A Duality in Proof Systems for Recursive Type Equality and for Bisimulation Equivalence on Cyclic Term Graphs [1]

Clemens Grabmayer [2]

Department of Mathematics and Computer Science
Vrije Universiteit Amsterdam
de Boelelaan 1081a, 1081 HV Amsterdam,
The Netherlands

Abstract

This paper is concerned with a proof-theoretic observation about two kinds of proof systems for regular cyclic objects. It is presented for the case of two formal systems that are complete with respect to the notion of "recursive type equality" on a restricted class of recursive types in μ-term notation. Here we show the existence of an immediate duality with a geometrical visualization between proofs in a variant of the coinductive axiom system due to M. Brandt and F. Henglein and "consistency-unfoldings" in a variant of a 'syntactic-matching' proof system for testing equations between recursive types due to Z. Ariola and J.W. Klop. This result makes it possible to argue for the soundness of the coinductive derivation rule present in the system of Brandt and Henglein in a new way and it leads to an independent soundness proof for the here considered variant of this system.

Finally we sketch an analogous result of a duality between a similar pair of proof systems for bisimulation equivalence on equational specifications of cyclic term graphs.

1 Introduction

The main part of this paper is concerned with an observation about two complete proof systems for the notion of "recursive type equality" on recursive types.

There are to our knowledge basically two different complete axiom systems known for recursive type equality: (i) A system due to R. Amadio and L. Cardelli given in [1] (1993) and (ii) a coinductively motivated axiom system introduced by M. Brandt and F. Henglein in [4] (1998). Apart from these axiomatizations it is also possible to consider (iii) a 'syntactic-matching' proof system for which a notion of

[1] This is a slightly extended and more detailed version of a paper submitted for the Workshop TERMGRAPH 2002, Barcelona, October 7, 2002. This version was completed at 15 July 2002.

[2] Email: `clemens@cs.vu.nl` ; homepage: `http://www.cs.vu.nl/~clemens` .

consistency with respect to this system is complete for recursive type equality. Such a system can be defined in a very similar way to one that has been introduced by Z. Ariola and J.W. Klop in [2] (1995) for the notion of bisimulation equivalence on equational representations of cyclic term graphs. For our purpose we will consider only 'normalized' variants without symmetry and transitivity rules of the Brandt-Henglein and syntactic-matching systems. In Section 3 these variant-systems will be defined and their respective soundness and completeness theorems stated.

It was noted by J.W. Klop that there appears to be a striking similarity between the activities of (a) trying to demonstrate the consistency of an equation between recursive types with respect to the syntactic-matching system and of (b) trying to find a derivation for the same equation in the system of Brandt and Henglein. This basic observation underlying the present paper will be described in Section 4 in relation to the introduced variant-systems by explaining it in the light of an example.

In order to extract a precise statement from this observation two formal prerequisites turn out to be necessary: Firstly, in Section 5 we will introduce an extension of the variant Brandt-Henglein system with some more coinductive rules. And secondly, in Section 6 we define so called "consistency-unfoldings" of given equations between recursive types in the variant 'syntactic-matching' system as certain formalizations of successful consistency-checks. With these notions our main theorem is then stated in Section 7: There exists even a "duality" between derivations in the extended variant-Brandt-Henglein system and corresponding consistency-unfoldings in the variant-syntactic-matching system via easily definable reflection mappings.

This relationship between the two considered proof systems can be geometrically visualized and allows us to give an alternative soundness proof for our variant of the Brandt-Henglein system and for its extension. By 'zooming' into a special case of this duality we furthermore show the existence of an analogous strong connection between derivations in the (not extended) variant-Brandt-Henglein system and corresponding consistency-unfoldings of a certain formally characterized kind in our syntactic-matching system.

In Section 8 we outline an analogous result for a similar pair of proof systems concerned with the bisimulation relation on equational specifications of cyclic term graphs.

2 Preliminaries on recursive types

Likewise as Brandt and Henglein in [4] we consider only [3] recursive types denoted by μ-terms in canonical form over the restricted class of finite types with $\rightarrow$ as the single type constructor. We assume a countably infinite set $TVar$ of *type variables* to be given and to underlie the following definition. The small Greek letters α and β (possibly with subscripts) will be used as syntactical variables for type variables and the letters τ, σ, ρ, χ for recursive types.

[3] Our results do not depend on the limitation to proof systems for recursive types *in canonical form* only. We followed [4] in the intention to avoid unnecessary technicalities here. The general case will be treated in [5].

Definition 2.1 (Recursive Types *can-μTp* in Canonical Form). The set *can-μTp* of *recursive types in canonical form* is generated by the following grammar:

$$\tau ::= \bot \mid \top \mid \alpha \mid \tau_1 \to \tau_2 \mid \mu\alpha.\underbrace{(\tau_1 \to \tau_2)}_{\text{where } \alpha \in \text{fv}(\tau_1 \to \tau_2)} \;. \tag{2.1}$$

The set of all equations $\tau = \sigma$ between recursive types τ and σ in canonical form will be denoted by *can-μTp–Eq*.

The recursive types in *can-μTp* are in "canonical form" due to the two requirements in the last disjunctive clause in grammar (2.1): For given $\alpha \in TVar$ the μ-operator may only be applied to a previously formed expression τ if τ is of the form $\tau_1 \to \tau_2$ and if α occurs free in $\tau_1 \to \tau_2$.

Contrary to [4] we do not implicitly identify recursive types in *can-μTp* that can be obtained from each other by a finite sequence of admissible renaming-steps for bound type variables, i.e. that are *variants* of each other. We will use the notation $\tau_1 \equiv_{\text{v}} \tau_2$ to express that τ_1 and τ_2 are variants of each other.

Via a natural transformation of μ-terms into cyclic term graphs described in (the extended version of) [2] it is possible to assign to every recursive type $\tau \in$ *can-μTp* a cyclic term graph $G(\tau)$, whose nodes have at most two outgoing edges and are labelled by either the binary function symbol $\to$ or by a symbol of arity zero in $\{\bot, \top\} \cup TVar$. Relying on this transformation the *tree unfolding* $\mathbf{Tree}(\tau)$ of an arbitrary recursive type $\tau \in$ *can-μTp* can be defined as the tree unfolding of $G(\tau)$. An alternative formal definition of $\mathbf{Tree}(\tau)$ can be found in [1].[4] The *leading symbol* $\mathcal{L}(\tau)$ of a recursive type $\tau \in$ *can-μTp* is defined as the symbol that labels the root in the tree unfolding $\mathbf{Tree}(\tau)$ of τ.[5]

Definition 2.2 (Recursive Type Equality (Strong Equivalence) $=_\mu$). Two recursive types $\tau, \sigma \in$ *can-μTp* are called *strongly equivalent* (symbolically denoted by: $\tau =_\mu \sigma$) iff they possess the same tree unfolding. More formally, the equivalence relation *recursive type equality* (also called *strong recursive type equivalence*) $=_\mu$ is defined by: For all $\tau, \sigma \in$ *can-μTp*

$$\tau =_\mu \sigma \;:\Longleftrightarrow\; \mathbf{Tree}(\tau) = \mathbf{Tree}(\sigma) \;.$$

An example for Definition 2.2 and for the underlying notion of the tree unfolding of a recursive type in *can-μTp* is given in Figure 1.

---

[4] The definition in [1] is slightly more general than then the one needed here because Amadio and Cardelli allow also recursive types *not in canonical form* like for example $\mu\alpha.(\mu\beta.\alpha)$.

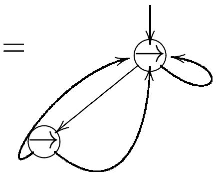
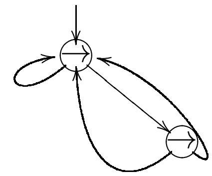
[5] Alternatively and more formally $\mathcal{L}(\tau)$ can be defined for all $\tau \in$ *can-μTp* by the 5 clauses $\mathcal{L}(\bot) := \bot$, $\mathcal{L}(\top) := \top$, $\mathcal{L}(\alpha) := \alpha$ (for all $\alpha \in TVar$) and $\mathcal{L}(\tau_1 \to \tau_2) := \mathcal{L}(\mu\alpha.(\tau_1 \to \tau_2)) := \to$ (for all $\alpha \in TVar$ and $\tau_1, \tau_2 \in$ *can-μTp*).

Figure 1 Example of two strongly equivalent recursive types.

We consider the recursive types in canonical form

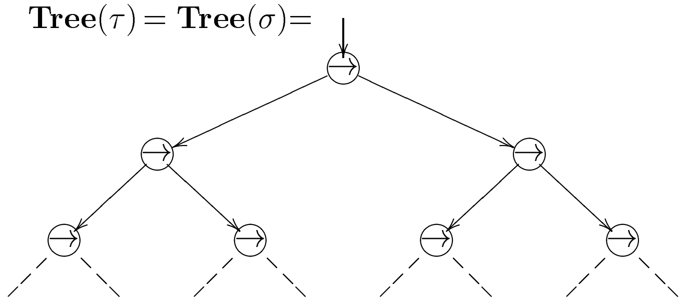
$$\tau :\equiv \ \mu\alpha.\,((\alpha \to \alpha) \to \alpha) \qquad \text{and} \qquad \sigma :\equiv \ \mu\alpha.\,(\alpha \to (\alpha \to \alpha)) \ .$$

These correspond respectively to the different cyclic term graphs

$G(\tau) =$  and $G(\sigma) =$  ,

but they possess the same tree unfolding of the form

$\mathbf{Tree}(\tau) = \mathbf{Tree}(\sigma) =$



.

Hence τ and σ are strongly equivalent, i.e. $\tau =_\mu \sigma$ holds, due to Definition 2.2.

3 The proof systems $\mathbf{HB}_\mathbf{0}^=$ and $\mathbf{AK}_\mathbf{0}^=$ for $=_\mu$

In this section we define the two proof systems on which our results will be based: A variant-system $\mathbf{HB}_\mathbf{0}^=$ of the coinductively motivated axiomatization for $=_\mu$ given by Brandt and Henglein in [4] and a proof system $\mathbf{AK}_\mathbf{0}^=$ suitable for consistency-checking similar to a system as defined by Ariola and Klop in [2]. We formulate these systems in natural-deduction style and for this and for later purposes we assume a countably infinite set Mk of assumption markers to be given.

Definition 3.1 (The axiom system $\mathbf{HB}_\mathbf{0}^=$ for $=_\mu$). The formal system $\mathbf{HB}_\mathbf{0}^=$ has the equations in *can-μTp–Eq* as its *formulas*. It contains the *axioms* (REFL), allows *marked assumptions* (Assm) and has the *derivation rules* VAR, FOLD$_l$, FOLD$_r$, ARROW and ARROW/FIX listed in Figure 2. The side-condition **I** on applications of ARROW/FIX requires that the class of discharged assumptions is actually inhabitated, i.e. non-empty.[6] A formula $\tau = \sigma$ is a *theorem* of $\mathbf{HB}_\mathbf{0}^=$ (symbolically denoted by $\vdash_{\mathbf{HB}_\mathbf{0}^=} \tau = \sigma$) iff there is a derivation $\mathcal{D}$ in $\mathbf{HB}_\mathbf{0}^=$ with conclusion $\tau = \sigma$ and with the property that all marked assumptions have been discharged at respective applications of ARROW/FIX in $\mathcal{D}$.

Apart from minor differences the system $\mathbf{HB}_\mathbf{0}^=$ can be considered as a 'normalized' version of the complete axiomatization for $=_\mu$ given in [4]. A distinctive role in

[6] The aim here is to create a clear-cut distinction between applications of ARROW and applications of ARROW/FIX for easing the reasoning about a later defined proof-transformation.

Figure 2 A normalized version $\mathbf{HB_0^=}$ of the coinductive axiomatization for recursive type equality $=_\mu$ given by Brandt and Henglein.

