Completeness of Kozen's Axiomatisation of the Propositional μ-Calculus

Igor Walukiewicz[1]
BRICS[2,3]
Department of Computer Science
University of Aarhus
Ny Munkegade
DK-8000 Aarhus C, Denmark
e-mail: igw@mimuw.edu.pl

(Extended abstract)

Abstract

We consider the propositional μ-calculus as introduced by Kozen [2]. In that paper a natural proof system was proposed and its completeness stated as an open problem. We show that the system is complete.

1 Introduction

In the paper introducing the propositional μ-calculus [2] Kozen proposed a very natural axiom system and showed that it proves negations of all unsatisfiable formulas of a special kind called *aconjunctive formulas*. In [9] another finitary axiomatisation was proposed and proved to be complete for the whole propositional μ-calculus. This solved one part of the problem posed in [2] but the question of the completeness of the original axiomatisation remained still open. We give an affirmative answer to this question.

There are other reasons, apart from curiosity, to investigate the problem of the completeness of Kozen's system. Axiomatisation proposed in [9] makes essential use of the small model theorem for the μ-calculus which makes it impossible to use it for extensions of the logic not enjoying the finite model theorem. The other reason is that Kozen's system is very natural, one may say as natural as the notion of Kripke structures. Hence it is good to know that the class of Kripke structures is a complete subclass of a quasi-variety defined by Kozen's system.

Unlike the proof in [9] this completeness proof does not use Martin's determinacy theorem or complicated automata theoretic results. For the proof new techniques "inside" the μ-calculus were developed. There are at least three innovations we use. The first is an alternative semantics of formulas given by tableau markings and the notion of tableau equivalence. This allows us to prove semantical equivalence of big and complicated formulas. The second development is the notion of disjunctive formula. It occurs that every formula is equivalent to a disjunctive formula and moreover disjunctive formulas are very easy to handle when satisfiability or provability is concerned. The third step is Lemma 5.7 which shows how to use tableau equivalence for constructing proofs.

The difficulty in proving completeness of the original system lays in the interplay of all the connectives of the μ-calculus. The problem encountered in [2] resulted from the mutual dependencies between conjunction and μ. In that paper the notion of *aconjunctive formula* was introduced in order to isolate the cases where the difficulties do not arise. In [9] a different method was proposed which nevertheless still used the same general idea of converting a tableau of an unsatisfiable formula into a proof of its unsatisfiability. This method works for different axiomatisation and its analysis shows that it cannot be easily extended to show completeness of the, weaker, Kozen's axiomatisation. This is somehow disturbing because tableau method [8] remains essentially the only know method of syntactic model construction.

An evidence that Kozen's system is nevertheless complete comes from an analysis of the role of a conjunction in the μ-calculus formulas. An insight to this can be gained by considering "operational interpretation" of the formulas.

Usually the semantics of the formulas of the μ-calculus is given denotationally, in a compositional

[1]This work was partially supported by Polish KBN grant No. 2 P301 009 06

[2]Basic Research in Computer Science, Centre of the Danish National Research Foundation.

[3]On the leave from: Institute of Informatics, Warsaw University, Banacha 2, 02-097 Warsaw, POLAND

way, by induction on the structure of a formula. A different look, presented in Section 3 below, is that a formula is an automaton-like device checking a property of the unwinding of a model from a given state. Operationally speaking we can say that a formula $q \wedge \langle a \rangle p$ checks that q holds in a state and that there is an edge labeled by a leading to a state satisfying p. Such an interpretation may look uninteresting for simple formulas but gets a bit more insightful in the general case. It gives us a characterisation of satisfiability of a formula in a given state of a model by means of *markings* of a tableau of the formula. This leads to the concept of tableau equivalence and the proof that if two formulas have equivalent tableaux then they are equivalent.

This "operational semantics" gives us intuitions about the role of each connective. If we are to check that $\alpha \vee \beta$ holds, we choose (nondeterministically) one of the disjuncts. If we are to check $\mu X.\alpha(X)$ we try equivalent formula $\alpha(\mu X.\alpha(X))$. When we check $\alpha \wedge \beta$ we must check that a state satisfies α and β. While a disjunction acts like a nondeterministic choice, a conjunction acts rather like an universal branching of an alternating automaton. Such an alternating behaviour of a conjunction is a source of many difficulties.

From automata theory we know that alternating automata are equivalent to nondeterministic ones [3]. This suggests an idea that every formula should be equivalent to a formula which does not have universal branching behaviour represented by a conjunction. Of course we cannot discard conjunctions completely from positive formulas as an example of formula $(\langle a \rangle p) \wedge (\langle b \rangle q)$ shows. Note that conjunction in this formula does not act as an universal branching. It is rather an implicit conjunction from (usual, not alternating) automata on trees where a transition relation forces one son to be labeled by a state q and the other one by q'. This implicit conjunction is the only form of conjunction that is present in fixpoint notation for the sets of trees defined by Niwiński [5]. In that paper it was proved that this fixpoint language has the same expressive power as *SnS*, monadic second order logic of n successors. Hence adding explicit conjunction to this language will not increase its expressive power.

This considerations lead to the concept of *disjunctive formulas* which are formulas where the role of conjunction is restricted so that it never acts as an universal branching. It turns out that every formula is equivalent to a disjunctive formula. Formulas of this kind have several interesting properties. Satisfiability checking can be done in linear time and there is an easy method of proving $\neg\varphi$ for every unsatisfiable disjunctive formula φ.

It should be noted that not all aconjunctive formulas (in the sense of [2]) are disjunctive formulas; also the other inclusion does not hold. To encompass both notions we introduce a class of *weakly aconjunctive formulas*. It turns out that we can repeat the arguments from [2] to show that for every unsatisfiable weakly aconjunctive formula φ, the formula $\neg\varphi$ is provable.

The last step is to show that for every formula φ there is an equivalent disjunctive formula $\widehat{\varphi}$ and the formula $\varphi \Rightarrow \widehat{\varphi}$ is provable. This is done by induction on the structure of φ with difficult cases for conjunction, the greatest fixpoint operator (ν) and the most difficult for the least fixpoint operator (μ). Proving formula $\neg(\varphi \wedge \neg\widehat{\varphi})$ is difficult because even if $\widehat{\varphi}$ is a disjunctive formula its negation may not be weakly aconjunctive and it is impossible to extend arguments from [2] far enough to encompass also this case. Fortunately it turns out that φ and $\widehat{\varphi}$ have equivalent tableaux which is a stronger statement then just saying that the two formulas are semantically equivalent. This fact together with disjunctiveness of $\widehat{\varphi}$ is essentially used in the construction of the proof of $\neg(\varphi \wedge \neg\widehat{\varphi})$.

Ideas used to derive "operational semantics" of formulas are very similar to those used by Streett and Emerson in [8] although there the authors were concerned with constructing a model for a formula rather than with the truth of a formula in a given state of a given model. Similar ideas can be also found in so called local model checking [7, 6], although we look at the problem from a different perspective. Disjunctive formulas for the μ-calculus were discovered independently by David Janin and the author. Properties of this formulas are analysed in the forthcoming joint paper. It is worth noting that a concept similar to disjunctive formulas but for process logic was used in [1].

Due to the length of the whole argument it was not possible to include the proofs in this abstract. In few cases we give a proof outline.

Acknowledgements

I would like to thank Dexter Kozen for the introduction to the μ-calculus and sharing with me his understanding of the subject. I am grateful to Damian Niwiński for many important comments and remarks on this work. Discussions with David Janin, Nils Klarlund and Dexter Kozen were a great help for me.

