mathtt{perm}(\Phi^{\overline{c},\overline{X}}|_Z)) \subseteq \mathtt{dom}(\mathtt{perm}(\nabla_1|_Z))$ e portanto $\Phi^{\overline{c},\overline{X}} \vdash \nabla_1$;

(iii) e como $\mathtt{dom}(\delta) = \{Y\}$ e $\mathtt{Var}(\Phi, X\sigma, Y\sigma, Z\sigma) = \{Z\}$, temos também que $\mathtt{dom}(\delta) \cap \mathtt{Var}(\Phi, X\sigma, Y\sigma, Z\sigma) = \emptyset$.

Portanto, $\mathtt{consistente}_\curlywedge$ retorna $\mathtt{falha}$, o que faz com que $\mathtt{desunif}_\curlywedge$ retorne $\emptyset$, e concluimos que $\mathcal{P}'_\curlywedge$ não possui solução.

Capítulo 5

Conclusão

Esta dissertação fez uso da linguagem nominal via ponto fixo para estender, no âmbito sintático, o trabalho existente sobre desunificação nominal, desenvolvido recentemente, mas sob a abordagem de *freshness*, por Ayala-Rincón et al. em [AFNV20]. Inicialmente, foi apresentada a sintaxe dos termos nominais e suas propriedades, para, em um segundo momento, tratar da unificação nominal via ponto fixo, aonde foi exibido um procedimento que decide sobre a solubilidade destes problemas, chamado $\texttt{unif}_{\curlywedge}$.

Uma vez que fizemos a escolha de implementar a relação de ponto fixo sem utilizar o quantificador $\mathsf{И}$, assumindo a possibilidade de sempre poder escolher nomes novos (fora do suporte), foi necessário reformular e verificar as demonstrações de todos os resultados. Além disso, também estabelecemos uma nova definição para o problema de *emparelhamento sob contexto* utilizando a abordagem de ponto fixo, que foi fundamental para obtenção dos resultados principais.

Por fim, usufruindo de tudo o que foi feito até então, foram desenvolvidas ferramentas importantes, como o algoritmo $\texttt{consistente}_{\curlywedge}$ que testa a consistência de pares com exceções, para se alcançar um dos resultados principais do trabalho, o Teorema 4.2, que fornece a correção para o algoritmo $\texttt{desunif}_{\curlywedge}$, o qual decide sobre a solubilidade de problemas de desunificação nominal via ponto fixo e computa uma representação finita e completa de soluções (Teorema 4.1).

Trabalhos Futuros. Pretendemos investigar extensões do problema de desunificação nominal via ponto fixo. Imediamente, vemos três possibilidades interessantes:

1. A exploração de extensões envolvendo teorias equacionais, na direção do trabalho proposto por Fernández [Fer93] que utiliza técnicas de *narrowing* para o tratamento de desunifição equacional, parece bastante promissora, uma vez que *Nominal Narrowing*

já foi estabelecido em trabalhos anteriores [AFN16]. Além disso, este trabalho teria potencial para aplicações práticas, por exemplo, na análise de segurança de protocolos criptográficos;

2. Por outro lado, generalizações do problema de desunificação nominal, utilizando quantificadores e disjunções, como é o caso do *Problema Equacional Nominal* introduzido em [AFNV21], utilizando a abordagem de ponto fixo, permitiria também o tratamento de teorias equacionais permutativas (aquelas que incluem comutatividade);

3. A generalização do chamado *Problema de AC-Complemento*, investigado em [Fer93], para sintaxe nominal seria tratável utilizando a abordagem nominal via ponto fixo.

Finalmente, acreditamos que este trabalho tem ainda muitas outras possibilidades de extensões e aplicações, como é o caso de combinação de teorias equacionais, generalização para desunificação de ordem superior, o estudo de aplicações em *sufficient completeness* [Com86], eliminação de negação em linguagens de programação lógica, formalizações, entre outros.

Bibliografia

[AdCSFN17] Mauricio Ayala-Rincón, Washington de Carvalho Segundo, Maribel Fernández, and Daniele Nantes-Sobrinho. On solving nominal fixpoint equations. In *Front. of Combining Systems - 11th Int. Symp., FroCoS 2017, Proc.*, volume 10483 of *LNCS*, pages 209–226. Springer, 2017.

[AFN16] Mauricio Ayala-Rincón, Maribel Fernández, and Daniele Nantes-Sobrinho. Nominal narrowing. In Delia Kesner and Brigitte Pientka, 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, 2016.

[AFN20] Mauricio Ayala-Rincón, Maribel Fernández, and Daniele Nantes-Sobrinho. On nominal syntax and permutation fixed points. *Logical Methods in Computer Science*, 16(1), 2020.

[AFNV20] Mauricio Ayala-Rincón, Maribel Fernández, Daniele Nantes-Sobrinho, and Deivid Vale. On solving nominal disunification constraints. In *Proc. LSFA 2019*, volume 348 of *ENTCS*, pages 3–22. Elsevier, 2020.

[AFNV21] Mauricio Ayala-Rincón, Maribel Fernández, Daniele Nantes-Sobrinho, and Deivid Vale. Nominal equational problems. In Stefan Kiefer and Christine Tasson, editors, *Foundations of Software Science and Computation Structures - 24th International Conference, FOSSACS 2021, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2021, Luxembourg City, Luxembourg, March 27 - April 1, 2021, Proceedings*, volume 12650 of *Lecture Notes in Computer Science*, pages 22–41. Springer, 2021.

[ARdCSF+21] Mauricio Ayala-Rincón, Washington de Carvalho-Segundo, Maribel Fernández, Gabriel Ferreira Silva, and Daniele Nantes-Sobrinho. Formalising nominal C-unification generalised with protected variables. *Mathematical Structures in Computer Science*, page 1–26, 2021.

[BB94] Wray L. Buntine and Hans-Jürgen Bürckert. On solving equations and disequations. *Journal of the ACM*, 41(4):591–629, 1994.

[BFNV21] Leonardo M. Batista, Maribel Fernández, Daniele Nantes-Sobrinho, and Deivid Vale. Nominal disunification via fixed-point constraints (work in progress). In *Informal Proceedings of the 35th International Workshop on Unification (UNIF)*, 2021.

[BN98] Franz Baader and Tobias Nipkow. *Term Rewriting And All That*. Cambridge University Press, 1998.

[BN21] Leonardo M. Batista and Daniele Nantes-Sobrinho. Desunificação nominal via ponto fixo. In *Anais do II Workshop Brasileiro de Lógica (WBL2021)*, 2021.

[CL89] Hubert Comon and Pierre Lescanne. Equational problems and disunification. *Journal of Symbolic Computation*, 7(3/4):371–425, 1989.

[Col84] Alain Colmerauer. Equations and inequations on finite and infinite trees. In *Proceedings of the International Conference on Fifth Generation Computer Systems, FGCS 1984, Tokyo, Japan, November 6-9, 1984*, pages 85–99. OHMSHA Ltd. Tokyo and North-Holland, 1984.

[Com86] Hubert Comon. Sufficient completness, term rewriting systems and "anti-unification". In Jörg H. Siekmann, editor, *8th International Conference on Automated Deduction, Oxford, England, July 27 - August 1, 1986, Proceedings*, volume 230 of *Lecture Notes in Computer Science*, pages 128–140. Springer, 1986.

[Com91] Hubert Comon. Disunification: A survey. In Jean-Louis Lassez and Gordon D. Plotkin, editors, *Computational Logic - Essays in Honor of Alan Robinson*, pages 322–359. The MIT Press, 1991.

[Fer93] Maribel Fernández. AC complement problems: Satisfiability and negation elimination. In Claude Kirchner, editor, *Rewriting Techniques and Applications, 5th International Conference, RTA-93, Montreal, Canada, June 16-18, 1993, Proceedings*, volume 690 of *Lecture Notes in Computer Science*, pages 358–373. Springer, 1993.

[FG07] Maribel Fernández and Murdoch Gabbay. Nominal rewriting. *Information and Computation*, 205(6):917–965, 2007.

[Gab09] Murdoch James Gabbay. Nominal algebra and the HSP theorem. *Journal of Logic and Computation*, 19(2):341–367, 2009.

[GP02] Murdoch Gabbay and Andrew M. Pitts. A new approach to abstract syntax with variable binding. *Formal Aspects Comput.*, 13(3-5):341–363, 2002.

[KL87] Claude Kirchner and Pierre Lescanne. Solving disequations. In *Proceedings of the Symposium on Logic in Computer Science (LICS '87), Ithaca, New York, USA, June 22-25, 1987*, pages 347–352. IEEE Computer Society, 1987.

[Lug94] Denis Lugiez. Higher order disunification: Some decidable cases. In Jean-Pierre Jouannaud, editor, *Constraints in Computational Logics, First International Conference, CCL'94, Munich, Germany, September 7-9, 1994*, volume 845 of *Lecture Notes in Computer Science*, pages 121–135. Springer, 1994.

[Pit13] Andrew M. Pitts. *Nominal Sets: Names and Symmetry in Computer Science*. Cambridge University Press, 2013.

[UPG04] Christian Urban, Andrew M. Pitts, and Murdoch Gabbay. Nominal unification. *Theoretical Computer Science*, 323(1-3):473–497, 2004.

Apêndice A

Resultados Técnicos

A.1 Resultados Técnicos Auxiliares

Nesta seção apresentamos algumas propriedades sobre composição de permutações e conjugados.

Lema A.1. Dada uma trasposição $(a\ b)$, temos que

(i) $(a\ b)^{\rho} = (\rho(a)\ \rho(b))$, para qualquer permutação $\rho \in \texttt{Perm}(\mathbb{A})$; e

(ii) $\rho \circ (a\ b) = (\rho(a)\ \rho(b)) \circ \rho$, para qualquer permutação $\rho \in \texttt{Perm}(\mathbb{A})$.

Demonstração. (i) De fato, suponha por contradição que exista $d \in \mathbb{A}$ tal que $(a\ b)^{\rho}(d) = d_1$ e $(\rho(a)\ \rho(b))(d) = d_2$, com $d_1 \neq d_2$.

- Se $d_1 = d$, então $d_2 \neq d$, e temos dois casos a anlisar:
  - se $d = \rho(a)$ e $d_2 = \rho(b)$, então $(a\ b)^{\rho}(d) = \rho(a\ b)\rho^{-1}(\rho(a)) = \rho(b)$. Como permutações são bijeções, temos que $d_1 = \rho(b) = d_2$, uma contradição;
  - se $d = \rho(b)$ e $d_2 = \rho(a)$, então $(a\ b)^{\rho}(d) = \rho(a\ b)\rho^{-1}(\rho(b)) = \rho(a)$. E analogamente ao caso anterior, temos que $d_1 = \rho(a) = d_2$, uma contradição.
- Se $d_2 = d$, então $d_1 \neq d$. Nesse caso, $d \notin \{\rho(a), \rho(b)\}$, o que implica que $\rho^{-1}(d) \notin \{a, b\}$. Logo, $(a\ b)^{\rho}(d) = \rho(a\ b)\rho^{-1}(d) = d$, isto é, $d_1 = d$, uma contradição.
- Caso $d_1 \neq d$ e $d_2 \neq d$, então temos outros dois casos:
  - $d = \rho(a)$ e $d_2 = \rho(b)$, e com isso $(a\ b)^{\rho}(d) = \rho(a\ b)\rho^{-1}(\rho(a)) = \rho(b)$, o que implica $d_1 = \rho(b) = d_2$, uma contradição;

– ou $d = \rho(b)$ $d_2 = \rho(a)$, e com isso $(a\ b)^\rho(d) = \rho(a\ b)\rho^{-1}(\rho(b)) = \rho(a)$, o que implica $d_1 = \rho(a) = d_2$, uma contradição.

