Closed combination of context-embedding iterative strategies

Walid Belkhir, Nicolas Ratier, Duy Duc Nguyen, Michel Lenczner

Closed combination of context-embedding iterative strategies *

Walid Belkhir[1,2], Nicolas Ratier[1], Duy Duc Nguyen[1] and Michel Lenczner[1]

[1] FEMTO-ST, Time and Frequency Department, University of Bourgogne-Franche-Comté, 25000, Besançon, France.
[2] INRIA Nancy - Grand Est, Pesto project, 54600 Villers-lès-Nancy, France.
{duyduc.nguyen, nicolas.ratier, michel.lenczner}@femto-st.fr

Abstract. This work is motivated by the challenging problem of the computer-aided generation of approximations (viewed as a series of transformations) of partial derivative equations. In this framework, the approximations posed over complex settings are incrementally constructed by extending an approximation posed on a simple setting and combining these extensions.
In order to formalize these extensions and their combination, we introduce a class of rewriting strategies, called context-embedding iterative strategies (CE-strategies, for short). Roughly speaking, the class of CE-strategies is constructed by means of adding contexts and an iteration operator allowing the definition of recursive strategies. We show that the class of CE-strategies is closed under combination with respect to a correctness-completeness criterion. It turns out that the class CE-strategies enjoy nice algebraic properties, namely, the combination is associative, has a neutral element, and all the elements are idempotents.

1 Introduction

The motivation of this work originates in an undergoing project for the modeling and simulation of complex systems in micro or nano-technologies, e.g. [12, 2, 3]. The systems under consideration are governed by partial differential equations (PDEs) and are too complex to be simulated by straightforward numerical methods, at least in the time-scale of design engineering. In mathematics the asymptotic methods, also called perturbation methods in physics, have been developed for more than seventy years for PDEs with the purpose of their transformation to "simpler" PDEs when they involve one or several small parameters. The latter can refer to geometry characteristics of the PDE domain or to coefficients. They are used in all fields where PDEs are used for modeling ranging from physics, biology, finance etc. As an illustration, we refer to the review paper [6] for some applications in mechanics of periodic media where the small parameter is the ratio of a periodic cell size to the full body size. Today, asymptotic methods are developed for producing models with drastically reduced simulation

* This work was supported by LABEX ACTION ANR-11-LABX-0001-01.

time together with keeping the essence of the model. Their main drawback preventing their dissemination in the engineering community is their case-by-case derivation. That is, for each new problem, the full process of model derivation is redone from the scratch even though the new problem shares many features with an already solved problem. Thus, they are implemented only in specialized software. We adopted an alternate approach by developing a software package called `MEMSALab` (for MEMS Array Lab) whose aim is to incrementally derive asymptotic models for input equations by taking into account their own features e.g. the scalar valued or vector valued solution, different estimates on the solutions and sources, thin structures, periodic structures, multiple nested scales, heterogeneity etc.

Our approach takes advantage of the modularity and the algebraic flavour of the asymptotic method of [10]. It relies on the so called *by-extension-combination method* [3] that we sketch. An asymptotic model derivation starts with an input PDE coming from any scientific field to which a derivation, also said proof, is applied ending to the expected model. This scheme is build for a reference case, which is the simplest that we can consider, so we call it the *reference scheme*. Then, it is complexified, we say extended, in several manners to take into account new features yielding new schemes. The input PDEs still arise from an application area but with additional features. Accordingly, the reference proof is extended in different ways to cover the new features. Applying the *extended proofs* to the enriched PDEs yields new asymptotic models. Finally, a new scheme for an input PDE covering a group of new features is built by *combination*. Precisely, its input PDE is still issued from a practical problem. Its proof is obtained by applying a combination of two or more extensions, built in the previous step, to the reference proof. Finally, applying the resulting proof to the input PDE yields an asymptotic model enjoying the groups of features. In summary, combining extensions related to new elementary features allows for building new proofs and therefore new asymptotic models in an incremental manner.

The concept of combination is not an isolated one, we identified works in several fields involving different techniques but with the same key idea. We refer to combination of logics [9, 4], algorithms, verification methods [5], and decision procedures [11]. These works share a common principle of incremental design of complex systems by integration of simple and heterogeneous subsystems.

The above idea of extension and their combination was introduced in [12], but in this seminal work the combination of extensions was done via composition, not allowing for conflicts between extensions. The complete principle of the extension-combination method was introduced in [3]. In this work, we have presented the design and implementation of a user language for the specification of rewriting strategies based proofs and extensions. We also stated computation rules for combinations of extensions. Although we considered combinations for a small class of usual rewriting strategies as `TopDown` and `BottomUp`, the question whether this class, or possibly a wider class, is closed under combination was left open, as well as the question of the correctness and soundness of the combination formulae.

Here, we address these two questions, in a framework involving more elementary operations but generating a wider class of rewriting strategies called *CE-strategies*. Although the idea of combination is kept the same, the tools and the techniques are different. The elementary extension operation on a term is still an enrichment by context insertion. However, the traversal strategies in a CE-strategy are built with a jump operator and an iterator/fixed-point operator instead of `TopDown` a more complex strategy. The resulting strategy language was inspired from [1] wherein it was shown that many syntactic iterative objects, like automata, games, logic, strategies etc, can be turned into a μ-calculus. In other words, instead of formulating the strategy language as in [7], the μ-calculus-like approach makes the strategy constructors more rudimentary and therefore tractable the question of language closure for combinations. Moreover, the formulae of combination of CE-strategies together with their verification is also much simplified.

Contributions. We address the key problem of extension and combination of proofs encountered in the field of computer aided asymptotic model derivation. Precisely, we identify an operation of combination over a class of extensions named CE-strategies, expressed as rewriting strategies that navigate along trees and insert contexts. We prove that this class is closed by combination after establishing an explicit formula of combinations. We introduce a correctness criteria that guarantees the validity of the combination formulas. All of them have been rigorously proved, which is an important piece of this work. We shown that usual traversal strategies as `TopDown` or `BottomUp` belong to the class of CE-strategies. Several nice algebraic properties of the CE-strategies are proved.

Organization of the paper. The paper is structured as follows. In Section 3 we introduce the class of elementary CE-strategies, which is a subclass of CE-strategies. It provides an illustration of the concept of unification and combination in simple cases and serves as a set of basic building blocks for the class of CE-strategies. The syntax and the semantics of the latter as well as their unification and combination are introduced in Section 4. Finally, in Section 5 we show that the unification and combination of CE-strategies is sound and complete, and state its main algebraic properties.

2 Preliminaries

We introduce preliminary definitions and notations.

Terms, contexts. Let $\mathcal{F} = \cup_{n\geq 0}\mathcal{F}_n$ be a set of symbols called *function symbols*. The *arity* of a symbol f in $\mathcal{F}_n$ is n and is denoted $ar(f)$. Elements of arity zero are called *constants* and often denoted by the letters a, b, c, etc. The set $\mathcal{F}_0$ of constants is always assumed to be not empty. Given a denumerable set $\mathcal{X}$ of *variable* symbols, the set of *terms* $\mathcal{T}(\mathcal{F},\mathcal{X})$, is the smallest set containing $\mathcal{X}$ and such that $f(t_1,\ldots,t_n)$ is in $\mathcal{T}(\mathcal{F},\mathcal{X})$ whenever $ar(f) = n$ and $t_i \in \mathcal{T}(\mathcal{F},\mathcal{X})$

for $i \in [1..n]$. Let the constant $\square \notin \mathcal{F}$, the set $\mathcal{T}_\square(\mathcal{F},\mathcal{X})$ of "*contexts*", denoted simply by $\mathcal{T}_\square$, is made with terms with symbols in $\mathcal{F} \cup \mathcal{X} \cup \{\square\}$ which includes exactly one occurence of $\square$. Evidently, $\mathcal{T}_\square(\mathcal{F},\mathcal{X})$ and $\mathcal{T}(\mathcal{F},\mathcal{X})$ are two disjoint sets. We shall write simply $\mathcal{T}$ (resp. $\mathcal{T}_\square$) instead of $\mathcal{T}(\mathcal{F},\mathcal{X})$ (resp. $\mathcal{T}_\square(\mathcal{F},\mathcal{X})$). We denote by $\mathcal{V}ar\,(t)$ the set of variables occurring in t.

Positions, prefix-order, combination of contexts. Let t be a term in $\mathcal{T}(\mathcal{F},\mathcal{X})$. A position in a tree is a sequence of integers of $\mathbb{N}_\epsilon^\omega = \{\epsilon\} \cup \mathbb{N} \cup (\mathbb{N} \times \mathbb{N}) \cup \cdots$. In particular we shall write $\mathbb{N}_\epsilon$ for $\{\epsilon\} \cup \mathbb{N}$. Given two positions $p = p_1p_2 \dots p_n$ and $q = q_1q_2 \dots q_m$, the *concatenation* of p and q, denoted by $p \cdot q$ or simply pq, is the position $p_1p_2 \dots p_nq_1q_2 \dots q_m$. The set of positions of the term t, denoted by $\mathcal{P}os\,(t)$, is a set of positions of positive integers such that, if $t \in \mathcal{X}$ is a variable or $t \in \mathcal{F}_0$ is a constant, then $\mathcal{P}os\,(t) = \{\epsilon\}$. If $t = f\,(t_1,...,t_n)$ then $\mathcal{P}os\,(t) = \{\epsilon\} \cup \bigcup_{i=1,n} \{ip \mid p \in \mathcal{P}os\,(t_i)\}$. The position ϵ is called the root position of term t, and the function or variable symbol at this position is called root symbol of t.

The prefix order defined as $p \leq q$ iff there exists p' such that $pp' = q$, is a partial order on positions. If $p' \neq \epsilon$ then we obtain the strict order $p < q$. We write $(p \parallel q)$ iff p and q are incomparable with respect to $\leq$. The binary relations $\sqsubset$ and $\sqsubseteq$ defined by $p \sqsubset q \quad \text{iff} \quad \big(p < q \text{ or } p \parallel q\big)$ and $p \sqsubseteq q \quad \text{iff} \quad \big(p \leq q \text{ or } p \parallel q\big)$, are total relations on positions.

For any $p \in \mathcal{P}os(t)$ we denote by $t_{|p}$ the subterm of t at position p, that is, $t_{|\epsilon} = t$, and $f(t_1,...,t_n)_{|iq} = (t_i)_{|q}$. For a term t, we shall denote by $\delta(t)$ the depth of t, defined by $\delta(t_0) = 0$, if $t_0 \in \mathcal{X} \cup \mathcal{F}^0$ is a variable or a constant, and $\delta(f(t_1,\dots,t_n)) = 1 + max(\delta(t_i))$, for $i = 1,\dots,n$. For any position $p \in \mathcal{P}os\,(t)$ we denote by $t\,[s]_p$ the term obtained by replacing the subterm of t at position p by s: $t[s]_\epsilon = s$ and $f(t_1,...,t_n)[s]_{iq} = f(t_1,...,t_i[s]_q,...,t_n)$.

For any $\tau, \tau' \in \mathcal{T}_\square$, we define the combination of two contexts by $\tau[\tau'] = \tau[\tau']_{\mathcal{P}os(t,\square)}$, where $\mathcal{P}os\,(t,\square)$ is the position of $\square$ in t. For any two tuples of contexts $\boldsymbol{\tau} = (\tau_1,\dots,\tau_n)$ and $\boldsymbol{\tau'} = (\tau'_1,\dots,\tau'_m)$ in $\mathcal{T}_\square^\omega = \mathcal{T}_\square \cup (\mathcal{T}_\square \times \mathcal{T}_\square) \cup \cdots$, we define the concatenation operation "$\cdot$" by $\boldsymbol{\tau} \cdot \boldsymbol{\tau'} = (\tau_1,\dots,\tau_n,\tau'_1,\dots,\tau'_m)$. The evaluation of a tuple of contexts $\boldsymbol{\tau} = \boldsymbol{\tau}_1 \cdot \ldots \cdot \boldsymbol{\tau}_n$, denoted as $\texttt{eval}(\boldsymbol{\tau})$, is inductively defined by

i.) if $\boldsymbol{\tau}_i = \boldsymbol{\tau}_{i+1}$, for some $i \in [1,\dots,n]$, then

$$\texttt{eval}(\boldsymbol{\tau}_1 \cdot \ldots \cdot \boldsymbol{\tau}_i \cdot \boldsymbol{\tau}_{i+1} \cdot \ldots \cdot \boldsymbol{\tau}_n) = \texttt{eval}(\boldsymbol{\tau}_1 \cdot \ldots \cdot \boldsymbol{\tau}_i \cdot \boldsymbol{\tau}_{i+2} \cdot \ldots \cdot \boldsymbol{\tau}_n),$$

ii.) otherwise,

$$\texttt{eval}\,((\tau_1,\dots,\tau_m)) = \begin{cases} \tau_1, & \text{if } n = 1 \\ \tau_1[\texttt{eval}\,((\tau_2,\dots,\tau_m))], & \text{if } n \geq 2. \end{cases}$$

A substitution is a mapping $\sigma : \mathcal{X} \to \mathcal{T}(\mathcal{F},\mathcal{X})$ such that $\sigma(x) \neq x$ for only finitely many xs. The finite set of variables that σ does not map to themselves is called the domain of σ: $\mathsf{Dom}(\sigma) \overset{def}{=} \{x \in \mathcal{X} \mid \sigma(x) \neq x\}$. If $\mathsf{Dom}(\sigma) = \{x_1,...,x_n\}$ then we write σ as: $\sigma = \{x_1 \mapsto \sigma\,(x_1)\,,...,x_n \mapsto \sigma\,(x_n)\}$. A substitution $\sigma : \mathcal{X} \to \mathcal{T}(\mathcal{F},\mathcal{X})$ uniquely extends to an endomorphism $\widehat{\sigma} : \mathcal{T}(\mathcal{F},\mathcal{X}) \to \mathcal{T}(\mathcal{F},\mathcal{X})$ defined by: $\widehat{\sigma}(x) = \sigma(x)$ for all $x \in \mathsf{Dom}(\sigma)$, and $\widehat{\sigma}(x) = x$ for all

$x \notin \mathsf{Dom}(\sigma)$, and $\widehat{\sigma}(f(t_1,\ldots,t_n)) = f(\widehat{\sigma}(t_1),\ldots,\widehat{\sigma}(t_n))$ for $f \in \mathcal{F}$. In what follows we do not distinguish between a substitution and its extension.

