Fibring Semantic Tableaux

Bernhard Beckert[1,⋆] and Dov Gabbay[2]

[1] University of Karlsruhe, Institute for Logic, Complexity and Deduction Systems, D-76128 Karlsruhe, Germany. E-mail: beckert@ira.uka.de
[2] Imperial College, Department of Computing, 180 Queen's Gate, London SW7 2BZ, UK. E-mail: dg@ic.ac.uk

Abstract. The methodology of *fibring* is a successful framework for combining logical systems based on combining their semantics. In this paper, we extend the fibring approach to *calculi* for logical systems: we describe how to uniformly construct a sound and complete tableau calculus for the combined logic from calculi for the component logics.
We consider semantic tableau calculi that satisfy certain conditions and are therefore known to be "well-behaved"—such that fibring is possible. The identification and formulation of conditions that are neither too weak nor too strong is a main contribution of this paper.
As an example, we fibre tableau calculi for first order predicate logic and for the modal logic K.

1 Introduction

The methodology of *fibring* is a successful framework for combining logical systems based on combining their semantics [7, 6, 8]. The basic idea is to combine the structures defining the semantics of two logics $\mathbf{L}_1$ and $\mathbf{L}_2$ such that the result can be used to define semantics for expressions from the combined languages of $\mathbf{L}_1$ and $\mathbf{L}_2$. The general assumption is that these structures have components like, for example, the worlds in Kripke structures; to build fibred structures, *fibring functions* $F_{(1,2)}$ are defined assigning to each constituent w of an $\mathbf{L}_1$-model $\mathbf{m}_1$ an $\mathbf{L}_2$-model $\mathbf{m}_2$. An $\mathbf{L}_2$-expression is evaluated in w, where its value is undefined, by instead evaluating it in $\mathbf{m}_2 = F_{(1,2)}(w)$. The full power of the fibring method is revealed when this process it iterated to define a semantics for the logic $\mathbf{L}_{[1,2]}$, where the operators of the component logics can occur arbitrarily nested in formulae. Fibring has been successfully used in many areas of logic to combine systems and define their semantics; for an overview see [7].

In this paper, we extend the fibring approach to *calculi* for logical systems: we describe how to uniformly construct a sound and complete tableau calculus for the combined logic from calculi for the component logics. Since tableau calculi are known for most "basic" logics [5] (including classical logic, modal logic, intuitionistic logic, and temporal logic), calculi can be obtained for all "complex" logics that can be constructed by fibring basic logics, such as modal predicate logic, intuitionistic temporal logic, etc.

⋆ This work was carried out during a visit at Imperial College, London, UK.

One cannot fibre just any proof procedures for two logics in a uniform way. First, "proving" can have different meanings in different logics: deciding (or semi-deciding) satisfiability or validity, computing a satisfying variable instantiation, etc. Second, it is not clear where to "plug in" the proof procedure for $\mathbf{L}_2$ into that for $\mathbf{L}_1$; a proof procedure may do something completely different from what (the definition of) the valuation function does that provides the truth value of a formula in a given model. For example, if the procedure P_1 is based on constructing a (counter) model, whereas the procedure P_2 uses a resolution calculus, they cannot be fibred (at least not uniformly).

Therefore, we consider semantic tableau calculi that satisfy certain conditions and are, thus, known to be "well-behaved"—such that fibring is possible (for some substructural logics, e.g. linear logic, no such "well-behaved" calculi exist). The identification and formulation of conditions that are neither too weak nor too strong is a main contribution of this paper.

If the components that are fibred satisfy these conditions, then the resulting calculus is automatically sound and complete. It may only be a semi-decision procedure, i.e., only terminate for unsatisfiable input formulae, even if its components are decision procedures; this, however, is not surprising because a fibred logic may be undecidable even if its components are decidable.

Related work includes [4], where a method for fibring tableau calculi for substructural implication logics has been presented. In [9], a method is described for fibring tableaux for modal logics to construct calculi for multi-modal logics; it can be seen as an instance of the general framework presented here.

We define the notion of a *logical system* in a very general way (Section 2); only indispensable properties of its syntax and semantics are part of the definition without which a useful tableau calculus for the logic cannot exist (or cannot be fibred with calculi for other logical systems).

Similarly, as few restrictions as possible are made regarding the type and form of tableau calculi. In particular, the calculus does not have to be analytical; and the tableau rules do not have to be given in form of rule schemata but can be described in an arbitrary way. The conditions that tableau calculi have to satisfy to be suitable for fibring are described in Section 3. We present two examples of calculi suitable for fibring in Sections 4 and 5: a calculus for first order predicate logic and a calculus for the modal logic K. In Section 6, the method of fibring logics is described in general and syntax and semantics of a fibred logic are defined, based on syntax and semantics of its component logics.

In Section 7, we present our uniform method for constructing a tableau calculus for a fibred logic from calculi for the component logics. The resulting calculus is shown to be sound and complete w.r.t. the semantics of the fibred logic and to be itself suitable for fibring with other calculi. The latter property makes it possible to iterate the fibring of tableau calculi and, thus, to construct a calculus for the fully fibred logic $\mathbf{L}_{[1,2]}$.

As an example, in Section 8, the calculi for first-order and for modal logic introduced in Sections 4 and 5 are fibred resulting in a calculus for modal predicate logic.

Finally, in Section 9 we draw conclusions from our work. Due to space restrictions, all proofs are omitted; they can be found in [2].

2 Logical Systems

In this section, we define the notion of a *logical system* in a very general way; only indispensable properties of its syntax and semantics are part of the definition without which a useful tableau calculus for the logic cannot exist (or cannot be fibred with calculi for other logical systems).

The logic has to have a model semantics that uses Kripke-style models, i.e., models consisting of *worlds* in which formulae are true or false; there are no restrictions on the relationship between these worlds. In fact, any kind of model can be considered to be a Kripke-style model with a *single* world (namely the model itself), including models of classical propositional and first-order logic. However, since the labels of tableau formulae are interpreted as worlds, if there is only one world in the models of a logic, then the interpretation of all labels is the same and they become useless for the calculus.

The restriction that only two-valued logics are considered is solely made for the sake of simplicity. All notions introduced in the following can easily be extended to many-valued logics (but no additional insight is gained).