2 Preliminary definitions

Let $Prop = \{p, q, \ldots\}$ be a set of propositional letters, $Var = \{X, Y, \ldots\}$ a set of variables and $Act = \{a, b, \ldots\}$ a set of actions. Formulas of the μ-calculus over these three sets can be defined by the following grammar:

$$\begin{aligned} F \quad := \quad & Var \mid Prop \mid \neg F \mid F \vee F \mid F \wedge F \mid \\ & \langle Act \rangle F \mid [Act] F \mid \mu\, Var.F \mid \nu\, Var.F \end{aligned}$$

Additionally we require that in formulas of the form $\mu X.\alpha(X)$ and $\nu X.\alpha(X)$, variable X occurs in $\alpha(X)$ only *positively*, i.e., under even number of negations. We will use σ to denote μ or ν. Propositional constants, variables and their negations will be called *literals.*

Formulas are interpreted in Kripke models of the form $\mathcal{M} = \langle S, R, \rho\rangle$, where: S is a nonempty set of states, $R : Act \to \mathcal{P}(S \times S)$ is a function assigning a binary relation on S to each action in Act and $\rho : Prop \to \mathcal{P}(S)$ is a function assigning a set of states to each propositional letter in $Prop$.

The meaning of a formula in a model is a set of states where it is true. For a given model $\mathcal{M}$ and a valuation $V : Var \to \mathcal{P}(S)$, the meaning of a formula α, denoted $\| \alpha \|_V^{\mathcal{M}}$, is defined by induction on the structure of α by the following clauses (we will omit superscript $\mathcal{M}$ when it causes no ambiguity):

$$
\begin{aligned}
\| X \|_V &= V(X)\\
\| p \|_V &= \rho(p)\\
\| \neg\alpha \|_V &= S - \| \alpha \|_V\\
\| \alpha \wedge \beta \|_V &= \| \alpha \|_V \cap \| \beta \|_V\\
\| \alpha \vee \beta \|_V &= \| \alpha \|_V \cup \| \beta \|_V\\
\| \langle a\rangle\alpha \|_V &= \{s : \exists t.(s,t) \in R(a) \wedge t \in \| \alpha \|_V\}\\
\| [a]\alpha \|_V &= \{s : \forall t.(s,t) \in R(a) \Rightarrow t \in \| \alpha \|_V\}\\
\| \mu X.\alpha(X) \|_V &= \bigcap\{T \subseteq S : \| \alpha \|_{V[T/X]} \subseteq T\}\\
\| \nu X.\alpha(X) \|_V &= \bigcup\{T \subseteq S : T \subseteq \| \alpha \|_{V[T/X]}\}
\end{aligned}
$$

When we write $\mathcal{M}, s, V \models \alpha$ we mean that $s \in \| \alpha \|_V^{\mathcal{M}}$.

Definition 2.1 (Positive, guarded formulas) *We call a formula* positive *iff all negations in the formula appear only before propositional constants and free variables.*

Variable X in $\mu X.\alpha(X)$ is called guarded *iff every occurrence of X in α is in the scope of some modality operator $\langle\ \rangle$ or $[\ ]$. We say that a formula is guarded iff every bound variable in the formula is guarded.*

Proposition 2.2 (Kozen) *Every formula is equivalent to a positive guarded formula.*

Next we will introduce tools which allow us to deal with occurrences of subformulas of a given formula. This tools are very similar to those used in [2] or [7]. We would like to have a different name (which will be a variable) for every fixpoint subformula of a given formula. We will also introduce a notion of a binding function which will associate subformulas to names.

Definition 2.3 (Binding) *We call a formula* well named *iff every variable is bound at most once in the formula and free variables are distinct from bound variables. For a variable X bound in a well named formula α there exists the unique subterm of α of the form $\sigma X.\beta(X)$, from now on called the* binding definition of X in α *and denoted $\mathcal{D}_\alpha(X)$. We will omit subscript α when it causes no ambiguity. We call X a* ν-variable *when $\sigma = \nu$, otherwise we call X a* μ-variable.

The function $\mathcal{D}_\alpha$ assigning to every bound variable its binding definition *in α will be called the* binding function *associated with α.*

Remark: Note that every formula is equivalent to a well-named one which can be obtained by some consistent renaming of bound variables. The substitution of a formula β for all free occurrences of a variable X in α, denoted $\alpha[\beta/X]$, can be made modulo some consistent renaming of bound variables of β, so that the obtained formula $\alpha[\beta/X]$ is still well-named.

Definition 2.4 (Dependency order) *Given a formula α we define the* dependency order *over the bound variables of α, denoted $\leq_\alpha$, as the least partial order relation such that if X occurs free in $\mathcal{D}_\alpha(Y)$ then $X \leq_\alpha Y$. We will say that a bound variable Y depends on a bound variable X in α when $X \leq_\alpha Y$.*

Example: In case $\alpha = \mu X.(b \to \{X\}) \vee \nu Y.(a \to \{Y\})$, variables X and Y are incomparable in $\leq_\alpha$ ordering. On the other hand if α is $\mu X.\nu Y.(a \to \{X\}) \vee \mu Z.(a \to \{Z \vee Y\})$ then $X \leq_\alpha Z$.

Definition 2.5 *Given a formula α with an associated binding function $\mathcal{D}_\alpha$, for every subformula β of α we define the* expansion *of β with respect to $\mathcal{D}_\alpha$ as:*

$$\langle\!|\beta|\!\rangle_{\mathcal{D}_\alpha} = \beta[\mathcal{D}_\alpha(X_n)/X_n]\cdots[\mathcal{D}_\alpha(X_1)/X_1]$$

where the sequence $(X_1, X_2, \ldots, X_n)$ is a linear ordering of all bound variables of α compatible with the dependency partial order, i.e. if $X_i \leq_\alpha X_j$ then $i \leq j$.

Definition 2.6 *In construction of our tableaux we will need to distinguish some occurrences of conjunction which should not be reduced by ordinary (and) rule. To do this we extend the syntax of the μ-calculus by allowing new construction of the form $(a \to \Psi)$, where a is an action and Ψ is a finite set of formulas. When semantics is concerned, we will consider such a formula as an abbreviation of a formula $\bigwedge\{\langle a\rangle\psi : \psi \in \Psi\} \wedge [a]\bigvee\Psi$. As usual the conjunction of the empty set is true and the disjunction is false.*

Remark: It is possible to express $[a]$ and $\langle a\rangle$ modalities with the construction introduced above. A formula $[a]\psi$ is equivalent to $(a \to \emptyset) \vee (a \to \{\psi\})$ and a

$$(and)\ \frac{\{\alpha,\beta,\Gamma\}}{\{\alpha\wedge\beta,\Gamma\}} \qquad (or)\ \frac{\{\alpha,\Gamma\}\quad\{\beta,\Gamma\}}{\{\alpha\vee\beta,\Gamma\}}$$

$$(\mu)\ \frac{\{\alpha(X),\Gamma\}}{\{\mu X.\alpha(X),\Gamma\}} \qquad (\nu)\ \frac{\{\alpha(X),\Gamma\}}{\{\nu X.\alpha(X),\Gamma\}}$$

$$(reg)\ \frac{\{\alpha(X),\Gamma\}}{\{X,\Gamma\}} \quad \text{whenever } X \text{ is a bound variable of } \varphi \text{ and } \mathcal{D}_\varphi(X)=\sigma X.\alpha(X)$$

$$(mod)\ \frac{\{\psi\}\cup\{\bigvee\Theta : (a\to\Theta)\in\Gamma, \Theta\neq\Psi\} \quad \text{for every } (a\to\Psi)\in\Gamma, \psi\in\Psi}{\Gamma}$$

Figure 1: The system $\mathcal{S}^\varphi$

formula $\langle a\rangle\psi$ to $(a\to\{\psi, true\})$. All the notions from this section like guarded formula, binding function etc. extend to formulas with this new construction.

Definition 2.7 *Formula of the form $(a\to\emptyset)$ will be called* terminal formula *because its meaning is that there are no a-transitions from a given state.*

Proviso: If not otherwise stated all formulas are assumed to be well named, positive, guarded and use $(a\to\{\Psi\})$ construct instead of $\langle a\rangle\psi$ and $[a]\psi$ modalities. By observations stated above this is not a restriction if semantics is concerned. It turns out that it is also not a restriction for provability arguments.