The *axioms* and possible *marked assumptions* in $\mathbf{HB_0^=}$:

$$(\text{REFL})\ \ \overline{\tau = \tau} \qquad\qquad (\text{Assm})\ \ (\tau = \sigma)^x \quad (\text{with } x \in Mk)\ .$$

The *derivation rules* of $\mathbf{HB_0^=}$:

$$\frac{\tau_0[\mu\alpha.\,\tau_0/\alpha] = \sigma}{\mu\alpha.\,\tau_0 = \sigma}\ \mathrm{FOLD}_l \qquad\qquad \frac{\tau = \sigma_0[\mu\beta.\,\sigma_0/\beta]}{\tau = \mu\beta.\,\sigma_0}\ \mathrm{FOLD}_r$$

$$\frac{\tau = \sigma}{\tau' = \sigma'}\ \mathrm{VAR}\quad \begin{array}{l}(\text{if } \tau' \equiv_{\mathrm{v}} \tau\\ \ \text{and } \sigma' \equiv_{\mathrm{v}} \sigma)\end{array} \qquad\qquad \frac{\tau_1 = \sigma_1 \qquad \tau_2 = \sigma_2}{\tau_1 \to \tau_2 = \sigma_1 \to \sigma_2}\ \mathrm{ARROW}$$

$$\frac{\begin{array}{c}\langle \tau_1 \to \tau_2 = \sigma_1 \to \sigma_2 \rangle^x \\ \mathcal{D}_1 \\ \tau_1 = \sigma_1\end{array} \qquad \begin{array}{c}\langle \tau_1 \to \tau_2 = \sigma_1 \to \sigma_2 \rangle^x \\ \mathcal{D}_2 \\ \tau_2 = \sigma_2\end{array}}{\tau_1 \to \tau_2 = \sigma_1 \to \sigma_2}\ \begin{array}{l}(\mathrm{ARROW/FIX})_x\\ (\text{if side-cond. } \mathbf{I})\end{array}\ .$$

both the original system in [4] as well as in $\mathbf{HB_0^=}$ is played by the rule ARROW/FIX, an application of which consists of the amalgamation of an application of the composition rule ARROW with an application of a fixed-point rule [7] FIX, at which open assumptions of the form of its conclusion can be discharged. No symmetry and transitivity rules are present in $\mathbf{HB_0^=}$ and the axioms (FOLD/UNFOLD) used in [4] have been reformulated into the two [8] rules $\mathrm{FOLD}_{l/r}$. $\mathbf{HB_0^=}$ is 'normalized' in the sense that it satisfies a version of the *subformula property*. Although lacking the expressivity of symmetry and transitivity rules the following also holds for $\mathbf{HB_0^=}$:

Theorem 3.2 (Sound- and Completeness of $\mathbf{HB_0^=}$ with respect to $=_\mu$). *The axiom system $\mathbf{HB_0^=}$ is sound and complete with respect to strong recursive type equivalence $=_\mu$, i.e. for all $\tau, \sigma \in$ can-μTp it holds that*

$$\vdash_{\mathbf{HB_0^=}} \tau = \sigma \quad \Longleftrightarrow \quad \tau =_\mu \sigma\ .$$

Both the soundness and the completeness of $\mathbf{HB_0^=}$ with respect to $=_\mu$ can be shown analogously as done by Brandt and Henglein in [4] for their system. [9]

[7] This rule is not part of $\mathbf{HB_0^=}$ nor of the system introduced in [4]. As Brandt and Henglein point out, the rule FIX is unsound in its general form, but it can be reformulated into a sound derivation rule for a formal system that axiomatizes $=_\mu$ by requiring a certain side-condition to be fulfilled for its applications.

[8] Here and later we allow two rules like FOLD_l and FOLD_r to be "bundled together" to rules $\mathrm{FOLD}_{l/r}$ in informal arguments: "... holds for a rule $\mathrm{FOLD}_{l/r}$" is intended to mean "... holds for a rule FOLD_l or for a rule FOLD_r" and "... holds for rules $\mathrm{FOLD}_{l/r}$" stands for "... holds for the rule FOLD_l and for the rule FOLD_r".

[9] Because of a very close relationship between $\mathbf{HB_0^=}$ and the definition of the tree unfolding of recursive types, Theorem 3.2 can also be shown in a more direct alternative way.

Figure 3 A normalized 'syntactic-matching' proof system $\mathbf{AK}_{\mathbf{0}}^{=}$ for checking the consistency of given equations with respect to $=_\mu$. This system is related to a one that was introduced by Ariola and Klop.

The *derivation rules* of $\mathbf{AK}_{\mathbf{0}}^{=}$:

$$\frac{\mu\alpha.\,\tau_0 = \sigma}{\tau_0[\mu\alpha.\,\tau_0/\alpha] = \sigma}\ \mathrm{UNFOLD}_l \qquad \frac{\tau = \mu\beta.\,\sigma_0}{\tau = \sigma_0[\mu\beta.\,\sigma_0/\beta]}\ \mathrm{UNFOLD}_r$$

$$\frac{\tau = \sigma}{\tau' = \sigma'}\ \mathrm{VAR}\quad \begin{array}{l}(\text{if } \tau' \equiv_{\mathrm{v}} \tau \\ \ \text{and } \sigma' \equiv_{\mathrm{v}} \sigma)\end{array} \qquad \frac{\tau_1 \to \tau_2 = \sigma_1 \to \sigma_2}{\tau_i = \sigma_i}\ \mathrm{DECOMP}\quad (i = 1, 2)\ .$$

We do not investigate in this paper the proof-theoretic relationship between the axiom system for $=_\mu$ introduced by Brandt and Henglein and our variant-system $\mathbf{HB}_{\mathbf{0}}^{=}$. However, we want to mention two facts in this respect that are proven in our forthcoming work [5]: (1) Every derivation $\mathcal{D}$ in $\mathbf{HB}_{\mathbf{0}}^{=}$ can be transformed in an easy and effective way into a derivation $\mathcal{D}'$ in the system of Brandt and Henglein such that $\mathcal{D}'$ has the same conclusion and the same open assumption classes as $\mathcal{D}$. But (2) an effective transformation of derivations between these two systems in the opposite direction is—although possible—not of an equally easy kind.

We continue with the definition of a proof system very similar to a 'syntactic-matching' system introduced by Ariola and Klop in Section 3.4 of [2].

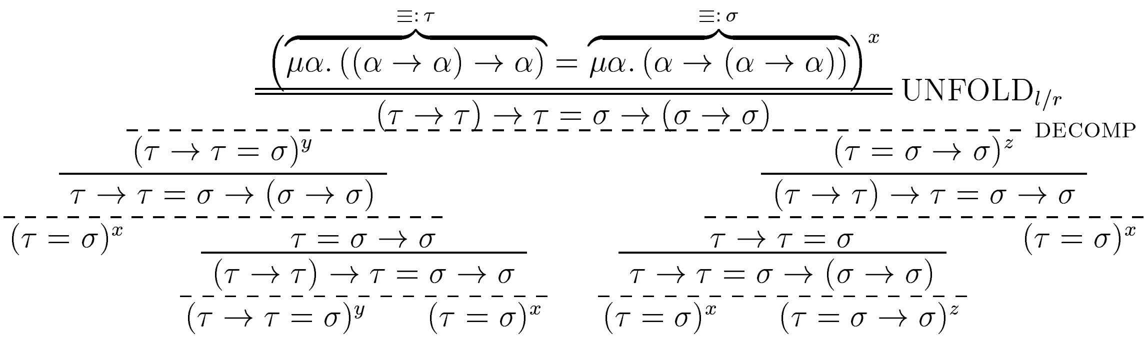
Definition 3.3 (A 'syntactic-matching' proof system $\mathbf{AK}_{\mathbf{0}}^{=}$ for $=_\mu$). The formal system $\mathbf{AK}_{\mathbf{0}}^{=}$ contains precisely all equations in *can-*μ*Tp–Eq* as its *formulas*. It contains *no axioms*. Its *derivation rules* are the rules VAR, UNFOLD_l, UNFOLD_r and DECOMP that are listed in Figure 3. We will use $\tau = \sigma \vdash_{\mathbf{AK}_{\mathbf{0}}^{=}} \chi_1 = \chi_2$ (for $\tau, \sigma, \chi_1, \chi_2 \in$ *can-*μ*Tp*) as notation for the assertion that there is a derivation in $\mathbf{AK}_{\mathbf{0}}^{=}$ from the assumption $\tau = \sigma$ with conclusion $\chi_1 = \chi_2$.

The conspicuous feature of this system is the decomposition rule DECOMP, which is a "destructive" counterpart of the "constructive" composition rules ARROW and ARROW/FIX in $\mathbf{HB}_{\mathbf{0}}^{=}$. Like $\mathbf{HB}_{\mathbf{0}}^{=}$ the system $\mathbf{AK}_{\mathbf{0}}^{=}$ does not contain symmetry and transitivity rules and is 'normalized' in the sense that it fulfills an "inverse subformula property".

Clearly, $\mathbf{AK}_{\mathbf{0}}^{=}$ does not axiomatize $=_\mu$, but a notion of consistency-checking with respect to $\mathbf{AK}_{\mathbf{0}}^{=}$ is sound and complete for $=_\mu$. For being able to state this properly, we need the following terminology: An equation $\tau = \sigma$ between recursive types is a *contradiction with respect to* $=_\mu$ iff $\mathcal{L}(\tau) \neq \mathcal{L}(\sigma)$, i.e. iff the leading symbols of τ and σ differ. Furthermore an equation $\tau = \sigma$ is called $\mathbf{AK}_{\mathbf{0}}^{=}$*-inconsistent* iff there exists a contradiction $\chi_1 = \chi_2$ with respect to $=_\mu$ such that $\tau = \sigma \vdash_{\mathbf{AK}_{\mathbf{0}}^{=}} \chi_1 = \chi_2$; otherwise we say that $\tau = \sigma$ is $\mathbf{AK}_{\mathbf{0}}^{=}$*-consistent*.

Theorem 3.4 (Soundness and Completeness with respect to $=_\mu$ of consistency-checking relative to $\mathbf{AK}_{\mathbf{0}}^{=}$). *Consistency with respect to* $\mathbf{AK}_{\mathbf{0}}^{=}$ *is*

Figure 4 Assemblage to a finite downwards-growing "tree of consequences" $\mathcal{C}$ of the 6 different possible initial segments of derivations from $\mu\alpha.\,((\alpha \to \alpha) \to \alpha) = \mu\alpha.\,(\alpha \to (\alpha \to \alpha))$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ without VAR-applications until *looping* occurs.



sound and complete in relation to $=_\mu$*: For all* $\tau, \sigma \in$ *can-*μTp *it holds that*

$$\tau = \sigma \;\; \textit{is } \mathbf{AK}_{\mathbf{0}}^{=}\textit{-consistent} \quad \Longleftrightarrow \quad \tau =_\mu \sigma \;\; .$$

Sketch of Proof. Both the soundness-part "$\Rightarrow$" and the completeness-part "$\Leftarrow$" are easy consequences of the fact that derivations $\mathcal{D}$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ from assumption $\tau = \sigma$ with conclusion $\chi_1 = \chi_2$ correspond to computations of $\chi_1, \chi_2 \in$ *can-*μTp with the property that [10] $\mathbf{Tree}(\tau)|p = \mathbf{Tree}(\chi_1)$ and $\mathbf{Tree}(\sigma)|p = \mathbf{Tree}(\chi_2)$ hold for some common position p in the tree unfoldings $\mathbf{Tree}(\tau)$ and $\mathbf{Tree}(\sigma)$ of τ and σ. □

4 The basic observation

It is our aim in this section to indicate the intuition behind the results of this paper by describing an observation about a concrete and simple example. Throughout this section we let τ and σ be the two strongly equivalent recursive types in *can-*μTp from the example in Figure 1.

Suppose that we want to prove that the equation $\tau = \sigma$ is indeed consistent with the system $\mathbf{AK}_{\mathbf{0}}^{=}$. Then we are obliged to show for every derivation $\mathcal{D}$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ from the assumption $\tau = \sigma$ that the conclusion of $\mathcal{D}$ is not a contradiction with respect to $=_\mu$. But since there are potentially infinitely many such derivations in $\mathbf{AK}_{\mathbf{0}}^{=}$, we might not be able to check all of them in a finite amount of time.

However, it turns out that in every derivation in $\mathbf{AK}_{\mathbf{0}}^{=}$ from $\tau = \sigma$ of depth ≥ 7, that does not contain applications of VAR, a loop arises, i.e. one formula occurs at two different places. What is more, the initial segments until looping occurs of all derivations from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ without VAR-applications can be arranged to the downwards-growing derivation-tree $\mathcal{C}$ depicted in Figure 4. Single and double lines in $\mathcal{C}$ separate the respective premises and conclusions of applications of $\mathrm{UNFOLD}_{l/r}$, whereas branchings at dashed lines in $\mathcal{C}$ stem from the two possible ways in which conclusions can be drawn at rules DECOMP in $\mathbf{AK}_{\mathbf{0}}^{=}$. The markers x, y and z used

[10] Here $\mathbf{Tree}(\tau)|p$ denotes the subtree of $\mathbf{Tree}(\tau)$ at position p.

for some formula occurrences in $\mathcal{C}$ are intended to highlight the looping in those $\mathbf{AK}_{\mathbf{0}}^{=}$-derivations initial segments of which constitute the branches of $\mathcal{C}$.

It is now possible to use the derivation-tree $\mathcal{C}$ in an easy inductive proof for the $\mathbf{AK}_{\mathbf{0}}^{=}$-consistency of the equality $\tau = \sigma$ by combining the following two properties of $\mathcal{C}$: Firstly, as inspection shows, $\mathcal{C}$ does not contain any contradictions with respect to $=_{\mu}$. And secondly, $\mathcal{C}$ can be considered to be the (positive) result of *loop-checking* for all possible derivations without VAR-applications from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$. – Let $\tilde{\mathcal{D}}$ be an arbitrary given derivation from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ without applications of VAR (the following argument has to be refined for derivations with applications of VAR). If looping does not occur in $\tilde{\mathcal{D}}$, then $|\tilde{\mathcal{D}}| \leq 7$ must hold and $\tilde{\mathcal{D}}$ has to be contained in one of the 6 different initial segments of $\mathbf{AK}_{\mathbf{0}}^{=}$-derivations from $\tau = \sigma$ gathered in $\mathcal{C}$; hence the conclusion of $\tilde{\mathcal{D}}$ must occur among the formulas in $\mathcal{C}$ and cannot be a contradiction. However, if looping does occur in $\tilde{\mathcal{D}}$, then by cutting out a loop from $\tilde{\mathcal{D}}$ we get a shorter derivation $\tilde{\mathcal{D}}_0$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ from $\tau = \sigma$ of smaller depth $|\tilde{\mathcal{D}}_0| < |\tilde{\mathcal{D}}|$, but with the same conclusion as $\tilde{\mathcal{D}}$. Therefore we can apply the induction hypotheses to $\tilde{\mathcal{D}}_0$ and conclude that the conclusion of $\tilde{\mathcal{D}}$ is no contradiction.