Definition 1. *Associated with a* logical system $\mathbf{L}$ *(a* logic *for short) is a set* Sig *of signatures*[1] *of* $\mathbf{L}$. *For each signature* $\Sigma \in Sig$*, syntax and semantics of the instance* $\mathbf{L}^\Sigma$ *of* $\mathbf{L}$ *are given by:*

Syntax: *A set* $Form^\Sigma$ *of* formulae *and a set* $Atom^\Sigma \subset Form^\Sigma$ *of* atomic formulae (atoms)*, where the sets* $Atom^\Sigma$ *and* $Form^\Sigma$ *are decidable.*

Semantics: *A set* $\mathcal{M}^\Sigma$ *of* models *where each model* $\mathbf{m} \in \mathcal{M}^\Sigma$ *(at least) contains (a) a set* W *of* worlds*, (b) an* initial world $w^0 \in W$*, and (c) a binary relation* $\models$ *between* W *and* $Form^\Sigma$*.*

If $w \models \phi$ *for some world* $w \in W$ *and some formula* $\phi \in Form^\Sigma$*, then* ϕ *is said to be* true *in* w*, else it is* false *in* w*. A formula* $\phi \in Form^\Sigma$ *is* satisfied *by a model* $\mathbf{m} \in \mathcal{M}^\Sigma$ *if (and only if) it is true in the initial world* w^0 *of* $\mathbf{m}$*. A set* $G \subset Form^\Sigma$ *of formulae is satisfied by* $\mathbf{m}$ *iff all its elements are satisfied by* $\mathbf{m}$*. A formula* $\phi \in Form^\Sigma$ *(a set* $G \subset Form^\Sigma$ *of formulae) is* satisfiable *if there is a model* $\mathbf{m} \in \mathcal{M}$ *satisfying* ϕ *(resp.* G*).*

Although usually non-atomic formulae are constructed from atomic formulae, and their truth value is determined by the truth value of the atoms they consist of, this is *not* part of the above definition. However, the existence of a tableau calculus for a logic $\mathbf{L}$ that is suitable for fibring implies that the truth value of a formula ϕ is strongly related to the truth values of certain atoms (that may or may not be sub-formulae of ϕ).

[1] We do not further specify what a signature is; *Sig* can be seen as a set of indices for distinguishing different instances of $\mathbf{L}$ (which *usually* differ in the symbols they use).

Tableau calculi allow to check the *satisfiability* of a formula; we only consider this property. It may or may not be possible in a certain logic to check whether a formula is valid in some model (true in all worlds) or is a tautology (valid in all models) by reducing this problem to a satisfiability problem; in many logics—though not in all—a formula is a tautology if its negation is not satisfiable.

Often, formulae are used in tableau calculi that are not part of the original but of an extended signature (e.g., formulae containing Skolem symbols):

Definition 2. *Given a logic* $\mathbf{L}$*, a signature* $\Sigma^* \in Sig$ *is an* extension *of a signature* $\Sigma \in Sig$ *(and* $\Sigma \in Sig$ *is a* restriction *of* $\Sigma^* \in Sig$*) if* $Form^{\Sigma} \subset Form^{\Sigma^*}$ *and* $Atom^{\Sigma} \subset Atom^{\Sigma^*}$*.*

In that case, a model $\mathbf{m} \in \mathcal{M}^{\Sigma}$ *is a* restriction *of a model* $\mathbf{m}^* \in \mathcal{M}^{\Sigma^*}$ *(to the signature* Σ*) if there is a function* f *that assigns to each world of* $\mathbf{m}$ *a world of* $\mathbf{m}^*$ *such that: (a) the initial world of* $\mathbf{m}^*$ *is assigned to the initial world of* $\mathbf{m}$*; and (b) for all formulae* $\phi \in Form^{\Sigma}$ *and worlds* w *of* $\mathbf{m}$*:* $w \models \phi$ *iff* $f(w) \models \phi$*.*

3 Tableau Calculi and the Conditions they Must Satisfy

As said above, only few restrictions are made regarding the type and form of tableau calculi. Any function that assigns to a tableau branch its (possible) extensions is regarded a tableau rule. Nevertheless, certain conditions have to be met, the first of which ensures that tableau rule applications do not transform the whole tableau in an arbitrary way:

Condition 1. Tableau rule applications have only *local* effects, in that they extend a single branch of a tableau, and do not alter or remove formulae already on the tableau.

The second assumption is that the applicability of a tableau rule to a branch and the result of its application are solely determined by the presence of certain formulae on the branch to which it is applied; no other pre-conditions are allowed such as, for example, the absence of certain formulae, the presence of formulae on different branches, or the order of formulae on a branch:

Condition 2. Whether a tableau branch B can be expanded in a certain way is solely determined by the presence of certain formulae on B (the *premiss* for that expansion).

Condition 2 implies that tableau branches are regarded as sets and that tableau rules are monotonic; thus, when formulae are added to the branch, previous tableau rule applications are not invalidated.

Conditions 1 and 2 intuitively prohibit "strange" behaviour of calculi. There are, however, useful calculi that violate these syntactical restrictions, including (a) calculi where variable substitutions are applied to the whole tableau, (b) calculi with resource restrictions that are not local to a branch (for example linear logic, where a formula can be "used up" globally), and (c) calculi using expansion rules that introduce *new* symbols, i.e., symbols that must not occur on the branch or even the whole tableau. At least the latter type of rules can often be replaced by similar rules satisfying Condition 2:

Example 3. In calculi for first-order predicate logic, often a tableau rule is used that allows to derive $\phi(c)$ from formulae of the form $(\exists x)(\phi(x))$, where c is a constant *new* to the tableau (or the branch); this rule violates Condition 2 because it demands the *absence* of formulae containing c.

If instead a special constant symbol c_ϕ is used, which does not have to be new, then the rule satisfies Condition 2 above. Soundness is preserved provided that c_ϕ is not introduced into the tableau in any other way than by skolemising $(\exists x)(\phi(x))$; in particular, the Skolem constant c_ϕ must not occur in the initial tableau (this is an adaptation of the rule for existential formulae presented in [3] to the ground case [1]).

As said before, we allow formulae from an extended signature Σ^* to be used in a tableau proof: Only the tableau formulae that are tested for satisfiability have to be taken from the non-extended signature Σ; they are put on the inital tableau. During the proof it is allowed, for example, to introduce Skolem symbols that are not elements of Σ. We proceed to formally define our (syntactical) notions of tableaux and tableau calculi:

Definition 4. *Given a logic* **L***, a signature* $\Sigma \in Sig$*, and a set* Lab *of labels, a* tableau formula $\sigma{:}\mathsf{S}\,\phi$ *consists of a label* $\sigma \in Lab$*, a truth value sign* $\mathsf{S} \in \{\mathsf{T}, \mathsf{F}\}$*, and a formula* $\phi \in Form^\Sigma$*; it is called* atomic *if* $\phi \in Atom^\Sigma$*. The set of all tableau formulae is denoted with* $TabForm^\Sigma$*. A* tableau *is a finitely branching tree whose nodes are labelled with tableau formulae. A* branch *of a tableau* T *is a maximal path in* T*. The set of formulae on a branch* B *is denoted with* $Form(B)$*.*

A tableau calculus $\mathcal{C}$ for a logic **L** has (different) "instances" $\mathcal{C}^\Sigma$ for each signature $\Sigma \in Sig$:

Definition 5. *A* tableau calculus $\mathcal{C}$ *for a logic* **L** *is, for each signature* $\Sigma \in Sig$*, specified by: (a) an extension* $\Sigma^* \in Sig$ *of the signature* Σ*; (b) a set* Lab *of labels and an initial label* $\sigma^0 \in Lab$*; (c) a tableau (expansion and closure) rule* $\mathcal{R}^\Sigma$*, i.e., a function that assigns to each finite set* $\Pi \subset TabForm^{\Sigma^*}$ *of tableau formulae (each premiss)—and thus to each tableau branch* B *with* $\Pi \subset Form(B)$*—a set* $\mathcal{R}^\Sigma(\Pi)$ *of (possible) conclusions, where a* conclusion *is a finite set of branch extensions or the symbol* $\bot$ *(branch closure), and a* branch extension *is a finite set of tableau formulae from* $TabForm^{\Sigma^*}$*. The rule* $\mathcal{R}^\Sigma$ *must satisfy the following conditions: (i)* $\mathcal{R}^\Sigma(\Pi)$ *may be infinite but has to be enumerable; (ii)* $\mathcal{R}^\Sigma(\Pi) \subset \mathcal{R}^\Sigma(\Pi \cup \Pi')$ *for all* $\Pi, \Pi' \subset TabForm^{\Sigma^*}$ *(monotonicity).*

In practice, tableau rules are often described by means of rule schemata. This fits perfectly in our framework, with the exception that different rule schemata are usually considered to define different rules, whereas we consider them to define different sub-cases of one (single) rule.

We now have everything at hand to define what the tableaux for a set G of formulae is and when a tableau is closed. The construction of tableaux for G is in general a non-deterministic process, since there may be any—even an infinite—number of possible conclusions that can be derived from a given premiss.

Definition 6. *Given a tableau calculus $\mathcal{C}$ for a logic* $\mathbf{L}$ *and a signature $\Sigma \in Sig$, the set of all tableaux for a finite set $\Gamma \subset TabForm^{\Sigma}$ of tableau formulae is inductively defined as follows: (1) A linear tableau whose nodes are labelled with the formulae in Γ is a tableau for Γ (an initial tableau). (2) Let T be a tableau for Γ, B a branch of T, and $C \neq \bot$ a conclusion in $\mathcal{R}^{\Sigma}(\Pi)$ for a premiss $\Pi \subset Form(B)$. Then a new tableau for Γ can be constructed from T as follows: the branch B is extended by a new sub-branch for each extension E in C, where the nodes in that sub-branch are labelled with the tableau formulae in E.*

T is a tableau for a finite set $G \subset Form^{\Sigma}$ of formulae if it is a tableau for the set $\{\sigma^0{:}\mathsf{T}\,\phi \mid \phi \in G\}$ of tableau formulae.

Definition 7. *Given a tableau calculus $\mathcal{C}$ for a logic* $\mathbf{L}$ *and a signature $\Sigma \in Sig$, a tableau branch B is* closed *iff $\bot \in \mathcal{R}^{\Sigma}(\Pi)$ for a premiss $\Pi \subset Form(B)$. A tableau is* closed *if all its branches are closed.*

Conditions 1 and 2 above, which are purely syntactical, still allow calculi to behave "strangely". Formulae could be added to the tableau that syntactically encode knowledge derived from a premiss Π, but whose semantics (i.e., truth value) has nothing to do with that of Π. An extreme example for this is that two symbols of the signature are used to encode the formulae in Π in a binary representation, and tableau rules are employed that operate on that binary representation. Such calculi—though they may be sound and complete—cannot be fibred in a uniform way as an understanding of the encoding would be needed. To assure a more "conservative" behaviour one could impose additional syntactical restrictions, for example only allow tableau rules that are analytic. However, the property of tableau rules that has to be guaranteed is more of a semantic nature: the result of a rule application must be semantically related to its premiss. The first semantical condition (Cond. 3) is part of our definition of the semantics of tableau formulae and tableaux (Def. 8):

Condition 3. The labels that are part of tableau formulae represent worlds in models, and the truth value signs encode truth and falsehood of a formula; they do not contain other information.

Definition 8. *Given a tableau calculus $\mathcal{C}$ for a logic* $\mathbf{L}$ *and a signature $\Sigma \in Sig$, a* tableau interpretation *for $\mathcal{C}^{\Sigma}$ is a pair $\langle \mathbf{m}, I \rangle$ where $\mathbf{m} \in \mathcal{M}^{\Sigma^*}$ is a model for the extended signature Σ^* and I is a partial function that assigns to labels $\sigma \in Lab^{\Sigma}$ worlds of $\mathbf{m}$ such that $I(\sigma^0) = w^0$ (i.e., I assigns to the initial label σ^0 the initial world w^0 of $\mathbf{m}$). A tableau interpretation $\langle \mathbf{m}, I \rangle$* satisfies *a tableau formula $\sigma{:}\mathsf{S}\,\phi \in Form^{\Sigma^*}$ iff $I(\sigma)$ is defined and (a) $\mathsf{S} = \mathsf{T}$ and ϕ is true in $I(\sigma)$ or (b) $\mathsf{S} = \mathsf{F}$ and ϕ is false in $I(\sigma)$. It* satisfies *a tableau branch B iff it satisfies* all *tableau formulae on B. It satisfies a tableau iff it satisfies* at least one *of its branches.*

Often, only a subset of all possible tableau interpretations is used to define the semantics of a tableaux. For example, to define the semantics of first-order tableaux, only tableau interpretations are used whose first part is an Herbrand model. In the following, the set of these tableau interpretations that are actually used to define the semantics of a calculus $\mathcal{C}^\Sigma$ is denoted with $\mathit{TabInterp}^\Sigma$.

The next four conditions we impose to make calculi "well-behaved", which are semantical, resemble the properties that a tableau calculus is shown to have in a classical soundness and completeness proof.

Condition 4. *Appropriateness of the set of tableau interpretations:* If a set $G \subset \mathit{Form}^\Sigma$ is satisfiable, then there is a tableau interpretation in $\mathit{TabInterp}^\Sigma$ that satisfies the initial tableau for G *(which is important for soundness)*; and, if $\langle \mathbf{m}^*, I \rangle$ is such a tableau interpretation, then $\mathbf{m}^*$ can be restricted to a model $\mathbf{m} \in \mathcal{M}^\Sigma$ that satisfies G *(which is important for completeness)*.

Condition 5. *Soundness of expansion (preliminary version):* If there is a tableau interpretation in $\mathit{TabInterp}^\Sigma$ satisfying a tableau T and T' is the result of applying the expansion rule to T then there is a tableau interpretation in $\mathit{TabInterp}^\Sigma$ satisfying T'.

Condition 6. *Soundness of Closure:* If a tableau branch is closed then it is *not* satisfied by any tableau interpretation in $\mathit{TabInterp}^\Sigma$.

Before Condition 7 can be formulated that establishes completeness of a calculus, the notion of a *fully expanded* tableau branch has to be defined. The definition relies on the fact that tableau rules are monotonic (Condition 2); without that property of tableau rules, it is difficult to define the notion of fully expanded branches in a uniform way. Intuitively a branch is fully expanded if no expansion rule application can add any new formulae to the branch.