Em todo caso, chega-se a um absurdo, logo $d_1 = d_2$, e $(a\ b)^\rho = (\rho(a)\ \rho(b))$, como queríamos.

(ii) Note que, usando o item anterior, temos

$$\rho \circ (a\ b) = \rho \circ (a\ b) \circ (\rho^{-1} \circ \rho) = (a\ b)^\rho \circ \rho \overset{\text{(i)}}{=} (\rho(a)\ \rho(b)) \circ \rho,$$

e o resultado segue.

□

Lema A.2. Sejam $\rho, \pi \in \mathtt{Perm}(\mathbb{A})$ e defina

$$\rho \cdot \mathtt{dom}(\pi) := \{\rho(a) \mid a \in \mathtt{dom}(\pi)\}.$$

Então $\mathtt{dom}(\pi^\rho) = \rho \cdot \mathtt{dom}(\pi)$.

Demonstração. Se $a \in \mathtt{dom}(\pi^\rho)$, então $\rho \circ \pi \circ \rho^{-1}(a) = \overline{a}$, para algum $\overline{a} \in \mathtt{dom}(\rho) \setminus \{a\}$. Tome $b = \rho^{-1}(a)$. Como $\rho \circ \pi \circ \rho^{-1}(a) \neq a$, então $\pi(b) \neq b$, isto é $b \in \mathtt{dom}(\pi)$. Note que, $\rho(b) = \rho(\rho^{-1}(a)) = a$, logo $a \in \rho \cdot \mathtt{dom}(\pi)$.

Por outro lado, suponha que $a \in \rho \cdot \mathtt{dom}(\pi)$. Então existe $b \in \mathtt{dom}(\pi)$ tal que $a = \rho(b)$. Note que $\pi^\rho(a) = \rho(\pi(\rho^{-1}(a))) = \rho(\pi(b)) \neq a$, pois $\pi(b) \neq b$ e ρ é uma bijeção. Logo, $a \in \mathtt{dom}(\pi^\rho)$, e concluimos que $\mathtt{dom}(\pi^\rho) = \rho \cdot \mathtt{dom}(\pi)$. □

Lema A.3. Sejam $\rho, \pi \in \mathtt{Perm}(\mathbb{A})$. Então

$$\rho \cdot \mathtt{dom}(\pi) = \big(\mathtt{dom}(\pi) \cup \mathtt{dom}(\rho)\big) \setminus \big\{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\big\}.$$

Demonstração. A prova será feita por análise dos casos de como $\mathtt{dom}(\pi)$ e $\mathtt{dom}(\rho)$ se relacionam:

1. $\mathtt{dom}(\pi) \cap \mathtt{dom}(\rho) = \emptyset$;

   Afirmação: $\rho \cdot \mathtt{dom}(\pi) = \mathtt{dom}(\pi)$.

   Nesse caso, $\rho(a) = a$ para todo $a \in \mathtt{dom}(\pi)$, isto é,

   $$\rho \cdot \mathtt{dom}(\pi) = \{\rho(a) \mid a \in \mathtt{dom}(\pi)\} = \{a \mid a \in \mathtt{dom}(\pi)\} = \mathtt{dom}(\pi),$$

   como foi afirmado.

Note que nesse caso, como $\mathtt{dom}(\rho^{-1}) = \mathtt{dom}(\rho)$ e $\mathtt{dom}(\pi) \cap \mathtt{dom}(\rho) = \emptyset$, temos que

$$\left\{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\right\} = \mathtt{dom}(\rho),$$

logo $\big(\mathtt{dom}(\pi) \cup \mathtt{dom}(\rho)\big) \setminus \left\{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\right\} = \mathtt{dom}(\pi)$, e o resultado segue.

2. $\mathtt{dom}(\pi) = \mathtt{dom}(\rho)$;

   Afirmação: $\rho \cdot \mathtt{dom}(\pi) = \mathtt{dom}(\pi)$.

   De fato, por um lado, $\rho(a) \in \mathtt{dom}(\pi)$ para todo $a \in \mathtt{dom}(\pi)$, isto é, $\rho \cdot \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\pi)$. Por outro lado, se $a \in \mathtt{dom}(\pi)$, então existe $b \in \mathtt{dom}(\rho)$ tal que $a = \rho(b) \in \rho \cdot \mathtt{dom}(\pi)$, ou seja, $\mathtt{dom}(\pi) \subseteq \rho \cdot \mathtt{dom}(\pi)$, e a afirmação é verdadeira.

   Note que nesse caso, temos que $\mathtt{dom}(\pi) \cup \mathtt{dom}(\rho) = \mathtt{dom}(\pi)$ e, como $\mathtt{dom}(\rho^{-1}) = \mathtt{dom}(\rho)$, então $\left\{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\right\} = \emptyset$, e com isso o resultado segue.

3. $\mathtt{dom}(\rho) \subset \mathtt{dom}(\pi)$;

   Afirmação: $\rho \cdot \mathtt{dom}(\pi) = \mathtt{dom}(\pi)$.

   De fato, por um lado, temos que $\rho(a) \in \mathtt{dom}(\pi)$ para todo $a \in \mathtt{dom}(\pi)$, pois

   - se $a \in \mathtt{dom}(\rho)$, temos que $\rho(a) \in \mathtt{dom}(\rho) \subset \mathtt{dom}(\pi)$; e
   - se $a \in \mathtt{dom}(\pi) \setminus \mathtt{dom}(\rho)$, temos que $\rho(a) = a \in \mathtt{dom}(\pi)$,

   logo $\rho \cdot \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\pi)$. Por outro lado,

   - se $a \in \mathtt{dom}(\pi)$ mas $a \notin \mathtt{dom}(\rho)$, então $a = \rho(a) \in \rho \cdot \mathtt{dom}(\pi)$;
   - se $a \in \mathtt{dom}(\rho) \subset \mathtt{dom}(\pi)$, então $\rho(a) \in \rho \cdot \mathtt{dom}(\pi)$,

   logo, $\mathtt{dom}(\pi) \subseteq \rho \cdot \mathtt{dom}(\pi)$, e afirmação segue.

   Note que, como $\mathtt{dom}(\rho) \subset \mathtt{dom}(\pi)$ e $\mathtt{dom}(\rho^{-1}) = \mathtt{dom}(\rho)$, temos que

   $$\mathtt{dom}(\pi) \cup \mathtt{dom}(\rho) = \mathtt{dom}(\pi) \ \text{ e } \ \left\{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\right\} = \emptyset,$$

   e com isso o resultado segue.

4. $\mathtt{dom}(\pi) \subset \mathtt{dom}(\rho)$;

   Afirmação: $\rho \cdot \mathtt{dom}(\pi) = \mathtt{dom}(\rho) \setminus \left\{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\right\}$.

   De fato, seja $b \in \rho \cdot \mathtt{dom}(\pi)$. Então $b = \rho(a)$, para algum $a \in \mathtt{dom}(\pi)$. Como $\mathtt{dom}(\pi) \subset \mathtt{dom}(\rho)$, temos que $b \in \mathtt{dom}(\rho)$. Porém, $\rho^{-1}(b) = \rho^{-1}(\rho(a)) = a \in \mathtt{dom}(\pi)$, ou seja,

$b \notin \{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\}$. Com isso, mostramos que $\rho \cdot \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\rho) \setminus \{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\}$.

Por outro lado, seja $b \in \mathtt{dom}(\rho) \setminus \{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\}$. Então $b \in \mathtt{dom}(\rho)$ e $\rho^{-1}(b) \in \mathtt{dom}(\pi)$. Logo, $\pi(\rho^{-1}(b)) \neq \rho^{-1}(b)$ e, como ρ é bijeção, segue que $\rho(\pi(\rho^{-1}(b))) \neq b$, isto é,

$$b \in \mathtt{dom}(\pi^{\rho}) \overset{\text{Lema A.2}}{=} \rho \cdot \mathtt{dom}(\pi),$$

e a afirmação segue.

Note que nesse caso, temos que $\mathtt{dom}(\pi) \cup \mathtt{dom}(\rho) = \mathtt{dom}(\rho)$ e com isso o resultado segue.

5. $\mathtt{dom}(\rho) \cap \mathtt{dom}(\pi) \neq \emptyset$, de modo que $\mathtt{dom}(\rho) \not\subset \mathtt{dom}(\pi)$ e $\mathtt{dom}(\pi) \not\subset \mathtt{dom}(\rho)$.

   Nesse caso, existem átomos que estão na interseção, átomos que estão apenas em $\mathtt{dom}(\rho)$ e átomos que estão apenas em $\mathtt{dom}(\pi)$.

   Denote $C := \{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\}$. Queremos mostrar que $\rho \cdot \mathtt{dom}(\pi) = \big(\mathtt{dom}(\pi) \cup \mathtt{dom}(\rho)\big) \setminus C$. Para isso, tome $b \in \rho \cdot \mathtt{dom}(\pi)$. Então $b = \rho(a)$ para algum $a \in \mathtt{dom}(\pi)$. Pela configuração dada por hipótese, temos duas possibilidades:

   - $a \in \mathtt{dom}(\pi) \setminus \mathtt{dom}(\rho)$, ou seja, $a \notin \mathtt{dom}(\rho)$, o que implica que $b = \rho(a) = a \in \mathtt{dom}(\pi)$. Além disso, $\rho^{-1}(b) = \rho^{-1}(\rho(a)) = a \in \mathtt{dom}(\pi)$, logo $b \notin C$;
   - ou $a \in \mathtt{dom}(\rho) \cap \mathtt{dom}(\pi)$, isto é, também temos $a \in \mathtt{dom}(\rho)$, o que implica que $b = \rho(a) \in \mathtt{dom}(\rho)$. Além disso, $\rho^{-1}(b) = \rho^{-1}(\rho(a)) = a \in \mathtt{dom}(\pi)$, logo $b \notin C$;

   Portanto, $b \in \big(\mathtt{dom}(\pi) \cup \mathtt{dom}(\rho)\big) \setminus C$.

   Por outro lado, seja $b \in \big(\mathtt{dom}(\pi) \cup \mathtt{dom}(\rho)\big) \setminus \{y \in \mathtt{dom}(\rho) \mid \rho^{-1}(y) \notin \mathtt{dom}(\pi)\}$. Então b é tal que $\rho^{-1}(b) \in \mathtt{dom}(\pi)$. Logo, $\pi(\rho^{-1}(b)) \neq \rho^{-1}(b)$ e, como ρ é bijeção, segue que $\rho(\pi(\rho^{-1}(b))) \neq b$, isto é,

$$b \in \mathtt{dom}(\pi^{\rho}) \overset{\text{Lema A.2}}{=} \rho \cdot \mathtt{dom}(\pi),$$

   e o resultado segue.