3 Tableau equivalence

In this section we will present a tool for proving equivalence of formulas. We will define the notion of a tableau for a formula and show that two formulas are equivalent if they have equivalent tableaux. In spite of the fact that the implication in the other direction does not hold, tableau equivalence turns out to be a very handy tool.

Definition 3.1 (Tableau rules) *For a formula φ and its binding function $\mathcal{D}_\varphi$ we define the system of tableau rules $\mathcal{S}^\varphi$ parameterised by φ or rather its binding function. The system is presented in Figure 1 (we use $\{\alpha,\Gamma\}$ as a shorthand for $\{\alpha\}\cup\Gamma$).*

Remark: (1) We see applications of rules as a process of reduction. Given a finite set of formulas Γ we want to derive, we look for the rule the conclusion of which matches our set. Then we apply the rule and obtain the assumptions of the instance of the rule in which Γ is the conclusion.

(2) There is no rule for reducing formulas of the form $\langle a\rangle\varphi$ or $[a]\varphi$ because we assume that this formulas are replaced by equivalent formulas using $(a\to\Phi)$ notation.

(3) The rule (mod) has as many assumptions as there are formulas in sets Φ, s.t., $(a\to\Phi)\in\Gamma$. For example

$$\frac{\{\varphi_1,\varphi_3\}\quad\{\varphi_2,\varphi_3\}\quad\{\varphi_1\vee\varphi_2,\varphi_3\}\quad\{\psi_1\}\quad\{\psi_2\}}{\{(a\to\{\varphi_1,\varphi_2\}),(a\to\{\varphi_3\}),(b\to\{\psi_1,\psi_2\})\}}$$

is an instance of the rule. We will call a son labeled by an assumption obtained by *reducing* an action a an *a-son.* In our example if a node n of a tableau is labeled by the conclusion of the rule then its son labeled by $\{\varphi_1,\varphi_3\}$ is an a-son of n and a son labeled by $\{\psi_1\}$ is a b-son of n.

Definition 3.2 (Tableaux) Tableau *for a formula φ is a pair $\langle T, L\rangle$, where T is a tree and L is a labeling function such that*

1. *the root of T is labeled by $\{\varphi\}$,*
2. *the sons of any internal node n are created and labeled according to the rules of system $\mathcal{S}_\varphi$. Additionally we require that rule (mod) is applied only when no other rule is applicable.*

As our tableaux may be infinite we will be interested not only in the form of the leaves but also in the internal structure of tableaux. We are now going to distinguish some nodes of tableaux and define a notion of trace which captures the idea of a history of a regeneration of a formula.

Definition 3.3 (Modal and choice nodes) *Leaves and nodes where reduction of modalities is performed, i.e., rule (mod) is used, will be called* modal nodes. *The root of the tableau and sons of modal nodes will be called choice nodes.*

If φ is a guarded formula then the sequence of all the choice nodes on the path of a tableau for φ induces a partition of the path into finite intervals beginning in choice nodes and ending in modal nodes. We will say that a modal node m is near *a choice node n iff they are both in the same interval, i.e., in the tableau there is a path from n to m without an application of rule (mod). Observe that in some cases choice node may be also modal node.*

Definition 3.4 (Trace) *Given a path $\mathcal{P}$ of a tableau $\mathcal{T} = \langle T, L\rangle$, a* trace *on $\mathcal{P}$ will be a function $\mathcal{T}r$ assigning a formula to every node in some initial segment of $\mathcal{P}$ (possibly to the whole $\mathcal{P}$), satisfying the following conditions:*

- *If $\mathcal{T}r(m)$ is defined then $\mathcal{T}r(m) \in L(m)$.*
- *Let m be a node with $\mathcal{T}r(m)$ defined and let $n \in \mathcal{P}$ be a son of m. If a rule applied in m does not reduce formula $\mathcal{T}r(m)$ then $\mathcal{T}r(n) = \mathcal{T}r(m)$. If $\mathcal{T}r(m)$ is reduced in m then $\mathcal{T}r(n)$ is one of the results of the reduction. This should be clear for all the rules except (mod). In case m is a modal node and n is labeled by $\{\psi\} \cup \{\bigvee \Theta : (a \to \Theta) \in \Gamma, \Theta \neq \Psi\}$ for some $(a \to \Psi) \in L(m)$ and $\psi \in \Psi$, then $\mathcal{T}r(n) = \psi$ if $\mathcal{T}r(m) = (a \to \Psi)$ and $\mathcal{T}r(n) = \bigvee \Theta$ if $\mathcal{T}r(m) = (a \to \Theta)$ for some $(a \to \Theta) \in \Gamma$, $\Theta \neq \Psi$. Traces from other formulas end in node m.*

Definition 3.5 (μ-trace) *We say that there is a* regeneration *of a variable X on a trace $\mathcal{T}r$ on some path iff for some node m and its son n on the path $\mathcal{T}r(m) = X$ and $\mathcal{T}r(n) = \alpha(X)$, where $\mathcal{D}(X) = \sigma X.\alpha(X)$.*

We call a trace μ-trace *iff it is an infinite trace (defined for the whole path) on which the smallest with respect to $\leq_\gamma$ ordering variable regenerated infinitely often is a μ-variable. Similarly a trace will be called a* ν-trace *iff it is an infinite trace where the smallest variable which regenerates infinitely often is a ν-variable.*

Remark: Every infinite trace is either a μ-trace or a ν-trace because all the rules except regenerations decrease the size of formulas and formulas are guarded hence every formula is eventually reduced.

We are now going to define what does it mean for two tableaux to be equivalent. It occurs that we can abstract from the order of application of non-modal rules. But the structure of a tree designated by modal nodes will be very important.

Definition 3.6 (Tableau equivalence) *We say that two tableaux $\mathcal{T}_1$ and $\mathcal{T}_2$ are* equivalent *iff there is a bijection $\mathcal{E}$ between choice and modal nodes of $\mathcal{T}_1$ and $\mathcal{T}_2$ such that:*

1. *$\mathcal{E}$ maps the root of $\mathcal{T}_1$ onto the root of $\mathcal{T}_2$, it maps choice nodes to choice nodes and modal nodes to modal nodes.*
2. *If n is a descendant of m then $\mathcal{E}(n)$ is a descendant of $\mathcal{E}(m)$. Moreover if for some action a, node n is an a-son of a modal node m then $\mathcal{E}(n)$ is an a-son of $\mathcal{E}(m)$.*
3. *For every modal node m, the sets of literals and terminal formulas occuring in $L(m)$ and in $L(\mathcal{E}(m))$ are equal.*
4. *There is a μ-trace on a path $\mathcal{P}$ of $\mathcal{T}_1$ iff there is a μ-trace on a path of $\mathcal{T}_2$ designated by the image of $\mathcal{P}$ under $\mathcal{E}$.*

Theorem 3.7 *If two formulas (which satisfy our proviso) have equivalent tableaux then they are semantically equivalent.*

4 Disjunctive formulas

Here we define a notion of *disjunctive formula* and show that every formula is equivalent to a disjunctive formula.

Definition 4.1 *A conjunction $\alpha_1 \wedge \ldots \wedge \alpha_n$ is called* special *iff every α_i is either a literal or a formula of a form $(a \to \Psi)$ and for every action a there is at most one conjunct of the form $(a \to \Psi)$ among $\alpha_1, \ldots, \alpha_n$.*

The set of disjunctive formulas, *$\mathcal{F}_d$ is the smallest set defined by the following clauses:*

1. *every literal is a disjunctive formula,*
2. *if $\alpha, \beta \in \mathcal{F}_d$ then $\alpha \vee \beta \in \mathcal{F}_d$; if moreover X occurs only positively in α and not in the context $X \wedge \gamma$ for some γ, then $\mu X.\alpha, \nu X.\alpha \in \mathcal{F}_d$,*
3. *$(a \to \Psi) \in \mathcal{F}_d$ if $\Psi \subseteq \mathcal{F}_d$,*
4. *a special conjunction of disjunctive formulas is a disjunctive formula.*

Remark: Modulo the order of application of (*and*) rules, disjunctive formulas have unique tableaux. Moreover on every infinite path there is one and only one infinite trace.