In order to give an indication about the particular relationship between the systems $\mathbf{AK}_{\mathbf{0}}^{=}$ and $\mathbf{HB}_{\mathbf{0}}^{=}$ that is described in this paper, we observe[11] the following: By reflecting the downwards-growing derivation-tree $\mathcal{C}$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ at a horizontal line it is possible to obtain an upwards-growing prooftree $\text{Refl}(\mathcal{C})$ in the system $\mathbf{HB}_{\mathbf{0}}^{=}$ with occurrences of open assumption classes. Thereby all applications of $\text{UNFOLD}_{l/r}$ in $\mathcal{C}$ are "reflected" into applications of $\text{FOLD}_{l/r}$ in $\text{Refl}(\mathcal{C})$ and all branchings DECOMP into applications of ARROW. To transform $\text{Refl}(\mathcal{C})$ into a derivation $\mathcal{D}$ in $\mathbf{HB}_{\mathbf{0}}^{=}$ without open assumptions it is merely necessary (1) to extend $\text{Refl}(\mathcal{C})$ above each of its leaves by one or two applications of $\text{FOLD}_{l/r}$, (2) to transfer respective assumption markers up to the new formulas at the top of the extended prooftree and (3) to redirect the bindings described by these markers to respective applications of ARROW below, thereby also changing these into ARROW/FIX-applications. In this way the derivation $\mathcal{D}$ in $\mathbf{HB}_{\mathbf{0}}^{=}$ without open assumption classes suggestively depicted in Figure 5 is reached.

And similarly, by reflecting the derivation $\mathcal{D}$ from Figure 5 at a horizontal line in an analogous way it is possible to get a downwards-growing derivation-tree $\text{Refl}(\mathcal{D})$ from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$, which—although slightly different from $\mathcal{C}$—like $\mathcal{C}$ can be taken as the basis of an inductive argument for showing the $\mathbf{AK}_{\mathbf{0}}^{=}$-consistency of $\tau = \sigma$.

This example suggests a very direct relationship between derivations in $\mathbf{HB}_{\mathbf{0}}^{=}$ without open assumption classes having conclusion $\tilde{\tau} = \tilde{\sigma}$ (for some $\tilde{\tau}, \tilde{\sigma} \in \textit{can-}\mu Tp$) and finite downwards-growing trees of consequences from the same equation $\tilde{\tau} = \tilde{\sigma}$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ that are the result of loop-checking and that facilitate easy inductive proofs for the consistency of $\tilde{\tau} = \tilde{\sigma}$ relative to $\mathbf{AK}_{\mathbf{0}}^{=}$.

[11] J.W. Klop noted this for a similar example in slightly different, but comparable proof systems.

Figure 5 The derivation $\mathcal{D}$ of $\mu\alpha.((\alpha\to\alpha)\to\alpha) = \mu\alpha.(\alpha\to(\alpha\to\alpha))$ in $\mathbf{HB}_{\mathbf{0}}^{=}$ without open assumption classes.

$$\dfrac{\dfrac{\dfrac{\dfrac{(\ldots)^x}{\tau=\sigma}\quad \dfrac{\dfrac{\dfrac{(\ldots)^y}{\tau\to\tau=\sigma}\quad\dfrac{(\ldots)^x}{\tau=\sigma}}{(\tau\to\tau)\to\tau=\sigma\to\sigma}}{\tau=\sigma\to\sigma}}{\tau\to\tau=\sigma\to(\sigma\to\sigma)}\,y}{\tau\to\tau=\sigma}\qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(\ldots)^x}{\tau=\sigma}\quad\dfrac{(\ldots)^z}{\tau=\sigma\to\sigma}}{\tau\to\tau=\sigma\to(\sigma\to\sigma)}}{\tau\to\tau=\sigma}\quad \dfrac{(\ldots)^x}{\tau=\sigma}}{(\tau\to\tau)\to\tau=\sigma\to\sigma}\,z}{\tau=\sigma\to\sigma}}{\dfrac{(\tau\to\tau)\to\tau=\sigma\to(\sigma\to\sigma)}{\underbrace{\mu\alpha.((\alpha\to\alpha)\to\alpha)}_{\equiv:\tau}=\underbrace{\mu\alpha.(\alpha\to(\alpha\to\alpha))}_{\equiv:\sigma}}\,\mathrm{FOLD}_{l/r}}\,(\text{ARR./FIX})_x$$

5 The extension e-$\mathbf{HB}_{\mathbf{0}}^{=}$ of $\mathbf{HB}_{\mathbf{0}}^{=}$

For obtaining a precise formulation of the observation in the previous section it will be helpful [12] to extend the system $\mathbf{HB}_{\mathbf{0}}^{=}$ by three more coinductive fixed-point rules.

Definition 5.1 (The extension e-$\mathbf{HB}_{\mathbf{0}}^{=}$ of the system $\mathbf{HB}_{\mathbf{0}}^{=}$). The extension **e-**$\mathbf{HB}_{\mathbf{0}}^{=}$ of the system $\mathbf{HB}_{\mathbf{0}}^{=}$ has the same *formulas* and *axioms* as $\mathbf{HB}_{\mathbf{0}}^{=}$, allows the same *marked assumptions* and contains all *derivation rules* of $\mathbf{HB}_{\mathbf{0}}^{=}$. Additionally, **e-**$\mathbf{HB}_{\mathbf{0}}^{=}$ possesses the rules VAR/FIX, FOLD_l/FIX and FOLD_r/FIX with applications of the respective form

$$\dfrac{\begin{matrix}[\,\tau=\sigma\,]^x\\ \mathcal{D}_0\\ \tau_0=\sigma_0\end{matrix}}{\tau=\sigma}\,(R/\mathrm{FIX})_x \quad \text{(if side-cond.(s) } \mathbf{I} \text{ (and } \mathbf{C} \text{ for } R=\mathrm{VAR})) \tag{5.1}$$

(with some $\tau,\sigma,\tau_0,\sigma_0 \in \mathit{can}\text{-}\mu Tp$ and $x\in Mk$), given that $\dfrac{\tau_0=\sigma_0}{\tau=\sigma}\,R$ is an application of a rule $R\in\{\,\mathrm{FOLD}_{l/r},\ \mathrm{VAR}\,\}$ and that the respectively necessary side-conditions described below are satisfied. At such applications the class $[\tau=\sigma]^x$ of open marked assumptions of the form $(\tau=\sigma)^x$ in $\mathcal{D}_0$ gets discharged. The side-condition **I** requires that the assumption class $\tau=\sigma$ in $\mathcal{D}_0$ is inhabitated (not empty). For applications of VAR/FIX the side-condition **C** demands furthermore that $\mathcal{D}_0$ is *contractive* with respect to the marked open assumptions $(\tau=\sigma)^x$, which means that for every thread in $\mathcal{D}_0$ from a marked open assumption $(\tau=\sigma)^x$ downwards at least one application of ARROW or ARROW/FIX is passed.

A formula $\tau=\sigma$ is a *theorem* of **e-**$\mathbf{HB}_{\mathbf{0}}^{=}$ (symbolically denoted by $\vdash_{\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}} \tau=\sigma$) iff there is a derivation $\mathcal{D}$ in **e-**$\mathbf{HB}_{\mathbf{0}}^{=}$ with conclusion $\tau=\sigma$ and with the property that all marked assumptions have been discharged at respective applications of $\mathrm{FOLD}_{l/r}$/FIX, of VAR/FIX or of ARROW/FIX.

[12] We will indicate later why this preparatory step indeed helps to obtain a more satisfying result.

It is easy to see that either of two following more special requirements $\mathbf{C}_1$ and $\mathbf{C}_2$ could have been used instead of the side-condition $\mathbf{C}$ for applications of VAR/FIX of the form (5.1) (with $R = \text{VAR}$) with an equivalent definition as the result: $\mathbf{C}_1$ is the condition "$\mathcal{D}_0$ contains at least one application of ARROW or ARROW/FIX" and $\mathbf{C}_2$ demands that "there is at least one application of a rule different from VAR in $\mathcal{D}_0$".

Although the system $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ is an extension of $\mathbf{HB}_{\mathbf{0}}^{=}$, no new theorems become derivable:

Theorem 5.2 (Equivalence of the systems $\mathbf{HB}_{\mathbf{0}}^{=}$ and $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$). *The system $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ is a conservative extension of $\mathbf{HB}_{\mathbf{0}}^{=}$ and hence*[13] *the systems $\mathbf{HB}_{\mathbf{0}}^{=}$ and $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ are equivalent (i.e. they possess the same theorems). More specifically, every derivation $\mathcal{D}$ in $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ can effectively be transformed into a derivation $\mathcal{D}'$ in $\mathbf{HB}_{\mathbf{0}}^{=}$ with the same conclusion and the same (if any) open assumption classes.*

Hint at the Proof. This theorem is a consequence of the fact that the rules $\text{FOLD}_{l/r}$/FIX and VAR/FIX of $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ are *admissible* rules of the system $\mathbf{HB}_{\mathbf{0}}^{=}$, i.e. rules that can effectively be eliminated from an arbitrary given derivation in $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ with the final result of a derivation in $\mathbf{HB}_{\mathbf{0}}^{=}$. The reason for this is that the "deductive power" of an application of $\text{FOLD}_{l/r}$/FIX or VAR/FIX in a derivation $\mathcal{D}$ can always be emulated by the "deductive power" of a respective application of ARROW/FIX in a derivation $\mathcal{D}'$ closely related to $\mathcal{D}$. We will demonstrate this only in the very special case of a derivation $\mathcal{D}$ in $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ ending with an application of FOLD_l/FIX that is itself immediately preceded by an application of ARROW: Let $\mathcal{D}$ be a derivation in $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ of the form

$$\dfrac{\dfrac{\begin{matrix}\langle\tau=\sigma\rangle^{\boldsymbol{x}}\\ \mathcal{D}_{01}\\ \tau_{01}[\tau/\alpha]=\sigma_1\end{matrix} \qquad \begin{matrix}\langle\tau=\sigma\rangle^{\boldsymbol{x}}\\ \mathcal{D}_{02}\\ \tau_{02}[\tau/\alpha]=\sigma_2\end{matrix}}{\tau_{01}[\tau/\alpha]\to\tau_{02}[\tau/\alpha]=\sigma_1\to\sigma_2}\,\text{ARROW}}{\underbrace{\mu\alpha.\,(\tau_{01}\to\tau_{02})}_{\equiv\tau}=\underbrace{\sigma_1\to\sigma_2}_{\equiv\sigma}}\,(\text{FOLD}_l/\text{FIX})_{\boldsymbol{x}}$$

and denote by $\mathcal{D}_0$ the sub-derivation of $\mathcal{D}$ that leads up to the application of FOLD_l/FIX at the bottom of $\mathcal{D}$. This application of FOLD_l/FIX can now be eliminated by extending $\mathcal{D}_0$ above each of the open assumptions $(\tau=\sigma)^x$ in $\mathcal{D}_0$ by an application of FOLD_l and by discharging the marked open assumptions $(\tau_{01}[\tau/\alpha]\to\tau_{02}[\tau/\alpha]=\sigma)^{\boldsymbol{x}}$ in the new leaves at an application of ARROW/FIX that arises by renaming from the penultimate rule application in $\mathcal{D}$, the application of ARROW. The result is the derivation $\mathcal{D}'$ of the following form:

[13] Since $\mathbf{HB}_{\mathbf{0}}^{=}$ and $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ have the same formulas.

$$\dfrac{\dfrac{\dfrac{(\tau_{01}[\tau/\alpha] \to \tau_{02}[\tau/\alpha] = \sigma)^{\boldsymbol{x}}}{\langle \tau = \sigma \rangle}\,\mathrm{FOLD}_l}{\tau_{01}[\tau/\alpha] = \sigma_1}\,\mathcal{D}_{01} \qquad \dfrac{\dfrac{(\tau_{01}[\tau/\alpha] \to \tau_{02}[\tau/\alpha] = \sigma)^{\boldsymbol{x}}}{\langle \tau = \sigma \rangle}\,\mathrm{FOLD}_l}{\tau_{02}[\tau/\alpha] = \sigma_2}\,\mathcal{D}_{02}}{\dfrac{\tau_{01}[\tau/\alpha] \to \tau_{02}[\tau/\alpha] = \sigma_1 \to \sigma_2}{\underbrace{\mu\alpha.\,(\tau_{01} \to \tau_{02})}_{\equiv \tau} = \underbrace{\sigma_1 \to \sigma_2}_{\equiv \sigma}}\,\mathrm{FOLD}_l}\,(\mathrm{ARROW/FIX})_{\boldsymbol{x}}$$

Similar effective eliminations can be carried out for all applications of $\mathrm{FOLD}_{l/r}$/FIX and VAR/FIX in arbitrary given $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$-derivations. □

As an immediate consequence of this theorem and of Theorem 3.2 we find the following corollary.

Corollary 5.3 (Sound- and Completeness of $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ with respect to $=_\mu$). *The axiom system* $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ *is sound and complete with respect to strong recursive type equivalence* $=_\mu$*, i.e. for all* $\tau, \sigma \in$ *can-*μTp *it holds that*

$$\vdash_{\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}} \tau = \sigma \quad \Longleftrightarrow \quad \tau =_\mu \sigma \ .$$

6 Consistency-Unfoldings

In a second step of the formulation of the observation in Section 4 into a precise statement we will formalize finite downwards-growing trees of consequences in $\mathbf{AK}_{\mathbf{0}}^{=}$ as "consistency-unfoldings", which allow to prove easily the $\mathbf{AK}_{\mathbf{0}}^{=}$-consistency of the formulas at their respective roots. – We have to give a definition of "partial consistency-unfoldings" first.