For two terms $t, t' \in \mathcal{T}$, we say that t matches t', written $t \ll t'$, iff there exists a substitution σ, such that $\sigma(t) = t'$. It turns out that if such a substitution exists, then it is unique. The most general unifier of the two terms u and u' is a substitution γ such that $\gamma(u) = \gamma(u')$ and, for any other substitution γ' satisfying $\gamma'(u) = \gamma'(u')$, we have that γ' is subsumed by γ. Besides, we shall write $u \wedge u'$ to denote the term $\gamma(u)$. The composition of functions will be denoted by "$\circ$". For a set A, the set of all functions from A to A will be denoted by $\mathfrak{F}(A)$. If l_1 and l_2 are lists, then we denote by $l_1 \sqcup l_2$ (resp. $l_1 \sqcap l_2$) their concatenation (resp. intersection). Sometimes we shall write $\sqcup_{i=1,n} e_i$ to denote the list $[e_1,\ldots,e_n]$. For any $n \in \mathbb{N}$ we simply denote by $[n]$ the interval $[1,\ldots,n]$.

3 Elementary CE-strategies and their combination

To define the elementary CE-strategies, we introduce two elementary strategies. For a position p and a tuple of context $\boldsymbol{\tau}$, the jump strategy $@p.\boldsymbol{\tau}$ applied to a term t inserts $\boldsymbol{\tau}$ at the position p of the input term t. The failing strategy $\varnothing$ fails when apply to any term. Their precise semantics are given in Definition below for Semantics of elementary CE-strategies.

Definition 1 (Elementary CE-strategies). *An elementary CE-strategy is either the failing strategy $\varnothing$ or the list $[@p_1.\boldsymbol{\tau}_1,\ldots,@p_n.\boldsymbol{\tau}_n]$, where $n \geq 1$, each p_i is a positions and each $\boldsymbol{\tau}_i$ is a tuple of contexts in $\mathcal{T}_\square^\omega$.*

We impose that the elementary CE-strategies respect some constraints on positions of insertions to avoid conflicts: the order of context insertions goes from the leave to the root.

Definition 2 (Well-founded elementary CE-strategy). *An elementary CE-strategy $E = [@p_1.\boldsymbol{\tau}_1,...,@p_n.\boldsymbol{\tau}_n]$ is well-founded iff*

i.) a position occurs at most one time in E, i.e. $p_i \neq p_j$ for all $i \neq j$, and

ii.) insertions at lower positions occur earlier in E, i.e. $i < j$ iff $p_i \sqsubset p_j$, for all $i, j \in [n]$.

In particular, the empty elementary CE-strategy $\varnothing$ is well-founded.

In all what follows we work only with the set of well-founded elementary CE-strategies, denoted by $\mathcal{E}$. For two elementary CE-strategies E' and E', we shall write $E = E'$ to mean that they are equal up to a permutation of their parallel positions. For a position p, we let $p.[@p_1.\boldsymbol{\tau}_1,\ldots,@p_n.\boldsymbol{\tau}_n] = [@pp_1.\boldsymbol{\tau}_1,\ldots,@pp_n.\boldsymbol{\tau}_n]$. We next define the semantics of an elementary CE-strategy as a function in $\mathfrak{F}(\mathcal{T} \cup \{\mathbb{F}\})$, with the idea that if the application of an elementary CE-strategy to a term fails, the result is $\mathbb{F}$.

Definition 3 (Semantics of elementary CE-strategies). *The semantics of an elementary CE-strategy E is a function $[\![E]\!]$ in $\mathfrak{F}(\mathcal{T} \cup \{\mathbb{F}\})$ inductively defined by:*

$$[\![\varnothing]\!](t) \stackrel{def}{=} \mathbb{F}, \qquad [\![@p.\boldsymbol{\tau}]\!](t) \stackrel{def}{=} \begin{cases} t[\mathtt{eval}(\boldsymbol{\tau})[t_{|p}]]_p & \textit{if } p \in \mathcal{P}os(t) \\ \mathbb{F} & \textit{otherwise}, \end{cases}$$

$$[\![E]\!](\mathbb{F}) \stackrel{def}{=} \mathbb{F}, \quad [\![(p_1, \boldsymbol{\tau}_1), \ldots, (p_n, \boldsymbol{\tau}_n)]\!](t) \stackrel{def}{=} ([\![@p_n.\boldsymbol{\tau}_n]\!] \circ \cdots \circ [\![@p_1.\boldsymbol{\tau}_1]\!])(t).$$

Example 1. We illustrate the idea and the interest of elementary CE-strategies through the simple example, presented in [3], of an extension of a mathematical expression encountered in an extension of a proof. The context $\boldsymbol{\tau} = \mathtt{list}(\square, j)$ depicted in Figure 1 captures the idea that the extension transforms a one-dimensional space coordinate variable x to an indexed multi-dimensional space coordinate variable x_j. The application of $@p.\boldsymbol{\tau}$ to the term $t = \partial_x v(x)$ at the position of p of the variable x (the parameter of the differential operator ∂) yields the term $[\![@p.\boldsymbol{\tau}]\!](t) = \partial_{x_j} v(x)$.

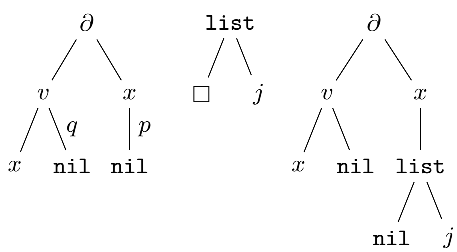

Fig. 1. Application of the elementary CE-strategy $@p.\tau$ (with the context $\tau = \mathtt{list}(\square, j)$) to the term $t = \partial_x v(x)$ at the position p, yielding the term $\partial_{x_j} v(x)$.

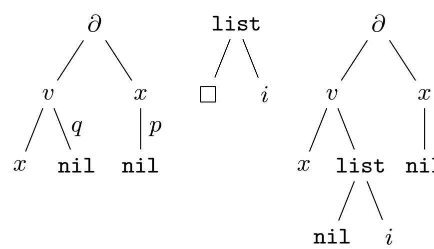

Fig. 2. Application of the elementary CE-strategy $@q.\tau'$ (with the context $\tau' = \mathtt{list}(\square, i)$) to the term $t = \partial_x v(x)$ at the position q, yielding the term $\partial_x v_i(x)$.

Besides, Figure 2 illustrates the elementary CE-strategy $@q.\boldsymbol{\tau}'$ and its application to the term $t = \partial_x v(x)$ at the position of the function v which yields the term $[\![@q.\boldsymbol{\tau}']\!](t) = \partial_x v_i(x)$. When an elementary CE-strategy $@p.\boldsymbol{\tau}$, where p is a position, is applied to a term t at the position p, the context $\boldsymbol{\tau}$ is inserted at the position p of t, and the subterm of t at the position p is inserted at $\square$. Figure 3 shows the combination of the two elementary CE-strategy $@p.\boldsymbol{\tau}$ and $@q.\boldsymbol{\tau}'$.

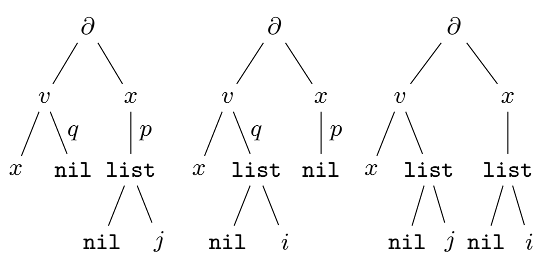

Fig. 3. The elementary CE-strategy $(@q.\tau) \curlyvee (@q.\tau')$, that is the combination of the two elementary CE-strategies $@q.\tau$ and $@q.\tau'$, and its application to the term $t = \partial_x v(x)$, yielding the term $\partial_{x_j} v_i(x)$.

Definition 4 (Unification of two elementary CE-strategies). *The unification of two elementary CE-strategies is the binary operation $\curlywedge : \mathcal{E} \times \mathcal{E} \longrightarrow \mathcal{E}$ defined as*

$$E \curlywedge E' = \begin{cases} E'' & \text{if } E \neq \varnothing \text{ and } E' \neq \varnothing \\ \varnothing & \text{if } E = \varnothing \text{ or } E' = \varnothing \end{cases}$$

where the first case $E = [@p_1.\boldsymbol{\tau}_1, \ldots, @p_n.\boldsymbol{\tau}_n]$, $E' = [@p'_1.\boldsymbol{\tau}'_1, \ldots, @p'_m.\boldsymbol{\tau}'_m]$ and $E'' = [@p''_1.\boldsymbol{\tau}''_1, \ldots, @p''_r.\boldsymbol{\tau}''_r]$ with sets of positions P, P' and $P'' = P \cup P'$ and the contexts $\boldsymbol{\tau}''_k$ defined as follows. For a position $p''_k \in P'' \setminus P \cap P'$,

$$\boldsymbol{\tau}''_k = \boldsymbol{\tau}_i \text{ if } p''_k = p_i \in P \qquad \text{and} \qquad \boldsymbol{\tau}''_k = \boldsymbol{\tau}'_j \text{ if } p''_k = p'_j \in P'.$$

Otherwise, $p''_k = p_i = p'_j \in P \cap P'$ for some i, j and $\boldsymbol{\tau}''_k = \boldsymbol{\tau}'_j \cdot \boldsymbol{\tau}_i$. Besides, the other of the positions in P'' is chosen so that E'' is well-founded.

Definition 5 (Combination of two elementary CE-strategies). *The combination of two elementary CE-strategies is a binary operation $\curlyvee : \mathcal{E} \times \mathcal{E} \longrightarrow \mathcal{E}$ defined for any E and E' in $\mathcal{E}$ by*

$$E \curlyvee E' = \begin{cases} E \curlywedge E' & \text{if } E \neq \varnothing \text{ and } E' \neq \varnothing \\ E & \text{if } E \neq \varnothing \text{ and } E' = \varnothing \\ E' & \text{if } E = \varnothing \text{ and } E' \neq \varnothing \\ \varnothing & \text{if } E = \varnothing \text{ and } E' = \varnothing \end{cases}$$

Proposition 1. *The following hold.*

1. *The set $\mathcal{E}$ of elementary CE-strategies together with the unification and combination operations enjoys the following properties.*
   (a) *The neutral element of the unification and combination is $@\epsilon.\Box$.*
   (b) *Every elementary CE-strategy E is idempotent for the unification and combination, i.e. $E \curlywedge E = E$ and $E \curlyvee E = E$.*
   (c) *The unification and combination are associative.*
2. *The unification and combination of elementary CE-strategies is non commutative.*

The idempotence follows from the equality $\mathtt{eval}(\boldsymbol{\tau} \cdot \boldsymbol{\tau}) = \mathtt{eval}(\boldsymbol{\tau})$, the associativity follows from the equality $\mathtt{eval}((\boldsymbol{\tau}_1 \cdot \boldsymbol{\tau}_2) \cdot \boldsymbol{\tau}_3) = \mathtt{eval}(\boldsymbol{\tau}_1 \cdot (\boldsymbol{\tau}_2 \cdot \boldsymbol{\tau}_3))$, and the non-commutativity is a consequence of $\mathtt{eval}(\boldsymbol{\tau}_1 \cdot \boldsymbol{\tau}_2) \neq \mathtt{eval}(\boldsymbol{\tau}_2 \cdot \boldsymbol{\tau}_1)$ in general, for any tuples of contexts $\boldsymbol{\tau}, \boldsymbol{\tau}_1, \boldsymbol{\tau}_2$ and $\boldsymbol{\tau}_3$.

4 The class of context-embedding strategies (CE-strategies)

We introduced the elementary CE-strategies to clarify the ideas behind contexts, their insertion as well as their combination. However, elementary CE-strategies are not satisfactory for practical applications, since the positions are generally not accessible and cannot be used on a regular basis in applications. So, we enrich this framework by introducing navigation strategies to form a class of *CE-strategies* that is closed under combination.

4.1 Specification of failure by Boolean formulas

The first enrichment of the elementary CE-strategies is to specify and handle the failure. Assume that we applied the elementary CE-strategy $E = [@p_1.\boldsymbol{\tau}_1, \ldots, @p_n.\boldsymbol{\tau}_n]$ to a term, and assume that one of the $@p_i.\boldsymbol{\tau}_i$ fails. In this case the whole elementary CE-strategy E fails. We shall relax this strong failure specification by allowing one to explicitly specify whether the application of a strategy to a term fails depending on the failure of the application of its sub-strategies. In this subsection we propose to specify the failure by means of Boolean formulas that we next introduce. For this purpose, to each position p in $\mathbb{N}_\epsilon^\omega$, we associate a Boolean *position-variable* denoted by $\widehat{p}$. The idea is that when we apply a CE-strategy, say $@p.\boldsymbol{\tau}$, to a term, then we get $\widehat{p} := \texttt{True}$ if this application succeeds, and $\widehat{p} := \texttt{False}$ if it fails. For instance, assume that we want that the application of the elementary CE-strategy $[@p_1.\boldsymbol{\tau}_1, @p_2.\boldsymbol{\tau}_2]$ succeeds if the application of $@p_1.\boldsymbol{\tau}_1$ succeeds *or* the application of $@p_2.\boldsymbol{\tau}_2$ succeeds. This is specified by the Boolean formula $\widehat{p}_1 \vee \widehat{p}_2$.

In what follows, the set of Boolean position-variables is denoted by $\widehat{\mathbb{N}}_\epsilon^\omega$.

Definition 6 (Boolean formulas over $\widehat{\mathbb{N}}_\epsilon^\omega$). *The set of Boolean formulas over $\widehat{\mathbb{N}}_\epsilon^\omega$, denoted by $\mathscr{B}ool(\widehat{\mathbb{N}}_\epsilon^\omega)$, is defined by the grammar:*

$$\mathcal{B} ::= \texttt{True} \mid \texttt{False} \mid \widehat{p} \mid \mathcal{B} \wedge \mathcal{B} \mid \mathcal{B} \vee \mathcal{B}$$

where $\widehat{p} \in \widehat{\mathbb{N}}_\epsilon^\omega$. The set of position-variables of $\phi \in \mathscr{B}ool(\widehat{\mathbb{N}}_\epsilon^\omega)$ will be denoted by $\mathcal{V}ar\,(\phi)$. A valuation is a mapping $\nu : \widehat{\mathbb{N}}_\epsilon^\omega \longrightarrow \{\texttt{True}, \texttt{False}\}$. We write $\nu \models \phi$ to mean that $\nu(\phi)$ holds.