Definition 9. *Given a tableau calculus $\mathcal{C}$ for a logic $\mathbf{L}$ and a signature $\Sigma \in \mathit{Sig}$, a tableau branch B is* fully expanded *if $E \subset \mathit{Form}(B)$ for all extensions E in all conclusions $C \in \mathcal{R}^\Sigma(\Pi)$ for all premisses $\Pi \subset \mathit{Form}(B)$.*

Condition 7. *Completeness:* If a tableau branch B is fully expanded and not closed then there is a tableau interpretation in $\mathit{TabInterp}^\Sigma$ satisfying B.

Conditions 4–7 ensure soundness and completeness of a tableau calculus:

Theorem 10. *If a tableau calculus $\mathcal{C}$ for a logic $\mathbf{L}$ satisfies Conditions 4–7 for all signatures $\Sigma \in \mathit{Sig}$ then the following holds for all finite sets $G \subset \mathit{Form}^\Sigma$: There is a closed tableau for G if and only if G is not satisfiable.*

To be suitable for fibring, a calculus has to satisfy two additional conditions. The first of these replaces Condition 5:

Condition 8. *Soundness of expansion:* If a tableau T is satisfied by a tableau interpretation in $\mathit{TabInterp}^\Sigma$ and T' is the result of applying the expansion rule to T, then T' is satisfied by *the same* tableau interpretation.

Intuitively, the reason why Condition 8 has to be used instead of Condition 5 is the following: Suppose T is a tableau for a fibred logic $\mathbf{L}_{(1,2)}$, the tableau interpretation $\langle \mathbf{m}_1, I_1 \rangle$ satisfies the $\mathbf{L}_1$-formulae on some branch B of T, the tableau interpretation $\langle \mathbf{m}_2, I_2 \rangle$ satisfies the $\mathbf{L}_2$-formulae on B, and together they form a tableau interpretation of the fibred logic $\mathbf{L}_{(1,2)}$ satisfying the whole branch B and, thus, the tableau T. Now, if the expansion rule for $\mathbf{L}_1$ only preserved satisfiability in *some* model, i.e., the $\mathbf{L}_1$-formulae on an extension B' of B were only satisfied by some different tableau interpretation $\langle \mathbf{m}'_1, I'_1 \rangle$, then a problem would arise if $\langle \mathbf{m}'_1, I'_1 \rangle$ and $\langle \mathbf{m}_2, I_2 \rangle$ are incompatible and do not form a fibred model.

Condition 9. If a tableau branch B is fully expanded then every tableau interpretation in $\mathit{TabInterp}^{\Sigma}$ satisfying the *atoms* on B satisfies *all* formulae on B.

This last condition ensures that the calculus is "analytical down to the atomic level". It is *not* a syntactical condition and it does *not* imply that the calculus is analytic in the classical sense. The condition is needed to ensure completeness when the calculus is used for fibring.

Example 11. In a tableau calculus for a modal logic that satisfies Condition 9, it must be possible to add the formula $\tau{:}\mathsf{T}\, p$ to a tableau branch containing $\sigma{:}\mathsf{T}\, \Box p$ for all labels τ representing a world reachable from the world represented by σ. In a tableau calculus for classical propositional logic it must be possible to expand a branch containing $\sigma{:}\mathsf{T}\, p \lor q$ by sub-branches containing $\sigma{:}\mathsf{T}\, p$ resp. $\sigma{:}\mathsf{T}\, q$, even if one of these atoms is *pure*, i.e., occurs only positively on the branch.

When the two calculi for propositional and for modal logic are fibred, then a propositional atom may indeed be a modal formula; even if it is pure (viewed as a propositional atom), it may be unsatisfiable as a modal formula. Thus, for example, a propositional calculus must expand the formula $\sigma{:}\mathsf{T}\, \Diamond(r \land \neg r) \lor q$ so that $\Diamond(r \land \neg r)$ can be passed on to the modal component of the fibred calculus, and its unsatisfiability can be detected.

Definition 12. *A tableau calculus $\mathcal{C}$ for a logic* $\mathbf{L}$ *is* suitable for fibring *if, for all signatures $\Sigma \in \mathit{Sig}$, there is a set $\mathit{TabInterp}^{\Sigma}$ of tableau interpretations such that Conditions 4–9 are satisfied (Condition 1–3 are part of the definition of tableau calculi resp. tableau interpretations).*

4 Example: First-order Predicate Logic

4.1 The Logical System of First-order Predicate Logic

To specify the logical system $\mathbf{L}_{\mathrm{PL1}}$ of first-order predicate logic, the set $\mathit{Sig}_{\mathrm{PL1}}$ of signatures and the syntax and semantics of $\mathbf{L}_{\mathrm{PL1}}$ have to be defined.

Signatures: The set $\mathit{Sig}_{\mathrm{PL1}}$ consists of all *first-order signatures* $\Sigma = \langle P_\Sigma, F_\Sigma \rangle$ where P_Σ is a set of predicate symbols and F_Σ is a set of function symbols. For skolemisation we do not use symbols from F_Σ but from a special infinite set F_Σ^{sko} of *Skolem function symbols* that is disjoint from F_Σ. The symbols in P_Σ, F_Σ and

F_{Σ}^{sko} may be used with any arity $n \geq 0$; in particular, function symbols can be used as constant symbols (arity 0).

Syntax: In addition to the predicate and function symbols there is an infinite set *Var* of *object variables.* The *logical operators* are $\vee$ (disjunction), $\wedge$ (conjunction), $\rightarrow$ (implication), and $\neg$ (negation), and the quantifiers $\forall$ and $\exists$. Terms, atoms, and formulae over a signature Σ are constructed as usual. As we use a calculus without free variables, $Form_{\mathrm{PL1}}^{\Sigma}$ is the set of all formulae over Σ not containing free variables, and $Atom_{\mathrm{PL1}}^{\Sigma} \subset Form_{\mathrm{PL1}}^{\Sigma}$ is the set of all ground atoms.

Semantics: A first-order *structure* $\langle D, \mathcal{I} \rangle$ for a signature Σ consists of a domain D and an interpretation $\mathcal{I}$, which gives meaning to the function and predicate symbols of Σ. A *variable assignment* is a mapping $\mu : Var \rightarrow D$ from the set of variables to the domain D. The *evaluation function val* is defined as usual; that is, given a structure $\langle D, \mathcal{I} \rangle$ and a variable assignment μ, it assigns to each formula $\phi \in Form^{\Sigma}$ a truth value $val_{\mathcal{I},\mu}(\phi) \in \{true, false\}$. As all models must contain a set of worlds (Def. 1), we define $\mathcal{M}_{\mathrm{PL1}}^{\Sigma}$ to consist of models where the initial and only world w^0 is a first-order structure. The relation $\models_{\mathrm{PL1}}$ is defined by: $w^0 \models_{\mathrm{PL1}} \phi$ iff, for all variable assignments μ, $val_{\mathcal{I},\mu}(\phi) = true$.