□

Lema A.4. Sejam $\pi \in \mathtt{Perm}(\mathbb{A})$ e um átomo c_1 tal que $c_1 \notin \mathtt{dom}(\pi)$. Então

$$\mathtt{dom}(\pi^{(c_1\ b)}) \subseteq \mathtt{dom}(\pi^{(b\ a)(c_1\ a)}) \cup \mathtt{dom}((a\ c_1)),$$

onde $a, b \in \mathbb{A}$ são arbitrários.

Demonstração. Denote $D := \mathtt{dom}(\pi^{(b\ a)(c_1\ a)}) \cup \mathtt{dom}((a\ c_1))$. Pelo Lema A.3,

$$\mathtt{dom}(\pi^{(c_1\ b)}) = \big(\mathtt{dom}(\pi) \cup \{b, c_1\}\big) \setminus \{b\}$$

e no pior dos casos

$$D = \big(\mathtt{dom}(\pi) \setminus \{a, b, c_1\}\big) \cup \{a, c_1\}.$$

Como $b \notin \mathtt{dom}(\pi^{(c_1\ b)})$, segue que $\mathtt{dom}(\pi^{(c_1\ b)}) \subseteq D$, e o resultado vale. □

Lema A.5. Sejam $\pi \in \mathtt{Perm}(\mathbb{A})$ e um átomo c_1 tal que $c_1 \notin \mathtt{dom}(\pi)$. Então

$$\mathtt{dom}(\pi^{(c_1\ a)}) \subseteq \mathtt{dom}((a\ b) \circ \pi) \cup \mathtt{dom}((b\ c_1)),$$

onde $a, b \in \mathbb{A}$ são arbitrários.

Demonstração. Denote $D := \mathtt{dom}((a\ b) \circ \pi) \cup \mathtt{dom}((b\ c_1))$. Pelo Lema A.3,

$$\mathtt{dom}(\pi^{(c_1\ a)}) = \big(\mathtt{dom}(\pi) \cup \{a, c_1\}\big) \setminus \{a\}.$$

Como $c_1 \in D$, o pior dos casos é quando $a, b \in \mathtt{dom}(\pi)$ de tal forma que $\pi(a) = b$ e $\pi(b) = a$, pois nesse caso $\mathtt{dom}((a\ b) \circ \pi) = \mathtt{dom}(\pi) \setminus \{a, b\}$. Porém, $b \in D$ devido a $\mathtt{dom}((b\ c_1)) \subseteq D$, e, de qualquer forma, $a \notin \mathtt{dom}(\pi^{(c_1\ a)})$. Logo, $\mathtt{dom}(\pi^{(c_1\ b)}) \subseteq D$, e o resultado vale □

Lema A.6 (Referente ao Lema 2.1)**.** Substituições e permutações comutam: $\pi \cdot (s\sigma) = (\pi \cdot s)\sigma$.

Demonstração. A prova será por indução sobre a estrutura de s.
(Base da indução)

Esse caso corresponde aos termos cuja estrutura não envolve símbolos de função ou abstrações de átomos.

- $s := a$

$$\pi \cdot (a\sigma) = \pi \cdot (a) = \pi(a) = (\pi(a))\sigma = (\pi \cdot a)\sigma$$

- $s := \pi' \cdot X$

$$\pi \cdot ((\pi' \cdot X)\sigma) = \pi \cdot (\pi' \cdot (X\sigma)) = (\pi \circ \pi') \cdot X\sigma = ((\pi \circ \pi') \cdot X)\sigma = (\pi \cdot (\pi' \cdot X))\sigma$$

(Passo indutivo)

Agora supomos como hipótese de indução (HI) que o resultado vale para termos com estrutura mais simples que s e mostramos que isso implica ser válido para s.

- $s := [a]s'$

$$\begin{aligned}\pi \cdot (([a]s')\sigma) &= \pi \cdot ([a](s'\sigma)) = [\pi(a)]\pi \cdot (s'\sigma) \\ &\overset{(HI)}{=} [\pi(a)](\pi \cdot s')\sigma = ([\pi(a)]\pi \cdot s')\sigma = (\pi \cdot [a]s')\sigma\end{aligned}$$

- $s := \mathtt{f}(s_1, \ldots, s_n)$

$$\begin{aligned}\pi \cdot ((\mathtt{f}(s_1, \ldots, s_n))\sigma) &= \pi \cdot \mathtt{f}(s_1\sigma, \ldots, s_n\sigma) = \mathtt{f}(\pi \cdot (s_1\sigma), \ldots, \pi \cdot (s_n\sigma)) \\ &\overset{(HI)}{=} \mathtt{f}((\pi \cdot s_1)\sigma, \ldots, (\pi \cdot s_n)\sigma) = (\mathtt{f}(\pi \cdot s_1, \ldots, \pi \cdot s_n))\sigma \\ &= (\pi \cdot \mathtt{f}(s_1, \ldots, s_n))\sigma\end{aligned}$$

□

A.2 Provas da Seção 2.3

Nesta seção apresentamos as provas completas dos resultados apresentados na Seção 2.3.

Lema A.7 (Referente ao Lema 2.3)**.**

(i) $\Upsilon \vdash \pi \curlywedge t$ se, e somente se, $\Upsilon \vdash \pi^\rho \curlywedge \rho \cdot t$, para qualquer permutação ρ.

(ii) $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$ se, e somente se, $\Upsilon \vdash \pi \cdot s \overset{\curlywedge}{\approx}_\alpha \pi \cdot t$, para qualquer permutação π.

Demonstração. (i) Por um lado, se $\Upsilon \vdash \pi^\rho \curlywedge \rho \cdot t$ para qualquer permutação ρ, então, em particular, vale para $\rho = \mathtt{Id}$, logo $\Upsilon \vdash \pi \curlywedge t$. Por outro lado, suponha que $\Upsilon \vdash \pi \curlywedge t$. Essa prova é por indução nas regras da Tabela 2.1, e consideramos os casos a partir da última regra aplicada na derivação.

- A última regra aplicada é $(\curlywedge\mathbf{a})$.
  Nesse caso, $t = a$ e $\pi(a) = a$. Com isso, temos que

$$\pi^\rho(\rho(a)) = \rho \circ \pi \circ \rho^{-1}(\rho(a)) = \rho(\pi(a)) = \rho(a).$$

  Isto é, $\rho(a)$ é ponto fixo de π^ρ. Logo, podemos usar a regra $(\curlywedge\mathbf{a})$ para concluir que $\Upsilon \vdash \pi^\rho \curlywedge \rho(a)$, e o resultado segue.

- A última regra aplicada é ($\curlywedge$**var**).

  Nesse caso, $t = \pi' \cdot X$, e $\mathtt{dom}(\pi^{(\pi')^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Queremos mostrar que $\Upsilon \vdash \pi^{\rho} \curlywedge (\rho \circ \pi') \cdot X$. Note que,

$$\begin{aligned}(\pi^{\rho})^{(\rho\circ\pi')^{-1}} &= (\rho \circ \pi')^{-1} \circ (\rho \circ \pi \circ \rho^{-1}) \circ (\rho \circ \pi') \\ &= ((\pi')^{-1} \circ \rho^{-1}) \circ (\rho \circ \pi \circ \rho^{-1}) \circ (\rho \circ \pi') \\ &= (\pi')^{-1} \circ \pi \circ \pi' \\ &= \pi^{(\pi')^{-1}}.\end{aligned}$$

  Logo, $\mathtt{dom}((\pi^{\rho})^{(\rho\circ\pi')^{-1}}) = \mathtt{dom}(\pi^{(\pi')^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ e agora basta usar a regra ($\curlywedge$**var**) e concluir o que queríamos.

- A última regra aplicada é ($\curlywedge\mathtt{f}$).

  Nesse caso, $t = \mathtt{f}(t_1, \ldots, t_n)$, e a hipótese de indução pressupõe que existem n provas $\Pi_1, \ldots, \Pi_n$ tais que, para cada $1 \leq i \leq n$ e para qualquer substituição ρ,

$$\frac{\Pi_i}{\Upsilon \vdash \pi^{\rho} \curlywedge \rho \cdot t_i}$$

  Usando a regra ($\curlywedge\mathtt{f}$), temos que $\Upsilon \vdash \pi^{\rho} \curlywedge \mathtt{f}(\rho \cdot t_1, \ldots, \rho \cdot t_n)$, ou seja, $\Upsilon \vdash \pi^{\rho} \curlywedge \rho \cdot \mathtt{f}(t_1, \ldots, t_n)$, e o resultado segue.

- A última regra aplicada é ($\curlywedge$**abs**).

  Nesse caso, $t = [a]t'$ e temos que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1) \cdot t'$, onde c_1 é um nome novo. Pela hipótese de indução, existe uma prova Π tal que, para qualquer substituição ρ,

$$\frac{\Pi}{\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(\rho \cdot t')} \vdash \pi^{\rho} \curlywedge \rho \cdot ((a\ c_1) \cdot t')}$$

  Note que pelo Lema A.1, temos que

$$\rho \circ (a\ c_1) = (\rho(a)\ \rho(c_1)) \circ \rho.$$

  Como $\rho(c_1) = c_1$, pois c_1 é novo, então

$$\frac{\dfrac{\Pi}{\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(\rho \cdot t')} \vdash \pi^{\rho} \curlywedge (\rho(a)\ c_1) \cdot (\rho \cdot t')}}{\Upsilon \vdash \pi^{\rho} \curlywedge [\rho(a)]\rho \cdot t'}\ (\curlywedge\mathbf{abs})$$

e com isso o resultado segue.

(ii) A prova desse item é análogo ao item anterior, e abaixo apresentamos um caso da indução nas regras da Tabela 2.2.

- A última regra aplicada é ($\overset{\curlywedge}{\approx}_\alpha$ **ab**).

  Nesse caso, $s = [a]s'$ e $t = [b]t'$, e temos que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} \vdash (a\ c_1) \curlywedge t$ e $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha (a\ b) \cdot t$, onde c_1 e c_2 são nomes novos. Note que, usando o Lema A.1 novamente, temos que

$$\rho \circ (a\ b) = (\rho(a)\ \rho(b)) \circ \rho \quad \text{e} \quad (a\ c_1)^\rho = (\rho(a)\ c_1)$$

  Com isso e pela hipótese de indução, existem provas Π_1 e Π_2 tais que

$$\dfrac{\dfrac{\dfrac{\Pi_1}{\Upsilon \vdash \rho \cdot s' \overset{\curlywedge}{\approx}_\alpha \rho \cdot ((a\ b) \cdot t')}}{\Upsilon \vdash \rho \cdot s' \overset{\curlywedge}{\approx}_\alpha (\rho(a)\ \rho(b)) \cdot (\rho \cdot t')} \qquad \dfrac{\dfrac{\Pi_2}{\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(\rho \cdot t')} \vdash (a\ c_1)^\rho \curlywedge \rho \cdot t'}}{\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(\rho \cdot t')} \vdash (\rho(a)\ c_1) \curlywedge \rho \cdot t'}}{\Upsilon \vdash [\rho(a)]\rho \cdot s' \overset{\curlywedge}{\approx}_\alpha [\rho(b)]\rho \cdot t'}\ (\overset{\curlywedge}{\approx}_\alpha \mathbf{ab})$$

  Logo, o resultado segue.