Definition 6.1 (Partial Consistency-Unfoldings in $\mathbf{AK}_{\mathbf{0}}^{=}$). For all recursive types $\tau, \sigma \in$ *can-*μTp a *partial consistency-unfolding* (a *p.c.u.*) $\mathcal{C}$ of the equation $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ is a finite downwards-growing "tree of consequences" of $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ that together with the assertion "$\mathcal{C}$ is a p.c.u. of $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$" can be formed by a finite number of applications of the following 5 generation rules. Thereby the notion of an *unbound leaf-occurrence of a marked formula* (an *u.l.o.m.f.*) in a p.c.u. is defined in parallel:[14]

(i) For all $\tau, \sigma \in$ *can-*μTp and $x \in Mk$ $\boxed{(\tau = \sigma)^x}$ is a p.c.u. $\mathcal{C}$ from $\tau = \sigma$. The occurrence of $(\tau = \sigma)^x$ in $\mathcal{C}$ is the single u.l.o.m.f. in $\mathcal{C}$. – Furthermore for all $\tau \in$ *can-*μTp $\boxed{\tau = \tau}$ is a p.c.u. of $\tau = \tau$, which contains no u.l.o.m.f.'s in $\mathcal{C}$.

[14] In the following clauses the addition "in $\mathbf{AK}_{\mathbf{0}}^{=}$" in statements like "$\mathcal{C}$ is a p.c.u. in $\mathbf{AK}_{\mathbf{0}}^{=}$" is always dropped. Auxiliary framed boxes are used to delimit the defined p.c.u.'s from the surrounding text. Here and later we will allow formulas $(\tau = \sigma)^{\boldsymbol{m}}$ with $\tau, \sigma \in$ *can-*μTp and a boldface-marker $\boldsymbol{m}$ to stand either (a) for the unmarked formula $\tau = \sigma$ or (b) for a marked formula $(\tau = \sigma)^x$ with some $x \in Mk$ such that this marker is then assumed to be denoted by $\boldsymbol{m}$.

(ii) For all $\tau, \sigma, \tau_0, \sigma_0 \in \mathit{can}\text{-}\mu Tp$
$$\boxed{\begin{array}{c} \dfrac{\tau = \sigma}{(\tau_0 = \sigma_0)^{\boldsymbol{m}_0}}\, R \\ \mathcal{C}_0 \end{array}}$$
is a p.c.u. $\mathcal{C}$ of $\tau = \sigma$ given that $\mathcal{C}_0$ is a p.c.u. of $\tau_0 = \sigma_0$ and that R is an application of a rule $\text{UNFOLD}_{l/r}$ or VAR. An u.l.o.m.f. in $\mathcal{C}$ is such an occurrence of a marked formula in $\mathcal{C}$ within its subtree $\mathcal{C}_0$ that corresponds to an u.l.o.m.f. in $\mathcal{C}_0$.

(iii) For all $\tau, \sigma, \tau_0, \sigma_0 \in \mathit{can}\text{-}\mu Tp$ and $x \in \mathit{Mk}$
$$\boxed{\begin{array}{c} \dfrac{(\tau = \sigma)^x}{(\tau_0 = \sigma_0)^{\boldsymbol{m}_0}}\, R \\ \mathcal{C}_0 \\ [\tau = \sigma]^x \end{array}}$$
is a p.c.u. $\mathcal{C}$ of $\tau = \sigma$ given that (1) $\mathcal{C}_0$ is a p.c.u. of $\tau_0 = \sigma_0$ in which the (indicated) class $[\tau = \sigma]^x$ of all u.l.o.m.f.'s of the form $(\tau = \sigma)^x$ is non-empty and that either (2a) R is an application of a rule $\text{UNFOLD}_{l/r}$ or (2b) R is an application of VAR and $\mathcal{C}_0$ contains at least one application of a rule different from VAR. – All occurrences of $(\tau = \sigma)^x$ within the subtree $\mathcal{C}_0$ of $\mathcal{C}$, that correspond to u.l.o.m.f.'s in $\mathcal{C}_0$, are *bound back* in $\mathcal{C}$ to the occurrence of $(\tau = \sigma)^x$ at the root. For all marked formulas $(\tilde{\tau} = \tilde{\sigma})^{\tilde{x}}$ different from $(\tau = \sigma)^x$ the unbound leaf-occurrences of this marked formula correspond uniquely and in an obvious way to the u.l.o.'s of $(\tilde{\tau} = \tilde{\sigma})^{\tilde{x}}$ within the subtree $\mathcal{C}_0$ of $\mathcal{C}$.

(iv)
$$\boxed{\begin{array}{c} \dfrac{\tau_{01} \to \tau_{02} = \sigma_{01} \to \sigma_{02}}{(\tau_{01} = \sigma_{01})^{\boldsymbol{m}_{01}} \qquad (\tau_{02} = \sigma_{02})^{\boldsymbol{m}_{02}}}\ \text{DECOMP} \\ \mathcal{C}_{01} \qquad\qquad \mathcal{C}_{02} \end{array}}$$
is a p.c.u. $\mathcal{C}$ of the formula $\tau_{01} \to \tau_{02} = \sigma_{01} \to \sigma_{02}$ for all $\tau_{01}, \tau_{02}, \sigma_{01}, \sigma_{02} \in \mathit{can}\text{-}\mu Tp$, given that $\mathcal{C}_{0i}$ is a p.c.u. of $\tau_{0i} = \sigma_{0i}$ for each $i \in \{1, 2\}$. The u.l.o.m.f.'s in $\mathcal{C}$ correspond uniquely and in an obvious way to the u.l.o.m.f.'s in either of its immediate subtrees $\mathcal{C}_{01}$ or $\mathcal{C}_{02}$.

(v)
$$\boxed{\begin{array}{c} \dfrac{(\tau_{01} \to \tau_{02} = \sigma_{01} \to \sigma_{02})^x}{(\tau_{01} = \sigma_{01})^{\boldsymbol{m}_{01}} \qquad (\tau_{02} = \sigma_{02})^{\boldsymbol{m}_{02}}}\ \text{DECOMP} \\ \mathcal{C}_{01} \qquad\qquad \mathcal{C}_{02} \\ \langle \tau = \sigma \rangle^x \qquad\qquad \langle \tau = \sigma \rangle^x \end{array}}$$
(with some $x \in \mathit{Mk}$ and with $\tau :\equiv \tau_{01} \to \tau_{02}$ and $\sigma :\equiv \sigma_{01} \to \sigma_{02}$) is a p.c.u. $\mathcal{C}$ of $\tau_{01} \to \tau_{02} = \sigma_{01} \to \sigma_{02}$ for all $\tau_{01}, \tau_{02}, \sigma_{01}, \sigma_{02} \in \mathit{can}\text{-}\mu Tp$ given that $\mathcal{C}_{0i}$ is a p.c.u. from $\tau_{0i} = \sigma_{0i}$ for each $i \in \{1, 2\}$ and that there is at least one unbound leaf-occurrence of the marked formula $(\tau_{01} \to \tau_{02} = \sigma_{01} \to \sigma_{02})^x$ in either $\mathcal{C}_{01}$ or in $\mathcal{C}_{02}$. – All occurrences of $(\tau = \sigma)^x$ within either of the immediate subtrees $\mathcal{C}_{01}$ and $\mathcal{C}_{02}$ of $\mathcal{C}$, that correspond to u.l.o.m.f.'s in $\mathcal{C}_{01}$ or $\mathcal{C}_{02}$, are *bound back* in $\mathcal{C}$ to the occurrence of $(\tau = \sigma)^x$ at the root (and hence are not u.l.o.m.f.'s in $\mathcal{C}$). For every marked formula $(\tilde{\tau} = \tilde{\sigma})^{\tilde{x}}$ different from $(\tau = \sigma)^x$ the unbound leaf-occurrences of this

marked formula correspond uniquely and in an obvious way to the u.l.o.'s of $(\tilde{\tau} = \tilde{\sigma})^{\tilde{x}}$ within either of the sub-p.c.u.'s $\mathcal{C}_{01}$ or $\mathcal{C}_{02}$ of $\mathcal{C}$.

The *depth* $|\mathcal{C}|$ of a p.c.u. $\mathcal{C}$ is defined as the depth of the underlying (derivation-) tree.

Definition 6.2 (Consistency-Unfoldings in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$). Let τ and σ be recursive types in canonical form. A partial consistency-unfolding $\mathcal{C}$ of $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$ is called a *consistency-unfolding* (a *c.u.*) of $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$ if and only if $\mathcal{C}$ does not contain any unbound leaf-occurrences of marked formulas.

According to these definitions the derivation-tree $\mathcal{C}$ depicted in Figure 4 can now be recognized as a p.c.u. in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$ without u.l.o.m.f.'s and hence as a consistency-unfolding of $\mu\alpha.\,((\alpha \to \alpha) \to \alpha) = \mu\alpha.\,(\alpha \to (\alpha \to \alpha))$ in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$. – An important statement about consistency-unfoldings is expressed in the following lemma that requires a somewhat technical, but not difficult proof.

Lemma 6.3 *Let $\tau, \sigma \in$ can-μTp and $\mathcal{C}$ be a consistency-unfolding of $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$. Then for all equations $\chi_1 = \chi_2$ occuring in $\mathcal{C}$ it holds that $\mathcal{L}(\chi_1) = \mathcal{L}(\chi_2)$, i.e. that χ_1 and χ_2 have the same leading symbols.*

It should perhaps be mentioned that if the hypotheses "let $\mathcal{C}$ be a c.u. ..." in Lemma 6.3 were replaced by "let $\mathcal{C}$ be a p.c.u. ...", then a wrong assertion would result. This can already be seen from the easy example of the contradiction $\bot = \top$ with respect to $=_\mu$ that occurs in the marked assumption $(\bot = \top)^x$, which by Definition 6.1 is a partial consistency-unfolding of $\bot = \top$ in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$.

The following theorem establishes the link motivated by the example in Section 4 between the notions of "$\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$-consistency" and "consistency-unfolding in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$".

Theorem 6.4 *For all recursive types $\tau, \sigma \in$ can-μTp it holds that:*

$$\tau = \sigma \text{ is } \mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}\text{-consistent} \iff \text{There exists a consistency-unfolding of } \tau = \sigma \text{ in } \mathbf{AK}_{\mathbf{0}}^{\mathbf{=}} . \tag{6.1}$$

Hint at the Proof. Let $\tau, \sigma \in$ *can-μTp*. The implication "$\Leftarrow$" in (6.1) follows by a generalization using Lemma 6.3 of the intuitive argumentation sketched in Section 4 for the example of the consistency-unfolding in Figure 4. The implication "$\Rightarrow$" in (6.1) follows by an analogous, in fact as good as 'dual' argument to that one used in a proof (following [4]) for the completeness of $\mathbf{HB}_{\mathbf{0}}^{\mathbf{=}}$ with respect to $=_\mu$: For an arbitrary given equation $\tau = \sigma$ between recursive types $\tau, \sigma \in$ *can-μTp* for which $\tau =_\mu \sigma$ holds a consistency-unfolding of $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$ can be reached by building up the "tree of consequences" of this equation in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$ in successive extension stages, cutting off branches always as soon as "looping" occurs or as soon as a formula $\chi = \chi$ has been encountered. There cannot be infinite branches in the arising derivation-tree due to the fact that the set of conclusions of derivations from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{\mathbf{=}}$ is always finite, if equations that arise from each other by applications of VAR are not counted separately. □

7 A duality between the proof systems e-$\mathbf{HB}_\mathbf{0}^=$ and $\mathbf{AK}_\mathbf{0}^=$

In a third step of our formalization of the observation in Section 4 we will now give a definition of a pair of reflection mappings $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$ between p.c.u.'s in $\mathbf{AK}_\mathbf{0}^=$ and derivations in e-$\mathbf{HB}_\mathbf{0}^=$.

Definition 7.1 (Reflection Mappings $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$). The *reflection mapping* $\mathcal{D}(\cdot)$ from partial consistency-unfoldings in $\mathbf{AK}_\mathbf{0}^=$ to derivations in $\mathbf{HB}_\mathbf{0}^=$ (with possibly open assumption-classes) is defined by induction on the depth $|\tilde{\mathcal{C}}|$ of a p.c.u. $\tilde{\mathcal{C}}$ in $\mathbf{AK}_\mathbf{0}^=$ according to 5 inductive clauses, which refer to the 5 cases in the inductive definition of p.c.u.'s in Definition 6.1; these clauses are indicated in Figure 6 through the arrows $\overset{\mathcal{D}(\cdot)}{\longmapsto}$ between the boxes on the left- and on the right-hand side. The definition of the *reflection mapping* $\mathcal{C}(\cdot)$ in the opposite direction can be carried out for all derivations $\tilde{\mathcal{D}}$ in e-$\mathbf{HB}_\mathbf{0}^=$ (with possibly open assumption classes) by induction on the depth $|\tilde{\mathcal{D}}|$ of $\tilde{\mathcal{D}}$ with clauses that apart from the base case distinguish the 8 cases of different rules in e-$\mathbf{HB}_\mathbf{0}^=$, applications of which may occur as the last rule application in $\tilde{\mathcal{D}}$ (if $|\tilde{\mathcal{D}}| > 0$). These in total 9 cases are described in the 5 inductive clauses of the definition of $\mathcal{C}(\cdot)$, which clauses are indicated through the arrows $\overset{\mathcal{C}(\cdot)}{\longmapsto}$ from right to left in Figure 6. For the second and the third clause in both definitions we use a bijective correspondence defined through the table

Rule $R^{(cu)}$ in $\mathbf{AK}_\mathbf{0}^=$	UNFOLD_l	UNFOLD_r	VAR
Rule $R^{(d)}$ in $\mathbf{HB}_\mathbf{0}^=$	FOLD_l	FOLD_r	VAR

between rules in $\mathbf{AK}_\mathbf{0}^=$ and rules in $\mathbf{HB}_\mathbf{0}^=$ respectively denoted by $R^{(cu)}$ and $R^{(d)}$.