4.2 Syntax and semantics of CE-strategies

Besides the specification of failure, the second enrichment of the elementary CE-strategies is the introduction of navigation strategies. Namely, we shall introduce the left-choice strategy constructor ($\oplus$), a restricted form of the composition (";"), and the fixed-point constructor ("μ") allowing the recursion in the definition of strategies. The resulting class is called the class of CE-strategies. In what follows we assume that there is a denumerable set of *fixed-point variables* denoted by $\mathcal{Z}$. Fixed-point variables in $\mathcal{Z}$ will be denoted by $X, Y, Z, \ldots$

Definition 7 (CE-strategies). *The class of CE-strategies is defined by the following grammar:*

$$\begin{aligned}\mathcal{S} \ ::= \varnothing \mid X \mid (u \to u); \mathcal{S} \mid \mathcal{S} \oplus \mathcal{S} \mid u \to u[\boldsymbol{\tau}] \mid \mu X.\mathcal{S} \mid @p.\mathcal{S} \mid @p.\boldsymbol{\tau} \mid \\ \langle [@p_1.\mathcal{S}_1 \ldots, @p_n.\mathcal{S}_n] \,|\, \phi \rangle\end{aligned}$$

where X is a fixed-point variable in $\mathcal{Z}$, and u is a term in $\mathcal{T}$, and $\boldsymbol{\tau}$ is a tuple of contexts in $\mathcal{T}_\square^\omega$ and $p, p_1, \ldots, p_n$ are positions in $\mathcal{P}os$, and ϕ is a Boolean formula in $\mathscr{B}ool(\widehat{\mathbb{N}}_\epsilon^\omega)$ with $\mathcal{V}ar\,(\phi) = \{\widehat{p}_1, \ldots, \widehat{p}_n\} \setminus \{\epsilon\}$. The set of CE-strategies will be denoted by $\mathcal{C}$.

The strategy $@p.S$ means to jump to the position p and to apply S there. The strategy $\langle [@p_1.S_1 \ldots, @p_n.S_n] \,|\, \phi \rangle$ consists in applying each of $@p_i.S_i$, which yields a valuation that sends the position-variable $\widehat{p_i}$ to `False` iff the application of $@p_i.S_i$ fails, then evaluating the Boolean formula ϕ. If this evaluation is false then the whole strategy $\langle [@p_1.S_1 \ldots, @p_n.S_n] \,|\, \phi \rangle$ fails, otherwise, every sub-strategy $@p_i.S_i$ that failed behaves like the identity, i.e. it does nothing, while the other non-failing sub-strategies $@p_j.S_j$ are applied. For example, if we apply the CE-strategy $S = \langle [@p_1.S_1, @p_2.S_2] \,|\, \widehat{p_1} \vee \widehat{p_2} \rangle$ to a term t, and $@p_1.S_1$ fails while $@p_2.S_2$ does not, we get an evaluation ν with $\nu(\widehat{p_1}) = \texttt{False}$ and $\nu(\widehat{p_1}) = \texttt{True}$. Since $\nu \models \widehat{p_1} \vee \widehat{p_2}$, then the result of the application of S to t is precisely the result of the application of $@p_2.S_2$ to t, making $@p_1.S_1$ behaving like the identity.

It's worth mentioning that the aim of incorporation of the Boolean formulas in CE-strategies is to make it expressive enough so we can write the standard traversal strategies (see Example 2). The fragment of CE-strategies without Boolean formulas remains closed under unification and combination.

We shall sometimes write $\mu X.S(X)$ instead of $\mu X.S$ to emphasize that the fixed-point variable X is free in S.

To define the semantics of CE-strategies we need to introduce an intermediary function $\eta : \mathfrak{F}(\mathcal{T} \cup \{\mathbb{F}\}) \to \mathcal{T} \cup \{\mathbb{F}\} \to \mathcal{T} \cup \{\mathbb{F}\}$, that stands for the *fail as identity*. It is defined for any function f in $\mathfrak{F}(\mathcal{T} \cup \{\mathbb{F}\})$ and any term $t \in \mathcal{T} \cup \{\mathbb{F}\}$ by

$$(\eta(f))(t) = \begin{cases} f(t) & \text{if } f(t) \neq \mathbb{F} \\ t & \text{otherwise.} \end{cases}$$

Beside, let $S^{i+1}(S') \overset{def}{=} S^i(S(S'))$, for all any CE-strategies $S(X)$ and S' in $\mathcal{C}$. A CE-strategy strategy is closed if all its fixed-point variables are bound.

Definition 8 (Semantics of CE-strategies). *The semantics of a closed CE-strategy S is a function $[\![S]\!]$ in $\mathfrak{F}(\mathcal{T} \cup \mathbb{F})$, which is defined inductively as follows.*

$$[\![\varnothing]\!](t) \overset{def}{=} \mathbb{F}.$$

$$[\![(u, s')]\!](t) \overset{def}{=} \begin{cases} [\![s']\!](t) & \textit{if } u \ll t, \\ \mathbb{F} & \textit{otherwise.} \end{cases} \qquad [\![@p.\boldsymbol{\tau}]\!](t) \overset{def}{=} \begin{cases} t[\boldsymbol{\tau}(t_{|p})]_p & \textit{if } p \in \mathcal{P}os(t), \\ \mathbb{F} & \textit{otherwise.} \end{cases}$$

$$[\![S_1 \oplus S_2]\!](t) \overset{def}{=} \begin{cases} [\![S_1]\!](t) & \textit{if } [\![S_1]\!](t) \neq \mathbb{F}, \\ [\![S_2]\!](t) & \textit{otherwise.} \end{cases} \qquad [\![\mu X.S(X)]\!](t) \overset{def}{=} [\![\bigoplus_{i=1,\delta(t)} S^i(\varnothing)]\!](t).$$

$$[\![@p.S]\!](t) \overset{def}{=} \begin{cases} t[[\![S]\!](t_{|p})]_p & \textit{if } [\![s]\!](t_{|p}) \neq \mathbb{F} \textit{ and } p \in \mathcal{P}os(t), \\ \mathbb{F} & \textit{otherwise.} \end{cases}$$

$$[\![\langle \bigsqcup_{i=1,n} @p_i.S_i \,|\, \phi \rangle]\!](t) \overset{def}{=} \begin{cases} \big(\eta([\![@p_n.S_n]\!]) \circ \cdots \circ \eta([\![@p_1.S_1]\!])\big)(t) & \textit{if } \mathcal{V}_{\mathfrak{f}}(S,t) \models \phi, \\ \mathbb{F} & \textit{otherwise,} \end{cases}$$

$$\textit{where } S = [@p_1.S_1, \cdots, @p_n.S_n], \textit{ and}$$
$$\mathcal{V}(S,t)(\widehat{p_i}) = \texttt{False} \textit{ iff } [\![@p_i.S_i]\!](t) = \mathbb{F}$$

For any CE-strategies S, S' in $\mathcal{C}$, we shall write $S \equiv S'$ iff $[\![S]\!] = [\![S']\!]$. To simplify the presentation, we shall write (u, S) instead of $(u \to u); S$ and we shall write $(u, \boldsymbol{\tau})$ instead of $u \to u[\boldsymbol{\tau}]$.

Example 2. We show how to encode some standard traversal strategies in our formalism using the fixed-point constructor. In what follows we assume that S is a CE-strategy. We recall that, when applied to a term t, the CE-strategy $\texttt{OneLeft}(S)$ tries to apply S to the subterm of t (if any) which is the closest to the root and on the far-left. The CE-strategy $\texttt{TopDown}(S)$ tries to apply S to the maximum of the sub-terms of t starting from the root of t, it stops when it is successfully applied. Hence,

$$\texttt{OneLeft}(S) = \mu X.\big(S \oplus \bigoplus_{f \in \mathcal{F},\, ar(f)=n} \big(f(x_1, \cdots, x_n), \bigoplus_{i=1,n} \langle [@i.X] \mid \bigvee_{i=1,n} \widehat{i} \rangle \big)\big),$$

$$\texttt{TopDown}(S) = \mu X.\big(S \oplus \bigoplus_{f \in \mathcal{F},\, ar(f)=n} \big(f(x_1, \cdots, x_n), \big\langle [@1.X, \cdots, @i.X] \mid \bigvee_{i=1,n} \widehat{i} \big\rangle\big)\big).$$

We generalize next the condition of well-foundedness from elementary CE-strategies to CE-strategies. Before that, it is helpful to view an CE-strategy as a tree with back-edges. A tree with back-edges is an oriented tree with possible edges going from a node to at most one of its ancestors in the tree.

Definition 9 (Well-founded CE-strategies.). *A CE-strategy S is well-founded iff*

i.) Every cycle in S passes through a position[3]*.*
ii.) All its sub-strategies of the form $\langle [@p_1.S_1, \ldots, @p_n.S_n, @q_1.\boldsymbol{\tau}_1, \ldots, @q_m, \boldsymbol{\tau}_m] \mid \phi \rangle$, where $n + m \geq 1$ and p_i, q_j are positions and $\boldsymbol{\tau}_i$ are tuples of contexts in $T^{\omega}_{\square}$ and S_i are CE-strategies, are subject to the following conditions:
   (a) $q_i \sqsubset q_j$, for all $i < j$, where $i, j \in [m]$, and
   (b) $p_i \parallel p_j$, for all $i \neq j$, where $i, j \in [n]$, and
   (c) $q_j \sqsubset p_i$, for all $j \in [m]$ and $i \in [n]$.

For instance the CE-strategy $\mu X.((f(x), \boldsymbol{\tau}) \oplus X)$ is not well-founded because the cycle that corresponds to the regeneration of the variable X does not cross a position delimiter, while the CE-strategy $\mu X.((f(x), \boldsymbol{\tau}) \oplus (@1.X))$ is well-founded. In all what follows we assume that the CE-strategies are well-founded. Notice that any CE-strategy is terminating. This is a direct consequence of Item (i) of the well-foundedness of CE-strategies, that is, every cycle in a well-founded CE-strategy passes through a position delimiter.

The set of Boolean formulas (resp. positions) of an CE-strategy S, will be denoted by $\Phi(S)$ (resp. $\mathcal{P}os(S)$). It is defined in a straightforward way.

[3] This constraint is similar to the one imposed on the modal μ-calculus formulas in which each cycle has to pass through a modality [1].

4.3 Canonical form of CE-strategies

Instead of the direct combination of CE-strategies, we shall first simplify the CE-strategies by turning each CE-strategy into an equivalent CE-strategy in the *canonical form*. A CE-strategy is in the canonical form if each of its Boolean formulas is a conjunction of position-variables, where each position-variable is in $\widehat{\mathbb{N}}_\epsilon$ instead of $\widehat{\mathbb{N}}_\epsilon^\omega$. The advantage of the use of canonical CE-strategies is that their combination is much simpler.

Definition 10 (Canonical form of CE-strategies). *An CE-strategy strategy S is in the canonical form iff any Boolean formula ϕ in $\Phi(S)$ is of the form $\phi = \bigwedge_i \widehat{p}_i$, where $\widehat{p}_i \in \widehat{\mathbb{N}}_\epsilon$. The set of CE-strategies in the canonical form is denoted by $\mathcal{C}^o$.*

It follows that if a CE-strategy strategy S is in the canonical form, then we have $\mathcal{P}os(S) \subset \mathbb{N}_\epsilon$.

Lemma 1. *Any CE-strategy can be turned into an equivalent CE-strategy in the canonical form.*

Proof. (Sketch) Firstly, we turn all the Boolean formulas of the CE-strategy into formulas in the disjunctive normal form. Then we express the disjunction in terms of the left-choice strategy. Secondly, we turn each position in $\mathbb{N}_\epsilon^\omega$ into a secession of positions in $\mathbb{N}_\epsilon$ by relying on the fact that the CE-strategy $@(ip).S$ is equivalent to $@i.(@p.S)$, where $i \in \mathbb{N}_\epsilon$ and $p \in \mathbb{N}_\epsilon^\omega$. $\square$

4.4 From CE-strategies to elementary CE-strategies

Out of a CE-strategy and a term it is possible to construct an elementary CE-strategy. The main purpose of this mapping is to formulate a correctness-completeness criterion for the unification and combination of CE-strategies in terms of elementary CE-strategies. Roughly speaking, this criterion imposes that the mapping of the combination of two CE-strategies is equivalent to the combination of their respective mapping. The definition of this mapping follows.

Definition 11. *Define the function $\Psi : \mathcal{C} \times \mathcal{T} \longrightarrow \mathcal{E}$, that associates to each closed CE-strategy S in $\mathcal{C}$ and a term t in $\mathcal{T}$ an elementary CE-strategy $\Psi(S,t)$ in $\mathcal{E}$ by*

$$\Psi(\varnothing, t) = \varnothing. \qquad \Psi(@p.\boldsymbol{\tau}, t) = @p.\boldsymbol{\tau}.$$

$$\Psi((u,\boldsymbol{\tau}),t) = \begin{cases} (\epsilon,\boldsymbol{\tau}) & \text{if } u \ll t, \\ \varnothing & \text{otherwise.} \end{cases} \qquad \Psi((u,S),t) = \begin{cases} \Psi(S,t) & \text{if } u \ll t, \\ \varnothing & \text{otherwise.} \end{cases}$$

$$\Psi(\langle \bigsqcup_{i\in[n]} @p_i.\boldsymbol{\tau}_i \mid \phi\rangle, t) = \bigsqcup_{i\in[n]} @p_i.\boldsymbol{\tau}_i. \qquad \Psi(@p.S,t) = @p \cdot \Psi(S, t_{|p}).$$

$$\Psi(S \oplus S', t) = \begin{cases} \Psi(S,t) & \text{if } \Psi(S,t) \neq \varnothing, \\ \Psi(S',t) & \text{otherwise.} \end{cases} \qquad \Psi(\mu X.S(X),t) = \Psi\big(\bigoplus_{i=1,\delta(t)} S^i(\varnothing), t\big).$$

If $S = \sqcup_{i\in[n]} @p_i.S_i$, *then*

$$\Psi\big(\langle S \mid \phi\rangle, t\big) = \begin{cases} \bigsqcup_{i\in[n]} @p_i.\eta(\Psi(S_i, t_{|p_i})) & \textit{if } \mathcal{V}(S,t) \models \phi, \\ \varnothing & \textit{otherwise.} \end{cases}$$

The application of the elementary CE*-strategy* $\Psi(S,t)$ *to the term* t *will be simply written as* $\Psi(S,t)(t)$ *instead of* $[\![\Psi(S,t)]\!](t)$.

It turns out that the function Ψ (Definition 11) preserves the semantics of CE-strategies in the following sense.

Lemma 2. *For any* CE*-strategy* S *in* $\mathcal{C}$ *and any term* t *in* $\mathcal{T}$, *we have* $[\![S]\!](t) = \Psi(S,t)(t)$.

The proof of this Lemma does not provide any difficulties since the definition of Ψ is close to the definition of the semantics of CE-strategies.

Lemma 3. *The function* Ψ *enjoys the following properties.*

i.) *For any elementary* CE*-strategies* E, E' *in* $\mathcal{E}$, *we have that* $E = E'$ *iff* $\Psi(E,t) = \Psi(E',t)$ *for any term* t.
ii.) *For any* CE*-strategies* S, S' *in* $\mathcal{C}$, *we have that* $S \equiv S'$ *iff* $\Psi(S,t) = \Psi(S',t)$ *for any term* t.