It turns out that every formula is equivalent to a disjunctive formula. This is unfortunately not a normal form because there may be many equivalent disjunctive formulas.

Theorem 4.2 *For every (well named, positive and guarded) formula φ and every regular tableau $\mathcal{T}$ for φ (i.e. a tableau which is a regular tree) there is a disjunctive formula $\widehat{\varphi}$ with the tableau equivalent to $\mathcal{T}$.*

Proof: A *tree with back edges* is a tree with added edges leading from some of the leaves to their ancestors. First we prove:

Lemma 4.3 *It is possible to construct a finite tree with back edges $\mathcal{T}_l = \langle T_l, L_l \rangle$, satisfying the following conditions:*

1. *$\mathcal{T}_l$ unwinds to $\mathcal{T}$.*
2. *Every node to which a back edge points can be assigned color magenta or navy in such a way that for every infinite path of the unwinding of $\mathcal{T}_l$ we have: there is a μ-trace on the path iff the closest to the root node of $\mathcal{T}_l$ through which the path goes infinitely often is colored magenta.*

Having such a tree one constructs from it a disjunctive formula $\widehat{\varphi}$ which has a tableau equivalent to $\mathcal{T}$. The construction starts in the leaves of the tree and proceeds to the root. All back edges leading to a node n are assigned the same variable X_n and the color of the node is used to decide which fixpoint operator should be used to close this variable when we reach n in our construction. □

5 Completeness

In this section we sketch the completeness proof of Kozen's axiom system [2]. Our plan is as follows. After recalling the system itself we introduce the notions of weakly aconjunctive formula and thin refutation. Next we show that a thin refutation for a formula φ can be converted into a proof of $\neg\varphi$. This together with the fact that every refutation of a weakly aconjunctive formula is thin gives us a slight generalisation of the main result from [2]. We will use thin refutations in the proof of our main theorem which states that for every formula φ there is an equivalent disjunctive formula $\widehat{\varphi}$ such that $\varphi \Rightarrow \widehat{\varphi}$ is provable. From this theorem follows the completeness theorem if one observes that for every unsatisfiable disjunctive formula $\widehat{\varphi}$, formula $\neg\widehat{\varphi}$ is provable.

Kozen's system for the μ-calculus consists of the axioms and rules of equational logic (including substitution of equals by equals, i.e., cut rule) and the axioms and rules presented in Figure 2 (notation $\gamma \leq \delta$ is a shorthand for $\gamma \vee \delta = \delta$).

Because we have put ν, $[\,]$ and $(a \to \Psi)$ constructs directly into the language we have to define them by equivalences:

$$[a]\alpha = \neg\langle a\rangle\neg\alpha \quad \nu X.\alpha(X) = \neg\mu X.\neg\alpha(\neg X)$$

$$(a \to \Psi) = \bigwedge\{\langle a\rangle\psi : \psi \in \Psi\} \wedge [a]\bigvee\Psi$$

We will use sequent notation. A sequent $\Gamma \vdash \Delta$ is a pair of finite sets of formulas. The meaning of $\Gamma \vdash \Delta$ is that $\Gamma \models \bigvee\Delta$, where $\bigvee\Delta$ is a disjunction of formulas from Δ. We write $\Gamma \models \alpha$ to mean that for every model $\mathcal{M}$, its state s and valuation *Val*, if $\mathcal{M}, s, Val \models \Gamma$ then $\mathcal{M}, s, Val \models \alpha$. The meaning of the sequent $\Gamma\vdash$ with the right hand side empty is that the conjunction of formulas from Γ is not satisfiable.

We will use the following admissible rules:

$$(\wedge) \quad \frac{\alpha, \beta, \Gamma\vdash}{\alpha\wedge\beta, \Gamma\vdash} \quad (\vee) \quad \frac{\alpha, \Gamma\vdash \quad \beta, \Gamma\vdash}{\alpha\vee\beta, \Gamma\vdash}$$

$$(\sigma) \quad \frac{\alpha(\sigma X.\alpha(X)), \Gamma\vdash}{\sigma X.\alpha(X), \Gamma\vdash}$$

$$(\langle\rangle) \quad \frac{\{\psi\} \cup \{\bigvee\Theta : (a \to \Theta) \in \Gamma, \Theta \neq \Psi\}\vdash}{(a \to \Psi), \Gamma\vdash} \quad \psi \in \Psi$$

$$(\leq) \quad \frac{\alpha\vdash\beta}{\varphi(\alpha)\vdash\varphi(\beta)} \quad X \text{ only positive in } \varphi(X)$$

First three of these rules are direct translations of the corresponding tableau rules. Rule $(\langle\rangle)$ corresponds to the rule (*mod*) but important difference is that unlike (*mod*) rule $(\langle\rangle)$ has only one assumption.

According to our proviso we restrict ourselves to positive, guarded formulas which use $(a \to \Psi)$ construct instead of $\langle a\rangle$ and $[a]$ modalities. It can be checked that every formula is provably equivalent to a formula of this form.

Definition 5.1 (Aconjunctiveness) *Let φ be a formula, $\mathcal{D}_\varphi$ be its binding function and let $\leq_\varphi$ be the dependency ordering (see Definition 2.3).*

— *We say that a variable X is* active *in ψ, a subformula of φ, iff there is a variable Y appearing in ψ with $X \leq_\varphi Y$.*

— *Let X be a variable with its binding definition $\mathcal{D}_\varphi(X) = \mu X.\gamma(X)$. Variable X is called aconjunctive iff for all subformulas of γ of the form $\alpha\wedge\beta$ it is not the case that X is active in α as well as in β.*

— *Variable X as above is called* weakly aconjunctive *iff for all subformulas of γ of the form $\alpha\wedge\beta$ if X is active both in α and β then $\alpha\wedge\beta$ is a special conjunction as defined in Definition 4.1.*

— *Formula φ is called* (weakly) aconjunctive *iff all μ-variables in φ are (weakly) aconjunctive.*

In the following we will be interested only in weakly aconjunctive formulas. Definition of aconjunctive formulas was recalled only to give a comparison of the two notions.

The next proposition states some closure properties of the set of weakly aconjunctive formulas. Observe

$(K1)$	axioms for Boolean algebra	$(K2)$	$\langle a\rangle\varphi \vee \langle a\rangle\psi = \langle a\rangle(\varphi \vee \psi)$
$(K3)$	$\langle a\rangle\varphi \wedge [a]\psi \leq \langle a\rangle(\varphi \wedge \psi)$	$(K4)$	$\langle a\rangle ff = ff$
$(K5)$	$\alpha(\mu X.\alpha(X)) \leq \mu X.\alpha(X)$	$(K6)$	$\dfrac{\alpha(\varphi) \leq \varphi}{\mu X.\alpha(X) \leq \varphi}$

Figure 2: Kozen's axiomatisation

that weakly aconjunctive formulas are not closed under taking the least fixpoint.