The well-definedness of $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$ as functions between the set of p.c.u.'s in $\mathbf{AK}_\mathbf{0}^=$ and the set of derivations in e-$\mathbf{HB}_\mathbf{0}^=$ with possibly open assumption classes can be shown by induction on the depth of the elements in the domain of the respective mapping.

We are now able to state our main theorem.

Theorem 7.2 (A Duality between derivations in e-$\mathbf{HB}_\mathbf{0}^=$ and consistency-unfoldings in $\mathbf{AK}_\mathbf{0}^=$). *There is a bijective functional relationship between derivations in* e-$\mathbf{HB}_\mathbf{0}^=$ *without open assumption classes and consistency-unfoldings in* $\mathbf{AK}_\mathbf{0}^=$ *via the reflection functions $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$ defined in Definition 7.1 in the following sense:*

(i) For every consistency-unfolding $\tilde{\mathcal{C}}$ of $\tau = \sigma$ in $\mathbf{AK}_\mathbf{0}^=$ *(with some $\tau, \sigma \in$ can-μTp) its reflection $\mathcal{D}(\tilde{\mathcal{C}})$ is a derivation in* e-$\mathbf{HB}_\mathbf{0}^=$ *with conclusion $\tau = \sigma$ and without open assumption classes.*

(ii) For every derivation $\tilde{\mathcal{D}}$ in e-$\mathbf{HB}_\mathbf{0}^=$ *without open assumption classes and with conclusion $\tau = \sigma$ (for some $\tau, \sigma \in$ can-μTp) its reflection $\mathcal{C}(\tilde{\mathcal{D}})$ is a consistency-unfolding of $\tau = \sigma$ in* $\mathbf{AK}_\mathbf{0}^=$.

(iii) The functions $\mathcal{D}(\cdot)$ of taking the reflection of a consistency-unfolding in $\mathbf{AK}_\mathbf{0}^=$ *and $\mathcal{C}(\cdot)$ of taking the reflection of a derivation in* e-$\mathbf{HB}_\mathbf{0}^=$ *without open assump-*

Figure 6 Inductive definition of the reflection mappings $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$ between partial consistency-unfoldings $\tilde{\mathcal{C}}$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ and derivations $\tilde{\mathcal{D}}$ in $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ (with possibly open assumption-classes).

$$\boxed{(\tau = \sigma)^{\boldsymbol{m}}} \quad \overset{\mathcal{D}(\cdot)}{\longmapsto} \underset{\mathcal{C}(\cdot)}{\longleftarrow\!\!\mid} \quad \boxed{(\tau = \sigma)^{\boldsymbol{m}}}$$

$$\boxed{\begin{array}{c}\dfrac{\tau = \sigma}{(\tau_0 = \sigma_0)^{\boldsymbol{m}_0}}\, R^{(cu)} \\ \tilde{\mathcal{C}}_0 \; (\equiv \mathcal{C}(\tilde{\mathcal{D}}_0))\end{array}} \quad \overset{\mathcal{D}(\cdot)}{\longmapsto} \underset{\mathcal{C}(\cdot)}{\longleftarrow\!\!\mid} \quad \boxed{\begin{array}{c}\tilde{\mathcal{D}}_0 \; (\equiv \mathcal{D}(\tilde{\mathcal{C}}_0)) \\ \dfrac{\tau_0 = \sigma_0}{\tau = \sigma}\, R^{(d)}\end{array}}$$

(for *corresponding* $R^{(cu)} \in \{\text{UNFOLD}_{l/r}, \text{VAR}\}$ and $R^{(d)} \in \{\text{FOLD}_{l/r}, \text{VAR}\}$)

$$\boxed{\begin{array}{c}\dfrac{(\tau = \sigma)^{x}}{(\tau_0 = \sigma_0)^{\boldsymbol{m}_0}}\, R^{(cu)} \\ \tilde{\mathcal{C}}_0 \; (\equiv \mathcal{C}(\tilde{\mathcal{D}}_0)) \\ [\tau = \sigma]^{x}\end{array}} \quad \overset{\mathcal{D}(\cdot)}{\longmapsto} \underset{\mathcal{C}(\cdot)}{\longleftarrow\!\!\mid} \quad \boxed{\begin{array}{c}[\tau = \sigma]^{x} \\ \tilde{\mathcal{D}}_0 \; (\equiv \mathcal{D}(\tilde{\mathcal{C}}_0)) \\ \dfrac{\tau_0 = \sigma_0}{\tau = \sigma}\, (R^{(d)}/\text{FIX})_x\end{array}}$$

(for *corresponding* $R^{(cu)} \in \{\text{UNFOLD}_{l/r}, \text{VAR}\}$ and $R^{(d)} \in \{\text{FOLD}_{l/r}, \text{VAR}\}$)

$$\boxed{\begin{array}{c}\dfrac{\tau_{01} \to \tau_{02} = \sigma_{01} \to \sigma_{02}}{(\tau_{01} = \sigma_{01})^{\boldsymbol{m}_{01}} \qquad (\tau_{02} = \sigma_{02})^{\boldsymbol{m}_{02}}}\, \text{DECOMP} \\ \tilde{\mathcal{C}}_{01} \; (\equiv \mathcal{C}(\tilde{\mathcal{D}}_{01})) \qquad \tilde{\mathcal{C}}_{02} \; (\equiv \mathcal{C}(\tilde{\mathcal{D}}_{02}))\end{array}} \quad \overset{\mathcal{D}(\cdot)}{\longmapsto} \underset{\mathcal{C}(\cdot)}{\longleftarrow\!\!\mid}$$

$$\overset{}{\longmapsto} \underset{}{\longleftarrow\!\!\mid} \quad \boxed{\begin{array}{c}\tilde{\mathcal{D}}_{01} \; (\equiv \mathcal{D}(\tilde{\mathcal{C}}_{01})) \qquad \tilde{\mathcal{D}}_{02} \; (\equiv \mathcal{D}(\tilde{\mathcal{C}}_{02})) \\ \dfrac{\tau_{01} = \sigma_{01} \qquad \tau_{02} = \sigma_{02}}{\tau_{01} \to \tau_{02} = \sigma_{01} \to \sigma_{02}}\, \text{ARROW}\end{array}}$$

$$\boxed{\begin{array}{c}\dfrac{(\overbrace{\tau_{01} \to \tau_{02}}^{\equiv \tau} = \overbrace{\sigma_{01} \to \sigma_{02}}^{\equiv \sigma})^{x}}{(\tau_{01} = \sigma_{01})^{\boldsymbol{m}_{01}} \qquad (\tau_{02} = \sigma_{02})^{\boldsymbol{m}_{02}}}\, \text{DECOMP} \\ \tilde{\mathcal{C}}_{01} \; (\equiv \mathcal{C}(\tilde{\mathcal{D}}_{01})) \qquad \tilde{\mathcal{C}}_{02} \; (\equiv \mathcal{C}(\tilde{\mathcal{D}}_{02})) \\ \langle \tau = \sigma \rangle^{x} \qquad \langle \tau = \sigma \rangle^{x}\end{array}} \quad \overset{\mathcal{D}(\cdot)}{\longmapsto} \underset{\mathcal{C}(\cdot)}{\longleftarrow\!\!\mid}$$

$$\overset{}{\longmapsto} \underset{}{\longleftarrow\!\!\mid} \quad \boxed{\begin{array}{c}\langle \tau = \sigma \rangle^{x} \qquad \langle \tau = \sigma \rangle^{x} \\ \tilde{\mathcal{D}}_{01} \; (\equiv \mathcal{D}(\tilde{\mathcal{C}}_{01})) \qquad \tilde{\mathcal{D}}_{02} \; (\equiv \mathcal{D}(\tilde{\mathcal{C}}_{02})) \\ \dfrac{\tau_{01} = \sigma_{01} \qquad \tau_{02} = \sigma_{02}}{\underbrace{\tau_{01} \to \tau_{02}}_{\equiv \tau} = \underbrace{\sigma_{01} \to \sigma_{02}}_{\equiv \sigma}}\, (\text{ARROW}/\text{FIX})_x\end{array}}$$

Figure 7 Example consisting of a consistency-unfolding $\tilde{\mathcal{C}}$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ and of a derivation $\tilde{\mathcal{D}}$ in $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ that are each other's *reflection* via the mappings $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$, i.e. for which it holds that $\mathcal{D}(\tilde{\mathcal{C}}) = \tilde{\mathcal{D}}$ and $\mathcal{C}(\tilde{\mathcal{D}}) = \tilde{\mathcal{C}}$:

$$\tilde{\mathcal{D}} := \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(\tau=\sigma)^x \qquad \overline{\bot=\bot}}{\tau\to\bot=\sigma\to\bot}\,\text{ARROW}}{\tau=\sigma\to\bot} \qquad \overline{\bot=\bot}}{\tau\to\bot=(\sigma\to\bot)\to\bot}\,\text{ARROW}}{\tau\to\bot=\sigma}\,\text{FOLD}_r}{\underbrace{\mu\alpha.\,(\alpha\to\bot)}_{\equiv:\,\tau}=\underbrace{\mu\beta.\,((\beta\to\bot)\to\bot)}_{\equiv:\,\sigma}}\,(\text{FOLD}_l/\text{FIX})_x$$

$$\tilde{\mathcal{C}} := \begin{array}{cl} (\mu\alpha.\,(\alpha\to\bot)=\mu\beta.\,((\beta\to\bot)\to\bot))^x & \text{UNFOLD}_l \\ \hline \tau\to\bot=\sigma & \text{UNFOLD}_r \\ \hline \tau\to\bot=(\sigma\to\bot)\to\bot & \text{DECOMP} \\ \hdashline \tau=\sigma\to\bot \qquad\qquad \bot=\bot & \\ \hline \tau\to\bot=\sigma\to\bot & \text{DECOMP} \\ \hdashline (\tau=\sigma)^x \qquad\qquad \bot=\bot & \end{array}$$

tion-classes are each other's inverse.

The very immediate kind of this bijective functional relationship and the possibility to visualize the reflection functions in a geometrical way is reason to call it a duality.

Sketch of Proof. All three items of the theorem (the third one can be split into the two assertions $\mathcal{D}\circ\mathcal{C}=\text{id}$ and $\mathcal{C}\circ\mathcal{D}=\text{id}$) can be shown by quite straightforward inductions using the inductive clauses in the definitions of $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$. In these inductions bookkeeping must be done as indicated in the below picture for respectively the set of open marked assumptions in an $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$-derivation $\tilde{\mathcal{D}}$ with conclusion $\tau=\sigma$ and for the classes of u.l.o.m.f.'s in a p.c.u. $\tilde{\mathcal{C}}$ of $\tau=\sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ (for arbitrary $\tau,\sigma\in \textit{can-}\mu Tp$):

$$\boxed{\begin{array}{c} (\tau=\sigma)^{\boldsymbol{m}} \\ \tilde{\mathcal{C}} \\ \{\,[\tau_i=\sigma_i]^{x_i}\,\}_{i=1,\ldots,n} \end{array}} \quad \begin{array}{c} \xmapsto{\mathcal{D}(\cdot)} \\ \xleftarrow[\mathcal{C}(\cdot)]{} \end{array} \quad \boxed{\begin{array}{c} \{\,[\tau_i=\sigma_i]^{x_i}\,\}_{i=1,\ldots,n} \\ \tilde{\mathcal{D}} \\ \tau=\sigma \end{array}}$$

Hereby the displayed family $\{\,[\tau_i=\sigma_i]^{x_i}\,\}_{i=1,\ldots,n}$ (with $n\in\mathbb{N}_0$, $\tau_i,\sigma_i\in \textit{can-}\mu Tp$ and $x_i\in Mk$ for $i=1,\ldots,n$) gathers in $\tilde{\mathcal{C}}$ precisely all n classes of u.l.o.m.f.'s and respectively assembles in $\tilde{\mathcal{D}}$ precisely all n open assumption classes. □

An example of a pair $(\tilde{\mathcal{D}},\tilde{\mathcal{C}})$ consisting of a derivation $\tilde{\mathcal{D}}$ in $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ without open assumption classes and of a consistency-unfolding $\tilde{\mathcal{C}}$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ that are each other's reflection via the operations $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$ is depicted in Figure 7.