□

O seguinte lema é um resultado auxiliar que nos permite derivar uma restrição de ponto fixo composta por um termo t e uma permutação π cujo suporte está contido no suporte de outras permutações que fixam t. Por exemplo, dado que $\Upsilon \vdash (a\ b) \curlywedge \mathtt{g}(e, X)$ e $\Upsilon \vdash (c\ d) \curlywedge \mathtt{g}(e, X)$, podemos deduzir $\Upsilon \vdash (a\ c) \curlywedge \mathtt{g}(e, X)$.

Lema A.8. Seja I um conjunto de índices. Se $\Upsilon \vdash \pi_i \curlywedge t$, para todo $i \in I$, e π é uma permutação tal que $\mathtt{dom}(\pi) \subseteq \bigcup_{i \in I} \mathtt{dom}(\pi_i)$, então $\Upsilon \vdash \pi \curlywedge t$.

Demonstração. A prova é por indução na estrutura de t. Abaixo, fazemos os casos da base, e o passo indutivo segue diretamente da hipótese de indução e será omitido.

- $t = a$

  Nesse caso, $\Upsilon \vdash \pi_i \curlywedge a$ e $\pi_i(a) = a$, para todo $i \in I$, isto é, $a \notin \cup_{i \in I} \mathtt{dom}(\pi_i)$. Como por hipótese $\mathtt{dom}(\pi) \subseteq \cup_{i \in I} \mathtt{dom}(\pi_i)$, segue que $\pi(a) = a$, e o resultado segue usando a regra ($\curlywedge$**a**).

- $t = \rho \cdot X$

Nesse caso, para todo $i \in I$, vale $\Upsilon \vdash \pi_i \curlywedge \rho \cdot X$ e $\mathtt{dom}({\pi_i}^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$, logo

$$\cup_{i \in I} \mathtt{dom}({\pi_i}^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)).$$

Como por hipótese $\mathtt{dom}(\pi) \in \cup_{i \in I} \mathtt{dom}(\pi_i)$, se $a \in \mathtt{dom}(\pi^{\rho^{-1}})$ então $\rho^{-1} \circ \pi \circ \rho(a) \neq a$, o que implica que $\rho(a) \in \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\pi_j)$, para algum $j \in I$.

Assim, $a \in \mathtt{dom}(\pi^{\rho^{-1}})$ implica que $\rho^{-1} \circ \pi_j \circ \rho(a) \neq a$, isto é, $a \in \mathtt{dom}({\pi_j}^{\rho^{-1}})$, para algum $j \in I$. Com isso, $\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}({\pi_j}^{\rho^{-1}}) \subseteq \cup_{i \in I} \mathtt{dom}({\pi_i}^{\rho^{-1}})$, e o resultado segue com uso da regra ($\curlywedge$**var**).

□

Proposição A.1 (Referente ao item (i) da Proposição 2.1)**.** Se $\Upsilon, \pi \curlywedge X \vdash \pi' \curlywedge s$ e $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ ou $X \notin \mathtt{Var}(s)$, então $\Upsilon \vdash \pi' \curlywedge s$.

Demonstração. A prova é por indução na estrutura de s.

- $s = a$

  Nesse caso, temos que $\Upsilon, \pi \curlywedge X \vdash \pi' \curlywedge a$, o que, pela inversão da regra ($\curlywedge$**a**), implica que $\pi'(a) = a$. Com isso, podemos derivar $\Upsilon, \pi \curlywedge X \vdash \pi' \curlywedge a$, usando a mesma regra, e o resulado segue.

- $s = \rho \cdot X$

  Nesse caso, $\Upsilon, \pi \curlywedge X \vdash \pi' \curlywedge \rho \cdot X$ e a inversão da regra ($\curlywedge$**var**) garante que

  $$\mathtt{dom}((\pi')^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \mathtt{dom}(\pi).$$

  Mas, como $X \in \mathtt{Var}(s)$, temos por hipótese que $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$, logo $\mathtt{dom}((\pi')^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ e, usando a regra ($\curlywedge$**var**), segue que $\Upsilon \vdash \pi' \curlywedge \rho \cdot X$.

  No caso em que $s = \rho \cdot Y$, com $Y \neq X$, temos da inversão de ($\curlywedge$**var**) que $\mathtt{dom}((\pi')^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_Y))$, e o resultado segue diretamente com o uso da regra ($\curlywedge$**var**).

- $s = \mathtt{f}(s_1, \ldots, s_n)$

  Nesse caso, $\Upsilon, \pi \curlywedge X \vdash \pi' \curlywedge \mathtt{f}(s_1, \ldots, s_n)$ e, pela inversão da regra ($\curlywedge\mathtt{f}$), temos que $\Upsilon, \pi \curlywedge X \vdash \pi' \curlywedge s_i$, para cada $1 \leq i \leq n$. Pela hipótese indutiva, segue que $\Upsilon \vdash \pi' \curlywedge s_i$, para cada $1 \leq i \leq n$, e o resultado é alcançado aplicando a regra ($\curlywedge\mathtt{f}$).

- $s = [a]s'$

Nesse caso, $\Upsilon, \pi \curlywedge X \vdash \pi' \curlywedge [a]s'$ e, pela inversão da regra ($\curlywedge$**abs**), temos que

$$\Upsilon, \pi \curlywedge X, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s')} \vdash \pi' \curlywedge (a\ c_1) \cdot s',$$

onde c_1 e c_2 são nomes novos. Pela hipótese indutiva, segue que

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s')} \vdash \pi' \curlywedge (a\ c_1) \cdot s',$$

e o resultado é alcançado aplicando a regra ($\curlywedge$**abs**).

□

Proposição A.2 (Referente ao item (ii) da Proposição 2.1)**.** Se $\Upsilon, \pi \curlywedge X \vdash s \overset{\curlywedge}{\approx}_\alpha t$ e $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ ou $X \notin \mathtt{Var}(s,t)$, então $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$.

Demonstração. Essa prova também é por indução na estrutura de s, que por sua vez induz a estrutura de t.

- $s = a$

  Nesse caso, temos que $\Upsilon, \pi \curlywedge X \vdash a \overset{\curlywedge}{\approx}_\alpha t$ e força-se $t = a$. Com isso, podemos derivar $\Upsilon, \pi \curlywedge X \vdash a \overset{\curlywedge}{\approx}_\alpha a$ usando a regra ($\overset{\curlywedge}{\approx}_\alpha$ **a**), e o resulado segue.

- $s = \rho_1 \cdot X$

  Nesse caso, $\Upsilon, \pi \curlywedge X \vdash \rho_1 \cdot X \overset{\curlywedge}{\approx}_\alpha t$ e força-se $t = \rho_2 \cdot X$. Logo, a inversão da regra ($\overset{\curlywedge}{\approx}_\alpha$ **var**) garante que

  $$\mathtt{dom}({\rho_2}^{-1} \circ \rho_1) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \mathtt{dom}(\pi).$$

  Mas, como $X \in \mathtt{Var}(s)$, temos por hipótese que $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$, logo $\mathtt{dom}({\rho_2}^{-1} \circ \rho_1) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ e, usando a regra ($\overset{\curlywedge}{\approx}_\alpha$ **var**), segue que $\Upsilon, \pi \curlywedge X \vdash \rho_1 \cdot X \overset{\curlywedge}{\approx}_\alpha \rho_2 \cdot X$.

  No caso em que $s = \rho_1 \cdot Y$, com $Y \neq X$, força-se $t = \rho_2 \cdot Y$ e temos da inversão de ($\curlywedge$**var**) que $\mathtt{dom}({\rho_2}^{-1} \circ \rho_1) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_Y))$, e o resultado segue diretamente com o uso da mesma regra.

- $s = \mathsf{f}(s_1, \ldots, s_n)$

  Nesse caso, $\Upsilon, \pi \curlywedge X \vdash \mathsf{f}(s_1, \ldots, s_n) \overset{\curlywedge}{\approx}_\alpha t$ e força-se $t = \mathsf{f}(t_1, \ldots, t_n)$. Pela inversão da regra ($\overset{\curlywedge}{\approx}_\alpha \mathsf{f}$), temos que $\Upsilon, \pi \curlywedge X \vdash s_i \curlywedge t_i$, para cada $1 \leq i \leq n$. Pela hipótese indutiva, segue que $\Upsilon \vdash s_i \curlywedge t_i$, para cada $1 \leq i \leq n$, e o resultado é alcançado aplicando a regra ($\curlywedge\mathsf{f}$).

- $s = [a]s'$

  Nesse caso, $\Upsilon, \pi \curlywedge X \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha t$ e temos duas situações induzidas à estrutura de t:

  - $t = [a]t'$

    Com isso, temos que $\Upsilon, \pi \curlywedge X \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha [a]t'$, e o resultado segue diretamente da inversão da regra $(\overset{\curlywedge}{\approx}_\alpha [\mathbf{a}])$, seguida da hipótese indutiva.

  - $t = [b]t'$

    Assim, pela inversão da regra $(\overset{\curlywedge}{\approx}_\alpha \mathbf{ab})$, temos que $\Upsilon, \pi \curlywedge X \vdash s' \overset{\curlywedge}{\approx}_\alpha (a\ b) \cdot t'$ e $\Upsilon,\ \pi \curlywedge X,\ \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (a\ c_1) \curlywedge t'$, onde c_1 e c_2 são nomes novos. Pela hipótese de indução, existem derivações Π_1 e Π_2 tais que

    $$\dfrac{\dfrac{\Pi_1}{\Upsilon \vdash s' \overset{\curlywedge}{\approx}_\alpha (a\ b) \cdot t'} \qquad \dfrac{\Pi_2}{\Upsilon,\ \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (a\ c_1) \curlywedge t'}}{\Upsilon \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha [b]t'}\ (\overset{\curlywedge}{\approx}_\alpha \mathbf{ab})$$

    E o resultado segue.

□

Lema A.9 (Referente ao Lema 2.4)**.** Se $\Upsilon \vdash \pi \curlywedge s$ e $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$, então $\Upsilon \vdash \pi \curlywedge t$.

Demonstração. Prova por indução na estrutura de s, que induz uma estrutura a t.
(Base da indução)

- $s = a$

  Nesse caso, $\Upsilon \vdash a \overset{\curlywedge}{\approx}_\alpha t$ e a última regra aplicável na derivação desse último sequente é $(\overset{\curlywedge}{\approx}_\alpha \mathbf{a})$, o que obriga $t = a$. Como $\Upsilon \vdash \pi \curlywedge a$, temos que $\pi(a) = a$, e o resultado segue usando a regra $(\curlywedge\mathbf{a})$.