Proposition 5.2 *If $\gamma(X)$ and δ are weakly aconjunctive formulas then $\gamma[\delta/X]$, $\nu X.\gamma(X)$ and $\delta \wedge \gamma(X)$ are also weakly aconjunctive formulas.*

Definition 5.3 (Refutation) Refutation $\mathcal{R}$ *for a formula φ is defined as a tableau, but this time we modify system $\mathcal{S}_\varphi$ by adding weakening rule and instead of (mod) rule we take rule ($\langle\rangle$). We also require that every leaf of $\mathcal{R}$ is labeled by a set containing some literal and its negation and there is a μ-trace on every infinite path of $\mathcal{R}$.*

We call a refutation thin *iff whenever a formula of the form $\alpha \wedge \beta$ is reduced in some node of the refutation and some variable is active in α as well as in β then either $\alpha \wedge \beta$ is a special conjunction or one of the conjuncts is immediately discarded by the use of weakening rule.*

It is quite easy to see that every refutation for a weakly aconjunctive formula is a thin refutation. In [4] it was shown that every unsatisfiable formula has a (not necessary thin) refutation. From this and the next theorem it follows that negations of unsatisfiable weakly aconjunctive formulas are provable. The theorem is stated more generally because in Lemma 5.7 we deal with thin refutations for possibly not weakly aconjunctive formulas. The proof of the theorem is essentially a repetition of the arguments from [2].

Theorem 5.4 *If there exists a thin refutation $\mathcal{R}$ for a formula φ then the sequent $\varphi\vdash$ is provable.*

Corollary 5.5 *For every unsatisfiable disjunctive formula $\widehat{\varphi}$ the formula $\neg\widehat{\varphi}$ is provable.*

It should be mentioned that there is also quite simple direct proof of the corollary which does not use Theorem 5.4.

To prove completeness theorem it is now enough to show that for every unsatisfiable formula φ there is an equivalent disjunctive formula $\widehat{\varphi}$ such that $\varphi\vdash\widehat{\varphi}$ is provable. Of course we could just take $\widehat{\varphi}$ to be *false* but then the proof of this fact would be exactly as difficult as showing completeness. So in general we will look for more complicated formulas than *false*. Because we will prove this fact by induction on φ we clearly need to consider also satisfiable formulas.

Theorem 5.6 *For every formula φ (satisfying our proviso) there exists an equivalent disjunctive formula $\widehat{\varphi}$ such that $\varphi\vdash\widehat{\varphi}$ is provable.*

Proof
The proof proceeds by induction on the structure of the formula φ.

Case: φ is a literal In this case $\widehat{\varphi}$ is just φ.

Case: $\varphi = \alpha \vee \beta$ By induction assumption there are disjunctive formulas $\widehat{\alpha}, \widehat{\beta}$ equivalent to α and β respectively. We let $\widehat{\alpha \vee \beta}$ to be $\widehat{\alpha} \vee \widehat{\beta}$. Of course $\alpha \vee \beta \vdash \widehat{\alpha} \vee \widehat{\beta}$ is provable because $\alpha\vdash\widehat{\alpha}$ and $\beta\vdash\widehat{\beta}$ are provable.

Case: $\varphi = (a \to \Phi)$ This case is very similar to the above.

Case: $\varphi = \nu X.\alpha(X)$ By the induction assumption there is a disjunctive formula $\widehat{\alpha}(X)$ equivalent to $\alpha(X)$. Of course $\nu X.\alpha(X)$ is equivalent to $\nu X.\widehat{\alpha}(X)$ and $\nu X.\alpha(X) \vdash \nu X.\widehat{\alpha}(X)$ is provable. Unfortunately $\nu X.\widehat{\alpha}(X)$ may not be a disjunctive formula. This is because X may occur in a context $X \wedge \gamma$ for some γ. Therefore we have to construct $\widehat{\varphi}$ from the scratch.

By Theorem 4.2 there is a disjunctive formula $\widehat{\varphi}$ which has the tableau equivalent to some regular tableau $\mathcal{T}$ for $\nu X.\widehat{\alpha}(X)$. By Theorem 3.7 the two formulas are equivalent. We are left to show that $\nu X.\widehat{\alpha}(X) \vdash \widehat{\varphi}$ is provable in Kozen's system. As every disjunctive formula is a weakly aconjunctive formula, by Proposition 5.2 we have that $\nu X.\widehat{\alpha}(X)$ is a weakly aconjunctive formula. Unfortunately we cannot directly apply Theorem 5.4

to $\nu X.\widehat{\alpha}(X), \neg\widehat{\varphi} \vdash$. This is because $\neg\widehat{\varphi}$ may not be a weakly aconjunctive formula. Nevertheless we know that the two formulas have equivalent tableaux and we can use this information.

Lemma 5.7 *Suppose that we have a weakly aconjunctive formula α and a disjunctive formula δ which have equivalent tableaux. In this case the sequent $\alpha \vdash \delta$ is provable.*

Proof

We would like to sketch how tableau equivalence is used to overcome the problem described above.

Let $\mathcal{T}_\alpha = \langle T_\alpha, L_\alpha\rangle$ and $\mathcal{T}_\delta = \langle T_\delta, L_\delta\rangle$ be tableaux for α and δ respectively. Let $\mathcal{E} : \mathcal{T}_\alpha \to \mathcal{T}_\delta$ be an equivalence function. We will construct a thin refutation $\mathcal{R}$ for $\alpha, \neg\delta \vdash$.

To facilitate the construction we will define a correspondence function $\mathcal{C}_\alpha$ which assigns to every considered node n of $\mathcal{R}$ (that is not to every node) a node $\mathcal{C}_\alpha(n)$ of $\mathcal{T}_\alpha$ such that:

$(*)\ \ L(n) = L_\alpha(\mathcal{C}_\alpha(n)) \cup \{\neg \bigwedge L_\delta(\mathcal{E}(\mathcal{C}_\alpha(n)))\}$

The root r of $\mathcal{R}$ will be of course labeled by $\alpha, \neg\delta \vdash$ and setting $\mathcal{C}_\alpha(r)$ to be the root of $\mathcal{T}_\alpha$ establishes condition $(*)$.

Now suppose that n is a node of $\mathcal{R}$ under construction and $\mathcal{C}(n)$ is a choice node such that $(*)$ holds.

As δ is a disjunctive formula $L_\delta(\mathcal{E}(\mathcal{C}_\alpha(n)))$ is one element set, say $\{\gamma\}$. Hence $L(n) = L_\alpha(\mathcal{C}_\alpha(n)) \cup \{\neg\gamma\}$. Let us apply as long as possible rules other than $(\langle\rangle)$ and weakening to all the formulas in $L(n)$ except $\neg\gamma$ in the same order as they were applied from $\mathcal{C}_\alpha(n)$. This way we obtain a part of a tree rooted in n with leaves $n_1, \ldots, n_k$. For every $j = 1, \ldots, k$ the label $L(n_j)$ contains $\neg\gamma$ and some set of formulas Γ_j to which only $(\langle\rangle)$ may be applicable. It is easy to see that every n_j corresponds to some modal node $\mathcal{C}_\alpha(n_j)$ near $\mathcal{C}_\alpha(n)$ with the property that $L_\alpha(\mathcal{C}_\alpha(n_j)) = \Gamma_j$.

Let us look at the path from $\mathcal{E}(\mathcal{C}_\alpha(n))$ to $\mathcal{E}(\mathcal{C}_\alpha(n_j))$ in $\mathcal{T}_\delta$. Because δ is a disjunctive formula, on this path first only (σ), (reg) and (or) rules may be applied and then we have zero or more applications of (and) rule. Let us apply dual rules to $\neg\gamma$: dual to (μ) is (ν), (reg) is self-dual, dual to (or) is (and) but when we apply this rule we immediately use weakening to leave only the conjunct which appears on the path to $\mathcal{E}(\mathcal{C}_\alpha(n_j))$. We do not apply (or) rules.

This way we arrive at a node m_j. If we define $\mathcal{C}_\alpha(m_j) = \mathcal{C}_\alpha(n_j)$ then its label can be presented as $L_\alpha(\mathcal{C}_\alpha(m_j)) \cup \{\neg \bigwedge L_\delta(\mathcal{E}(\mathcal{C}_\alpha(m_j)))\}$. Hence $(*)$ is satisfied and $\mathcal{C}_\alpha(m_j)$ is a modal node. We can now apply similar reasoning to prolong $\mathcal{R}$ from each of the constructed nodes.