4.2 A Tableau Calculus for First-order Predicate Logic

To describe our calculus $\mathcal{C}_{\mathrm{PL1}}$ for first-order predicate logic $\mathbf{L}_{\mathrm{PL1}}$, we have to define, for each signature $\Sigma \in Sig_{\mathrm{PL1}}$, the extension Σ^* to be used for constructing tableaux, the set of labels, the initial label, and the expansion and closure rule.

Extended signature: Since the function symbols in F_{Σ}^{sko} are used for skolemisation, the extended signature Σ^* is $\langle P_{\Sigma}, F_{\Sigma} \cup F_{\Sigma}^{sko} \rangle$.

Labels: The models of first-order logic consist of only one world; it is represented by the label $*$. Thus, $Lab^{\Sigma} = \{*\}$, and $*$ is the initial label.

Expansion and closure rule: The set of tableau formulae in $TabForm^{\Sigma}$ that are not literals is divided into four classes as shown on the right: α for formulae of conjunctive type, β for formulae of disjunctive type, γ for quantified formulae of universal type, and δ for quantified formulae of existential type (unifying notation). To comply with Condition 1, which does not allow the application of substitutions (to the whole tableau), we use the classical *ground* version of tableaux for first-order logic (universally quantified variables are replaced by *ground* terms when the γ-rule is applied.) To comply with Condition 2, we use a δ-rule that does not introduce a *new* Skolem function symbol. Rather, each class of δ-formulae identical up to variable renaming is assigned its own unique Skolem symbol:

α	$\alpha_1, \quad \alpha_2$
$*{:}\mathsf{T}\,(\phi \wedge \psi)$	$*{:}\mathsf{T}\,\phi,\ *{:}\mathsf{T}\,\psi$
$*{:}\mathsf{F}\,(\phi \vee \psi)$	$*{:}\mathsf{F}\,\phi,\ *{:}\mathsf{F}\,\psi$
$*{:}\mathsf{F}\,(\phi \rightarrow \psi)$	$*{:}\mathsf{T}\,\phi,\ *{:}\mathsf{F}\,\psi$
$*{:}\mathsf{T}\,\neg\phi$	$*{:}\mathsf{F}\,\phi,\ *{:}\mathsf{F}\,\phi$
$*{:}\mathsf{F}\,\neg\phi$	$*{:}\mathsf{T}\,\phi,\ *{:}\mathsf{T}\,\phi$

β	$\beta_1, \quad \beta_n$
$*{:}\mathsf{T}\,(\phi \vee \psi)$	$*{:}\mathsf{T}\,\phi,\ *{:}\mathsf{T}\,\psi$
$*{:}\mathsf{F}\,(\phi \wedge \psi)$	$*{:}\mathsf{F}\,\phi,\ *{:}\mathsf{F}\,\psi$
$*{:}\mathsf{F}\,(\phi \rightarrow \psi)$	$*{:}\mathsf{F}\,\phi,\ *{:}\mathsf{T}\,\psi$

$\gamma(x)$	$\gamma_1(x)$
$*{:}\mathsf{T}\,(\forall x)(\phi(x))$	$*{:}\mathsf{T}\,\phi(x)$
$*{:}\mathsf{F}\,(\exists x)(\phi(x))$	$*{:}\mathsf{F}\,\phi(x)$

$\delta(x)$	$\delta_1(x)$
$*{:}\mathsf{F}\,(\forall x)(\phi(x))$	$*{:}\mathsf{F}\,\phi(x)$
$*{:}\mathsf{T}\,(\exists x)(\phi(x))$	$*{:}\mathsf{T}\,\phi(x)$

Definition 13. *Given a signature $\Sigma \in Sig_{\mathrm{PL1}}$, the function sko assigns to each δ-formula $\phi \in TabForm^{\Sigma^*}$ a symbol $sko(\phi) \in F^{sko}_{\Sigma}$ such that (a) $sko(\phi) > f$ for all $f \in F^{sko}_{\Sigma}$ occurring in ϕ, where $>$ is an arbitrary but fixed ordering on F^{sko}_{Σ}, and (b) for all δ-formulae $\phi, \phi' \in TabForm^{\Sigma^*}$ the symbols $sko(\phi)$ and $sko(\phi')$ are identical if and only if ϕ and ϕ' are identical up to renaming of quantified variables.*

The purpose of condition (a) in the above definition of *sko* is to avoid cycles like: $sko(\phi) = f$, f occurs in ϕ', $sko(\phi') = g$, and g occurs in ϕ.

The expansion and closure rule $\mathcal{R}_{\mathrm{PL1}}$ of our calculus $\mathcal{C}_{\mathrm{PL1}}$ is formally defined as follows: For all premisses $\Pi \subset TabForm^{\Sigma^*}_{\mathrm{PL1}}$, the set $\mathcal{R}^{\Sigma}_{\mathrm{PL1}}(\Pi)$ of possible conclusions is the smallest set containing the following conclusions (where $\alpha, \beta, \gamma, \delta$ denote tableau formulae of the corresponding type): (a) $\{\{\alpha_1, \alpha_2\}\}$ for all $\alpha \in \Pi$, (b) $\{\{\beta_1\}, \{\beta_2\}\}$ for all $\beta \in \Pi$, (c) $\{\{\gamma_1(t)\}\}$ for all $\gamma \in \Pi$ and all ground terms t over Σ^*, (d) $\{\{\delta_1(c)\}\}$ for all $\delta \in \Pi$ where $c = sko(\delta)$ (Def. 13), (e) $\perp$ if $*{:}\mathsf{T}\,\phi, *{:}\mathsf{F}\,\phi \in \Pi$ for any $\phi \in Form^{\Sigma^*}_{\mathrm{PL1}}$.

Semantics: We define the semantics of $\mathcal{C}_{\mathrm{PL1}}$-tableaux using tableau interpretations that are *canonical* in the following sense:

Definition 14. *A tableau interpretation for $\mathcal{C}_{\mathrm{PL1}}$ is* canonical *if its first-order structure $\langle D, \mathcal{I}\rangle$ satisfies the following conditions: (a) D is the set of all ground terms over Σ^*; (b) for all δ-formulae $\delta(x) \in TabForm^{\Sigma^*}$ and all variable assignments μ: if $val_{I,\mu}(\delta(x)) = true$ then $val_{I,\mu}(\delta_1(c)) = true$ where $c = sko(\delta)$.*

Using the set $TabInterp^{\Sigma}_{\mathrm{PL1}}$ of canonical tableau interpretations, the calculus $\mathcal{C}_{\mathrm{PL1}}$ satisfies Conditions 4–9. In particular, if a tableau T is satisfied by a canonical tableau interpretation, then all tableaux constructed from T are satisfied by the same interpretation; and every fully expanded tableau branch that is not closed is satisfied by a canonical interpretation.