- $s = \rho_1 \cdot X$

  Nesse caso, $\Upsilon \vdash \rho_1 \cdot X \overset{\curlywedge}{\approx}_\alpha t$ e como no caso anterior, força-se $t = \rho_2 \cdot X$. Com isso, temos $\Upsilon \vdash \rho_1 \cdot X \overset{\curlywedge}{\approx}_\alpha \rho_2 \cdot X$, e $\mathtt{dom}({\rho_2}^{-1} \circ \rho_1) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Além disso, $\mathtt{dom}(\pi^{{\rho_1}^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$, pois $\Upsilon \vdash \pi \curlywedge \rho_1 \cdot X$, e logo

  $$\underbrace{\mathtt{dom}({\rho_2}^{-1} \circ \rho_1) \cup \mathtt{dom}(\pi^{{\rho_1}^{-1}})}_{S} \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)).$$

  Queremos mostrar que $\Upsilon \vdash \pi \curlywedge \rho_2 \cdot X$, e para isso afirmamos que $\mathtt{dom}(\pi^{{\rho_2}^{-1}}) \subseteq S$. De fato, suponha por contradição que $\mathtt{dom}(\pi^{{\rho_2}^{-1}}) \not\subseteq S$. Nesse caso, existe $a \in \mathtt{dom}(\pi^{{\rho_2}^{-1}})$ tal que $a \notin S$. Então

1. ${\rho_2}^{-1} \circ \rho_1(a) = a$, o que implica $\rho_1(a) = \rho_2(a)$, e
2. ${\pi^{\rho_1}}^{-1}(a) = a$, o que implica

$$\begin{aligned}
{\rho_1}^{-1} \circ \pi \circ \rho_1(a) = a \;&\Rightarrow\; \pi(\rho_1(a)) = \rho_1(a) \\
&\Rightarrow\; \pi(\rho_2(a)) = \rho_2(a) \\
&\Rightarrow\; {\rho_2}^{-1} \circ \pi \circ \rho_2(a) = a \\
&\Rightarrow\; {\pi^{\rho_2}}^{-1}(a) = a \\
&\Rightarrow\; a \notin \mathtt{dom}({\pi^{\rho_2}}^{-1}),\ \text{uma contradição.}
\end{aligned}$$

Logo a afirmação é verdadeira, e concluimos que $\mathtt{dom}({\pi^{\rho_2}}^{-1}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. O resultado segue usando a regra ($\curlywedge$**var**).

(Passo indutivo)

- $s = \mathsf{f}(s_1, \ldots, s_n)$

  Esse caso segue diretamente da hipótese de indução, e sua prova será omitida.

- $s = [a]s'$

  Nesse caso, $\Upsilon \vdash \pi \curlywedge [a]s'$, o que fornece

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s')} \vdash \pi \curlywedge (a\ c_1) \cdot s', \tag{A.1}$$

  onde c_1 e c_2 são nomes novos. Como por hipótese $\Upsilon \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha t$, duas situações são possíveis:

  1. a última regra aplicada foi ($\overset{\curlywedge}{\approx}_\alpha$ [**a**]) e força-se $t = [a]t'$;
     Com isso $\Upsilon \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha [a]t'$ e, pela invertibilidade, $\Upsilon \vdash s' \overset{\curlywedge}{\approx}_\alpha t'$.

     Observação A.1. Se $\Upsilon' \vdash u \overset{\curlywedge}{\approx}_\alpha v$, então, por uma simples indução nessa derivação, temos que $\mathtt{Var}(u) = \mathtt{Var}(v)$, para qualqer contexto Υ' e quaisquer $u, v \in T(\Sigma, \mathbb{X}, \mathbb{A})$.

     Pela propriedade de enfraquecimento (Prop. 2.2), $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash s' \overset{\curlywedge}{\approx}_\alpha t'$, logo podemos usar o sequente em (A.1), a observação acima – para trocar $\mathtt{Var}(s')$ por $\mathtt{Var}(t')$ – e a hipótese de indução para concluir que

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1) \cdot t',$$

e usando a regra ($\curlywedge$**abs**) segue que $\Upsilon \vdash \pi \curlywedge [a]t'$.

2. a última regra aplicada foi ($\overset{\curlywedge}{\approx}_\alpha$ **ab**) e força-se $t = [b]t'$;
   Com isso $\Upsilon \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha [b]t'$ e, pela invertibilidade, temos

$$\Upsilon \vdash s' \overset{\curlywedge}{\approx}_\alpha (a\ b)\cdot t' \tag{A.2}$$

$$\text{e } \Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (a\ c_1) \curlywedge t'. \tag{A.3}$$

Queremos mostrar que $\Upsilon \vdash \pi \curlywedge [b]t'$, mas para isso, precisamos primeiramente mostrar que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (b\ c_1)\cdot t'$.
De (A.2) e pela propriedade de enfraquecimento, temos

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash s' \overset{\curlywedge}{\approx}_\alpha (a\ b)\cdot t'.$$

Note que $\mathtt{Var}(t') = \mathtt{Var}((a\ b)\cdot t')$, pois $(a\ b)\cdot t'$ é apenas uma abreviação e não interfere no conjunto de variáveis de t'. Com isso, podemos usar o sequente em (A.1), a obs. A.1 – para trocar $\mathtt{Var}(s')$ por $\mathtt{Var}(t')$ – e a hipótese de indução para obter $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1)\cdot((a\ b)\cdot t')$, ou equivalentemente por equivariância (Lema 2.3),

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi^{(b\ a)(c_1\ a)} \curlywedge t'. \tag{A.4}$$

Observe que, pelo Lema A.4, temos

$$\mathtt{dom}(\pi^{(c_1\ b)}) \subseteq \mathtt{dom}(\pi^{(b\ a)(c_1\ a)}) \cup \mathtt{dom}((a\ c_1)),$$

e com (A.3) e (A.4) podemos usar o Lema A.8 para concluir que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi^{(c_1\ b)} \curlywedge t'$ o que equivale a $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (b\ c_1)\cdot t'$, como queríamos.
O resultado segue ao aplicar a regra ($\curlywedge$**abs**).

□

Lema A.10 (Referente ao Lema 2.5)**.** Suponha que c_1 e c_2 sejam átomos que não ocorrem em Υ, π, s e t. Assim,

(i) $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge t$ se, e somente se, $\Upsilon \vdash \pi \curlywedge t$; e

(ii) $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s,t)} \vdash s \overset{\curlywedge}{\approx}_\alpha t$ se, e somente se, $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$.

Demonstração. A volta de ambos os itens segue diretamente da propriedade de Enfraquecimento (Proposição 2.2). Abaixo, fazemos a prova da ida.

(i) A prova será por indução na derivação de $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge t$.

- A última regra aplicada foi ($\curlywedge$**a**)
  Nesse caso, $t = a$ e com isso $\overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} = \emptyset$. Logo, na verdade, temos $\Upsilon \vdash \pi \curlywedge a$ e o resultado segue.
- A última regra aplicada foi ($\curlywedge$**var**)
  Nesse caso, $\Upsilon, (c_1\ c_2) \curlywedge X \vdash \pi \curlywedge \rho \cdot X$ e

$$\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{c_1, c_2\}.$$

  Como c_1 e c_2 são novos, $\mathtt{dom}(\pi^{\rho^{-1}}) \cap \{c_1, c_2\} = \emptyset$. Com isso,

$$\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)),$$

  e usando a regra ($\curlywedge$**var**) obtemos $\Upsilon \vdash \pi \curlywedge \rho \cdot X$.
- A última regra aplicada foi ($\curlywedge\mathtt{f}$)
  Nesse caso, $\Upsilon, \overline{(c_1\ c_2) \curlywedge \cup_{j=1}^{n} \mathtt{Var}(t_j)} \vdash \pi \curlywedge \mathtt{f}(t_1, \ldots, t_n)$ e, pela invertibilidade da regra, temos que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \cup_{j=1}^{n} \mathtt{Var}(t_j)} \vdash \pi \curlywedge t_i$, para todo $1 \leq i \leq n$. Mas pela propriedade de fortalecimento (Proposição 2.1), temos apenas que

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t_i)} \vdash \pi \curlywedge t_i, \text{ para todo } 1 \leq i \leq n.$$

  Assim, pela hipótese de indução, $\Upsilon \vdash \pi \curlywedge t_i$, para todo $1 \leq i \leq n$, e o resultado segue usando a regra ($\curlywedge\mathtt{f}$).
- A última regra aplicada foi ($\curlywedge$**abs**)
  Nesse caso, $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge [a]t'$ e, pela invertibilidade da regra, temos que $\Upsilon, \overline{(c_1\ c_2), (c_1'\ c_2') \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1') \cdot t'$. Pela hipótese de indução $\Upsilon, \overline{(c_1'\ c_2') \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ c_1') \cdot t'$ e o resultado segue usando a regra ($\curlywedge$**abs**).

(ii) A prova também é por indução, mas, dessa vez, na derivação de $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s,t)} \vdash s \overset{\curlywedge}{\approx}_\alpha t$, e o resultado segue com raciocínio inteiramente análogo ao feito no item anterior.

□

Proposição A.3 (Referente à Proposição 2.3)**.** Sejam Υ e Υ' contextos e suponha que $\Upsilon \vdash \Upsilon'\sigma$. Então,

(i) se $\Upsilon' \vdash \pi \curlywedge s$, então $\Upsilon \vdash \pi \curlywedge s\sigma$; e

(ii) se $\Upsilon' \vdash s \overset{\curlywedge}{\approx}_\alpha t$, então $\Upsilon \vdash s\sigma \overset{\curlywedge}{\approx}_\alpha t\sigma$.

Demonstração. A prova de ambos os itens é por indução nas regras de derivação. Abaixo, mostramos os casos mais interessantes.

(i) A última regra aplicada foi

- ($\curlywedge$**var**)
  Nesse caso, $s = \rho \cdot X$ e temos $\Upsilon' \vdash \pi \overset{\curlywedge}{\approx}_\alpha \rho \cdot X$, o que implica pela invertibilidade que
  $$\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon'|_X)) = \bigcup_{\pi' \in \mathtt{perm}(\Upsilon'|_X)} \mathtt{dom}(\pi').$$
  Por hipótese, $\Upsilon \vdash \Upsilon'\sigma$, isto é, $\Upsilon \vdash \pi' \curlywedge X\sigma$, para todo $\pi' \curlywedge X \in \Upsilon'$. Logo, pelo Lema A.8 e por Equivariância, temos que $\Upsilon \vdash \pi^{\rho^{-1}} \curlywedge X$, e por Equivariância o resultado segue.
- ($\curlywedge$**abs**)
  Nesse caso, $s = [a]s'$ e temos que
  $$\dfrac{\begin{array}{c}\vdots\\ \Upsilon', \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s')} \vdash \pi \curlywedge (a\ c_1) \cdot s'\end{array}}{\Upsilon' \vdash \pi \curlywedge [a]s'}$$
  Por hipótese $\Upsilon \vdash \Upsilon'\sigma$ e portanto,
  $$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s'\sigma)} \vdash \Upsilon'\sigma, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s'\sigma)}.$$
  Assim, pela hipótese de indução e o fato de permutação comutar com substituição, temos $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(s'\sigma)} \vdash \pi \curlywedge (a\ c_1) \cdot (s'\sigma)$, e o resultado segue usando a regra ($\curlywedge$**abs**).