This example makes it also very easy to explain why—with the aim of establishing a satisfying duality result—we have chosen to extend the system $\mathbf{HB}_{\mathbf{0}}^{=}$ first by some more rules to the system $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ before only later defining mutual transformations between p.c.u.'s in $\mathbf{AK}_{\mathbf{0}}^{=}$ and derivations in the extended system: If we had not done so, then we would not have been able to discharge the open marked assumption $(\tau = \sigma)^x$ in a derivation $\mathrm{Refl}(\tilde{\mathcal{C}})$ in $\mathbf{HB}_{\mathbf{0}}^{=}$ that arises by plain reflection from the c.u. $\tilde{\mathcal{C}}$ from Figure 7. Similarly as described for the example used in Section 4 we would have had to enlarge $\mathrm{Refl}(\tilde{\mathcal{C}})$ above this marked assumption by two additional applications (one of FOLD_l and one of FOLD_r) before being able to discharge the newly occuring open marked assumption $(\tau \to \bot = (\sigma \to \bot) \to \bot)^x$ in a thereby created derivation $\mathrm{Refl}(\tilde{\mathcal{C}})^*$ at an application of ARROW/FIX that results by renaming from the bottommost application of ARROW in $\mathrm{Refl}(\tilde{\mathcal{C}})$.[15] But due to the presence of the rules $\mathrm{FOLD}_{l/r}$/FIX in $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ it is in fact possible to transform the plain reflection $\mathrm{Refl}(\tilde{\mathcal{C}})$ of $\tilde{\mathcal{C}}$ into the derivation $\tilde{\mathcal{D}}$ in Figure 7 by merely renaming the bottommost application of FOLD_l in $\mathrm{Refl}(\tilde{\mathcal{C}})$ into FOLD_l/FIX and by discharging the open marked assumption $(\tau = \sigma)^x$ at this application.

This look at the example from Figure 7 can make it clear why it is actually not possible to find a bijective and equally immediate correspondence as stated in Theorem 7.2 between *arbitrary* consistency-unfoldings in $\mathbf{AK}_{\mathbf{0}}^{=}$ and derivations in $\mathbf{HB}_{\mathbf{0}}^{=}$.

But the duality statement in Theorem 7.2 leaves open the question how the particular class of those c.u.'s in $\mathbf{AK}_{\mathbf{0}}^{=}$ that are the images under the reflection function $\mathcal{C}(\cdot)$ of derivations in the basic system $\mathbf{HB}_{\mathbf{0}}^{=}$ can formally be characterized. Closer examination shows that such c.u.'s are always of the particular form, that leaf-occurrences in them of marked formulas are exclusively bound back to upper premises of branchings DECOMP; we stipulate that such c.u.'s *fulfill the property* **D**. This observation gives rise to the following specialized version of Theorem 7.2, which can also be proved in a very straightforward way.

Theorem 7.3 (A Duality between derivations in $\mathbf{HB}_{\mathbf{0}}^{=}$ and consistency-unfoldings in $\mathbf{AK}_{\mathbf{0}}^{=}$ with the property D). *The restrictions $\mathcal{D}_0(\cdot) := \mathcal{D}|_A(\cdot)$ and $\mathcal{C}_0(\cdot) := \mathcal{C}|_B(\cdot)$ of the reflection functions $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$ to respectively the set A of partial consistency-unfoldings in $\mathbf{AK}_{\mathbf{0}}^{=}$ with the property* **D** *and to the set B of derivations in $\mathbf{HB}_{\mathbf{0}}^{=}$ (possibly with open assumption classes) yield a duality statement with assertions analogous to items (i), (ii) and (iii) in Theorem 7.2 between consistency-unfoldings in $\mathbf{AK}_{\mathbf{0}}^{=}$ with the property* **D** *and derivations in $\mathbf{HB}_{\mathbf{0}}^{=}$ without open assumption classes.*

This theorem is illustrated in Figure 8 by a derivation in $\mathbf{HB}_{\mathbf{0}}^{=}$ without open assumption classes and by a consistency-unfolding in $\mathbf{AK}_{\mathbf{0}}^{=}$ with the property **D** that are each other's reflection via the reflection mappings $\mathcal{D}(\cdot)$ and $\mathcal{C}(\cdot)$.

Our "duality"-results, Theorem 7.2 and Theorem 7.3, are able to provide a pre-

[15] The derivation $\mathrm{Refl}(\tilde{\mathcal{C}})^*$ described here is actually equal to the derivation $\tilde{\mathcal{D}}$ in $\mathbf{HB}_{\mathbf{0}}^{=}$ depicted in Figure 8 [strictly speaking, $\mathrm{Refl}(\tilde{\mathcal{C}})^*$ is equal to one of the 4 $\mathbf{HB}_{\mathbf{0}}^{=}$-derivations that are denoted by the prooftree $\tilde{\mathcal{D}}$ depicted in Figure 8 (since for the two pairs of successive $\mathrm{FOLD}_{l/r}$-applications the order in which these two applications actually follow each other has not been fixed there)].

Figure 8 Example consisting of a consistency-unfolding $\tilde{\mathcal{C}}$ in $\mathbf{AK_0^=}$ with the property $\mathbf{D}$ and of a derivation $\tilde{\mathcal{D}}$ in $\mathbf{HB_0^=}$ that are each other's *reflection* via the restrictions $\mathcal{D}_0(\cdot) := \mathcal{D}|_A(\cdot)$ and $\mathcal{C}_0(\cdot) := \mathcal{C}|_B(\cdot)$ (cf. Theorem 7.3) of the reflection mappings $\mathcal{C}(\cdot)$ and $\mathcal{D}(\cdot)$, i.e. for which it holds that $\mathcal{D}_0(\tilde{\mathcal{C}}) = \mathcal{D}|_A(\tilde{\mathcal{C}}) = \tilde{\mathcal{D}}$ and $\mathcal{C}_0(\tilde{\mathcal{D}}) = \mathcal{C}|_B(\tilde{\mathcal{D}}) = \tilde{\mathcal{C}}$.

$$
\tilde{\mathcal{D}} := \left\{
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{
        \dfrac{
          \dfrac{\big(\tau \to \bot = (\sigma \to \bot) \to \bot\big)^x}{\tau = \sigma}
          \qquad
          \dfrac{}{\bot = \bot}
        }{\tau \to \bot = \sigma \to \bot}\,\text{ARROW}
      }{\tau = \sigma \to \bot}
      \qquad
      \dfrac{}{\bot = \bot}
    }{\tau \to \bot = (\sigma \to \bot) \to \bot}\,(\text{ARR./FIX})_x
  }{\underbrace{\mu\alpha.\,(\alpha \to \bot)}_{\equiv:\,\tau} = \underbrace{\mu\beta.\,((\beta \to \bot) \to \bot)}_{\equiv:\,\sigma}}\,\text{FOLD}_{l/r}
}{}
\right.
$$

$$
\tilde{\mathcal{C}} := \left\{
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{
        \dfrac{\mu\alpha.\,(\alpha \to \bot) = \mu\beta.\,((\beta \to \bot) \to \bot)}{\big(\tau \to \bot = (\sigma \to \bot) \to \bot\big)^x}\,\text{UNFOLD}_{l/r}
      }{\tau = \sigma \to \bot \qquad\qquad \bot = \bot}\,\text{DECOMP}
    }{\tau \to \bot = \sigma \to \bot}
  }{\tau = \sigma \qquad \bot = \bot}\,\text{DECOMP}
}{\big(\tau \to \bot = (\sigma \to \bot) \to \bot\big)^x}
\right.
$$

cise formal connection between the Brandt-Henglein-like axiomatizations $\mathbf{HB_0^=}$ and $\mathbf{e\text{-}HB_0^=}$ for $=_\mu$ and the 'syntactic-matching' proof system $\mathbf{AK_0^=}$ that is similar to a system introduced by Ariola and Klop. But probably the main significance of these statements consists in the fact that they can help to understand the soundness of the reasoning formalized through a coinductive rule like ARROW/FIX in a direct way with a geometrical visualization.

In particular, they make it possible to attribute some precise meaning to the informal explanation given by M. Brandt in the sentence

> "The intuition [of the reasoning formalized by rules like ARROW/FIX, C.G.] being that if you can not find hard evidence proving that the judgement is false then it must be true."

(cited from the abstract of [3]): Suppose we have given a derivation $\tilde{\mathcal{D}}$ in $\mathbf{HB_0^=}$ of the form

$$
\dfrac{
  \begin{array}{c} \langle \tau = \sigma \rangle^x \\ \tilde{\mathcal{D}}_{01} \\ \tau_{01} = \sigma_{01} \end{array}
  \qquad
  \begin{array}{c} \langle \tau = \sigma \rangle^x \\ \tilde{\mathcal{D}}_{02} \\ \tau_{02} = \sigma_{02} \end{array}
}{\underbrace{\tau_{01} \to \tau_{02}}_{\equiv:\,\tau} = \underbrace{\sigma_{01} \to \sigma_{02}}_{\equiv:\,\sigma}}\,(\text{ARROW/FIX})_x \quad . \tag{7.1}
$$

We will try to understand the inference formalized by the bottommost application of ARROW/FIX in $\tilde{\mathcal{D}}$ in the light of the above cited sentence. Furthermore we want to detect the reason why no "harm" does arise by discharging all open marked assumptions in $\tilde{\mathcal{D}}_{01}$ and $\tilde{\mathcal{D}}_{02}$ of the form of the conclusion of $\tilde{\mathcal{D}}$ at the application of ARROW/FIX at the bottom of $\tilde{\mathcal{D}}$.

By building the reflection of $\tilde{\mathcal{D}}$ via the reflection mapping $\mathcal{C}(\cdot)$ defined in Definition 7.1 we arrive at the p.c.u. $\mathcal{C}(\tilde{\mathcal{D}})$ of the form

$$\begin{array}{cc} \multicolumn{2}{c}{(\overbrace{\tau_{01} \to \tau_{02}}^{\equiv \tau} = \overbrace{\sigma_{01} \to \sigma_{02}}^{\equiv \sigma})^{x}} \\ \hdashline (\tau_{01} = \sigma_{01})^{\boldsymbol{m}_{01}} & (\tau_{02} = \sigma_{02})^{\boldsymbol{m}_{02}} \\ \tilde{\mathcal{C}}_{01}\ (\equiv \mathcal{C}(\tilde{\mathcal{D}}_{01})) & \tilde{\mathcal{C}}_{02}\ (\equiv \mathcal{C}(\tilde{\mathcal{D}}_{02})) \\ \langle \tau = \sigma \rangle^{x} & \langle \tau = \sigma \rangle^{x} \end{array} \ \text{DECOMP} \qquad (7.2)$$

and can use some of our acquired knowledge about (partial) consistency-unfoldings: It is easy to prove (similar to a proof for Lemma 6.3) that no contradictions with respect to $=_\mu$ can possibly occur in $\mathcal{C}(\tilde{\mathcal{D}})$ between the root and such leaf-occurrences of marked formulas $(\tau = \sigma)^x$ that are bound back to the root. Hence all different branches $b_1, \ldots, b_n$ in $\mathcal{C}(\tilde{\mathcal{D}})$ from the root downwards to u.l.o.m.f.'s $(\tau = \sigma)^x$ in either $\tilde{\mathcal{C}}_{01}$ or $\tilde{\mathcal{C}}_{02}$ correspond to derivations $\mathcal{D}_1, \ldots, \mathcal{D}_n$ from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ in which *no contradictions* with respect to $=_\mu$ are encountered and during which at least one full loop was passed through. From this it follows that for the purpose of showing the $\mathbf{AK}_{\mathbf{0}}^{=}$-consistency of $\tau = \sigma$ all those derivations $\mathcal{D}$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ from the assumption $\tau = \sigma$, that have one of the derivations $\mathcal{D}_1, \ldots, \mathcal{D}_n$ as their initial segment, do not have to be taken into further account: If such a derivation $\mathcal{D}$ had a contradiction with respect to $=_\mu$ as its conclusion, then a shorter derivation $\mathcal{D}_0$ (in $\mathbf{AK}_{\mathbf{0}}^{=}$ from the assumption $\tau = \sigma$) that resulted from $\mathcal{D}$ by cutting out the loop at its beginning would also lead to a contradiction.

The n different threads[16] $\Theta_1, \ldots, \Theta_n$ within the derivation $\tilde{\mathcal{D}}$ of (7.1) from one of the marked assumptions $(\tau = \sigma)^x$ down to the conclusion $\tau = \sigma$ of $\tilde{\mathcal{D}}$ correspond uniquely—under the reflection mapping $\mathcal{C}(\cdot)$—to the above described n branches $b_1, \ldots, b_n$ in the p.c.u. $\mathcal{C}(\tilde{\mathcal{D}})$ in (7.2) and hence to the derivations $\mathcal{D}_1, \ldots, \mathcal{D}_n$ of from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$, in which no contradiction with respect to $=_\mu$ is encountered and during which a loop is passed through. Thus the inference formalized by the bottommost application of ARROW/FIX in $\tilde{\mathcal{D}}$ can be justified on the grounds that (a) along the derivations $\mathcal{D}_1, \ldots, \mathcal{D}_n$ from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ that result as mirror images from the threads $\Theta_1, \ldots, \Theta_n$ in $\tilde{\mathcal{D}}$ *no evidence for the* $\mathbf{AK}_{\mathbf{0}}^{=}$*-inconsistency of* $\tau = \sigma$ is found and that (b) the open marked assumptions $(\tau = \sigma)^x$ in either $\tilde{\mathcal{D}}_{01}$ or in $\tilde{\mathcal{D}}_{02}$ are allowed to be discharged at the bottom of $\tilde{\mathcal{D}}$ because of the "meaning" given to $\tilde{\mathcal{D}}$ through the p.c.u. $\mathcal{C}(\tilde{\mathcal{D}})$ relative to the concept of "consistency with respect to $\mathbf{AK}_{\mathbf{0}}^{=}$".