Theorem 15. *The tableau calculus $\mathcal{C}_{\mathrm{PL1}}$ for $\mathbf{L}_{\mathrm{PL1}}$ is suitable for fibring.*

5 Example: The Logic $\mathbf{L}_{\mathbf{K}}$ of Modalities

5.1 The Logical System $\mathbf{L}_{\mathbf{K}}$

As a second example, we use the modal logic **K** without binary logical connectives; that is, all formulae are of the form $\circ_1 \cdots \circ_n\, p$ $(n \geq 0)$, where p is a propositional variable and $\circ_i$ is one of the modalities $\Box, \Diamond$ or the negation symbol $-$ (which is used to avoid confusion with first-oder negation $\neg$). We call this logic $\mathbf{L}_{\mathrm{K}}$. The missing connectives are not needed, since $\mathbf{L}_{\mathrm{K}}$ is later fibred with first-oder logic where they are available (Sect. 8).

Signatures: A signature Σ in Sig_{K} is an enumerable non-empty set of primitive propositions.

Syntax: The formulae in $Form^{\Sigma}_{\mathrm{K}}$ consist of a single element of Σ prefixed by a sequence of the logical operators $\Box, \Diamond, -$. The set $Atom^{\Sigma}_{\mathrm{K}}$ is identical to Σ.

Semantics: The semantics of $\mathbf{L}_{\mathrm{K}}$ is defined in the usual way using Kripke structures: A model $\mathbf{m}$ in $\mathcal{M}_{\mathrm{K}}^{\Sigma}$ consists of (a) a non-empty set W of worlds, one of which is the initial world w^0, (b) a binary reachability relation on W, and (c) a valuation V, which is a mapping from Σ to subsets of W. Thus, $V(p)$ is the set of worlds at which p is "true". For primitive propositions p, the relation $\models_{\mathrm{K}}$ is defined by: $w \models_{\mathrm{K}} p$ iff $w \in V(p)$; for complex formulae it is recursively defined by: (a) $w \models_{\mathrm{K}} -\phi$ iff not $w \models_{\mathrm{K}} \phi$, (b) $w \models_{\mathrm{K}} \Box\phi$ iff $w' \models_{\mathrm{K}} \phi$ for *all* w' reachable from w, and (c) $w \models_{\mathrm{K}} \Diamond\phi$ iff $w' \models_{\mathrm{K}} \phi$ for *some* w' reachable from w.

5.2 A Tableau Calculus for the Logic $\mathbf{L}_{\mathrm{K}}$

We define a calculus $\mathcal{C}_{\mathrm{K}}$ for $\mathbf{L}_{\mathrm{K}}$ that uses sequences of natural numbers as labels; the world named by $\sigma.n$ is reachable from the world named by σ.

Extended signature: No extension of the signature is needed, thus $\Sigma = \Sigma^*$.

Labels: The set Lab_{K} of labels is for all Σ inductively defined by: the initial label 1 is a label, and if σ is a label then so is $\sigma.n$ for all natural numbers n.

Expansion and closure rule: To comply with Condition 2, we use a π-rule that does not introduce a *new* label but—similar to the δ-rule in Section 4.2—uses a label that is uniquely assigned to the formula to which the rule is applied.

The expansion and closure rule of our calculus $\mathcal{R}_{\mathrm{K}}^{\Sigma}$ for the logic $\mathbf{L}_{\mathrm{K}}$ is formally defined as follows: For all premisses $\Pi \subset TabForm_{\mathrm{K}}^{\Sigma}$, the set $\mathcal{R}_{\mathrm{K}}(\Pi)$ of possible conclusions is the smallest set containing the following conclusions (where *goedel* is any bijection from $Form_{\mathrm{K}}^{\Sigma}$ to the set of natural numbers): (a) $\{\{\sigma.n{:}\mathsf{T}\,\phi\}\}$ for all $\sigma{:}\mathsf{T}\,\Box\phi \in \Pi$ and all labels of the form $\sigma.n$ occurring in Π, (b) $\{\{\sigma.n{:}\mathsf{F}\,\phi\}\}$ for all $\sigma{:}\mathsf{F}\,\Diamond\phi \in \Pi$ and all labels of the form $\sigma.n$ occurring in Π, (c) $\{\{\sigma.n{:}\mathsf{F}\,\phi\}\}$ for all $\sigma{:}\mathsf{F}\,\Box\phi \in \Pi$ where $n = goedel(\phi)$, (d) $\{\{\sigma.n{:}\mathsf{T}\,\phi\}\}$ for all $\sigma{:}\mathsf{T}\,\Diamond\phi \in \Pi$ where $n = goedel(\phi)$, (e) $\{\{\sigma{:}\mathsf{F}\,\phi\}\}$ for all $\sigma{:}\mathsf{T}\,-\phi \in \Pi$, (f) $\{\{\sigma{:}\mathsf{T}\,\phi\}\}$ for all $\sigma{:}\mathsf{F}\,-\phi \in \Pi$, (g) $\perp$ if $\sigma{:}\mathsf{T}\,\phi, \sigma{:}\mathsf{F}\,\phi \in \Pi$ for any $\phi \in Form_{\mathrm{K}}^{\Sigma}$.

Semantics: The set $TabInterp_{\mathrm{K}}^{\Sigma}$ contains *canonical* tableau interpretation satisfying the following condition:

Definition 16. *A tableau interpretation $\langle \mathbf{m}, I\rangle$ for $\mathbf{L}_{\mathrm{K}}$ is* canonical *if: (a) if $I(\sigma)$ is defined and satisfies $\sigma{:}\mathsf{T}\,\Diamond\phi$, then $I(\sigma.n)$ is defined and satisfies $\sigma{:}\mathsf{T}\,\phi$ where $n = goedel(\phi)$; and (b) for all numers n, if $w = I(\sigma)$ and w'=$I(\sigma.n)$ are defined, then the world w' is reachable from w.*

Theorem 17. *The tableau calculus $\mathcal{C}_{\mathrm{K}}$ for $\mathbf{L}_{\mathrm{K}}$ is suitable for fibring.*

6 Fibring Logical Systems

To fibre two logics $\mathbf{L}_1$ and $\mathbf{L}_2$ means to consider a logic whose formulae are constructed from symbols and operators from both logics [7, 8]. In a first step we consider a logic $\mathbf{L}_{(1,2)}$ where $\mathbf{L}_2$-formulae can occur inside $\mathbf{L}_1$-formulae but not vice versa.

Example 18. If $\mathbf{L}_1 = \mathbf{L}_{\mathrm{PL1}}$ and $\mathbf{L}_2 = \mathbf{L}_{\mathrm{K}}$, then $(\forall x)(p(x))$, $\Box q$, $(\forall x)(\Box p(x))$, and $(\forall x)(\Box p(x)) \to (\exists x)(\Diamond q(x))$ are formulae of $\mathbf{L}_{(1,2)}$, but $\Box(\forall x)(p(x))$ is not.

The logic $\mathbf{L}_{[1,2]} \equiv \mathbf{L}_{[2,1]}$ that is the full combination of $\mathbf{L}_1$ and $\mathbf{L}_2$, where expressions from the two logics can be nested arbitrarily, can be handled by inductively repeating the construction presented in this section. Similarly, it is possible to combine three or more logics.