5 Unification and combination of CE-strategies

The problem of the closure of the class of CE-strategies under combination asks whether, for any two CE-strategies $S, S' \in \mathcal{C}$, there exists a third CE-strategy $S'' \in \mathcal{C}$ such that for every term $t \in \mathcal{T}$, we have that $\Psi(S,t) \curlyvee \Psi(S',t) = \Psi(S'',t)$. This can be understood as a correctness-completeness criterion of the combination of CE-strategies given in terms of the elementary CE-strategies. We define the combination of CE-strategies by means of their unification. Instead of unifying/combining CE-strategies directly, we unify/combine their canonical forms. We omit the symmetric cases in the following definition.

Definition 12 (Unification of canonical CE-strategies). *The unification of* CE*-strategies in the canonical form is a binary operation* $\curlywedge : \mathcal{C}^o \times \mathcal{C}^o \longrightarrow \mathcal{C}^o$ *inductively defined as follows.*

$$\begin{array}{ll} \varnothing \curlywedge S = \varnothing. & S \curlywedge \varnothing = \varnothing. \\ @i.\boldsymbol{\tau} \curlywedge @i.\boldsymbol{\tau}' = @i.(\boldsymbol{\tau}\cdot\boldsymbol{\tau}'). & @i.\boldsymbol{\tau} \curlywedge @j.\boldsymbol{\tau}' = [@i.\boldsymbol{\tau}, @j.\boldsymbol{\tau}'],\ \textit{if } j \sqsubset i. \\ @i.\boldsymbol{\tau} \curlywedge @i.S = @i.(@\epsilon.\boldsymbol{\tau} \curlywedge S). & @i.\boldsymbol{\tau} \curlywedge @j.S = [@i.\boldsymbol{\tau}, @j.S,],\ \textit{if } j \sqsubset i. \end{array}$$

$$\begin{array}{ll} (u,\boldsymbol{\tau}) \curlywedge @i.\boldsymbol{\tau}' = (u, @\epsilon.\boldsymbol{\tau} \curlywedge @i.\boldsymbol{\tau}'), & (u,\boldsymbol{\tau}) \curlywedge @i.\boldsymbol{\tau}' = \varnothing, \\ \qquad \textit{if } i \in [ar(u)] \cup \{\epsilon\}. & \qquad \textit{if } i \notin [ar(u)] \cup \{\epsilon\}. \\ @i.\boldsymbol{\tau} \curlywedge (u,S) = (u, (@i.\boldsymbol{\tau}) \curlywedge S), & @i.\boldsymbol{\tau} \curlywedge (u,S) = \varnothing, \\ \qquad \textit{if } [ar(u)] \cup \{\epsilon\}. & \qquad \textit{if } i \notin [ar(u)] \cup \{\epsilon\}. \\ (u,\boldsymbol{\tau}) \curlywedge (u',S') = (u \wedge u', (@\epsilon.\boldsymbol{\tau} \curlywedge S')). & (u,S) \curlywedge (u',S') = (u \wedge u', S \curlywedge S'). \end{array}$$

For the rest, assume $\mathcal{L} = \bigsqcup_{i\in\mathcal{I}} @i.S_i$ *and* $\mathcal{L}' = \bigsqcup_{j\in\mathcal{J}} @j.S'_j$.
Let $\mathcal{L}_1 = \bigsqcup_{i\in\mathcal{I}\cap\mathcal{J}} @i.(S_i \curlywedge S'_i)$ *and* $\mathcal{L}_2 = \bigsqcup_{i\in\mathcal{I}\setminus\mathcal{J}} @i.S_i$ *and* $\mathcal{L}_3 = \bigsqcup_{i\in\mathcal{J}\setminus\mathcal{I}} @i.S'_i$. *Define*

$$\langle \mathcal{L} \,|\, \phi\rangle \curlywedge \langle \mathcal{L}' \,|\, \phi'\rangle = \big\langle \mathcal{L}_1 \sqcup \mathcal{L}_2 \sqcup \mathcal{L}_3 \,|\, \phi \wedge \phi'\big\rangle. \quad (u, S) \curlywedge \langle \mathcal{L} \,|\, \phi\rangle = \big(u, S \curlywedge \langle \mathcal{L} \,|\, \phi\rangle\big).$$
$$(S_1 \oplus S_2) \curlywedge S = (S_1 \curlywedge S) \oplus (S_2 \curlywedge S).$$

For the fixed-point CE*-strategies,*

$$\mu X.S(X) \curlywedge \mu X'.S'(X') = \mu Z.S''\big(\mu X.S(X), \mu X'.S'(X'), Z\big),$$
$$\text{where } S''(X, X', Z) = [S(X) \curlywedge S'(X')]_{|X\curlywedge X':=Z}, \text{ and } Z \text{ is fresh.}$$
$$(\mu X.S(X)) \curlywedge S' = S''(\mu X.S(X)), \text{ where } S''(X) = S(X) \curlywedge S' \text{ and } S'.$$

Comments. We comment on the key points in Definition 12. The unification of (u, S) with (u', S') is naturally $(u \wedge u', S \curlywedge S')$ since we want to merge them. The idea behind the unification of $\mu X.S(X)$ with $\mu X'.S'(X')$ is to unfold $\mu X.S(X)$ (resp. $\mu X'.S'(X')$) to $S(\varnothing) \oplus S\big(\mu X.S(X)\big)$ (resp. $S'(\varnothing) \oplus S'(\mu X'.S'(X'))$) and to combine the resulting CE-strategy. This is achieved by firstly unifying $S(X)$ with $S'(X')$, where clearly the fixed-point variable X (resp. X') is free in $S(X)$ (resp. $S'(X')$). The resulting CE-strategy $S''(X, X', X \curlywedge X')$ contains three free fixed-point variables. The key point is to view $X \curlywedge X'$ as a fresh fixed-point variable, say Z, and to bind it to the full expression $S''(\mu X.S(X), \mu X'.S'(X'), Z)$, meaning that Z corresponds exactly to the CE-strategy that we are defining.

Example 3. Let $S(X) = (u, \boldsymbol{\tau}) \oplus @1.X$ and $S'(X') = (u', \boldsymbol{\tau}') \oplus @1.X'$, be two CE-strategies. We compute $\mu X.S(X) \curlywedge \mu X'.S'(X')$. Firstly, the unification $(*)$ of $S(X)$ and $S'(X')$ yields:

$$\begin{aligned}
(*) &= S(X) \curlywedge S'(X')\\
&= ((u, \boldsymbol{\tau}) \oplus @1.X) \curlywedge ((u', \boldsymbol{\tau}') \oplus @1.X')\\
&= ((u, \boldsymbol{\tau}) \curlywedge (u', \boldsymbol{\tau}')) \oplus (@1.X \curlywedge (u', \boldsymbol{\tau}')) \oplus ((u', \boldsymbol{\tau}) \curlywedge @1.X') \oplus (@1.X \curlywedge @1.X')\\
&= (u \wedge u', \boldsymbol{\tau}' \cdot \boldsymbol{\tau}) \oplus (u, [@1.X', @\epsilon.\boldsymbol{\tau}]) \oplus (u', [@1.X, @\epsilon.\boldsymbol{\tau}]) \oplus (@1.(X \curlywedge X')).
\end{aligned}$$

Hence, combination of $\mu X.S(X)$ and $\mu X'.S'(X')$ is

$$\begin{aligned}
(\mu X.S(X)) \curlywedge (\mu X'.S'(X')) = \mu Z.\big(&(u \wedge u', \boldsymbol{\tau}' \cdot \boldsymbol{\tau}) \oplus (u, [@1.(\mu X'.S'(X')), @\epsilon.\boldsymbol{\tau}]) \oplus\\
&(u', [@1.(\mu X.S(X)), @\epsilon.\boldsymbol{\tau}']) \oplus (@1.Z)\big).
\end{aligned}$$

Definition 13 (Combination of canonical CE-strategies). *The combination of CE-strategies in the canonical form is a binary operation* $\curlyvee : \mathcal{C}^o \times \mathcal{C}^o \longrightarrow \mathcal{C}^o$*, defined for any* S *and* S' *in* $\mathcal{C}^o$ *by* $S \curlyvee S' \stackrel{def}{=} (S \curlywedge S') \oplus S \oplus S'$.

The unification and combination of CE-strategies can defined in terms of their canonical form.

Definition 14 (Unification and combination of CE-strategies). *Let* S, S' *be two CE-strategies in* $\mathcal{C}$ *and* $\tilde{S}, \tilde{S}' \in \mathcal{C}^o$ *their canonical form, respectively. The unification (resp. combination) of* S *and* S' *is defined by* $S \curlywedge S' \stackrel{def}{=} \tilde{S} \curlywedge \tilde{S}'$ *(resp.* $S \curlyvee S' \stackrel{def}{=} \tilde{S} \curlyvee \tilde{S}'$*).*

5.1 The correction and completeness of the unification and combination of CE-strategies

Now we are ready to state the main results of this paper. Namely, the unification and combination of CE-strategies is sound and complete.

Theorem 1. *For every term $t \in \mathcal{T}$ and for every CE-strategies S and S' in the canonical form in $\mathcal{C}^o$, we have that $\Psi(S \curlywedge S', t) = \Psi(S, t) \curlywedge \Psi(S', t)$.*

Similarly, the combination of the CE-strategies is sound and complete. The following Theorem is a consequence of Theorem 1.

Theorem 2. *For every term $t \in \mathcal{T}$, for every CE-strategies S and S' in the canonical form in $\mathcal{C}^o$, we have that $\Psi(S \curlyvee S', t) = \Psi(S, t) \curlyvee \Psi(S', t)$.*

Since each CE-strategy can be turned into an equivalent CE-strategy in the canonical form (Lemma 1) and since the image of two equivalent CE-strategies under the homomorphism Ψ is identical (Item ii. of Lemma 3), then Theorems 1 and 2 hold for the class of CE-strategies as well.

Besides, thanks to the fact that the function Ψ is an homomorphism (in the first argument), one can transfer all the properties of the combination and unification of elementary CE-strategies (stated in Proposition 1) to CE-strategies.

Proposition 2. *The following hold.*

1. *The set $\mathcal{C}$ of CE-strategies together with the unification and combination operations enjoy the following properties.*
   - *(a) The neutral element of the unification and combination is $@\epsilon.\Box$.*
   - *(b) Every CE-strategy S is idempotent for the unification and combination, i.e. $S \curlywedge S = S$ and $S \curlyvee S = S$.*
   - *(c) The unification and combination of CE-strategies are associative.*
2. *The unification and combination of CE-strategies is non commutative.*
3. *For any CE-strategies S and S' in $\mathcal{C}$, and for any term t in $\mathcal{T}$, we have that*

$$\begin{array}{lcl} \Psi(S \curlywedge S', t) = \varnothing & \textit{iff} & \Psi(S, t) = \varnothing \;\; \textit{or} \;\; \Psi(S', t) = \varnothing. \\ \Psi(S \curlyvee S', t) = \varnothing & \textit{iff} & \Psi(S, t) = \varnothing \;\textit{and}\; \Psi(S', t) = \varnothing. \end{array}$$

4. *The unification and combination of CE-strategies is a congruence, that is, for any CE-strategies S_1, S_2, S in $\mathcal{C}$, we have that:*

$$\begin{array}{lll} \textit{If } S_1 \equiv S_2 & \textit{then} & S_1 \curlywedge S \equiv S_2 \curlywedge S \;\; \textit{and} \;\; S \curlywedge S_1 \equiv S \curlywedge S_2. \\ \textit{If } S_1 \equiv S_2 & \textit{then} & S_1 \curlyvee S \equiv S_2 \curlyvee S \;\; \textit{and} \;\; S \curlyvee S_1 \equiv S \curlyvee S_2. \end{array}$$

6 Conclusion and future work

We addressed the problem of extension and combination of proofs encountered in the field of computer aided asymptotic model derivation. We identified a class of rewriting strategies of which the operations of unification and combination were defined and proved correct. The design of this class is inspired by the formalism

μ-calculus. On the one hand the jumping into an immediate position of the tree together with a Boolean formula that specifies the failure are morally similar to the diamond and box modalities ($\langle\cdot\rangle$ and $[\cdot]$) of the propositionsal modal μ-calculus [1]. On the other hand we use of the fixed-point operator which is finer and more powerful than the `repeat` constructor used e.g. in [7].

The CE-strategies are indeed modular in the sense that they navigate in the tree without modifying it, then they insert contexts. This makes our formalism flexible since it allows one to modify and enrich the navigation part and/or the insertion part without disturbing the set-up.

Although the CE-strategies can be viewed as algebraic infinite trees [8], our technique of the unification and combination involving μ-terms and their unfolding is new. Therefore, we envision consequences of these results on the study of the syntactic (or modulo a theory) unification and the pattern-matching of infinite trees once the infinite trees are expressed as μ-terms in the same way we expressed the CE-strategies. Thus, a rewriting language that transforms algebraic infinite trees can be elaborated.

References

1. André Arnold and Damian Niwiński. *Rudiments of μ-calculus*. Studies in logic and the foundations of mathematics. London, Amsterdam, 2001.
2. Walid Belkhir, Alain Giorgetti, and Michel Lenczner. A symbolic transformation language and its application to a multiscale method. *Journal of Symbolic Computation*, 65:49–78, 2014.
3. Walid Belkhir, Nicolas Ratier, Duy Duc Nguyen, Bin Yang, Michel Lenczner, Frédéric Zamkotsian, and Horatiu Cirstea. Towards an automatic tool for multi-scale model derivation illustrated with a micro-mirror array. In *SYNASC 2015*, pages 47–54. IEEE Computer Society, 2015.
4. Christoph Benzmüller. A top-down approach to combining logics. In *ICAART*, pages 346–351. SciTePress Digital Library, 2013.
5. Jasmin Christian Blanchette and Nikolai Kosmatov, editors. *Tests and Proofs - 9th International Conference, TAP 2015*, volume 9154 of *LNCS*. Springer, 2015.
6. Nicolas Charalambakis. Homogenization techniques and micromechanics. A survey and perspectives. *Applied Mechanics Reviews*, 63(3):030803, 2010.
7. Horatiu Cirstea, Claude Kirchner, Luigi Liquori, and Benjamin Wack. Rewrite strategies in the rewriting calculus. *ENTCS*, 86(4):593 – 624, 2003. WRS'03.
8. Bruno Courcelle. Fundamental properties of infinite trees. *TCS*, 25(2):95 – 169, 1983.
9. Silvio Ghilardi and Luigi Santocanale. Algebraic and model theoretic techniques for fusion decidability in modal logics. In *LPAR'03*, volume 2850, pages 152–166.
10. M. Lenczner and R. C. Smith. A two-scale model for an array of AFM's cantilever in the static case. *Mathematical and Computer Modelling*, 46(5-6):776–805, 2007.
11. Zohar Manna and Calogero G. Zarba. Combining decision procedures. In *Formal Methods at the Crossroads. From Panacea to Foundational Support, 10th Anniversary Colloquium of UNU/IIST, Revised Papers*, pages 381–422, 2002.
12. Bin Yang, Walid Belkhir, and Michel Lenczner. Computer-aided derivation of multi-scale models: A rewriting framework. *International Journal for Multiscale Computational Engineering.*, 12(2):91–114, 2014.