(ii) A última regra aplicada foi

- ($\overset{\curlywedge}{\approx}_\alpha$ **var**)
  Nesse caso, temos que $\Upsilon' \vdash \rho_1 \cdot X \overset{\curlywedge}{\approx}_\alpha \rho_2 \cdot X$ e, pela invertibilidade da regra,
  $$\mathtt{dom}({\rho_2}^{-1} \circ \rho_1) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon'|_X)) = \bigcup_{\pi' \in \mathtt{perm}(\Upsilon'|_X)} \mathtt{dom}(\pi').$$
  Por hipótese, $\Upsilon \vdash \Upsilon'\sigma$, isto é, $\Upsilon \vdash \pi' \curlywedge X\sigma$, para todo $\pi' \curlywedge X \in \Upsilon'$. Logo, pelo Lema A.8 temos que $\Upsilon \vdash {\rho_2}^{-1} \circ \rho_1 \curlywedge X\sigma$, que por usa vez é equivalente a $\Upsilon \vdash$

$(\rho_2{}^{-1} \circ \rho_1) \cdot X\sigma \overset{\curlywedge}{\approx}_\alpha X\sigma$, pelo Teorema da Correção. O resultado segue por equivariância.

- $(\overset{\curlywedge}{\approx}_\alpha \mathbf{ab})$

  Nesse caso, $s = [a]s'$, $t = [b]t'$ e temos que

$$\frac{\begin{array}{c}\vdots\\ \Upsilon' \vdash s' \overset{\curlywedge}{\approx}_\alpha (a\ b) \cdot t'\end{array} \qquad \begin{array}{c}\vdots\\ \Upsilon', \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash (a\ c_1) \curlywedge t'\end{array}}{\Upsilon' \vdash [a]s' \overset{\curlywedge}{\approx}_\alpha [b]t'}$$

  Por hipótese $\Upsilon \vdash \Upsilon'\sigma$ e portanto,

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t'\sigma)} \vdash \Upsilon'\sigma, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t'\sigma)}.$$

  Assim, pela hipótese de indução e o fato de permutação comutar com substituição, temos $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t'\sigma)} \vdash \pi \curlywedge (a\ c_1) \cdot (t'\sigma)$. Mas, também pela hipótese de indução, temos que $\Upsilon \vdash s'\sigma \overset{\curlywedge}{\approx}_\alpha (a\ b) \cdot (t'\sigma)$ e o resultado segue usando a regra $(\overset{\curlywedge}{\approx}_\alpha \mathbf{ab})$.

□

A.3 Provas da Seção 2.4

Nesta seção apresentamos as provas detalhadas dos resultados referentes ao suporte de um termo nominal.

Exemplo A.1. Seja $\Upsilon = \{(d_1\ d_2) \curlywedge X, (d_1\ d_3) \curlywedge X\}$ e $P \subset \mathbb{A} \setminus \underbrace{\{d_1, d_2, d_3\}}_{\mathtt{dom}(\mathtt{perm}(\Upsilon|_X))}$ finito. Vamos analisar o suporte de $\Upsilon \vdash \rho \cdot X$ para quando $\rho = (b_1\ b_2)$, ou $\rho = (d_1\ b_2)$ ou $\rho = (d_3\ b_3)$ onde $b_1, b_2, b_3 \in P$:

- $\Upsilon \vdash (b_1\ b_2) \cdot X$ (esse exemplo ilustra o item (ii) do Lema A.11):

  Observe que para toda permutação π tal que $\mathtt{dom}(\pi) \subseteq \{d_1, d_2, d_3\}$, temos que $\pi(a) = a$, para todo $a \in P$. Além disso, $\Upsilon \vdash \pi \curlywedge (b_2\ b_1) \cdot X$, pois $\pi^{(b_1\ b_2)} = \pi$ e, portanto, $\mathtt{dom}(\pi^{(b_2\ b_1)}) \subseteq \{d_1, d_2, d_3\}$. Logo, o menor tal conjunto P é canditado a suporte de $\Upsilon \vdash (b_1\ b_2) \cdot X$.

- $\Upsilon \vdash (d_1\ b_2) \cdot X$ (esse exemplo ilustra o item (iii) do Lema A.11):

Note que $(d_1\ d_2)(a) = a$, para todo $a \in P$. Mas, $\Upsilon \nvdash (d_1\ d_2) \curlywedge (d_1\ b_2) \cdot X$, já que $(d_1\ d_2)^{(b_2\ d_1)} = (b_2\ d_2)$ e, $\mathtt{dom}((d_1\ d_2)^{(b_2\ d_1)}) = \{b_2, d_2\} \not\subseteq \{d_1, d_2, d_3\}$, logo P não pode ser o suporte.

Agora, seja $Q \subset \mathbb{A} \setminus \{d_2, d_3\} = (\mathbb{A} \setminus \underbrace{\{d_1, d_2, d_3\}}_{\mathtt{dom}(\pi)}) \cup \underbrace{\{d_1, b_2\}}_{\mathtt{dom}(\rho)}$ finito:

Note que $\pi = (d_2\ d_3)$ que fixa todo $a \in Q$ e vale $\Upsilon \vdash (d_2\ d_3) \curlywedge (d_1\ b_2) \cdot X$, já que $\mathtt{dom}(\pi^{(d_1\ b_2)}) = \mathtt{dom}(\pi) = \{d_2, d_3\} \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Então $\mathtt{supp}_\Upsilon((d_1\ b_2) \cdot X) \subseteq Q$. Note também que $\mathtt{dom}(\pi) \cap Q = \emptyset$.

- $\Upsilon \vdash (d_3\ b_3) \cdot X$ (esse exemplo ilustra o item (iii) do Lema A.11):

  Considere um subconjunto $P' \subset \mathbb{A} \setminus \{d_1, d_2\} = (\mathbb{A} \setminus \underbrace{\{d_1, d_2, d_3\}}_{\mathtt{dom}(\pi)}) \cup \underbrace{\{d_3, b_3\}}_{\mathtt{dom}(\rho)}$ finito.

  Note que $\pi = (d_1\ d_2)$ é tal que $\pi(a) = a$, para todo $a \in P'$. Além disso, temos que $\Upsilon \vdash \pi \curlywedge (d_3\ b_3) \cdot X$, pois $\mathtt{dom}((d_1\ d_2)^{(d_3\ b_3)}) = \{d_1, d_2\} \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

  Logo, $\mathtt{supp}_\Upsilon((d_3\ b_3) \cdot X) \subseteq P'$.

Lema A.11. (i) Se $\rho = \mathtt{Id}$, então $\mathtt{supp}_\Upsilon(\rho \cdot X) \subset \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

(ii) Se $\rho \neq \mathtt{Id}$ e $\mathtt{dom}(\rho) \cap \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) = \emptyset$, então $\mathtt{supp}_\Upsilon(\rho \cdot X) \subset \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

(iii) Sejam

- $A = \mathtt{dom}(\rho) \cap \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$;
- $B = \mathtt{dom}(\rho) \cap \big(\mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))\big)$;
- $C = \{x \mid x \in A \text{ e } \rho^{-1}(x) \in \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))\}$;
- $D = \{x \mid x \in B \text{ e } \rho^{-1}(x) \in \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))\}$.

Se $\rho \neq \mathtt{Id}$ e $A \neq \emptyset$, então

$$\mathtt{supp}_\Upsilon(\rho \cdot X) \subset \begin{cases} \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) & \text{, se } B = \emptyset; \\ \big(\mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D)\big) \cup C & \text{, caso contrário.} \end{cases}$$

Demonstração. (i) Suponha que $\rho = \mathtt{Id}$.

Seja π tal que $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Então, $\pi(a) = a$ para todo $a \in \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$ e além disso, $\Upsilon \vdash \pi \curlywedge \rho \cdot X$, já que $\mathtt{dom}(\pi^{\rho^{-1}}) = \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

Logo, $\mathtt{supp}_\Upsilon(\rho \cdot X) \subset \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

(ii) $\rho \neq \mathtt{Id}$ e $\mathtt{dom}(\rho) \cap \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) = \emptyset$:

Nesse caso, temos que $\mathtt{dom}(\rho) \subset \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Note que para todo π tal que $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$, valem:

- $\pi(a) = a$, para todo $a \in \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$; e
- $\mathtt{dom}(\pi^{\rho^{-1}}) = \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

Logo, $\mathtt{supp}_\Upsilon(\rho \cdot X) \subset \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

(iii) $\rho \neq \mathtt{Id}$ e $A = \mathtt{dom}(\rho) \cap \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \neq \emptyset$:

- $B = \mathtt{dom}(\rho) \cap (\mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))) = \emptyset$

  Nesse caso, $\mathtt{dom}(\rho) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

  Seja π tal que $\mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Note que $\pi(a) = a$, para todo $a \in \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Além disso, $\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$, o que implica $\Upsilon \vdash \pi \curlywedge \rho \cdot X$. Logo, $\mathtt{supp}_\Upsilon(\rho \cdot X) \subset \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

- $B = \mathtt{dom}(\rho) \cap (\mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))) \neq \emptyset$

  Nesse caso, ρ tem átomos em $\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)))$ e átomos fora de $\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)))$. Considere os conjuntos

  - $C = \{x \mid x \in A \text{ e } \rho^{-1}(x) \in \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))\}$;
  - $D = \{x \mid x \in B \text{ e } \rho^{-1}(x) \in \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))\}$.

  e seja π tal que $\mathtt{dom}(\pi) \subseteq \big(\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D\big) \setminus C$. Então $\pi(a) = a$ para todo $a \in \big(\mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D)\big) \cup C$.

  Além disso, se $x \in \mathtt{dom}(\pi)$, então

  - $x \in \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \setminus C$, e nesse caso $\rho^{-1}(x) \in \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$;
  - ou $x \in D \setminus C$, e nesse caso também temos $\rho^{-1}(x) \in \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

  Com isso $\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$, o que implica $\Upsilon \vdash \pi \curlywedge \rho \cdot X$. Portanto, $\mathtt{supp}_\Upsilon(\rho \cdot X) \subset \big(\mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D)\big) \cup C$.

□

Lema A.12. $\mathtt{supp}_\Upsilon(\mathtt{f}(t_1, \ldots, t_n)) = \bigcup_{i=1}^{n} \mathtt{supp}_\Upsilon(t_i)$.

Demonstração. Seja π uma permutação tal que $\pi(a) = a$, para todo $a \in \bigcup_{i=1}^{n} \mathtt{supp}_\Upsilon(t_i)$. Então, vale $\Upsilon \vdash \pi \curlywedge t_i$, para cada $1 \leq i \leq n$ e, usando a regra ($\curlywedge\mathtt{f}$) (da Tabela 2.1), temos que $\Upsilon \vdash \pi \curlywedge \mathtt{f}(t_1, \ldots, t_n)$. Ou seja, $\mathtt{supp}_\Upsilon(\mathtt{f}(t_1, \ldots, t_n)) \subseteq \cup_{i=1}^{n} \mathtt{supp}_\Upsilon(t_i)$.