We consider $\mathbf{L}_{(1,2)}$ to be a special case of $\mathbf{L}_1$: it contains the formulae of $\mathbf{L}_2$ as (additional) atoms. And, in each world w of an $\mathbf{L}_1$-model, the truth value of the additional atoms, which are $\mathbf{L}_2$-formulae, is the same as that in the initial world of an $\mathbf{L}_2$-model assigned to w. Thus, an $\mathbf{L}_{(1,2)}$-model consists of an $\mathbf{L}_1$-model $\mathbf{m}_1$ and a *fibring function* F that assigns to each world w in $\mathbf{m}_1$ an $\mathbf{L}_2$-model. Intuitively, when an $\mathbf{L}_2$-formula is to be evaluated in w, where its value is undefined, it is evaluated in $\mathbf{m}_2 = F(w)$ instead. In most cases, certain restrictions have to be imposed on F to make sure that the fibred models define the desired semantics. These restrictions are given in form of a relation $\mathcal{P}$ between $\mathbf{L}_1$-models, $\mathbf{L}_1$-worlds, and $\mathbf{L}_2$-models; a fibring function can be used for an $\mathbf{L}_1$-model $\mathbf{m}_1$ if $\mathcal{P}(\mathbf{m}_1, w, F_{(1,2)}(w))$ holds for all worlds w of $\mathbf{m}_1$.

Example 19. A proposition may be represented by different atoms p_1 and p_2 in $\mathbf{L}_1$ and $\mathbf{L}_2$. Then, for the semantics defined by the fibred models to be useful, one imposes the restriction that, if p_1 is true in a world w of $\mathbf{m}_1$ then p_2 is true in the initial world of $F(w)$.

Definition 20. *Logics* $\mathbf{L}_1, \mathbf{L}_2$ *are* suitable for fibring *iff, for all* $\Sigma_1 \in Sig_1$ *and* $\Sigma_2 \in Sig_2$*, there is a signature* $\Sigma_{(1,2)} \in Sig_1$ *such that* $Form_2^{\Sigma_2} \subset Atom_1^{\Sigma_{(1,2)}}$.

Let $\mathcal{P}$ *be a* restricting relation *between* $\mathbf{L}_1$*-models,* $\mathbf{L}_1$*-worlds, and* $\mathbf{L}_2$*-models. Then, the fibred logic* $\mathbf{L}_{(1,2)}$ *is given by:*

Signatures: $Sig_{(1,2)} = \{\Sigma_{(1,2)} \mid \Sigma_1 \in Sig_1, \Sigma_2 \in Sig_2\}$.

Syntax: *For all* $\Sigma_{(1,2)} \in Sig_{(1,2)}$, $Form_{(1,2)}^{\Sigma_{(1,2)}}$ *is identical to* $Form_1^{\Sigma_{(1,2)}}$ *and* $Atom_{(1,2)}^{\Sigma_{(1,2)}}$ *is identical to* $Atom_1^{\Sigma_{(1,2)}}$.

Semantics: *A model* $\mathbf{m}_{(1,2)} \in \mathcal{M}_{(1,2)}^{\Sigma_{(1,2)}}$ *consists of an* $\mathbf{L}_1$*-model* $\mathbf{m}_1 \in \mathcal{M}_1^{\Sigma_{(1,2)}}$ *and a fibring function* F *that assigns to each world* w *in* $\mathbf{m}_1$ *an* $\mathbf{L}_2$*-model* $\mathbf{m}_2$ *in* $\mathcal{M}_2^{\Sigma_2}$ *such that (a)* $\mathcal{P}(\mathbf{m}_1, w, \mathbf{m}_2)$*, and (b)* $w \models_1 \phi$ *iff* $F(w) \models_2 \phi$ *for all* $\phi \in Form_2^{\Sigma_2}$. *We define* $\models_{(1,2)} = \models_1$, $W_{(1,2)} = W_1$*, and* $w^0_{(1,2)} = w^0_1$.

Example 21. To fibre $\mathbf{L}_{\mathrm{PL1}}$ and $\mathbf{L}_{\mathrm{K}}$, we assume that there is an $\mathbf{L}_{\mathrm{K}}$-signature Σ_{K} for every $\mathbf{L}_{\mathrm{PL1}}$-signature Σ_{PL1} such that the atoms over Σ_{PL1} are the primitive propositions in Σ_{K}. Then, $\Sigma_{(\mathrm{PL1,K})}$ is an $\mathbf{L}_{\mathrm{PL1}}$-signature such that the predicate symbols are of the form $\circ_1 \cdots \circ_n\, p$ $(n \geq 0)$ where $\circ_i \in \{\Box, \Diamond, -\}$ and p is a predicate symbol in Σ_{PL1}.

The fibred logic $\mathbf{L}_{(\mathrm{PL1,K})}$ is a first-order modal logic, where the modal operators can only occur on the atomic level. If, however, the fibring process is iterated, then the result is full modal predicate logic, because then the logical connectives $\vee, \wedge, \neg$ of $\mathbf{L}_{\mathrm{PL1}}$ can be used inside modal formulae.

7 Fibring Tableau Calculi

In this section, we describe how to construct—in a uniform way—a calculus for a fibred logic $\mathbf{L}_{(1,2)}$ from two calculi $\mathcal{C}_1$ and $\mathcal{C}_2$ for $\mathbf{L}_1$ and $\mathbf{L}_2$.

Expanding a tableau can be seen as an attempt to construct a model for the formula in the root node. If the tableau is closed, then there is no model and the formula in the root node is unsatisfiable. A tableau formula $\sigma{:}\mathsf{T}\,\phi$ represents the fact that, in the constructed model, ϕ is true in the world corresponding to σ.

Now, we have to construct a fibred model and, thus, to represent knowledge about a fibred model by tableau formulae. Therefore, labels now are either of the form $\sigma_1 \in Lab_1$ denoting a world in the $\mathbf{L}_1$-model or of the form $(\sigma_1;\sigma_2)$ (where $\sigma_1 \in Lab_1$ and $\sigma_2 \in Lab_2$) denoting a world in the $\mathbf{L}_2$-model that is assigned by the fibring function to the world represented by σ_1 in the $\mathbf{L}_1$-model. A tableau formula $\sigma_1{:}\mathsf{T}\,\phi$ still means that ϕ is true in $I_1(\sigma_1)$; a tableau formula $(\sigma_1;\sigma_2){:}\mathsf{T}\,\phi$ means that ϕ is true in the world $I_2(\sigma_2)$ of the model assigned to $I_1(\sigma_1)$.

The combined calculus does not construct separate tableaux for $\mathbf{L}_1$- and $\mathbf{L}_2$-formulae but a single tableau, using a unified (set of) tableau rule(s).