Appendix: detailed proofs for reviewers

7 Proofs for preliminary section 2

If $\pi = \{\widehat{p}_1, \ldots, \widehat{p}_n\}$ is a set of variable-positions, then we shall write $\bigwedge \pi$ (resp. $\bigvee \pi$) for the Boolean formula $\widehat{p}_1 \wedge \ldots \wedge \widehat{p}_n$ (resp. $\widehat{p}_1 \vee \ldots \vee \widehat{p}_n$). In particular, $\bigwedge \emptyset = \bigvee \emptyset = \texttt{False}$.

Fact 3 *Let u, t be two terms and γ, γ' two substitutions. We have that, if $\gamma(u) \ll t$ and γ is subsumed by γ', then $\gamma'(u) \ll t$ as well.*

Lemma 4. *Let u, u', t be terms in $\mathcal{T}$. Then,*

$$(u \wedge u') \ll t \qquad \textit{iff} \qquad u \ll t \textit{ and } u' \ll t.$$

Proof. For the direction ($\Rightarrow$), let γ be the most general unifier of u and u', and α be a substitution such that $\alpha(u \wedge u') = t$. This means that $\alpha(\gamma(u)) = t$ and $\alpha(\gamma(u')) = t$. That is, $\gamma(u) \ll t$ and $\gamma(u') \ll t$.

For the direction ($\Leftarrow$), let σ and σ' be substitutions such that $\sigma(u) = t$ and $\sigma'(u') = t$. Consider the decomposition $\sigma = \sigma_1 \uplus \sigma_2$ and $\sigma' = \sigma'_1 \uplus \sigma'_2$, where $\mathsf{Dom}(\sigma_1) \cap \mathsf{Dom}(\sigma'_1) = \emptyset$ and $\mathsf{Dom}(\sigma_2) = \mathsf{Dom}(\sigma'_2)$. Since $\sigma(u) = \sigma'(u')$, it follows that $\sigma_2 = \sigma'_2$. But this means that $\sigma_2(u) = \sigma_2(u')$, and $\sigma_2(u) \ll t$. In other words, u and u' can be unified. Let γ be the most general unifier of u and u'. But since $\sigma_2(u) \ll t$ and σ_2 is subsumed by γ, then it follows from Fact 3 that $\gamma(u) \ll t$. $\square$

8 Proofs and formal definitions for section 4

8.1 CE-strategies as tree-with back edges

It is helpful to view CE-strategies as trees with back-edges. For instance, the tree-like structure of the CE-strategy $S(X) = (f(x), \boldsymbol{\tau}) \oplus (@1.X)$, where the fixed-point variable X is free, is depicted on the left of Figure 4. While the tree with- back-edge related to $\mu X.S(X)$ is depicted on the right.

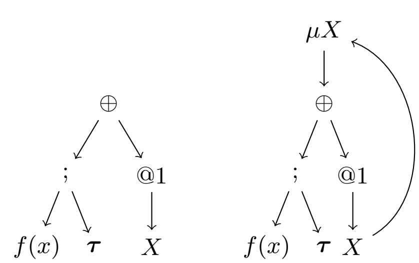

Fig. 4. The tree-like structure of the CE-strategy $S(X) = (f(x), \boldsymbol{\tau}) \oplus (@1.X)$ (left) and $\mu X.S(X)$ (right).

8.2 Set of Boolean formulas and positions of a CE-strategy

Definition 15 (Set of Boolean formulas and positions of a CE-strategy). *The set of Boolean formulas (resp. positions) of an CE-strategy S, denoted by $\Phi(S)$ (resp. $\mathcal{P}os(S)$), is inductively defined by*

$$
\begin{aligned}
\Phi(@p.\boldsymbol{\tau}) &= \emptyset & \mathcal{P}os(@p.\boldsymbol{\tau}) &= \{p\} \\
\Phi((u,\boldsymbol{\tau})) &= \emptyset & \mathcal{P}os((u,\boldsymbol{\tau})) &= \emptyset \\
\Phi(X) &= \emptyset & \mathcal{P}os(X) &= \emptyset \\
\Phi((u,S)) &= \Phi(S) & \mathcal{P}os((u,S)) &= \mathcal{P}os(S) \\
\Phi(\bigoplus_{i\in[n]} S_i) &= \bigcup_{i\in[n]} \Phi(S_i) & \mathcal{P}os(\bigoplus_{i\in[n]} S_i) &= \bigcup_{i\in[n]} \mathcal{P}os(S_i) \\
\Phi(\langle \bigsqcup_{i\in[n]} @p_i.S_i \mid \phi\rangle) &= \{\phi\} \cup \bigcup_{i\in[n]} \Phi(S_i) & \mathcal{P}os(\langle \bigsqcup_{i\in[n]} @p_i.S_i \mid \phi\rangle) &= \bigcup_{i\in[n]} \{p_i\} \cup \mathcal{P}os(S_i) \\
\Phi(\mu X.S(X)) &= \Phi(S(X)) & \mathcal{P}os(\mu X.S(X)) &= \mathcal{P}os(S(X))
\end{aligned}
$$

8.3 Depth of CE-strategies

Taking into account that the structure of a CE-strategy is no longer a tree but a tree with back-edges that may contains cycles, we slightly modify the standard measure of the depth of trees in order to capture both the number of nested loops, caused by the nested application of the constructor μ, and the distance from the root of the tree to the leaves. Many proofs will be done by induction with respect to this measure.

Definition 16 (Depth of a CE-strategy). *The depth of an CE-strategy is function $\Delta : \mathcal{C} \longrightarrow \mathbb{N} \times \mathbb{N}$ defined inductively as follows.*

$$
\begin{aligned}
\Delta(@p.\boldsymbol{\tau}) &= (0,0) \\
\Delta((u,\boldsymbol{\tau})) &= (0,0) \\
\Delta(X) &= (0,0) \\
\Delta((u,S)) &= (0,1) + \Delta(S) \\
\Delta(S_1 \oplus \ldots \oplus S_n) &= (0,1) + max(\Delta(S_1),\ldots,\Delta(S_n)) \\
\Delta(\langle @p_n.S_1,\cdots,@p_n.S_n \mid \phi\rangle) &= (0,1) + max(\Delta(S_1),\ldots,\Delta(S_n)) \\
\Delta(\mu X.S(X)) &= (1,0) + \Delta(S(X))
\end{aligned}
$$

We shall denoted by $<, \leq$ and $>, \geq$ the related lexicographic orders on $\mathbb{N} \times \mathbb{N}$.

Notice that if a CE-strategy S is iteration-free, i.e. it does not contain the constructor μ, then its depth $\Delta(S) = (0,n)$, for some $n \in \mathbb{N}$.

8.4 Canonical form of CE-strategies

Lemma 5. *(i.e. Lemma 1) Any CE-strategy can be turned into an equivalent CE-strategy in the canonical form.*

Proof. Firstly, we turn all the Boolean formulas of the CE-strategy into formulas in the disjunctive normal form. Then we express the disjunction in terms of the left-choice strategy. (Lemmas 6 and 7). Thus we obtain an equivalent CE-strategy in which all the Boolean formulas are conjunctions of position-variables. Secondly, we turn each position in $\mathbb{N}_\epsilon^\omega$ into a secession of positions in $\mathbb{N}_\epsilon$ (Lemma 8) by relying on the fact that the CE-strategy $@(ip).S$ is equivalent to $@i.(@p.S)$, where $i \in \mathbb{N}_\epsilon$ and $p \in \mathbb{N}_\epsilon^\omega$. □

Lemma 6. *Let $p_1, \dots, p_n$ be parallel positions in $\mathbb{N}_\epsilon^\omega$, and $S_1, \dots, S_n$ be CE-strategies, with $n \geq 1$. Let $\pi, \pi' \subseteq \{\widehat{p}_1, \dots, \widehat{p}_n\}$ with $\pi \cup \pi' = \{\widehat{p}_1, \dots, \widehat{p}_n\}$ and let $\phi = \bigwedge \pi$ and $\phi' = \bigwedge \pi'$ be Boolean formulas. Let $\mathcal{S} = [@p_1.S_1, \dots, @p_n.S_n]$. Then we have the equivalence*

$$\begin{aligned}\langle \mathcal{S} \mid \phi \vee \phi' \rangle \equiv & \langle \mathcal{S} \mid \phi \wedge \phi' \rangle \oplus \\ & \bigoplus_{\substack{\wp' \subset \pi' \\ \wp' = |\pi'|-1}} \langle \mathcal{S}_{|\pi \cup \wp'} \mid \phi \wedge {\phi'}_{|\wp'} \rangle \oplus \cdots \oplus \bigoplus_{\substack{\wp' \subset \pi' \\ |\wp'| = 1}} \langle \mathcal{S}_{|\pi \cup \wp'} \mid \phi \wedge {\phi'}_{|\wp'} \rangle \oplus \\ & \bigoplus_{\substack{\wp \subset \pi \\ |\wp| = |\pi|-1}} \langle \mathcal{S}_{|\wp \cup \pi'} \mid \phi_{|\wp} \wedge \phi' \rangle \oplus \cdots \oplus \bigoplus_{\substack{\wp \subset \pi \\ |\wp| = 1}} \langle \mathcal{S}_{|\wp \cup \pi'} \mid \phi_{|\wp} \wedge \phi' \rangle \end{aligned} \tag{1}$$

Proof. Recall that

$$[\![\langle \mathcal{S} \mid \phi \vee \phi' \rangle]\!](t) = \begin{cases} (\eta([\![@p_n.S_n]\!]) \circ \cdots \circ \eta([\![@p_1.S_1]\!])(t) & \text{if } \mathcal{V}(\mathcal{S}, t) \models \phi \vee \phi', \\ \mathbb{F} & \text{otherwise.} \end{cases}$$

We discuss four cases depending on whether $\mathcal{V}(\mathcal{S}, t) \models \phi$ or $\mathcal{V}(\mathcal{S}, t) \models \phi'$.

1. If $\mathcal{V}(\mathcal{S}, t) \models \phi$ and $\mathcal{V}(\mathcal{S}, t) \models \phi'$, then in this case

$$\mathcal{V}(\mathcal{S}, t) \models \phi \wedge \phi' \qquad \text{and} \qquad [\![\langle \mathcal{S} \mid \phi \vee \phi' \rangle]\!](t) = [\![\langle \mathcal{S} \mid \phi \wedge \phi' \rangle]\!](t).$$

   Thus Eq. (1) holds.
2. If $\mathcal{V}(\mathcal{S}, t) \models \phi$ and $\mathcal{V}(\mathcal{S}, t) \not\models \phi'$, then we must show that

$$[\![\langle \mathcal{S} \mid \phi \wedge \phi' \rangle]\!](t) = \mathbb{F}, \tag{2}$$

and

$$\exists! \wp' \subset \pi', \qquad [\![\langle \mathcal{S} \mid \phi \vee \phi' \rangle]\!](t) = [\![\langle \mathcal{S}_{|\pi \cup \wp'} \mid \phi \wedge {\phi'}_{|\wp'} \rangle]\!](t), \tag{3}$$

and

$$\forall \varrho' \subset \pi', \text{ where } |\varrho'| \geq |\wp'| \text{ and } \varrho' \neq \wp', \qquad [\![\langle \mathcal{S}_{|\pi \cup \varrho'} \mid \phi \wedge {\phi'}_{|\varrho'} \rangle]\!](t) = \mathbb{F}. \tag{4}$$

   However, Eq. (2) follows from the fact that $\mathcal{V}(\mathcal{S}, t) \models \phi$ and $\mathcal{V}(\mathcal{S}, t) \not\models \phi'$.

To prove Eq. (3), we let

$$\wp' \stackrel{def}{=} \{\widehat{p}_i \in \pi \;\mid\; \mathcal{V}(S,t)(\widehat{p}_i) = \texttt{True}\} \cap \pi'.$$

Hence, $\mathcal{V}(\mathcal{S},t) \models \phi \vee \phi'$ if and only if $\mathcal{V}(\mathcal{S}_{|\pi\cup\wp'},t) \models \phi \wedge \phi'_{|\wp'}$. Besides,

$$\forall \widehat{p}_i \in \pi \cup \pi', \quad \eta([\![@p_i.S_i]\!])(t) = \begin{cases} [\![@p_i.S_i]\!](t) & \text{if } \mathcal{V}(\mathcal{S},t)(\widehat{p}_i) = \texttt{True} \\ t & \text{otherwise,} \end{cases}$$

and

$$\forall \widehat{p}_i \in \pi \cup \wp', \quad \eta([\![@p_i.S_i]\!])(t) = [\![@p_i.S_i]\!](t) \quad \text{and} \quad \mathcal{V}(\mathcal{S}_{|\pi\cup\wp'},t)(\widehat{p}_i) = \texttt{True}.$$

Summing up, Eq. (3) holds.
To prove Eq. (4), we notice that there exists $\widehat{p} \in \varrho'$ such that $\mathcal{V}(S,t)(\widehat{p}) = \texttt{False}$, and hence $\mathcal{V}(S_{\pi\cup\varrho'},t) \not\models \phi'_{|\varrho'}$, making $[\![\langle \mathcal{S}_{|\pi\cup\varrho'} \mid \phi \wedge \phi'_{|\varrho'}\rangle]\!](t) = \mathbb{F}$. Thus Eq. (1) holds.

3. If $\mathcal{V}(\mathcal{S},t) \not\models \phi$ and $\mathcal{V}(\mathcal{S},t) \models \phi'$, then this case is similar to the case when $\mathcal{V}(\mathcal{S},t) \models \phi$ and $\mathcal{V}(\mathcal{S},t) \not\models \phi'$ discussed above in Item 2.
4. If $\mathcal{V}(\mathcal{S},t) \not\models \phi$ and $\mathcal{V}(\mathcal{S},t) \not\models \phi'$, then in this case

$$\begin{aligned} [\![\langle \mathcal{S} \mid \phi \vee \phi'\rangle]\!](t) &= \mathbb{F}, \quad \text{and} \\ \forall \wp' \subset \pi', [\![\langle \mathcal{S}_{|\pi\cup\wp'} \mid \phi \wedge \phi'_{|\wp'}\rangle]\!](t) &= \mathbb{F}, \quad \text{and } \forall \wp \subset \pi, [\![\langle \mathcal{S}_{|\wp\cup\pi'} \mid \phi_{|\wp} \wedge \phi'\rangle]\!](t) = \mathbb{F}, \end{aligned}$$

making the Eq. (1) hold.