Por outro lado, seja π uma permutação tal que $\pi(a) = a$, para todo $a \in \mathtt{supp}_\Upsilon(\mathtt{f}(t_1,\ldots,t_n))$. Então, vale $\Upsilon \vdash \pi \curlywedge \mathtt{f}(t_1,\ldots,t_n)$ e, pela propriedade de Inversão (Lema 2.2), temos que $\Upsilon \vdash \pi \curlywedge t_i$, para cada $1 \leq i \leq n$. Ou seja, $\mathtt{supp}_\Upsilon(t_i) \subseteq \mathtt{supp}_\Upsilon(\mathtt{f}(t_1,\ldots,t_n))$, para todo $1 \leq i \leq n$, ou mais precisamente, $\bigcup_{i=1}^{n} \mathtt{supp}_\Upsilon(t_i) \subseteq \mathtt{supp}_\Upsilon(\mathtt{f}(t_1,\ldots,t_n))$. □

Proposição A.4. $\mathtt{supp}_\Upsilon([a]t) = \mathtt{supp}_{\Upsilon,\overline{(c_1\ c_2)\curlywedge \mathtt{Var}(t)}}((a\ c_1)\cdot t)$, para c_1, c_2 nomes novos com relação a $\Upsilon \vdash [a]t$.

Demonstração. Seja π tal que $\pi(c) = c$ para todo $c \in \mathtt{supp}_\Upsilon([a]t)$. Então, $\Upsilon \vdash \pi \curlywedge [a]t$, e pelo Lema da Inversão (Lema 2.2), segue que $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge (a\ c_1)\cdot t$. Portanto, $\mathtt{supp}_{\Upsilon,\overline{(c_1\ c_2)\curlywedge \mathtt{Var}(t)}}(\pi \curlywedge (a\ c_1)\cdot t) \subseteq \mathtt{supp}_\Upsilon([a]t)$.

Reciprocamente, seja π tal que $\pi(c) = c$ para todo $c \in \mathtt{supp}_{\Upsilon,\overline{(c_1\ c_2)\curlywedge \mathtt{Var}(t)}}((a\ c_1)\cdot t)$. Então, $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge (a\ c_1)\cdot t$, que implica, $\Upsilon \vdash \pi \curlywedge [a]t$;

Portanto $\mathtt{supp}_\Upsilon([a]t) \subseteq \mathtt{supp}_{\Upsilon,\overline{(c_1\ c_2)\curlywedge \mathtt{Var}(t)}}((a\ c_1)\cdot t)$. □

Lema A.13. Seja π tal que $\pi(c) = c$, para todo $c \in \mathtt{supp}_\Upsilon(t) \setminus \{d\}$, e d não ocorre em Υ, t. Então $\Upsilon, \overline{(c_1\ d) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge t$, para c_1 arbitrário que não ocorre em Υ, t.

Demonstração. Existem duas possibilidades:

- $d \notin \mathtt{supp}_\Upsilon(t)$.

  Então, $\pi(c) = c$ para todo $c \in \mathtt{supp}_\Upsilon(t)$ implica que $\Upsilon \vdash \pi \curlywedge t$. Por enfraquecimento (Proposição 2.2) temos que $\Upsilon, \overline{(d\ c_1) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge t$, e o resultado segue.

- $d \in \mathtt{supp}_\Upsilon(t)$.

  A prova é por indução sobre t.

  1. $t = \rho \cdot X$

     Seja π tal que $\pi(c) = c$ para todo $c \in \mathtt{supp}_\Upsilon(\rho \cdot X) \setminus \{d\}$, e d não ocorre em Υ, ρ.

     - $d \notin \mathtt{dom}(\pi)$.
       Neste caso, $\pi(c') = c'$ para todo $c' \in \mathtt{supp}_\Upsilon(\rho \cdot X)$ (inclusive d).
       Então $\Upsilon \vdash \pi \curlywedge \rho \cdot X$ e, por inversão, temos $\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{c_1, d\}$. Portanto, $\Upsilon, \overline{(d\ c_1) \curlywedge \mathtt{Var}(t)} \vdash \pi \curlywedge \rho \cdot X$
     - $d \in \mathtt{dom}(\pi)$.
       Afirmação: $d \in \mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{c_1, d\}$.
       Caso contrário existiria $a \in \pi$ $(a \neq d)$ tal que $\rho(a) \notin \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Mas tal a também ocorre no caso acima, e teríamos uma contradição com o fato de $\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

2. $t = a$

   Este caso é trivial.

3. $t = \mathtt{f}(t_1, \ldots, t_n)$

   Este caso segue por hipótese de indução.

4. $t = [a]t'$

   Seja π tal que $\pi(c) = c$ para todo $c \in \mathtt{supp}_\Upsilon([a]t') \setminus \{d\}$, e d não ocorre em $\Upsilon, [a]t'$. Pela Proposição A.4, $\mathtt{supp}_\Upsilon([a]t') = \mathtt{supp}_{\Upsilon, \overline{(d_1\ d_2)}}((a\ d_1) \cdot t')$, para d_1, d_2 nomes novos. Desta forma, $\pi(c) = c$ para todo $c \in \mathtt{supp}_{\Upsilon, \overline{(d_1\ d_2) \curlywedge \mathtt{Var}(t')}}((a\ d_1) \cdot t') \setminus \{d\}$. Por hipótese de indução,

   $$\Upsilon, \overline{(d_1\ d_2) \curlywedge \mathtt{Var}(t')}, \overline{(d\ c_1) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (a\ d_1) \cdot t'$$

   que implica, pela regra $\curlywedge - abs$,

   $$\Upsilon, \overline{(d\ c_1) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge [a]t'$$

   e o resultado segue.

□

Lema A.14. $\mathtt{supp}_\Upsilon([b]t) = \mathtt{supp}_\Upsilon(t) \setminus \{b\}$.

Demonstração. Por indução em t:

1. $t = \rho \cdot X$:

   Neste caso, consideraremos C e D como na Proposição A.11.

   Afirmação: $\mathtt{supp}_\Upsilon([b](\rho \cdot X)) \subset \underbrace{(\mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D \cup \{b\}) \cup (C \setminus \{b\})}_{P}$.

   - $b \in \mathtt{dom}(\rho)$:

     Tome π tal que $\pi(a) = a$ para todo $a \in P$.

     Então $\mathtt{dom}(\pi) \subseteq (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X) \cup D) \setminus C) \cup \{b\}$. A seguir c_1, c_2 são nomes novos com relação a π, ρ e Υ.

     - $b \in \mathtt{dom}(\pi)$:

       Como $c_1 \notin \mathtt{dom}(\pi) \cup \mathtt{dom}(\rho)$ and $\mathtt{dom}(\pi) \cap \mathtt{dom}(\rho) = \{b\} \cup D$, temos que $\pi^{((b\ c_1) \circ \rho)^{-1}} = \pi^{(b\ c_1)}$. Além disso,

       $$\mathtt{dom}(\pi^{(b\ c_1)}) \subseteq (\mathtt{dom}(\pi) \setminus \{b\}) \cup \rho(D) \cup \{c_1\} \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{c_1, c_2\}$$

- $b \notin \mathtt{dom}(\pi)$:
  Como $c_1 \notin \mathtt{dom}(\pi) \cup \mathtt{dom}(\rho)$, segue que $\pi^{((b\ c_1)\circ\rho)^{-1}} = \pi^{\rho^{-1}} = \pi$. Logo, $\mathtt{dom}(\pi^{((b\ c_1)\circ\rho)^{-1}}) = \mathtt{dom}(\pi) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.
  Assim, $\Upsilon \vdash \pi \curlywedge [b](\rho \cdot X)$ segue das equivalências abaixo.

$$\begin{aligned}
\Upsilon \vdash \pi \curlywedge [b](\rho \cdot X) &\Leftrightarrow \Upsilon, (c_1\ c_2) \curlywedge X \vdash \pi \curlywedge (b\ c_1)(\rho \cdot X) \\
&\Leftrightarrow \Upsilon, (c_1\ c_2) \curlywedge X \vdash \pi \curlywedge \underbrace{((b\ c_1) \circ \rho)} \cdot X \\
&\Leftrightarrow \mathtt{dom}(\pi^{((b\ c_1)\circ\rho)^{-1}}) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{c_1, c_2\}
\end{aligned} \tag{A.5}$$

- $b \notin \mathtt{dom}(\rho)$:
  Tome π tal que $\pi(a) = a$ para todo $a \in (\mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{b\} \cup D)) \cup C$. Então $\mathtt{dom}(\pi) \subseteq (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{b\} \cup D) \setminus C$.
  - $b \in \mathtt{dom}(\pi)$:
    Então $\mathtt{dom}(\pi^{((b\ c_1)\circ\rho)^{-1}}) \subseteq (\mathtt{dom}(\pi) \setminus \{b\}) \cup \{c_1\} \cup \rho(D) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{c_1, c_2\}$. Pelas equivalências em (A.5) segue que $\Upsilon \vdash \pi \curlywedge [b](\rho \cdot X)$.
  - $b \notin \mathtt{dom}(\pi)$:
    Neste caso $\pi^{((b\ c_1)\circ\rho)^{-1}} = \pi^{\rho^{-1}}$, e portanto $\mathtt{dom}(\pi^{\rho^{-1}}) \subseteq \mathtt{dom}(\pi) \cup \rho(D) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Pelas equivalências em (A.5) segue que $\Upsilon \vdash \pi \curlywedge [b](\rho \cdot X)$.

Pela Proposição A.11, temos que

$$\mathtt{supp}_\Upsilon(\rho \cdot X) \subset (\mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D)) \cup C.$$

E pela afirmação do início:

$$\mathtt{supp}_\Upsilon([b](\rho \cdot X)) = (\mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup \{b\} \cup D)) \cup (C \setminus \{b\}).$$

Portanto, $\mathtt{supp}_\Upsilon(([b](\rho \cdot X)) = \mathtt{supp}_\Upsilon(\rho \cdot X) \setminus \{b\}$, e o resultado segue.

2. $t = a$

   Afirmação: $\mathtt{supp}_\Upsilon([b]a) = \{a\}$.

   Note que para todo π tal que $\pi(a) = a$ temos que:

$$\begin{aligned}\Upsilon \vdash \pi \curlywedge [b]a &\Leftrightarrow \Upsilon \vdash \pi \curlywedge (b\ c_1)\cdot a \\ &\Leftrightarrow \Upsilon \vdash \pi \curlywedge a \\ &\Leftrightarrow \pi(a) = a\end{aligned} \tag{A.6}$$

Logo, $\mathtt{supp}_\Upsilon([b]a) = \{a\} = \mathtt{supp}_\Upsilon(a) \setminus \{b\}$, e o resultado segue.