By extending the above argumentation slightly it is easy to see: All m different threads $\Theta'_1, \ldots, \Theta'_m$ in the derivation $\tilde{\mathcal{D}}$ depicted in (7.1) from a leaf at the top labelled with either an axiom (REFL) or a marked assumption, that is *discharged* in $\tilde{\mathcal{D}}$, downwards to the conclusion of $\tilde{\mathcal{D}}$ correspond uniquely via reflection to m derivations $\mathcal{D}'_1, \ldots, \mathcal{D}'_m$ from $\tau = \sigma$ in $\mathbf{AK}_{\mathbf{0}}^{=}$ during which no contradiction with

[16] Due to the side-cond. **I** on appl.'s of ARROW/FIX there must exist at least one such thread.

respect to $=_\mu$ does occur. And furthermore, contradictions with respect to $=_\mu$ can only occur in such derivations $\mathcal{D}$ from $\tau = \sigma$ in $\mathbf{AK}_\mathbf{0}^=$ that possess an initial segment $\mathcal{D}_0$ that is related [17] via reflection to a thread Θ in $\tilde{\mathcal{D}}$ from an *open* marked assumption downwards to the conclusion of $\tilde{\mathcal{D}}$.

Rather more formally than done so in the above discussion the duality theorem, Theorem 7.2, enables us to carry out the following alternative proof for the soundness part in Theorem 3.2, in which the soundness assertion for $\mathbf{HB}_\mathbf{0}^=$ with respect to $=_\mu$ is 'reduced' to the soundness assertion for $\mathbf{AK}_\mathbf{0}^=$ with respect to $=_\mu$.

Alternative [18] Soundness Proof for $\mathbf{HB}_\mathbf{0}^=$ with respect to $=_\mu$. Suppose that $\tau = \sigma$ is a theorem of $\mathbf{HB}_\mathbf{0}^=$, where $\tau, \sigma \in \mathit{can}\text{-}\mu Tp$. This means that there exists a derivation $\mathcal{D}$ in $\mathbf{HB}_\mathbf{0}^=$ with conclusion $\tau = \sigma$ and without open assumption classes; let $\mathcal{D}$ be chosen as such a derivation. Then due to Theorem 7.2 the reflection $\mathcal{C}(\mathcal{D})$ of $\mathcal{D}$ is a consistency-unfolding of $\tau = \sigma$ in $\mathbf{AK}_\mathbf{0}^=$ (which c.u.—as we remark by the way—fulfills the property **D** due to Theorem 7.3). Hence by Theorem 6.4 the equation $\tau = \sigma$ is consistent with respect to $\mathbf{AK}_\mathbf{0}^=$. And from this Theorem 3.4, which states the soundness of $\mathbf{AK}_\mathbf{0}^=$ with respect to $=_\mu$, implies that τ and σ are strongly equivalent. □

The soundness of the extension $\mathbf{e\text{-}HB}_\mathbf{0}^=$ of $\mathbf{HB}_\mathbf{0}^=$ with respect to $=_\mu$ can be shown by a completely analogous [19] proof. – Although the argumentation used for the above proof can be carried out in the opposite direction as well and is able to demonstrate also the completeness of $\mathbf{e\text{-}HB}_\mathbf{0}^=$ with respect to $=_\mu$, this does not really constitute an *alternative* completeness proof for $\mathbf{e\text{-}HB}_\mathbf{0}^=$ independent from such a completeness proof for $\mathbf{HB}_\mathbf{0}^=$ that (as hinted for Theorem 3.2) can be derived from the one described in [4]. This is because the problem of showing the direction "$\Rightarrow$" of (6.1) for Theorem 6.4 (which implication is used in such an argument for the completeness of $\mathbf{e\text{-}HB}_\mathbf{0}^=$) is in fact a problem of a "dual" kind to showing the completeness of $\mathbf{e\text{-}HB}_\mathbf{0}^=$: In view of Theorem 7.2 and, more precisely, in view of its proof the activity of trying to build a derivation in $\mathbf{e\text{-}HB}_\mathbf{0}^=$ with conclusion $\tau = \sigma$ for two given recursive types $\tau, \sigma \in \mathit{can}\text{-}\mu Tp$ corresponds uniquely to the activity of trying to build a consistency-unfolding of $\tau = \sigma$ in $\mathbf{AK}_\mathbf{0}^=$.

8 A duality in proof systems for bisimulation equivalence on cyclic term graphs

In this section we want to sketch how our duality result about two proof systems for recursive type equality can be transferred to similar proof systems concerned with bisimulation equivalence on equational representations of cyclic term graphs.

[17] Due to the "influence" of possible VAR-applications in $\mathcal{D}_0$ the word "related" cannot be replaced by "corresponds uniquely" here.
[18] By this we mean an alternative proof compared to one that follows from and is derived from the soundness proof given in [4] with respect to $=_\mu$ for the system given there.
[19] More precisely, only the two appearances of "$\mathbf{HB}_\mathbf{0}^=$" in the proof have to be replaced by "$\mathbf{e\text{-}HB}_\mathbf{0}^=$" and the addition in brackets "(which c.u.... fulfills the property **D** ...)" has to be dropped.

In the aim to limit technicalities and to follow [2] we will only consider equational specifications of cyclic term graphs without free variables. We are assuming a countably infinite set *RVar* of *recursion variables* to underlie the following definition. In this section we will let small Greek letters $\alpha, \beta, \ldots$ vary through recursion variables.

Definition 8.1 (Canonical Term Graph Specifications). Let Σ be a first-order signature. A *canonical term graph specification* (a *c.t.g.s.*) is an equational specification of the form $\langle\, \alpha_0 \,|\, \{\alpha_0 = t_0, \ldots, \alpha_n = t_n\} \,\rangle$, where $n \in \mathbb{N}$, $\alpha_0, \ldots, \alpha_n$ are pairwisely different recursion variables in *RVar* and for all i with $0 \leq i \leq n$ the terms t_i are of the form $t_i \equiv F(\alpha_{i1}, \ldots, \alpha_{in_i})$ for some function symbol $F \in \Sigma$ of arity n_i and variables $\alpha_{i1}, \ldots, \alpha_{in_i} \in \{\alpha_0, \ldots, \alpha_n\}$. An equation $\alpha_i = t_i$ for $i \in \{1, \ldots, n\}$ is called *useless* iff the recursion variable α_i is not *reachable* from the *root* α_0 in the obvious sense. We will use the letters g and h to vary through c.t.g.s.'s and denote by $\mathcal{TGS}(\Sigma)$ the set of all c.t.g.s.'s over Σ.

Bisimilarity between c.t.g.s.'s is defined in [4] as follows:

Definition 8.2 (Bisimulation Equivalence $\underline{\leftrightarrow}$ on c.t.g.s.'s). Let Σ be a signature. Let g and h be canonical term graph specifications over Σ of the form $g = \langle\, \alpha_0 \,|\, \{\alpha_0 = t_0, \ldots, \alpha_n = t_n\} \,\rangle$ and $h = \langle\, \alpha'_0 \,|\, \{\alpha'_0 = t'_0, \ldots, \alpha'_{n'} = t'_{n'}\} \,\rangle$.

(a) R is called a *bisimulation* between g and h if and only if
   (i) R is a relation with domain $\{\alpha_0, \ldots, \alpha_n\}$ and codomain $\{\alpha'_0, \ldots, \alpha'_{n'}\}$;
   (ii) $\alpha_0 \, R \, \alpha'_0$;
   (iii) if $\alpha_i \, R \, \alpha'_j$ for some i, j with $0 \leq i \leq n$ and $0 \leq j \leq n'$, and given that $t_i \equiv F(\alpha_{i1}, \ldots, \alpha_{in_i})$ and $t'_j \equiv F'(\alpha'_{j1}, \ldots, \alpha'_{jn'_j})$ with some $n_i, n'_j \in \mathbb{N}_0$, then $F \equiv F'$ (and hence $n_i = n'_j$) and $\alpha_{i1} \, R \, \alpha'_{j1}, \ldots, \alpha_{in_i} \, R \, \alpha'_{jn'_j}$ must hold.

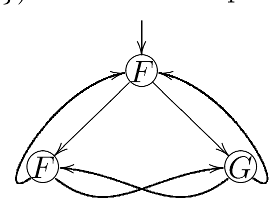
(b) We say that g and h are *bisimilar* (symbolically denoted by $g \,\underline{\leftrightarrow}\, h$) iff there exists a bisimulation between g and h.

We continue with an example for the notions defined in Definition 8.1 and Definition 8.2.

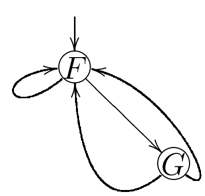
Example 8.3 We consider the two canonical term graph specifications

$$g := \langle\, \alpha_0 \,|\, E_g \,\rangle := \langle\, \alpha_0 \,|\, \{\alpha_0 = F(\alpha_1, \alpha_2),\ \alpha_1 = F(\alpha_0, \alpha_2),\ \alpha_2 = G(\alpha_1, \alpha_0)\} \,\rangle \quad (8.1)$$
$$h := \langle\, \beta_0 \,|\, E_h \,\rangle := \langle\, \beta_0 \,|\, \{\beta_0 = F(\beta_0, \beta_1),\ \beta_1 = G(\beta_0, \beta_0)\} \,\rangle \quad (8.2)$$

in $\mathcal{TGS}(\{F, G\})$. These correspond respectively to the two cyclic term graphs



and



.

It is easy to check that $R := \{(\alpha_0, \beta_0), (\alpha_1, \beta_0), (\alpha_2, \beta_1)\}$ is a bisimulation between g and h according to Definition 8.2. Hence $g \,\underline{\leftrightarrow}\, h$ holds, i.e. g and h are bisimilar.

Figure 9 A Brandt-Henglein-like axiomatization $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$ without symmetry and transitivity rules of bisimulation equivalence between canonical term graph specifications over signature Σ.

The *axioms* and possible *marked assumptions* in $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$:

$$\text{(REFL)}\ \frac{}{\langle\,\alpha\mid\{\alpha=C,\ldots\}\,\rangle=\langle\,\beta\mid\{\beta=C,\ldots\}\,\rangle}\ \text{(if } C \text{ is a constant symbol in } \Sigma) \qquad \text{(Assm)}\ \ (g=h)^{x}\ \text{(with } x\in Mk)\ .$$

The *derivation rules* of $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$: Rules COMP and rules COMP/FIX with

$$\frac{\overbrace{\langle\,\alpha\mid E_0\,\rangle}^{=g}=h}{\langle\,\alpha\mid E_0\uplus\{\alpha_i=t_i\}\,\rangle=h}\ \mathrm{GC}_l^{-1}\ \text{(if } \alpha_i \text{ is unreachable in } g) \qquad \frac{g=\overbrace{\langle\,\beta\mid E_0\,\rangle}^{=h}}{g=\langle\,\beta\mid E_0\uplus\{\beta_i=s_i\}\,\rangle}\ \mathrm{GC}_r^{-1}\ \text{(if } \beta_i \text{ is unreachable in } h)$$

$$\frac{\begin{array}{ccc}\langle\!\langle\,\langle\,\alpha\mid E_g\,\rangle=\langle\,\beta\mid E_h\,\rangle\,\rangle\!\rangle^{x} & & \langle\!\langle\,\langle\,\alpha\mid E_g\,\rangle=\langle\,\beta\mid E_h\,\rangle\,\rangle\!\rangle^{x}\\ \mathcal{D}_1 & & \mathcal{D}_n\\ \langle\,\alpha_1\mid E_g\,\rangle=\langle\,\beta_1\mid E_h\,\rangle & \ldots & \langle\,\alpha_n\mid E_g\,\rangle=\langle\,\beta_n\mid E_h\,\rangle\end{array}}{\langle\,\alpha\mid\underbrace{\{\alpha=F(\alpha_1,\ldots,\alpha_n)\}\uplus E_g^{(0)}}_{=E_g}\,\rangle=\langle\,\beta\mid\underbrace{\{\beta=F(\beta_1,\ldots,\beta_n)\}\uplus E_h^{(0)}}_{=E_h}\,\rangle}\ (\text{COMP/FIX})_x\ \text{(if s.-c. } \mathbf{I})$$

An axiomatization $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$ for $\underline{\leftrightarrow}$, which is very similar to the 'normalized' variant $\mathbf{HB}_{\mathbf{0}}^{=}$ of the axiom system for $=_\mu$ by Brandt and Henglein, is depicted in Figure 9. Similarly as it was defined for its counterpart in $\mathbf{HB}_{\mathbf{0}}^{=}$, the rule ARROW/FIX, applications of the rule COMP/FIX in $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$ are subjected to the side condition **I**: This requirement demands that the discharged assumption class is in fact non-empty (to distinguish such applications from ones of the "plain" COMP-rule). The rules $GC_{l/r}^{-1}$ formalize the inverse operation of *garbage collection* (of *useless* equations) on c.t.g.s.'s. The following theorem, which is very straightforward to prove, holds for $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$:

Theorem 8.4 (Sound- and Completeness of $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$ with respect to $\underline{\leftrightarrow}$). *The axiom system $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$ is sound and complete with respect to bisimulation equivalence $\underline{\leftrightarrow}$ on canonical term graph specifications, i.e. for all c.t.g.s.'s g and h it holds:*

$$\vdash_{\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}} g=h \quad\Longleftrightarrow\quad g\,\underline{\leftrightarrow}\,h\ .$$

A 'syntactic matching' proof system $\mathbf{AK}_{\mathbf{0}}^{\leftrightarrow}$ for $\underline{\leftrightarrow}$ is depicted in Figure 10, which system is of a similar kind as the system $\mathbf{AK}_{\mathbf{0}}^{=}$ for equational testing with respect to $=_\mu$. The rules $\mathrm{GC}_{l/r}$ in $\mathbf{AK}_{\mathbf{0}}^{\leftrightarrow}$ formalize the operations of garbage collection (of useless equations) on c.t.g.s.'s. A notion of consistency with respect to $\mathbf{AK}_{\mathbf{0}}^{\leftrightarrow}$ is sound and complete for $\underline{\leftrightarrow}$. We need the following terminology: An equation $\tilde{g}=\tilde{h}$

Figure 10 A 'syntactic-matching' proof system $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ for testing bisimulation equivalence on equations between canonical term graph specifications.