$\square$

Lemma 7. *Let $p_1, \ldots, p_n$ be parallel positions in $\mathbb{N}_\epsilon^\omega$, and $S_1, \ldots, S_n$ be CE-strategies, with $n \geq 1$. Let $\pi = \{\widehat{p}_1, \ldots, \widehat{p}_n\}$ and Let $\mathcal{S} = [@p_1.S_1, \ldots, @p_n.S_n]$. Then we have the equivalence*

$$\langle \mathcal{S} \mid \bigvee \pi\rangle \equiv \Big(\bigoplus_{\substack{\wp\subseteq\pi \\ |\wp|=|\pi|}} \langle \mathcal{S}_{|\wp} \mid \bigwedge \wp\rangle \Big) \oplus \cdots \oplus \Big(\bigoplus_{\substack{\wp\subset\pi \\ |\wp|=0}} \langle \mathcal{S}_{|\wp} \mid \bigwedge \wp\rangle \Big) \tag{5}$$

Proof. We recall that

$$\langle \mathcal{S} \mid \bigvee \pi\rangle(t) = \begin{cases} \big(\eta([\![@p_n.S_n]\!]) \circ \cdots \circ \eta([\![@p_1.S_1]\!])\big)(t) & \text{if } \mathcal{V}(\mathcal{S},t) \models \bigvee \pi, \\ \mathbb{F} & \text{otherwise.} \end{cases}$$

Out of the valuation $\mathcal{V}(\mathcal{S},t)$, we shall show that there exists a unique $\wp \subseteq \pi$ such that

$$[\![\langle \mathcal{S} \mid \bigvee \pi\rangle]\!](t) = [\![\langle \mathcal{S}_{|\wp} \mid \bigwedge \wp\rangle]\!](t), \tag{6}$$

and that for all $\wp' \subseteq \pi$ where $|\wp'| \geq |\wp|$ and $\wp' \neq \wp$, we have that

$$[\![\langle \mathcal{S}_{|\wp'} \mid \bigwedge \wp'\rangle]\!](t) = \mathbb{F}. \tag{7}$$

For this purpose, we define $\wp$ by

$$\wp \stackrel{def}{=} \{\widehat{p} \in \pi \;\mid\; \mathcal{V}(\mathcal{S}, t) = \texttt{True}\}.$$

Therefore, $\mathcal{V}(\mathcal{S}, t) \models \bigvee \pi$ iff $\mathcal{V}(\mathcal{S}_{|\wp}, t) \models \bigwedge \wp$, and

$$\forall \widehat{p_i} \in \pi, \qquad \eta([\![@p_i.S_i]\!])(t) = \begin{cases} [\![@p_i.S_i]\!](t) & \text{if } \mathcal{V}(\mathcal{S}, t) \models \bigvee \pi \\ t & \text{otherwise} \end{cases}$$

and

$$\forall \widehat{p_i} \in \wp, \mathcal{V}(\mathcal{S}_{|\wp}, t) \models \bigwedge \wp \text{ and } \quad \eta([\![@p_i.S_i]\!])(t) = [\![@p_i.S_i]\!](t).$$

Hence Eq. (6) holds. And Eq. (7) follows from the fact that there exists $\widehat{q} \in \wp'$ such that $\mathcal{V}(\mathcal{S}, t)(\widehat{q}) = \texttt{False}$, thus $\mathcal{V}(\mathcal{S}_{|\wp'}, t)(\widehat{q}) = \texttt{False}$ and $\mathcal{V}(\mathcal{S}_{|\wp'}, t) \not\models \bigwedge \wp'$. □

Lemma 8. *Each CE-strategy in which every Boolean formulas is a conjunction of position-variables in $\widehat{\mathbb{N}}_\epsilon^\omega$, can be turned into an equivalent CE-strategy in which every Boolean formulas is a conjunction of position-variables in $\widehat{\mathbb{N}}_\epsilon$.*

Proof. Let S be CE-strategy strategy. The idea is simple. If there are no Boolean formulas in the CE-strategy, then we rely on the observation that the CE-strategy $@(ip).S'$ is equivalent to $@i.(@p.S')$ where $i \in \mathbb{N}_\epsilon$ and $p \in \mathbb{N}_\epsilon^\omega$. Which means that we use the reduction rule

$$@(ip).S' \quad \rightharpoondown \quad @i.(@p.S') \tag{8}$$

to put the CE-strategy in the canonical form. We generalize the rule (8) to take into account the presence of Boolean formulas as follows. Let

$$\begin{aligned} \mathcal{S} &= \big(\bigsqcup_j @\widehat{1p_j}.S_j^1\big) \sqcup \cdots \sqcup \big(\bigsqcup_j @\widehat{np_j}.S_j^n\big), \text{ and} \\ \mathcal{S}_1 &= @\widehat{1}.\big\langle\big(\bigsqcup_j @\widehat{p_j}.S_j^1\big) \mid \bigwedge_j \widehat{p_j}\big\rangle, \text{ and} \\ \mathcal{S}_n &= @\widehat{n}.\big\langle\big(\bigsqcup_j @\widehat{p_j}.S_j^1\big) \mid \bigwedge_j \widehat{p_j}\big\rangle \end{aligned}$$

Then we define the reduction rule

$$\Big\langle \mathcal{S} \mid \bigwedge_i \bigwedge_j \widehat{ip_j} \Big\rangle \quad \rightharpoondown \quad \big\langle [S_1, \ldots, S_n] \mid \bigwedge_i \widehat{i} \big\rangle.$$

Since all the Boolean formulas are conjunctions of position-variables, then we have

$$\Big\langle \mathcal{S} \mid \bigwedge_i \bigwedge_j \widehat{ip_j} \Big\rangle \equiv \big\langle [S_1, \ldots, S_n] \mid \bigwedge_i \widehat{i} \big\rangle.$$

□

8.5 Properties of the function Ψ

Lemma 9 (Ψ preserves the semantics, i.e. Lemma 2). *For any CE-strategy S in $\mathcal{C}$ and any term t in $\mathcal{T}$,*

$$[\![S]\!](t) = \Psi(S,t)(t) \tag{9}$$

Proof. The proof is by induction on $\Delta(S)$, the depth of S.

Basic case: $\Delta(S) = (0,0)$. We distinguish three cases depending on S.

1. If $S = \varnothing$, then this case is trivial.
2. If $S = @p.\boldsymbol{\tau}$. This case is trivial since $\Psi(S,t) \stackrel{def}{=} S$.
3. If $S = (u, \boldsymbol{\tau})$. In this case

$$[\![S]\!](t) = \begin{cases} \boldsymbol{\tau}[t] & \text{if } u \ll t \\ \mathbb{F} & \text{otherwise,} \end{cases}$$

and on the other hand,

$$\Psi(S,t) = \begin{cases} @\epsilon.\boldsymbol{\tau} & \text{if } u \ll t \\ \varnothing & \text{otherwise} \end{cases} \quad \text{hence} \quad \Psi(S,t)(t) = \begin{cases} \boldsymbol{\tau}[t] & \text{if } u \ll t \\ \mathbb{F} & \text{otherwise} \end{cases}$$

That is, $[\![S]\!](t) = \Psi(S,t)(t)$.

Induction case: $\Delta(S) > (0,0)$. We distinguish three cases depending on S.

1. If S is a left-choice of the form

$$S = S_1 \oplus S_2$$

then,

$$[\![S]\!](t) = \begin{cases} [\![S_1]\!](t) & \text{if } [\![S_1]\!](t) \neq \mathbb{F}, \\ [\![S_2]\!](t) & \text{otherwise.} \end{cases}$$

and

$$\Psi(S_1 \oplus S_2, t) \stackrel{def}{=} \begin{cases} \Psi(S_1,t) & \text{if } \Psi(S_1,t) \neq \varnothing, \\ \Psi(S_2,t) & \text{otherwise.} \end{cases}$$

Since $\Psi(S_1,t) = \varnothing$ iff $\Psi(S_1,t)(t) = \mathbb{F}$, we get

$$\Psi(S_1 \oplus S_2, t)(t) \stackrel{def}{=} \begin{cases} \Psi(S_1,t)(t) & \text{if } \Psi(S_1,t)(t) \neq \mathbb{F}, \\ \Psi(S_2,t)(t) & \text{otherwise.} \end{cases}$$

From the induction hypothesis we have that $[\![S_i]\!](t) = \Psi(S_i,t)(t)$ for $i = 1,2$. Hence, $[\![S_1 \oplus S_2]\!](t) = \Psi(S_1 \oplus S_2, t)(t)$.

2. If S is of the form

$$S = \langle [@p_1.S_1, \ldots, @p_n.S_n, @q_1.\boldsymbol{\tau}_1, \ldots, @q_m.\boldsymbol{\tau}_m] \mid \phi \rangle, \quad n \geq 1, m \geq 0,$$

then let

$$f = \eta([\![@p_n.S_n]\!]) \circ \cdots \circ \eta([\![@p_1.S_1]\!]) \quad \text{and} \quad f' = [\![@q_m.\boldsymbol{\tau}_m]\!] \circ \cdots \circ [\![@q_1.\boldsymbol{\tau}_1]\!]$$

On the one hand

$$\begin{aligned} [\![S]\!](t) &\stackrel{def}{=} \begin{cases} (f' \circ f)(t) & \text{if } \mathcal{V}(S,t) \models \phi \\ \mathbb{F} & \text{otherwise} \end{cases} \\ &= \begin{cases} \big(\eta([\![@p_n.S_n]\!]) \circ \cdots \circ \eta([\![@p_1.S_1]\!])\big)(t) & \text{if } \mathcal{V}(S,t) \models \phi \\ \mathbb{F} & \text{otherwise} \end{cases} \end{aligned}$$

On the other hand, let

$$\begin{aligned} \mathcal{L} &= [\eta(@p_1.\Psi(S_1, t_{|p_1})), \ldots, \eta(@p_n.\Psi(S_n, t_{|p_n}))], \\ &\text{and} \\ \mathcal{L}' &= [@q_1.\boldsymbol{\tau}_1, \ldots, @q_m.\boldsymbol{\tau}_m]. \end{aligned}$$

Thus

$$\Psi(S,t) \stackrel{def}{=} \begin{cases} \mathcal{L} \sqcup \mathcal{L}' & \text{if } \mathcal{V}(S,t) \models \phi, \\ \varnothing & \text{otherwise.} \end{cases}$$

Hence,

$$\Psi(S,t)(t) = \begin{cases} ([\![\mathcal{L}']\!] \circ [\![\mathcal{L}]\!])(t) & \text{if } \mathcal{V}(S,t) \models \phi, \\ \mathbb{F} & \text{otherwise.} \end{cases}$$

It remains to show that, for any term t in $\mathcal{T}$,

$$f(t) = [\![\mathcal{L}]\!](t) \qquad \text{and} \qquad f'(t) = [\![\mathcal{L}']\!](t).$$

But $f' = [\![\mathcal{L}']\!]$, and thus it remains to show that

$$\forall i \in [n], \qquad [\![@p_i.S_i]\!](t) = [\![@p_i.\Psi(S_i, t_{|p_i})]\!](t). \tag{10}$$

However,

$$[\![@p_i.S_i]\!](t) = \begin{cases} t\big[[\![S_i]\!](t_{|p_i})\big]_{p_i} & \text{if } p_i \in \mathcal{P}os(t) \\ \mathbb{F} & \text{otherwise} \end{cases}$$

and

$$[\![@p_i.\Psi(S_i, t_{|p_i})]\!](t) = \begin{cases} t\big[\Psi(S_i, t_{|p_i})(t_{|p_i})\big]_{p_i} & \text{if } p_i \in \mathcal{P}os(t) \\ \mathbb{F} & \text{otherwise} \end{cases}$$

From the induction hypothesis we have $[\![S_i]\!](t_{|p_i}) = \Psi(S_i, t_{|p_i})(t_{|p_i})$. Therefore, the Eq. (10) holds.

3. If S is of the form $S = \mu X.S(X)$, then the claims follows from the fact that

$$[\![\mu X.S(X)]\!](t) = [\![\bigoplus_{i=1,\delta(t)} S^i(\varnothing)]\!](t) \quad \text{and} \quad \Psi(\mu X.S(X), t) = \Psi(\bigoplus_{i=1,\delta(t)} S^i(\varnothing), t),$$

   by applying the induction hypothesis, since

$$\Delta(\bigoplus_{i=1,\delta(t)} S^i(\varnothing)) < \Delta\big(\mu X.S(X)\big),$$

   because if $\Delta\big(\bigoplus_{i=1,\delta(t)} S^i(\varnothing)\big) = (n, m)$, for some $n, m \in \mathbb{N}$, then $\Delta\big(\mu X.S(X)\big) = (n+1, m')$, for some $m' > m$.

This ends the proof of Lemma 2. □

Lemma 10 (i.e. Lemma 3). *The function Ψ enjoys the following properties.*

i.) For any elementary CE*-strategies E, E' in $\mathcal{E}$, we have that*

$$E = E' \qquad \textit{iff} \qquad \Psi(E,t) = \Psi(E',t),$$

for any term t.

ii.) For any CE*-strategies S, S' in $\mathcal{C}$, we have that*

$$S \equiv S' \qquad \textit{iff} \qquad \Psi(S,t) = \Psi(S',t),$$

for any term t.

Proof. We only prove Item *ii.)*, the other item follows immediately from the definition of Ψ. On the one hand, from the definition of $\equiv$ we have that

$$S \equiv S' \qquad \text{iff} \qquad [\![S]\!](t) = [\![S']\!](t), \;\; \forall t \in \mathcal{T}.$$

However, it follows from Lemma 2 that

$$[\![S]\!](t) = \Psi(S,t)(t) \qquad \text{and} \qquad [\![S']\!](t) = \Psi(S',t)(t).$$

Therefore,

$$\Psi(S,t)(t) = \Psi(S',t)(t), \forall t \in \mathcal{T}.$$

Since, both $\Psi(S,t)$ and $\Psi(S',t)$ are elementary CE-strategies, it follows from Item i.) of this Lemma that $\Psi(S,t) = \Psi(S',t)$.

9 Proofs for section 5

9.1 Correctness and Completeness of the unification and combination of CE-strategies.

Theorem 4 (i.e. Theorem 1). *For every term $t \in \mathcal{T}$, for every CE-strategies S and S' in the canonical form in $\mathcal{C}^o$, we have that*

$$\Psi(S \curlywedge S', t) = \Psi(S, t) \curlywedge \Psi(S', t)$$

Proof. The proof is by a double induction on $\Delta(S)$ and $\Delta(S')$. We recall that if there are two symmetric cases, we only prove one of them. We make an induction on $\Delta(S)$.