This way we obtain a thin refutation for $\alpha, \neg\delta \vdash$. By construction all finite paths of the constructed tree end with axioms. For every infinite path $\mathcal{P}$ we have two possibilities. There may be a μ-trace on a path of $\mathcal{T}_\alpha$ designated by the image of $\mathcal{P}$ under $\mathcal{C}_\alpha$. If it is so then by construction the same trace appears on $\mathcal{P}$. If there is no μ-trace on $\mathcal{C}_\alpha(\mathcal{P})$ then, by the definition of tableau equivalence, there cannot be a μ-trace on $\mathcal{E}(\mathcal{C}_\alpha(\mathcal{P}))$. Hence there is a ν-trace on $\mathcal{E}(\mathcal{C}_\alpha(\mathcal{P}))$ which negated in $\mathcal{R}$ becomes a μ-trace.

This means that every finite path of $\mathcal{R}$ ends in a set containing a literal and its negation and on every infinite path there is a μ-trace. Hence $\mathcal{R}$ is a refutation. It is also a thin refutation because of the way we have constructed it and the fact that α is a weakly aconjunctive formula. Hence by Theorem 5.4 the sequent $\alpha \wedge \neg\delta \vdash$ is provable.

□

Case: $\varphi = \alpha \wedge \beta$ The argument is very similar to the one from the previous case. Lemma 5.7 is also used here.

Case: $\varphi = \mu X.\alpha(X)$ This case is more complicated than the case for the greatest fixpoint. As in that case we have by the induction assumption a disjunctive formula $\widehat{\alpha}(X)$ equivalent to $\alpha(X)$. Unfortunately this time $\mu X.\widehat{\alpha}(X)$ may not be a weakly aconjunctive formula so we cannot use the same argument as in the last two cases. Let us nevertheless try to carry on and see where modifications are needed.

By Theorem 4.2 there is a disjunctive formula $\widehat{\varphi}$ which has the tableau equivalent to some regular tableau $\mathcal{T}$ for $\mu X.\widehat{\alpha}(X)$. By Theorem 3.7 the two formulas are equivalent. We are left to show that $\mu X.\widehat{\alpha}(X) \vdash \widehat{\varphi}$ is provable in Kozen's system. Because $\mu X.\widehat{\alpha}(X)$ may not be an aconjunctive formula we cannot just use Lemma 5.7 to $\mu X.\widehat{\alpha}(X)$ and $\widehat{\varphi}$. What we could do is to use Park's induction rule (P) if only we could prove the sequent $\widehat{\alpha}(\widehat{\varphi}) \vdash \widehat{\varphi}$.

By Proposition 5.2 we know that $\widehat{\alpha}(\widehat{\varphi})$ is a weakly aconjunctive formula but we meet another obstacle preventing us from using Lemma 5.7. We don't know weather $\widehat{\alpha}(\widehat{\varphi})$ and $\widehat{\varphi}$ have equivalent tableaux. Actually it may happen that the two formulas do not have equivalent tableaux.

What we need is some weaker notion of correspondence between tableaux but it should be strong enough to give us something like Lemma 5.7. Below we propose such a notion which we call *tableau consequence*. This notion will be defined in terms of games on tableaux. To simplify the definition of these games we will introduce a notion of a *wide tableau*.

$$(wmod)\quad \frac{\{\psi\}\cup\{\bigvee\Theta : (a\to\Theta)\in\Gamma, \Theta\neq\Psi\} \quad \text{for every } (a\to\Psi)\in\Gamma, \psi\in\Psi \qquad \{\bigvee\Theta : (a\to\Theta)\in\Gamma\}}{\Gamma}$$

Figure 3: Rule (*wmod*)

Definition 5.8 Wide tableaux *are constructed according to the same rules as tableaux but rule (mod) is replaced by (wmod) presented in Figure 3.*

Compared to (mod) rule (wmod) has new assumptions, one for each action a such that there is a formula of the form $(a \to \Psi)$ *in* Γ. *We will call sons of the old type* $\langle a\rangle$-sons. *The sons of the new type will be called* $[a]$-sons. *Observe that for every action a we can have at most one* $[a]$*-son of a node.*

Remark: For example:

$$\frac{\{\alpha_1\} \quad \{\alpha_2\} \quad \{\alpha_1\vee\alpha_2\} \quad \{\beta_1\} \quad \{\beta_2\} \quad \{\beta_1\vee\beta_2\}}{\{(a\to\{\alpha_1,\alpha_2\}),(b\to\{\beta_1,\beta_2\})\}}$$

is an instance of (*wmod*) rule. There are two $\langle a\rangle$-sons, two $\langle b\rangle$-sons, one $[a]$-son and one $[b]$-son.

Rule (*mod*) with the same conclusion would have only four assumptions, $[a]$ and $[b]$-sons would be missing. In rule (*wmod*) we make explicit the universal requirements of the conclusion. The meaning of $(a \to \{\alpha_1,\alpha_2\})$ is that there is a state reachable by action a where α_1 is satisfied (represented by $\langle a\rangle$-son labeled $\{\alpha_1\}$), there is a state where α_2 is satisfied (represented by $\langle a\rangle$-son labeled $\{\alpha_2\}$) and all the states reachable by action a must satisfy $\alpha_1\vee\alpha_2$, this is represented by $[a]$-son labeled $\{\alpha_1\vee\alpha_2\}$.

Definition 5.9 (Game) *For a given pair of wide tableaux* $(\widetilde{\mathcal{W}},\mathcal{W})$, *where* $\widetilde{\mathcal{W}}=\langle\widetilde{T},\widetilde{L}\rangle$ *and* $\mathcal{W}=\langle T,L\rangle$, *we define a two player game* $\mathcal{G}(\widetilde{\mathcal{W}},\mathcal{W})$ *with the following rules.*

- *The starting position is a pair of the roots of both tableaux.*
- *If a position of a play is* $(\widetilde{n},n)$, *both nodes being choice nodes of* $\widetilde{\mathcal{W}}$ *and* $\mathcal{W}$ *respectively, then player I chooses a modal node* $\widetilde{m}$ *near* $\widetilde{n}$ *and player II replies by choosing a modal node m near n with the property that every literal and terminal formula (see Definition 2.7) from* $L(m)$ *appears in* $\widetilde{L}(\widetilde{m})$.
- *If a position is* $(\widetilde{m},m)$, *a pair of modal nodes from* $\widetilde{\mathcal{W}}$ *and* $\mathcal{W}$ *respectively, then player I can choose a son n of m and player II has to respond with a son* $\widetilde{n}$ *of* $\widetilde{m}$ *of the same kind. That is if n is a* $\langle a\rangle$*-son then* $\widetilde{n}$ *must be a* $\langle a\rangle$*-son and if n is a* $[a]$*-son so must be* $\widetilde{n}$.

The game may end in a finite number of steps because one of the players cannot make a move. In this case the opposite player wins. When the game has infinitely many steps we get as the result two infinite paths: $\widetilde{\mathcal{P}}$ *from* $\widetilde{\mathcal{W}}$ *and* $\mathcal{P}$ *from* $\mathcal{W}$. *Player I wins if there is no* μ*-trace on* $\widetilde{\mathcal{P}}$ *but there is a* μ*-trace on* $\mathcal{P}$, *otherwise player II is the winner.*

Definition 5.10 (Tableau consequence) *A strategy* $\mathcal{S}$ *for the second player in the game* $\mathcal{G}(\widetilde{\mathcal{W}},\mathcal{W})$ *is a partial function giving for a position consisting of two choice nodes* $(\widetilde{n},n)$ *and a modal node* $\widetilde{m}$ *near* $\widetilde{n}$ *a modal node* $\mathcal{S}(\widetilde{m},n)$ *near n. If* $(\widetilde{m},m)$ *is a pair of modal nodes and n is a son of m then the strategy gives us a son* $\mathcal{S}(\widetilde{m},n)$ *of* $\widetilde{m}$. *A strategy is called* winning *for II iff it guarantees that player II wins the game no matter what the moves of player I are. This also implies that the strategy is defined for appropriate positions.*

We will say that a wide tableau $\mathcal{W}$ *is a* consequence *of a wide tableau* $\widetilde{\mathcal{W}}$ *iff player II has a wining strategy in* $\mathcal{G}(\widetilde{\mathcal{W}},\mathcal{W})$.