The *derivation rules* of $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$:

$$\frac{\langle\, \alpha \mid E_0 \uplus \{\alpha_i = t_i\}\,\rangle = h}{\underbrace{\langle\, \alpha \mid E_0\,\rangle}_{=g} = h}\ \mathrm{GC}_l \qquad \frac{g = \langle\, \beta \mid E_0 \uplus \{\beta_i = s_i\}\,\rangle}{g = \underbrace{\langle\, \beta \mid E_0\,\rangle}_{=h}}\ \mathrm{GC}_r$$

(if α_i is unreachable in g) (if β_i is unreachable in h)

$$\frac{\langle\, \alpha \mid \overbrace{\{\alpha = F(\alpha_1,\ldots,\alpha_n)\} \uplus E_g^{(0)}}^{=: E_g}\,\rangle = \langle\, \beta \mid \overbrace{\{\beta = F(\beta_1,\ldots,\beta_n)\} \uplus E_h^{(0)}}^{= E_h}\,\rangle}{\langle\, \alpha_i \mid E_g\,\rangle = \langle\, \beta_i \mid E_h\,\rangle}\ \begin{array}{l}\text{DECOMP}\\ (\text{for } 1 \le i \le n)\end{array}$$

between two c.t.g.s.'s $\tilde{g}$ and $\tilde{h}$ is called $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$-*consistent* iff no contradiction with respect to $\underline{\leftrightarrow}$ is derivable in $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ from $\tilde{g} = \tilde{h}$. And furthermore an equation $\tilde{g} = \tilde{h}$ between two c.t.g.s.'s $\tilde{g} = \langle\, \alpha_0 \mid \{\alpha_0 = t_0, \ldots\}\,\rangle$ and $\tilde{h} = \langle\, \alpha'_0 \mid \{\alpha'_0 = t'_0, \ldots\}\,\rangle$ is agreed to be a *contradiction with respect to* $\underline{\leftrightarrow}$ iff it holds that $t_0 \equiv F(\alpha_{01}, \ldots, \alpha_{0n_0})$ and $t'_0 \equiv G(\alpha'_{01}, \ldots, \alpha'_{0n'_0})$ for some $n_0, n'_0 \in \mathbb{N}_0$, variables $\alpha_{01}, \ldots, \alpha_{0n_0}$, $\alpha'_{01}, \ldots, \alpha'_{0n'_0}$ and *different* symbols $F, G \in \Sigma$ (i.e. $F \not\equiv G$). Relying on these notational agreements the following theorem holds, which is again easy to show.

Theorem 8.5 (Soundness and Completeness with respect to $\underline{\leftrightarrow}$ of consistency-checking relative to $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$).
The 'syntactic-matching' system $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ is sound and complete with respect to $\underline{\leftrightarrow}$ for the notion of checking consistency relative to this system: For all canonical term graph specifications g and h it holds:

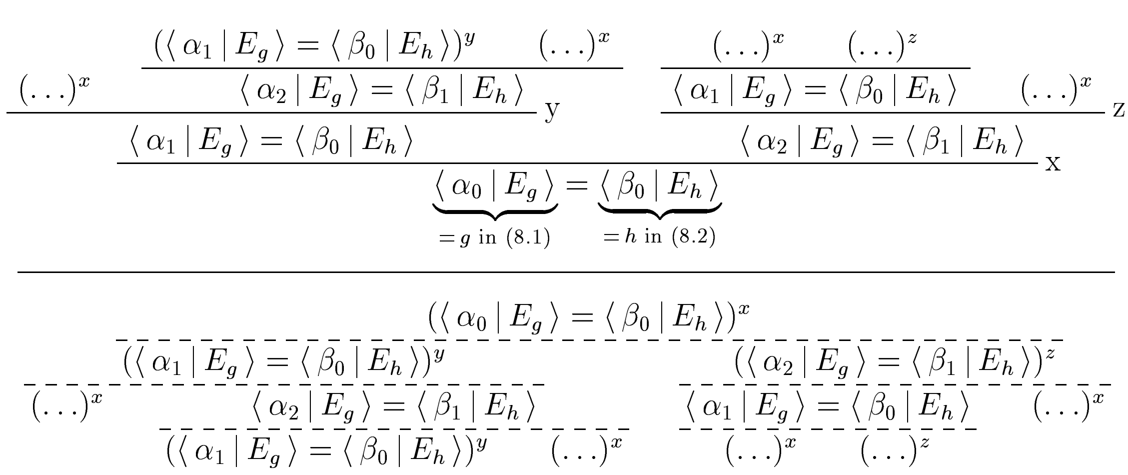
$$g = h \ \textit{ is } \mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}\textit{-consistent} \quad \Longleftrightarrow \quad g \,\underline{\leftrightarrow}\, h \ .$$

Now it is very straightforward to define the notion of p.c.u.'s and consistency-unfoldings in $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ of equations between c.t.g.s.'s analogously to Definitions 6.1 and 6.2. And furthermore also reflection mappings $\mathcal{C}(\cdot)$ and $\mathcal{D}(\cdot)$ between p.c.u.'s in $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ and derivations in $\mathbf{HB}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ can be defined very similar to (and in fact easier than in) Definition 7.1. In this way we are lead to the following counterpart of Theorem 7.2 for the two proof systems considered here.

Theorem 8.6 (A Duality between derivations in $\mathbf{HB}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ and consistency-unfoldings in $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$). *There is a bijective functional relationship between derivations in $\mathbf{HB}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ without open assumption classes and consistency-unfoldings in $\mathbf{AK}_{\mathbf{0}}^{\underline{\leftrightarrow}}$ via reflection mappings $\mathcal{C}(\cdot)$ and $\mathcal{D}(\cdot)$: This means that completely analogous statements to that in items (i), (ii) and (iii) of Theorem 7.2 are true.*

In Figure 11 the assertion of this theorem is exemplified for the c.t.g.s.'s g and h of Example 8.3 by a suggestively typeset pair $(\tilde{\mathcal{D}}, \tilde{\mathcal{C}})$ of a derivation $\tilde{\mathcal{D}}$ for $g = h$

Figure 11 Example consisting of a derivation in $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$ without open assumption classes and of a consistency-unfolding in $\mathbf{AK}_{\mathbf{0}}^{\leftrightarrow}$ that are are each other's "reflection". (The canonical term graph specifications g and h are taken from Example 8.3).



in $\mathbf{HB}_{\mathbf{0}}^{\leftrightarrow}$ without open assumption classes and a consistency-unfolding of $g = h$ in $\mathbf{AK}_{\mathbf{0}}^{\leftrightarrow}$, where $\tilde{\mathcal{C}}$ and $\tilde{\mathcal{D}}$ are each other's "mirror image" via reflection mappings $\mathcal{C}(\cdot)$ and $\mathcal{D}(\cdot)$.

9 Conclusion

In the main part of this paper we have motivated and developed a precise formal relationship between two different proof systems concerned with recursive type equality $=_\mu$ on a restricted class of recursive types in μ-term notation with only type constructor $\to$. We showed the existence of a bijective correspondence that can geometrically be visualized between (1) derivations without open assumptions in an extension $\mathbf{e\text{-}HB}_{\mathbf{0}}^{=}$ of a 'normalized' version $\mathbf{HB}_{\mathbf{0}}^{=}$ of the axiomatization for $=_\mu$ by Brandt and Henglein and (2) what we defined as consistency-unfoldings in a proof system $\mathbf{AK}_{\mathbf{0}}^{=}$ à la Ariola and Klop for equational testing with respect to $=_\mu$. This correspondence takes place via two reflection mappings $\mathcal{C}(\cdot)$ and $\mathcal{D}(\cdot)$ that formalize effective transformations and that are inverse to each other. Its particularly immediate kind gave us reason to call it a duality. – By "developing on fine-grained film" and analyzing the image of the set of $\mathbf{HB}_{\mathbf{0}}^{=}$-derivations under the reflection mapping $\mathcal{C}(\cdot)$ we found that our correspondence result can be specialized to the assertion of a duality taking place via appropriate restrictions $\mathcal{C}_0(\cdot)$ and $\mathcal{D}_0(\cdot)$ of the reflection mappings $\mathcal{C}(\cdot)$ and $\mathcal{D}(\cdot)$ also between (1′) derivations without open assumption classes in our basic Brandt-Henglein system $\mathbf{HB}_{\mathbf{0}}^{=}$ and (2′) such consistency-unfoldings in $\mathbf{AK}_{\mathbf{0}}^{=}$ that fulfill the particular property **D**.

Apart from establishing a precise formal link between the systems $\mathbf{HB}_{\mathbf{0}}^{=}$ and $\mathbf{AK}_{\mathbf{0}}^{=}$ by tying together closely the notions of "derivability in $\mathbf{HB}_{\mathbf{0}}^{=}$" and "consistency with respect to $\mathbf{AK}_{\mathbf{0}}^{=}$", the main significance of the duality results consists perhaps in the following: They can be used to understand and justify the soundness of the—at least at first sight—seemingly paradoxical reasoning formalized by the rule ARROW/FIX in the variant-Brandt-Henglein system $\mathbf{HB}^{=}_{\mathbf{0}}$. In fact, our results facilitated an alternative soundness proof for the system $\mathbf{HB}^{=}_{\mathbf{0}}$ that is independent from the one given in [4] and that proceeds by 'reducing' the soundness of $\mathbf{HB}^{=}_{\mathbf{0}}$ to the soundness of the system $\mathbf{AK}^{=}_{\mathbf{0}}$.

We did not investigate in this paper the proof-theoretic relationship between the axiom system (here denoted by) $\mathbf{HB}^{=}$ for recursive type equality introduced by Brandt and Henglein and our variant-system $\mathbf{HB}^{=}_{\mathbf{0}}$. The symmetry and transitivity rules present in $\mathbf{HB}^{=}$ are not part of the formal system $\mathbf{HB}^{=}_{\mathbf{0}}$ for which a version of the subformula property is true. It can be shown that every $\mathbf{HB}^{=}$-derivation without open assumption classes can be 'normalized' in a certain effective way by 'working away' all applications of symmetry and transitivity rules with the result of derivation in $\mathbf{HB}^{=}_{\mathbf{0}}$ with the same conclusion and no open assumption classes. For this as well as for a detailed study of proof-theoretic transformations between the here formally introduced or merely mentioned proof systems for recursive types and a number of further variant-systems we want to refer to our forthcoming work [5].

In the last section we indicated that the described duality result is not specific to the two considered proof systems for recursive types: We sketched an analogous duality theorem for a similar pair of proof systems concerned with the notion of bisimulation equivalence on equational specifications of cyclic term graphs.

We have come to realize only very recently that the notion of a consistency-unfolding, the definition of which was devised very much in an 'ad hoc'-manner for the special purpose at hand here, does bear an obvious analogy with the concept of a 'closed analytic tableau' as introduced by R. Smullyan. And in fact, the duality statements developed here lend themselves for being reformulated with respect to an—in each case—suitably defined tableau calculus as assertions about an immediate functional relationship between proofs in a respective Brandt-Henglein system and so called 'syntactic-matching tableaux' in the tableau system. Preliminary formulations of results in this direction regarding proof systems for recursive types can be found on the slides [6] of a recent talk.

Acknowledgement

I want to thank J.W. Klop for suggesting a study about the proof-theoretic connections between proof systems for cyclic objects, for providing me with a couple of initial ideas and for reading some versions of this paper. And I am also very much indebted to Bas Luttik for his careful reading of drafts for the submitted version of this paper and for many suggestions of improvements he kindly offered.

References

[1] Amadio, R.M., Cardelli, L.: "Subtyping Recursive Types", *ACM Transactions on Programming Languages and Systems 15 (4)*, pp. 575–631, 1993.

[2] Ariola, Z.M., Klop, J.W.: "Equational Term Graph Rewriting", *Fundamenta Informaticae 26 (3,4)*, pp. 207–240, June 1996; extended version as: *Vrije Universiteit Amsterdam Technical Report IR–391*, September 1995.

[3] Brandt, M.: "Recursive Subtyping: Axiomatizations and Computational Interpretations", Master's Thesis, DIKU, Department of Computer Science, University of Copenhagen, August 1997; available on the internet *TOPPS Technical Report D-352* from: `ftp://ftp.diku.dk/diku/semantics/papers/D-352.ps.gz` .

[4] Brandt, M., Henglein, F.: "Coinductive Axiomatization of Recursive Type Equality and Subtyping", *Fundamenta Informaticae 33*, pp. 1–30, 1998.

[5] Grabmayer, C.: "Proof-Theoretic Interconnections between Proof Systems for Recursive Type Equality", forthcoming as *Vrije Universiteit Amsterdam Technical Report*, 2002.

[6] Grabmayer, C.: "Proving Equality for Recursive Types—A Duality between 'syntactic-matching' tableaux and Brandt-Henglein derivations", *slides* for a talk given at the *Process Algebra Meeting (PAM)*, CWI Amsterdam, May 8, 2002; presently available from: `http://www.cs.vu.nl/~clemens/PAM.ps.gz` .