<u>**Base case:** $\Delta(S) = (0, 0)$</u>**.** We make an induction on $\Delta(S')$.

<u>**Base case:** $\Delta(S') = (0, 0)$</u>**.** We distinguish three cases depending on the structure of S and S'.

1. The cases when $(S, S') = (\varnothing, \varnothing)$ or $(S, S') = (@i.\boldsymbol{\tau}, @j.\boldsymbol{\tau}')$ are trivial whether $i = j$ or not.

2. If $(S, S') = \big(@i.\boldsymbol{\tau}, (u, \boldsymbol{\tau}')\big)$, where $i \in \mathbb{N}_\epsilon \setminus \{\epsilon\}$, then in this case

$$S \curlywedge S' = (u, [@i.\boldsymbol{\tau}, @\epsilon.\boldsymbol{\tau}']) \text{ and}$$
$$\Psi(S \curlywedge S', t) = \begin{cases} [@i.\boldsymbol{\tau}, @\epsilon.\boldsymbol{\tau}'] & \text{if } u \ll t \\ \varnothing & \text{otherwise.} \end{cases}$$

On the other hand,

$$\Psi(S, t) = \begin{cases} @\epsilon.\boldsymbol{\tau} & \text{if } u \ll t \\ \varnothing & \text{otherwise.} \end{cases} \qquad \text{and} \qquad \Psi(S', t) = @i.\boldsymbol{\tau}'$$

Hence

$$\begin{aligned} \Psi(S, t) \curlywedge \Psi(S', t) &= \begin{cases} [@i.\boldsymbol{\tau}, @\epsilon.\boldsymbol{\tau}'] & \text{if } u \ll t \\ \varnothing & \text{otherwise.} \end{cases} \\ &= \Psi(S, t) \curlywedge \Psi(S', t) \end{aligned}$$

3. If $S = @\epsilon.\boldsymbol{\tau}$ and $S' = (u, \boldsymbol{\tau}')$, then this case is similar to the previous one except that the insertion of the tuples of the contexts $\boldsymbol{\tau}$ and $\boldsymbol{\tau}'$ occurs at the root position instead of two different positions. We have that

$$S \curlywedge S' = (u, \boldsymbol{\tau}' \cdot \boldsymbol{\tau}) \quad \text{and} \quad \Psi(S \curlywedge S', t) = \begin{cases} @\epsilon.(\boldsymbol{\tau}' \cdot \boldsymbol{\tau}) & \text{if } u \ll t \\ \varnothing & \text{otherwise.} \end{cases}$$

On the other hand,

$$\Psi(S,t) = \begin{cases} @\epsilon.\boldsymbol{\tau} & \text{if } u \ll t \\ \varnothing & \text{otherwise.} \end{cases} \qquad \text{and} \qquad \Psi(S',t) = @\epsilon.\boldsymbol{\tau}'$$

Hence

$$\begin{aligned} \Psi(S,t) \curlywedge \Psi(S',t) &= \begin{cases} @\epsilon.(\boldsymbol{\tau}' \cdot \boldsymbol{\tau}) & \text{if } u \ll t \\ \varnothing & \text{otherwise.} \end{cases} \\ &= \Psi(S,t) \curlywedge \Psi(S',t) \end{aligned}$$

Induction step: $\Delta(S') > (0,0)$**.** We distinguish six cases depending on the structure of S and S'.

1. If $S = (@i, \boldsymbol{\tau})$ and $S' = (u', R')$, where $i \in \mathbb{N}_\epsilon$, then in this case

$$S \curlywedge S' = \big(u', (@i.\boldsymbol{\tau}) \curlywedge R'\big),$$

and

$$\Psi(S \curlywedge S', t) = \begin{cases} \Psi\big((@i.\boldsymbol{\tau}) \curlywedge R', t\big) & \text{if } u' \ll t, \\ \varnothing & \text{otherwise.} \end{cases}$$

Since $\Delta(R') < \Delta(S')$, it follows from the induction hypothesis that

$$\begin{aligned} \Psi(S \curlywedge S', t) &= \begin{cases} \Psi(@i.\boldsymbol{\tau}, t) \curlywedge \Psi(R', t\big) & \text{if } u' \ll t \\ \varnothing & \text{otherwise.} \end{cases} \\ &= \begin{cases} @i.\boldsymbol{\tau} \quad \curlywedge \Psi(R', t\big) & \text{if } u' \ll t \text{ and } i \in \mathcal{P}os(t) \\ \varnothing & \text{otherwise.} \end{cases} \end{aligned}$$

On the other hand,

$$\Psi(S,t) = \begin{cases} @i.\boldsymbol{\tau} & \text{if } i \in \mathcal{P}os(t) \\ \varnothing & \text{otherwise.} \end{cases} \quad \text{and} \quad \Psi(S',t) = \begin{cases} \Psi(R',t) & \text{if } u' \ll t \\ \varnothing & \text{otherwise} \end{cases}$$

Hence the unification $\Psi(S,t) \curlywedge \Psi(S',t)$ is defined by

$$\begin{aligned} \Psi(S,t) \curlywedge \Psi(S',t) &= \begin{cases} @i.\boldsymbol{\tau} \curlywedge \Psi(R',t) & \text{if } \Psi(R',t\big) \text{ and } u' \ll t \\ \varnothing & \text{otherwise} \end{cases} \\ &= \Psi(S,t) \curlywedge \Psi(S',t). \end{aligned}$$

2. If $S = (@i, \boldsymbol{\tau})$ and $S' = \langle \bigsqcup_{j \in J} @j.S_j \mid \phi \rangle$, where $i \in \mathbb{N}_\epsilon$, then we only discuss the case when $i \in J$, the case when $i \notin I$ is immediate. In this case, let

$$\mathcal{S} = \bigsqcup_{j \in J \setminus \{i\}} @j.S_j \sqcup (@i.\boldsymbol{\tau} \curlywedge @i.S_i)$$

and

$$S \curlywedge S' = \langle \mathcal{S} \mid \phi \rangle, \text{ and}$$

$$\Psi(S \curlywedge S', t) = \begin{cases} \bigsqcup_{j \in J \setminus \{i\}} @j.\Psi\big(S_j, t_{|j}\big) \sqcup \Psi\big((@i.\boldsymbol{\tau} \curlywedge @i.S_i), t_{|i}\big) & \text{if } \mathcal{V}(\mathcal{S}, t) \models \phi, \\ \varnothing & \text{otherwise} \end{cases}$$

$$= \begin{cases} \bigsqcup_{j \in J \setminus \{i\}} @j.\Psi\big(S_j, t_{|j}\big) \sqcup \big(\Psi\big(@i.\boldsymbol{\tau}, t_{|i}\big) \curlywedge \Psi(@i.S_i, t_{|i})\big) & \text{if } \mathcal{V}(\mathcal{S}, t) \models \phi, \\ \varnothing & \text{otherwise} \end{cases}$$

Since $\Delta(S_i) < \Delta(S', t)$, it follows from the induction hypothesis that

$$\Psi(S \curlywedge S', t) = \begin{cases} \bigsqcup_{j \in J \setminus \{i\}} @j.\Psi\big(S_j, t_{|j}\big) \sqcup \big(@i.\boldsymbol{\tau} \curlywedge \Psi(@i.S_i, t_{|i})\big) & \text{if } \mathcal{V}(\mathcal{S}, t) \models \phi, \\ \varnothing & \text{otherwise} \end{cases}$$

On the other hand,

$$\Psi(S, t) = \begin{cases} @i.\boldsymbol{\tau} & \text{if } i \in \mathcal{P}os(t), \\ \varnothing & \text{otherwise} \end{cases} \quad \text{and} \quad \Psi(S', t) = \begin{cases} \bigsqcup_{j \in J} @j.\Psi\big(S_j, t_{|j}\big) & \text{if } \mathcal{V}(\mathcal{S}, t) \models \phi, \\ \varnothing & \text{otherwise} \end{cases}$$

and since $i \in J$, the unification of $\Psi(S, t)$ and $\Psi(S', t)$ is

$$\Psi(S, t) \curlywedge \Psi(S', t) =$$

$$\begin{cases} \bigsqcup_{j \in J \setminus \{i\}} @j.\Psi\big(S_j, t_{|j}\big) \sqcup (@i.\boldsymbol{\tau} \curlywedge @i.\Psi\big(S_i, t_{|i}\big)) & \text{if } i \in \mathcal{P}os(t) \text{ and } \mathcal{V}(\mathcal{S}, t) \models \phi, \\ \varnothing & \text{otherwise} \end{cases}$$

Since ϕ is a conjunction of position-variables, and $i \in J$, which means $i \in \mathcal{V}ar(\mathcal{S})$, then

$$\mathcal{V}(\mathcal{S}, t) \models \phi \qquad \text{iff} \qquad i \in \mathcal{P}os(t) \text{ and } \mathcal{V}(\mathcal{S}, t) \models \phi.$$

That leads to $\Psi(S \curlywedge S', t) = \Psi(S, t) \curlywedge \Psi(S', t)$.

3. If $S = (@i.\boldsymbol{\tau})$ and $S' = \mu Z.R(Z)$, where $i \in \mathbb{N}_\epsilon$, then in this case

$$S \curlywedge S' = S''\big(\mu Z.R(Z)\big), \text{ where } S''(Z) = (@i.\boldsymbol{\tau}) \curlywedge R(Z), \text{ and}$$

$$\begin{aligned} \Psi(S \curlywedge S', t) &= \Psi(S''(\mu Z.R(Z)), t) \\ &= \Psi\Big(S''\big(\bigoplus_{i=1,\delta(t)} R^i(\varnothing)\big), t\Big) \\ &= \Psi\Big((@i.\boldsymbol{\tau}) \curlywedge R\big(\bigoplus_{i=1,\delta(t)} R^i(\varnothing)\big), t\Big) \\ &= \Psi\Big((@i.\boldsymbol{\tau}) \curlywedge \bigoplus_{i=1,\delta(t)} R^{i+1}(\varnothing)\big), t\Big) \\ &= \Psi\Big((@i.\boldsymbol{\tau}) \curlywedge \bigoplus_{i=1,\delta(t)} R^{i}(\varnothing)\big), t\Big) \end{aligned}$$

If we assume that $\Delta\big(\bigoplus_{i=1,\delta(t)} R^i(\varnothing)\big) = (n, m)$, for some $n, m \in \mathbb{N}$, then $\Delta\big(\mu Z.R(Z)\big) = (n+1, m')$, for some $m' \in \mathbb{N}$. Meaning that $\Delta\big(\bigoplus_{i=1,\delta(t)} R^i(\varnothing)\big) < \Delta\big(\mu Z.R(Z)\big)$. Thus it follows from the induction hypothesis that

$$\Psi(S \curlywedge S', t) = \Psi((@i.\boldsymbol{\tau}), t) \curlywedge \Psi\big(\bigoplus_{i=1,\delta(t)} R^i(\varnothing), t\big)$$

On the hand,

$$\begin{aligned}\Psi(S', t) &= \Psi(\mu Z.R, t)\\ &= \Psi\big(\bigoplus_{i=1,\delta(t)} R^i(\varnothing), t\big)\end{aligned}$$

Hence,

$$\Psi(S \curlywedge S', t) = \Psi(S, t) \curlywedge \Psi(S', t).$$

4. If $S = (u, \boldsymbol{\tau})$ and $S' = (u', R')$, then in this case

$$\begin{aligned} S \curlywedge S' &= (u \wedge u', (@\epsilon.\boldsymbol{\tau}) \curlywedge S')\\ &\text{and}\\ \Psi(S \curlywedge S', t) &= \begin{cases} \Psi\big((@\epsilon.\boldsymbol{\tau}) \curlywedge S', t\big) & \text{if } (u \wedge u') \ll t\\ \varnothing, \text{ otherwise.} \end{cases}\\ &= \begin{cases} \Psi(@\epsilon.\boldsymbol{\tau}, t) \curlywedge \Psi(S', t) & \text{if } (u \wedge u') \ll t\\ \varnothing, \text{ otherwise.} \end{cases}\end{aligned}$$

On the other hand,

$$\Psi(S, t) = \begin{cases} @\epsilon.\boldsymbol{\tau} & \text{if } u \ll t\\ \varnothing & \text{otherwise} \end{cases} \quad \text{and} \quad \Psi(S', t) = \begin{cases} \Psi(S', t) & \text{if } u' \ll t\\ \varnothing, \text{ otherwise.} \end{cases}$$

Since

$$(u \wedge u') \ll t \qquad \text{iff} \qquad u \ll t \text{ and } u' \ll t \qquad \text{(Lemma 4)}$$

We get

$$\Psi(S \curlywedge S', t) = \Psi(S, t) \curlywedge \Psi(S', t).$$

5. If $S = (u, \boldsymbol{\tau})$ and $S' = \langle \bigsqcup_{i\in I} @i.S'_i \,|\, \phi'\rangle$, then we only prove the case when $I \subset \mathbb{N}_\epsilon \setminus \{\epsilon\}$, the case when $\epsilon \in I$ is similar. We have that

$$S \curlywedge S' = \big(u, \langle \bigsqcup_{i\in I} @i.S'_i \sqcup (@\epsilon.\boldsymbol{\tau}) \,|\, \phi'\rangle\big), \text{ and}$$

$$\begin{aligned}\Psi(S \curlywedge S', t) &= \begin{cases} \Psi\big(\langle \bigsqcup_{i\in I} @i.S'_i \sqcup (@\epsilon.\boldsymbol{\tau}) \,|\, \phi'\rangle, t\big) & \text{if } u \ll t \\ \varnothing, \text{ otherwise} \end{cases} \\ &= \begin{cases} \bigsqcup_{i\in I} @i.\Psi(S'_i, t_{|i}) \sqcup (@\epsilon.\boldsymbol{\tau}) & \text{if } u \ll t \text{ and } \mathcal{V}(\bigsqcup_{i\in I} @i.\Psi(S'_i), t) \models \phi' \\ \varnothing, \text{ otherwise} \end{cases} \\ &= \Psi(S, t) \curlywedge \Psi(S', t).\end{aligned}$$

6. If $S = (u, \boldsymbol{\tau})$ and $S' = \mu Z.R(Z)$, then this case is similar to the case where $S = @i.\boldsymbol{\tau}$ discussed before.

Induction step: $\Delta(S) > (0, 0)$. We make an induction on $\Delta(S')$.

Base case: $\Delta(S') = (0, 0)$. This case is symmetric to the case where $\Delta(S) = (0, 0)$ and $\Delta(S') > (0, 0)$ discussed before.