The definition of the game is based on the following intuition about wide tableaux. Wide tableau for a formula describes "operationally" semantics of a formula. In order to satisfy formulas in a choice node n we must provide a state which satisfies the label of one of the modal nodes near n. The sons of a modal node describe the transitions from a hypothetical state satisfying its label. $\langle a\rangle$-sons describe which a-successors are required and the $[a]$-son describes what are general conditions all a-successors must satisfy. In this way tableau of a formula describes all possible models of the formula.

The game is defined so that whenever *II* has a winning strategy from a position $(\widetilde{n},n)$ then every model of the label of $\widetilde{n}$, $\widetilde{L}(\widetilde{n})$, is also a model of the label of n, $L(n)$. If $\widetilde{n}$ and n are both choice nodes then a model for $\widetilde{L}(\widetilde{n})$ must satisfy the label of one of the modal nodes near $\widetilde{n}$. Hence for every modal node near $\widetilde{n}$ we must show a modal node near n which label is implied by it. If $\widetilde{n}$ and n are both modal nodes then every $\langle a\rangle$-son of n describes a state the existence of which is required in order to satisfy $L(n)$. We must show that the existence of such a state is also required by $\widetilde{L}(\widetilde{n})$. The $[a]$-son of n represents the general requirements on states reachable by action a imposed by $L(n)$. We must show that they are implied by the general requirements in $\widetilde{L}(\widetilde{n})$

The following lemma can be proved using exactly the same method as in Lemma 5.7

Lemma 5.11 *Suppose that we have a weakly aconjunctive formula α and a disjunctive formula δ such that there is a wide tableau for δ which is a consequence of a wide tableau for α. In this case the sequent $\alpha \vdash \delta$ is provable.*

To finish the completeness proof it is enough to show that there is a wide tableau for $\widehat{\varphi}$ which is a consequence of a wide tableau for $\widehat{\alpha}(\widehat{\varphi})$. First we will need some wide tableaux for $\mu X.\widehat{\alpha}(X)$ and $\widehat{\varphi}$. The following lemma guarantees existence of these.

Lemma 5.12 *For a given pair of equivalent tableaux $\mathcal{T}_1$ and $\mathcal{T}_2$ for formulas φ_1 and φ_2 respectively, we can construct wide tableaux $\mathcal{W}_1$ for φ_1 and $\mathcal{W}_2$ for φ_2 such that $\mathcal{W}_2$ is a consequence of $\mathcal{W}_1$ and $\mathcal{W}_1$ is a consequence of $\mathcal{W}_2$.*

Lemma 5.13 *Let $\mathcal{W}, \widehat{\mathcal{W}}$ be a pair of wide tableaux for $\mu X.\widehat{\alpha}(X)$ and $\widehat{\varphi}$ respectively constructed from $\mathcal{T}$ and $\widehat{\mathcal{T}}$ as in Lemma 5.12. There is a wide tableau $\widetilde{\mathcal{W}}$ for $\widehat{\alpha}(\widehat{\varphi})$ of which $\mathcal{W}$ is a consequence.*

Proof

Let L and $\widehat{L}$ by labeling functions of $\mathcal{W}$ and $\widehat{\mathcal{W}}$ respectively. Let $\mathcal{S}$ be a winning strategy in the game $\mathcal{G}(\widehat{\mathcal{W}}, \mathcal{W})$. There is a very close match between $\mathcal{W}$ and a wide tableau for $\widehat{\alpha}(\mu X.\widehat{\alpha}(X))$ as the only tableau rule applicable to $\{\mu X.\widehat{\alpha}(X)\}$ is (μ). Let us denote by $\mathcal{W}$ also the wide tableau for $\widehat{\alpha}(\mu X.\widehat{\alpha}(X))$. Let us use notation $\beta[\widehat{\varphi}/\mu X.\widehat{\alpha}(X)]$ to stand for the obvious replacement, it will be always the case that no free variable in $\mu X.\widehat{\alpha}(X)$ or $\widehat{\varphi}$ is bound by the context β.

To define a strategy from a position $(\widetilde{m}, m)$ we will use some additional information about the position. This information will come from the three functions $p(\widetilde{m})$, $nd(\widetilde{m})$ and $\widehat{nd}(\widetilde{m})$. The first function assigns a priority, that is a natural number or ∞, to every formula in $\widetilde{L}(\widetilde{m})$. The functions $nd(\widetilde{m})$ and $\widehat{nd}(\widetilde{m})$ are partial functions which assign nodes of $\mathcal{W}$ and $\widehat{\mathcal{W}}$ respectively to finite priorities from the image of $p(\widetilde{m})$, i.e., from the set $\{q \in \mathcal{N} : p(\widetilde{m})^{-1}(q) \neq \emptyset\}$.

This situation is presented in Figure 4. We have a position $(\widetilde{m}, m)$ of the main play of the game $\mathcal{G}(\widetilde{\mathcal{W}}, \mathcal{W})$ and for every natural number in the image of $p(\widetilde{m})$ we have a sub-play of $\mathcal{G}(\widehat{\mathcal{W}}, \mathcal{W})$ in a position given by functions $\widehat{nd}(\widetilde{m})$ and $nd(\widetilde{m})$. To make a move in the main play we consult strategy $\mathcal{S}$ for sub-plays.

For every considered position $(\widetilde{m}, m)$ the following three conditions will be satisfied:

(*i*) for every $q \in \mathcal{N}$ in the image of $p(\widetilde{m})$: $p(\widetilde{m})^{-1}(q) \subseteq \widehat{L}(\widehat{nd}(\widetilde{m})(q)) \subseteq \bigcup\{p(\widetilde{m})^{-1}(q') : q' \leq q\}$,

(*ii*) $L(m) \subseteq \bigcup_{q\in\mathcal{N}} L(nd(\widetilde{m})(q)) \cup \{\psi[\mu X.\widehat{\alpha}(X)/\widehat{\varphi}] : \psi \in p^{-1}(\widetilde{m})(\infty)\}$

(*iii*) for every $q \in \mathcal{N}$ in the image of $p(\widetilde{m})$, strategy $\mathcal{S}$ is defined for the position $(\widehat{nd}(\widetilde{m})(q), nd(\widetilde{m})(q))$.

The idea of a strategy in $\mathcal{G}(\widetilde{\mathcal{W}}, \mathcal{W})$ is to consult $\mathcal{S}$ in every step and the above three conditions allow us to do this. Whenever a position is a pair of modal nodes $(\widetilde{m}, m)$ and I chooses a son of m, this son is designated by some formulas $\xi \in \Xi$ and $(a \to \Xi) \in L(m)$. By condition (*ii*) we know that either $(a \to \Xi)[\widehat{\varphi}/\mu X.\widehat{\alpha}(X)] \in \widetilde{L}(\widetilde{m})$ or there is a priority $q \in \mathcal{N}$ such that $(a \to \Xi) \in nd(\widetilde{m})(q)$. In the first case we take a son of $\widetilde{m}$ designated by $\xi[\widehat{\varphi}/\mu X.\widehat{\alpha}(X)]$ and $(a \to \Xi)[\widehat{\varphi}/\mu X.\widehat{\alpha}(X)]$. In the second case by (*iii*) we can take an a-son of $\widehat{nd}(\widetilde{m})(q)$ which is a response of $\mathcal{S}$ to choosing a son of $nd(\widetilde{m})(q)$ designated by ξ and $(a \to \Xi)$. By (*i*) we can find an appropriate son of $\widetilde{m}$. The scheme of the reasoning in case when a position is a pair of choice nodes is similar.