3. $t = \mathsf{f}t'$

   Seja π tal que $\pi(a) = a$ para todo $a \in \mathtt{supp}_\Upsilon([b]\mathsf{f}t')$. Então vale $\Upsilon \vdash \pi \curlywedge [b]ft'$ e temos as seguintes equivalências: d_1, d_2 são nomes novos.

$$\begin{aligned}\Upsilon \vdash \pi \curlywedge [b]\mathsf{f}t' &\Leftrightarrow \Upsilon, \overline{(d_1\ d_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (b\ d_1)\cdot \mathsf{f}t' \\ &\Leftrightarrow \Upsilon, \overline{(d_1\ d_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge \mathsf{f}((b\ d_1)\cdot t') \\ &\Leftrightarrow \Upsilon, \overline{(d_1\ d_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (b\ d_1)\cdot t' \\ &\Leftrightarrow \Upsilon \vdash \pi \curlywedge [b]t' \\ &\Leftrightarrow \Upsilon \vdash \pi \curlywedge \mathsf{f}([b]t')\end{aligned}$$

   Logo,

   $\mathtt{supp}_\Upsilon([b]\mathsf{f}t') = \mathtt{supp}_\Upsilon(\mathsf{f}([b]t')) = \mathtt{supp}_\Upsilon([b]t') = \mathtt{supp}_\Upsilon(t') \setminus \{b\} = \mathtt{supp}_\Upsilon(\mathsf{f}t') \setminus \{b\}$.

4. $t = [a]t'$

   Afirmação: $\mathtt{supp}_\Upsilon([b][a]t') = \mathtt{supp}_\Upsilon([a]t') \setminus \{b\}$.

   - Seja π tal que $\pi(c) = c$ para todo $c \in \mathtt{supp}_\Upsilon([a]t') \setminus \{b\}$. Então $c \in \mathtt{supp}_\Upsilon([a]t')$ e $c \notin \{b\}$. Como $c \in \mathtt{supp}_\Upsilon([a]t')$, então vale $\Upsilon \vdash \pi \curlywedge [a]t'$. Sejam c_1, c_2 nomes novos com relação a Υ, π, a e t'.

$$\Upsilon \vdash \pi \curlywedge [a]t' \Rightarrow \Upsilon \vdash \pi^{(b\ c_1)} \curlywedge [a](b\ c_1)\cdot t', \text{ por equivariância}$$

     - $b \notin \mathtt{dom}(\pi)$:
       Neste caso, $\pi^{(b\ c_1)} = \pi$, e então:

$$\begin{aligned}\Upsilon \vdash \pi^{(b\ c_1)} \curlywedge [a](b\ c_1)\cdot t' &\Rightarrow \Upsilon \vdash \pi \curlywedge [a](b\ c_1)\cdot t' \\ &\Rightarrow \Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge [a](b\ c_1)\cdot t' \\ &\Rightarrow \Upsilon \vdash \pi \curlywedge [b][a]\cdot t'\end{aligned}$$

por enfraquecimento (Lema 2.2) e inversão das regras de derivabilidade de $\curlywedge$. Logo, $c \in \mathtt{supp}_\Upsilon([b][a]t')$.

– $b \in \mathtt{dom}(\pi)$:
Neste caso, $c_1 \in \pi^{(b\ c_1)}$ e $\pi^{(b\ c_1)}(c) = c$, para todo $c \in \mathtt{supp}_\Upsilon([a]t') \setminus \{c_1\}$. Então $\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi^{(b\ c_1)} \curlywedge [a]t'$ pelo Lema A.13. Novamente, por equivariância (Lema 2.3), seguindo de inversão, obtemos:

$$\begin{aligned}\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi^{(b\ c_1)} \curlywedge [a] \cdot t' &\Rightarrow \Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge [a](b\ c_1) \cdot t' \\ &\Rightarrow \Upsilon \vdash \pi \curlywedge [b][a] \cdot t'\end{aligned}$$

Logo, $c \in \mathtt{supp}_\Upsilon([b][a]t')$, e portanto, $\mathtt{supp}_\Upsilon([a]t') \setminus \{b\} \subseteq \mathtt{supp}_\Upsilon([b][a]t')$.

– Por hipótese de indução,

$$\mathtt{supp}_\Upsilon([a]t') = \mathtt{supp}_\Upsilon(t') \setminus \{a\}.$$

Seja π tal que $\pi(c) = c$ para todo $c \in \mathtt{supp}_\Upsilon([a]t')$. Então, $\Upsilon \vdash \pi \curlywedge [a]t'$, e $\pi(c) = c$, para todo $c \in \mathtt{supp}_\Upsilon(t') \setminus \{a\}$. Observe que $\pi^{(b\ c_1)}(c) = c$, para todo $c \in (\mathtt{supp}_\Upsilon(t') \setminus \{a\}) \setminus \{b\}$. Desta forma, $\Upsilon \vdash \pi^{(b\ c_1)} \curlywedge [a]t'$
Assim, $\pi(c) = c$, para todo $c \in \mathtt{supp}_{\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')}}((b\ c_1) \cdot t') \setminus \{a\}$, o que implica

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (b\ c_1) \cdot t'$$

□

Teorema A.1 (Referente ao Teorema 2.4)**.**

(i) Se $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$ então $\mathtt{supp}_\Upsilon(s) = \mathtt{supp}_\Upsilon(t)$

(ii) $\Upsilon \vdash \pi \curlywedge t$ se, e somente se, $\mathtt{dom}(\pi) \cap \mathtt{supp}_\Upsilon(t) = \emptyset$.

Demonstração. (i) Seja π uma permutação tal que $\pi(a) = a$, para todo $a \in \mathtt{supp}_\Upsilon(s)$. Pela Definição 2.9, $\Upsilon \vdash \pi \curlywedge s$. Como por hipótese $\Upsilon \vdash s \overset{\curlywedge}{\approx}_\alpha t$, segue pelo Lema 2.4 que $\Upsilon \vdash \pi \curlywedge t$. Logo, $\mathtt{supp}_\Upsilon(t) \subseteq \mathtt{supp}_\Upsilon(s)$. Analogamente, temos que $\mathtt{supp}_\Upsilon(s) \subseteq \mathtt{supp}_\Upsilon(t)$, e com isso, a igualdade segue.

(ii) A prova é por indução sobre t:

($\Rightarrow$) Suponha que $\Upsilon \vdash \pi \curlywedge t$.

(a) $t = \rho \cdot X$.

Então, a hipótese fica $\Upsilon \vdash \pi \curlywedge \rho \cdot X$, e portanto, $\mathtt{dom}(\pi^\rho) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Suponha que $\rho \neq \mathtt{Id}$:

- $\mathtt{dom}(\pi) \cap \mathtt{dom}(\rho) = \emptyset$.
  Então $\pi^\rho = \pi$ e $\mathtt{dom}(\pi) = \mathtt{dom}(\pi^\rho) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.
  Pelo Teorema 2.3, $\mathtt{supp}_\Upsilon(\rho \cdot X) \subset (\mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D)) \cup C$, e portanto, $\mathtt{supp}_\Upsilon(\rho \cdot X) \cap \mathtt{dom}(\pi) = \emptyset$.
- $\mathtt{dom}(\pi) \cap \mathtt{dom}(\rho) \neq \emptyset$.
  Então $\mathtt{dom}(\pi^\rho) \subseteq \mathtt{dom}(\pi) \cup \mathtt{dom}(\rho) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.
  Suponha, por absurdo, que $a \in \mathtt{dom}(\pi) \cap \mathtt{supp}_\Upsilon(\rho \cdot X) \neq \emptyset$. Pelo Teorema 2.3, segue que $a \in \mathtt{supp}_\Upsilon(\rho \cdot X) \subset (\mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D)) \cup C$. Vamos analisar os dois casos:
  i. $a \in \mathbb{A} \setminus (\mathtt{dom}(\mathtt{perm}(\Upsilon|_X)) \cup D)$.
     (*) Se $a \in \mathtt{dom}(\pi) \setminus \mathtt{dom}(\rho)$ então $\pi^\rho = \pi$ e $a \in \mathtt{dom}(\pi^\rho) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.
     Absurdo, pois tomamos $a \in \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.
     (*) Se $a \in \mathtt{dom}(\pi) \cap \mathtt{dom}(\rho)$ então $\rho(a) \in \mathtt{dom}(\pi^\rho) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon_X))$.
     Absurdo, pois $a \in \mathbb{A} \setminus D$.
  ii. $a \in C$
     Neste caso, $a \in C = \{a \mid a \in A \text{ e } \rho(a) \in \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))\}$, onde $A = \mathtt{dom}(\rho) \cap \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.
     Então, $\rho(a) \in \mathtt{dom}(\pi^\rho) \subseteq \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$. Absurdo, pois de $a \in C$, temos que $\rho(a) \in \mathbb{A} \setminus \mathtt{dom}(\mathtt{perm}(\Upsilon|_X))$.

  Logo, $\mathtt{dom}(\pi) \cap \mathtt{supp}_\Upsilon(\rho \cdot X) = \emptyset$.

O caso em que $\rho = \mathtt{Id}$ é trivial.

(b) $t = a$.

Suponha que $\Upsilon \vdash \pi \curlywedge a$. Então $\pi(a) = a$ e $a \notin \mathtt{dom}(\pi)$.

Por definição, $\mathtt{supp}_\Upsilon(a) = \{a\}$. Portanto, $\mathtt{dom}(\pi) \cap \mathtt{supp}_\Upsilon(a) = \emptyset$.

(c) $t = \mathsf{f}(t_1, t_2, \ldots, t_n)$.

Suponha que $\Upsilon \vdash \pi \curlywedge \mathsf{f}(t_1, \ldots, t_n)$, então $\Upsilon \vdash \pi \curlywedge t_1, \ldots, \Upsilon \vdash \pi \curlywedge t_n$. Por hipótese de indução, $\mathtt{dom}(\pi) \cap \mathtt{supp}_\Upsilon(t_i) = \emptyset$, para cada $i = 1, \ldots, n$. Pelo Teorema 2.3, segue que $\mathtt{dom}(\pi) \cap \mathtt{supp}_\Upsilon(\mathsf{f}(t_1, \ldots, t_n)) = \emptyset$.

(d) $t = [b]t'$

Suponha que $\Upsilon \vdash \pi \curlywedge [b]t'$. Então,

$$\Upsilon, \overline{(c_1\ c_2) \curlywedge \mathtt{Var}(t')} \vdash \pi \curlywedge (b\ c_1) \cdot t'.$$

onde c_1, c_2 são nomes novos, que não ocorrem em Υ, π nem em t'.
Por hipótese de indução, $\mathtt{dom}(\pi) \cap \mathtt{supp}_{\Upsilon, \overline{(c_1\ c_2)\curlywedge \mathtt{Var}(t')}}((b\ c_1) \cdot t') = \emptyset$.
Pelos Lemas A.13 e A.14:

$$\mathtt{supp}_{\Upsilon}([b]t') = \mathtt{supp}_{\Upsilon}(t') \setminus \{b\} = \mathtt{supp}_{\Upsilon, \overline{(c_1\ c_2)\curlywedge \mathtt{Var}(t')}}((b\ c_1) \cdot t').$$

Logo, $\mathtt{dom}(\pi) \cap \mathtt{supp}_{\Upsilon}([b]t') = \emptyset$, e o resultado segue.

$\square$