Induction step: $\Delta(S') > (0, 0)$. We distinguish four cases.

1. If $S = (u, R)$ and $S' = (u', R')$, then in this case

$$S \curlywedge S' = (u \wedge u', R \curlywedge R') \quad \text{and} \quad \Psi(S \curlywedge S', t) = \begin{cases} \Psi(R \curlywedge R', t) & \text{if } (u \wedge u') \ll t \\ \varnothing & \text{otherwise} \end{cases}$$

Since

$$(u \wedge u') \ll t \qquad \text{iff} \qquad u \ll t \text{ and } u' \ll t \qquad \text{(Lemma 4)}$$

and since $\Delta(R) < \Delta(S)$ and $\Delta(R') < \Delta(S')$, it follows from the induction hypothesis that

$$\Psi(S \curlywedge S', t) = \begin{cases} \Psi(R, t) \curlywedge \Psi(R', t) & \text{if } u \ll t \text{ and } u' \ll t \\ \varnothing & \text{otherwise} \end{cases}$$

On the other hand,

$$\Psi(S, t) = \begin{cases} \Psi(R, t) & \text{if } u \ll t \\ \varnothing & \text{otherwise} \end{cases} \quad \text{and} \quad \Psi(S', t) = \begin{cases} \Psi(R', t) & \text{if } u' \ll t \\ \varnothing & \text{otherwise} \end{cases}$$

Therefore $\Psi(S, t) \curlywedge \Psi(S', t) = \Psi(S \curlywedge S', t)$.

2. If S and S' are lists of position delimiters of the form

$$S = \langle \bigsqcup_{i\in\mathcal{I}} @i.S_i \,|\, \phi\rangle \qquad \text{and} \qquad S' = \langle \bigsqcup_{j\in\mathcal{J}} @j.S'_j \,|\, \phi'\rangle$$

where $\mathcal{I}, \mathcal{J} \subset \mathbb{N}_\epsilon$, then let

$$\mathcal{S} = \bigsqcup_{i \in \mathcal{I}} @i.S_i \qquad \text{and} \qquad \mathcal{S}' = \bigsqcup_{j \in \mathcal{J}} @j.S'_j$$

On the one hand we have that

$$\Psi(S,t) = \begin{cases} \bigsqcup_{i\in\mathcal{I}} @i.\eta\big(\Psi(S_i,t_i)\big) & \text{if } \mathcal{V}(\mathcal{S},t) \models \phi \\ \varnothing & \text{otherwise,} \end{cases}$$

and

$$\Psi(S',t) = \begin{cases} \bigsqcup_{j\in\mathcal{J}} @j.\eta\big(\Psi(S'_j,t_j)\big) & \text{if } \mathcal{V}(\mathcal{S}',t) \models \phi' \\ \varnothing & \text{otherwise.} \end{cases}$$

Hence, the unification $\Psi(S,t) \curlywedge \Psi(S',t)$ is defined by

$$\Psi(S,t) \curlywedge \Psi(S',t) =$$
$$\begin{cases} \bigsqcup_{i\in\mathcal{I}} @i.\eta\big(\Psi(S_i,t_i)\big) \curlywedge \bigsqcup_{j\in\mathcal{J}} @j.\eta\big(\Psi(S'_j,t_{|j})\big) & \text{if } \mathcal{V}(\mathcal{S},t) \models \phi \text{ and } \mathcal{V}(\mathcal{S}',t) \models \phi' \\ \varnothing & \text{otherwise.} \end{cases}$$

On the other hand, let

$$\mathcal{S}'' = \bigsqcup_{i\in\mathcal{I}\cap\mathcal{J}} @i.(S_i \curlywedge S'_i) \sqcup \bigsqcup_{i\in\mathcal{I}\setminus\mathcal{J}} @i.S_i \sqcup \bigsqcup_{i\in\mathcal{J}\setminus\mathcal{I}} @i.S'_i$$

and thus the combination $S \curlywedge S'$ is defined by

$$S \curlywedge S' \stackrel{def}{=} \big\langle \mathcal{S}'' \,|\, \phi \wedge \phi' \big\rangle.$$

To simplify the presentation, let

$$\widetilde{\mathcal{S}}'' = \bigsqcup_{i\in\mathcal{I}\cap\mathcal{J}} @i.\eta\big(\Psi((S_i \curlywedge S'_i),t)\big) \sqcup \bigsqcup_{i\in\mathcal{I}\setminus\mathcal{J}} @i.\eta\big(\Psi(S_i,t)\big) \sqcup \bigsqcup_{i\in\mathcal{J}\setminus\mathcal{I}} @i.\eta(\Psi(S'_i,t)).$$

Thus $\Psi(S \curlywedge S', t)$ can be written as

$$\Psi(S \curlywedge S', t) = \begin{cases} \widetilde{\mathcal{S}}'' & \text{if } \mathcal{S}'' \models \phi \wedge \phi' \\ \varnothing & \text{otherwise.} \end{cases}$$

3. $\langle \bigsqcup_{i\in\mathcal{I}} @i.S_i \,|\, \phi \rangle \curlywedge (u', S') = \big(u', \langle \bigsqcup_{i\in\mathcal{I}} @i.S_i \,|\, \phi \curlywedge S' \rangle\big)$

4. If $S = \mu X.S(X)$ and $S' = \mu X'.S'(X')$, then we have

$$(\mu X.S(X)) \curlywedge (\mu X'.S'(X')) \stackrel{def}{=} \mu Z.S''(\mu X.S(X), \mu X'.S'(X'), Z)$$
$$\text{where } S''(X, X', X \curlywedge X') \stackrel{def}{=} S(X) \curlywedge S'(X').$$

Let $t \in \mathcal{T}$ and let $n = \delta(t)$. To show that

$$\Psi\big(\mu X.S(X), t\big) \curlywedge \Psi\big(\mu X'.S'(X'), t\big) = \Psi\big(\mu Z.S''(\mu X.S(X), \mu X'.S'(X'), Z), t\big)$$

it is sufficient to show that

$$\bigoplus_{i\in[n]} \widetilde{S}^i(\varnothing) = \bigoplus_{i\in[n]} \bigoplus_{i\in[n]} S^i(\varnothing) \curlywedge S'^i(\varnothing) \tag{11}$$

We have that for all $i \geq 1$,

$$\begin{aligned} S^i(\varnothing) \curlywedge S'^i(\varnothing) &\overset{def}{=} S(S^{i-1}(\varnothing)) \curlywedge S'(S'^{i-1}(\varnothing)) \\ &\overset{def}{=} S''\big(S^{i-1}(\varnothing), S'^{i-1}(\varnothing), S^{i-1}(\varnothing) \curlywedge S'^{i-1}(\varnothing)\big) \\ &\overset{def}{=} \widetilde{S}^i(\varnothing) \end{aligned}$$

hence,

$$\begin{aligned} [\![\mu X.S(X)]\!] \curlywedge [\![\mu X'.S'(X')]\!] &\overset{def}{=} \big(\bigoplus_{i\in[n]} S^i(\varnothing)\big) \curlywedge \big(\bigoplus_{i\in[n]} S'^i(\varnothing)\big) \\ &= S''\big(\bigoplus_{i\in[n-1]} S^i(\varnothing), \bigoplus_{i\in[n-1]} S'^i(\varnothing), \bigoplus_{i\in[n-1]} S^i(\varnothing) \curlywedge \bigoplus_{i\in[n-1]} S'^i(\varnothing)\big) \\ &= \bigoplus_{i\in[n]} \widetilde{S}^i(\varnothing) \\ &= [\![\mu X.S(X) \curlywedge \mu X'.S'(X')]\!] \end{aligned}$$

Hence,

$$\begin{aligned} \Psi\big(\mu X.S(X) \curlywedge \mu X'.S'(X'), t\big) &= \Psi\big(\big(\bigoplus_{i\in[n]} S^i(\varnothing)\big) \curlywedge \big(\bigoplus_{i\in[n]} S'^i(\varnothing)\big), t\big) \\ &= \Psi\big(\bigoplus_{i\in[n]} S^i(\varnothing), t\big) \curlywedge \Psi\big(\bigoplus_{i\in[n]} S'^i(\varnothing), t\big) \\ &= \Psi(\mu X.S(X), t) \curlywedge \Psi(\mu X'.S'(X'), t) \end{aligned}$$

5. The cases of $(\mu X.S(X)) \curlywedge S'$ and $S \curlywedge \mu X'.S'(X')$ are similar to the previous one.

6. If $S = S_1 \oplus S_2$ then we recall that

$$\Psi(S_1 \oplus S_2, t) \equiv \Psi(S_1, t) \oplus \Psi(S_2, t),$$

and

$$\begin{aligned} \Psi(S \curlywedge S', t) &\overset{def}{=} \Psi\big((S_1 \curlywedge S') \oplus (S_2 \curlywedge S'), t\big) \\ &= \Psi(S_1 \curlywedge S', t) \oplus \Psi(S_2 \curlywedge S', t). \end{aligned}$$

Hence,

$$\begin{aligned}\Psi(S,t) \curlywedge \Psi(S',t) &= \big(\Psi(S_1,t) \oplus \Psi(S_2,t)\big) \curlywedge \Psi(S',t) \\ &= \big(\Psi(S_1,t) \curlywedge \Psi(S',t)\big) \oplus \big(\Psi(S_2,t) \curlywedge \Psi(S',t)\big).\end{aligned}$$

Since $\Delta(S_i) < \Delta(S)$, for $i = 1, 2$, it follows from the induction hypothesis that

$$\begin{aligned}\Psi(S,t) \curlywedge \Psi(S',t) &= \Psi(S_1 \curlywedge S',t) \oplus \Psi(S_2 \curlywedge S',t) \\ &= \Psi(S \curlywedge S',t).\end{aligned}$$

This ends the proof of Theorem 4. □

Theorem 5 (i.e. Theorem 2). *For every term $t \in \mathcal{T}$, for every* CE*-strategies S and S' in the canonical form in $\mathcal{C}^o$, we have that*

$$\Psi(S \curlyvee S',t) = \Psi(S,t) \curlyvee \Psi(S',t). \tag{12}$$

Proof.

$$\begin{aligned}\Psi(S \curlyvee S',t) &= \Psi\big((S \curlywedge S') \oplus S \oplus S',t\big) && \text{(Def. 13 of } \curlyvee) \\ &= \Psi(S \curlywedge S',t) \oplus \Psi(S,t) \oplus \Psi(S',t) && \text{(Def. 11 of } \Psi) \\ &= (\Psi(S,t) \curlywedge \Psi(S',t)) \oplus \Psi(S,t) \oplus \Psi(S',t) && \text{(Theorem 1)} \\ &= \Psi(S,t) \curlyvee \Psi(S',t) && \text{(Def. 13 of } \curlyvee)\end{aligned}$$

Proposition 3 (i.e. Proposition 2). *The following hold.*

1. *The set $\mathcal{C}$ of* CE*-strategies together with the unification and combination operations enjoy the following properties.*
   (a) *The neutral element of the unification and combination is $@\epsilon.\Box$.*
   (b) *Every* CE*-strategy S is idempotent for the unification and combination, i.e. $S \curlywedge S = S$ and $S \curlyvee S = S$.*
   (c) *The unification and combination of* CE*-strategies are associative.*
2. *The unification and combination of* CE*-strategies is non commutative.*
3. *For any* CE*-strategies S and S' in $\mathcal{C}$, and for any term t in $\mathcal{T}$, we have that*

   $\Psi(S \curlywedge S',t) = \varnothing$ *iff* $\Psi(S,t) = \varnothing$ *or* $\Psi(S',t) = \varnothing$.

   $\Psi(S \curlyvee S',t) = \varnothing$ *iff* $\Psi(S,t) = \varnothing$ *and* $\Psi(S',t) = \varnothing$.

4. *The unification and combination of* CE*-strategies is a congruence, that is, for any* CE*-strategies S_1, S_2, S in $\mathcal{C}$, we have that:*

   If $S_1 \equiv S_2$ *then* $S_1 \curlywedge S \equiv S_2 \curlywedge S$ *and* $S \curlywedge S_1 \equiv S \curlywedge S_2$.

   If $S_1 \equiv S_2$ *then* $S_1 \curlyvee S \equiv S_2 \curlyvee S$ *and* $S \curlyvee S_1 \equiv S \curlyvee S_2$.

Proof. We only prove the last Item. To prove the associativity of the both unification and combination for CE-strategies we rely on the associativity of the unification and combination of elementary CE-strategies (Proposition 1) together with the property of the function Ψ (Theorems 1 and 2).

Let S_1, S_2 and S_3 be CE-strategies in $\mathcal{C}$. It follows from Item *iii.)* of Lemma 3 that in order to prove that

$$S_1 \curlyvee (S_2 \curlyvee S_3) \equiv (S_1 \curlyvee S_2) \curlyvee S_3,$$

it suffices to prove that, for any term $t \in \mathcal{T}$, we have that

$$\Psi\big(S_1 \curlyvee (S_2 \curlyvee S_3), t\big) = \Psi\big((S_1 \curlyvee S_2) \curlyvee S_3, t\big).$$

But this follows from an easy computation:

$$\begin{aligned}
\Psi\big(S_1 \curlyvee (S_2 \curlyvee S_3), t\big) &= \Psi(S_1, t) \curlyvee \Psi(S_2 \curlyvee S_3, t) && \text{(Theorem 2)}\\
&= \Psi(S_1, t) \curlyvee (\Psi(S_2, t) \curlyvee \Psi(S_3, t)) && \text{(Theorem 2)}\\
&= (\Psi(S_1, t) \curlyvee \Psi(S_2, t)) \curlyvee \Psi(S_3, t) && \text{(Proposition 1)}\\
&= \Psi(S_1 \curlyvee S_2, t) \curlyvee \Psi(S_3, t) && \text{(Theorem 2)}\\
&= \Psi\big((S_1 \curlyvee (S_2 \curlyvee S_3), t\big) && \text{(Theorem 2)}
\end{aligned}$$

On the one hand, if follows from Theorem 2 that

$$\Psi(S_1 \curlywedge S, t) = \Psi(S_1, t) \curlywedge \Psi(S, t).$$

On the other hand, since $S_1 \equiv S_2$, it follows from Item *iii.)* of Lemma 3 that

$$\Psi(S_1, t) = \Psi(S_2, t).$$

Hence we get

$$\begin{aligned}
\Psi(S_1 \curlywedge S, t) &= \Psi(S_2, t) \curlywedge \Psi(S, t)\\
&= \Psi(S_2 \curlywedge S, t) && \text{(Theorem 2)}
\end{aligned}$$

Again, from Item *iii.)* of Lemma 3, we get

$$S_1 \curlywedge S \equiv S_2 \curlywedge S.$$

The proof of the remaining claims is similar.