To show that the strategy is winning one assumes conversely that there is a play when *II* plays according to the strategy and looses. By the definition of the strategy *II* cannot lose in finite number of steps, hence the result of the game are to infinite paths, $\widetilde{\mathcal{P}}$ from $\widetilde{\mathcal{W}}$ and $\mathcal{P}$ from $\mathcal{W}$. Because I won the game there is a μ-trace T on $\mathcal{P}$ and there is no μ-trace on $\widetilde{\mathcal{P}}$.

Condition (*ii*) allows us to define for every considered node n of $\mathcal{P}$ a priority $p(n)(T(n))$. The strategy is defined in such a way that this priority can only decrease, so from some point, say node n_0, it must be constant, say equal $q \in \mathcal{N}$ (the case $q = \infty$ is easier). Let $n_0, n_1, \ldots$ be successive choice nodes on $\mathcal{P}$ and $\widetilde{n}_0, \widetilde{n}_1, \ldots$ corresponding choice nodes of $\widetilde{\mathcal{P}}$. For every $i \in \mathcal{N}$ by (*ii*) we have $T(n_i) \in L(nd(\widetilde{n}_i)(q))$. It turns out that the same variables are regenerated on T and the trace designated by T on the path $\mathcal{P}' = \{nd(\widetilde{n}_i)(q)\}_{i\in\mathcal{N}}$. Hence there is a μ-trace on $\mathcal{P}'$. On the other hand by (*i*) $\widehat{L}(\widehat{nd}(\widetilde{n}_i)(q)) \subseteq \widetilde{L}(\widetilde{n}_i)$ and as there is no μ-trace on $\widetilde{\mathcal{P}}$ the unique trace on the path $\widehat{\mathcal{P}}$ designated by $\{\widehat{nd}(\widetilde{n}_i)(q)\}_{i\in\mathcal{N}}$ is not a μ-trace. But

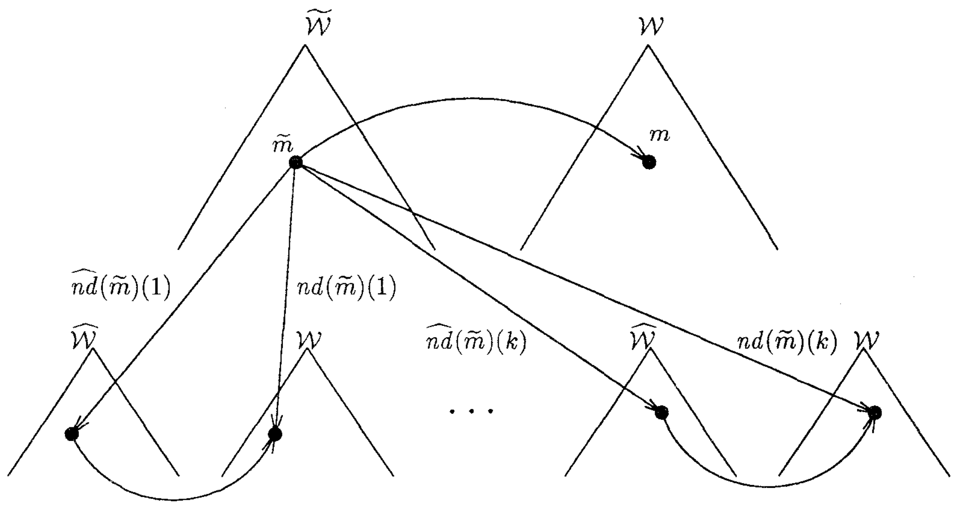

Figure 4: Auxiliary functions

a pair of paths $\widehat{\mathcal{P}}$ and $\mathcal{P}'$ is the result of the sub-play for priority q where II played according to $\mathcal{S}$. This is a contradiction because we have assumed that $\mathcal{S}$ is winning. □

Summarising the case of the proof for $\varphi = \mu X.\alpha(X)$. By induction assumption we have a disjunctive formula $\widehat{\alpha}(X)$ equivalent to $\alpha(X)$ and know that $\alpha(X) \vdash \widehat{\alpha}(X)$ is provable. By Theorem 4.2 we obtain a disjunctive formula $\widehat{\varphi}$ which has a tableau $\widehat{\mathcal{T}}$ equivalent to some regular tableau $\mathcal{T}$ for $\mu X.\widehat{\alpha}(X)$. By Theorem 3.7, formula $\widehat{\varphi}$ is equivalent to φ. By Lemma 5.12 there are wide tableaux: $\mathcal{W}$ for $\mu X.\widehat{\alpha}(X)$ and $\widehat{\mathcal{W}}$ for $\widehat{\varphi}$ such that $\mathcal{W}$ is a consequence of $\widehat{\mathcal{W}}$ and vice versa. By Lemma 5.13 $\mathcal{W}$ is a consequence of $\widetilde{\mathcal{W}}$, a wide tableau for $\widehat{\alpha}(\widehat{\varphi})$. Hence, as the tableau consequence relation is transitive, $\widehat{\mathcal{W}}$ is a consequence of $\widetilde{\mathcal{W}}$. Now by Proposition 5.2 $\widehat{\alpha}(\widehat{\varphi})$ is a weakly aconjunctive formula and $\widehat{\varphi}$ is by definition a disjunctive formula. By Lemma 5.11 the sequent $\widehat{\alpha}(\widehat{\varphi}) \vdash \widehat{\varphi}$ is provable. Then $\mu X.\widehat{\alpha}(X) \vdash \widehat{\varphi}$ is provable by rule (P) and $\varphi \vdash \mu X.\widehat{\alpha}(X)$ is provable by induction assumption, hence the sequent $\varphi \vdash \widehat{\varphi}$ is provable.

□

Theorem 5.14 (Completeness) *For every unsatisfiable formula φ sequent $\varphi \vdash$ is provable.*

Proof

Let φ be an unsatisfiable formula. As every formula is provably equivalent to a positive, guarded and well named formula we may assume that φ is such a formula. Form Theorem 5.6 follows that there is a disjunctive formula $\widehat{\varphi}$ equivalent to φ and the sequent $\varphi \vdash \widehat{\varphi}$ is provable. By Proposition 5.5 sequent $\widehat{\varphi} \vdash$ is provable. □

References

[1] D. Harel, D Kozen, and R. Parikh. Process logic: Expressiveness, decidability and completeness. *Journal of Computer and System Sciences*, 25:144–201, 1982.

[2] Dexter Kozen. Results on the propositional mu-calculus. *Theoretical Computer Science*, 27:333–354, 1983.

[3] D.E. Muller and P.E. Schupp. Alternating automata on infinite trees. *Theoretical Computer Science*, 54:267–276, 1987.

[4] D. Niwiński and I. Walukiewicz. Games for μ-calculus. Technical Report TR 94-03(192), Institute of Informatics, Warsaw Univeristy, February 1994. To appear in TCS.

[5] Damian Niwiński. Fixed points vs. infinite generation. In *Proc. 3rd. IEEE LICS*, pages 402–409, 1988.

[6] Colin S. Stirling. Modal and teporal logics for processes. to appear in LNCS.

[7] Colin S. Stirling and David J. Walker. Local model checking in the modal mu-calculus. *Theoretical Computer Science*, 89:161–171, 1991.

[8] Robert S. Street and E. Allan Emerson. An automata theoretic procedure for the propositional mu-calculus. *Information and Computation*, 81:249–264, 1989.

[9] Igor Walukiewicz. On completeness of the μ-calculus. In *LICS '93*, pages 136–146, 1993.