The only additional assumption we have to make is that the extension of the restricting relation $\mathcal{P}$ (Def. 20) to tableau interpretations can be characterised using *finite* sets of tableau formulae:

Definition 22. *Let $\mathbf{L}_1$ and $\mathbf{L}_2$ be logics suitable for fibring, let $\mathcal{C}_1$ and $\mathcal{C}_2$ be calculi for $\mathbf{L}_1, \mathbf{L}_2$, let $\mathcal{P}$ be a restricting relation (Def. 20), and let $\Sigma_1 \in Sig_1$ and $\Sigma_2 \in Sig_2$. A function $\mathcal{P}^T$ that assigns to a* finite *subset Π of $TabForm_1^{\Sigma_1^*}$ and a label $\sigma_1 \in Lab_1$ a* finite *set $\mathcal{P}^T(\Pi,\sigma_1)$ of $\mathbf{L}_2$-tableau formulae over the non-extended signature Σ_2* characterises *$\mathcal{P}$ if the following holds for all finite or infinite sets $\tilde{\Pi} \subset TabForm_1^{\Sigma_1^*}$, all labels $\sigma_1 \in Lab_1$, and all tableau interpretations $\langle \mathbf{m}_1, I_1\rangle \in TabInterp_1^{\Sigma_1}$ and $\langle \mathbf{m}_2, I_2\rangle \in TabInterp_2^{\Sigma_2}$:*

$\mathcal{P}(\mathbf{m}_1, I_1(\sigma_1), \mathbf{m}_2)$ holds if and only if (a) $I_1(\sigma_1)$ is defined, (b) $I_1(\sigma_1) \models_1 \tilde{\Pi}$, and (c) $\langle \mathbf{m}_1, I_1\rangle$ satisfies $\mathcal{P}^T(\Pi,\sigma_1)$ for all finite subsets Π of $\tilde{\Pi}$.

Of course, the fibred calculus can only be implemented if the function $\mathcal{P}^T$ is computable; for a semi-decision procedure, it is sufficient if $\mathcal{P}^T(\Pi,\sigma_1)$ is enumerable for all Π and σ_1.

Example 23. The following function can be used to characterise the (simple) restriction from Example 19: $\mathcal{P}^T(\Pi,\sigma_1) = \{\sigma_2^0{:}\mathsf{S}\,p_2 \mid \sigma_1{:}\mathsf{S}\,p_1 \in \Pi\}$ where σ_2^0 is the initial label of $\mathcal{C}_2$.

The expansion and closure rule of the fibred calculus $\mathcal{C}_{(1,2)}$ constructed from $\mathcal{C}_1$ and $\mathcal{C}_2$ has four components: (1) the expansion rule of $\mathcal{C}_1$, which can be applied to $\mathbf{L}_1$-tableau formulae; (2) the expansion rule of $\mathcal{C}_2$, which can be applied to $\mathbf{L}_2$-tableau formulae with a label of the form $(\sigma_1;\sigma_2)$; (3) a transition rule that allows to derive $(\sigma_1;\sigma_2^0){:}\mathsf{S}\,\phi_2$ from $\sigma_1{:}\mathsf{S}\,\phi_2$ if ϕ_2 is an $\mathbf{L}_2$-formula (in that case ϕ_2 has to be expanded by the $\mathcal{C}_2$-rule), i.e., if an $\mathbf{L}_2$-formula ϕ_2 is true in an $\mathbf{L}_1$-world $w = I_1(\sigma_1)$ then it is true in the initial world of the $\mathbf{L}_2$-model assigned to w; (4) a rule implementing the restriction relation, i.e., if the formulae in Π occur on a branch and $\sigma_2{:}\mathsf{S}\,\phi_2 \in \mathcal{P}^T(\Pi,\sigma_1)$ then $(\sigma_1;\sigma_2){:}\mathsf{S}\,\phi_2$ may be added.

Definition 24. *Let $\mathbf{L}_1, \mathbf{L}_2$ be logics suitable for fibring; let $\mathcal{C}_1, \mathcal{C}_2$ be calculi for logics $\mathbf{L}_1, \mathbf{L}_2$, and let these calculi be suitable for fibring; let $\mathcal{P}$ be a restricting relation characterised by the function $\mathcal{P}^T$ (Def. 22). Then, the fibred calculus $\mathcal{C}_{(1,2)}$ is, for all $\Sigma_1 \in Sig_1, \Sigma_2 \in Sig_2$, defined by:*

Extended Signature: *The extension of $\Sigma_{(1,2)}$ is the signature $\Sigma^*_{(1,2)}$ that is associated with Σ^*_1 and Σ^*_2 according to Definition 20.*

Labels: $Lab_{(1,2)}^{\Sigma_{(1,2)}} = Lab_1 \cup \{(\sigma_1;\sigma_2) \mid \sigma_1 \in Lab_1^{\Sigma_1}, \sigma_2 \in Lab_2^{\Sigma_2}\}$*; the initial label $\sigma^0_{(1,2)}$ is the initial label σ^0_1 if $\mathcal{C}_1$.*

Expansion and closure rule: *For all premisses $\Pi \subset TabForm_{(1,2)}^{\Sigma^*_{(1,2)}}$, the set $\mathcal{R}_{(1,2)}(\Pi)$ is the smallest set containing:*

1. *the conclusions in $\mathcal{R}_1(\Pi_1)$ where Π_1 consists of all tableau formulae of the form $\sigma_1{:}\mathsf{S}\,\phi$ in Π such that $\phi \in Form_1^{\Sigma^*_1}$* (expansion rule of $\mathcal{C}_1$),
2. *for all $\sigma_1 \in Lab_1^{\Sigma_{(1,2)}}$, the conclusions that can be constructed from the conclusions in $\mathcal{R}_2(\Pi_{2,\sigma_1})$ replacing σ_2 by $(\sigma_1;\sigma_2)$; the set Π_{2,σ_1} consists of all tableau formulae of the form $\sigma_2{:}\mathsf{S}\,\phi$ such that $(\sigma_1;\sigma_2){:}\mathsf{S}\,\phi$ is in Π and $\phi \in Form_2^{\Sigma^*_2}$* (expansion rule of $\mathcal{C}_2$),
3. *the conclusion $\{\{(\sigma_1;\sigma^0_2){:}\mathsf{S}\,\phi\}\}$ for all tableau formulae of the form $\sigma_1{:}\mathsf{S}\,\phi$ in Π such that $\phi \in Form_2^{\Sigma^*_2}$* (transition rule),
4. *for all $\sigma_1 \in Lab_1^{\Sigma_{(1,2)}}$ and all subsets Π_1 of Π (see point 1 above), the conclusion $\{\mathcal{P}^T(\Pi_1,\sigma_1)\}$* (restriction relation).

Theorem 25. *The fibred calculus $\mathcal{C}_{(1,2)}$ that is constructed according to Definition 24 is suitable for fibring, i.e., it satisfies Conditions 4–9 in Section 3.*

Corollary 26. *The fibred calculus $\mathcal{C}_{(1,2)}$ that is constructed according to Definition 24 is a sound and complete calculus for $\mathbf{L}_{(1,2)}$, i.e., there is a closed tableau for $G \in Form^{\Sigma_{(1,2)}}$ if and only if G is not satisfiable.*

8 Fibring Calculi for Predicate and Modal Logic

As an example, we fibre the calculi $\mathcal{C}_{\mathrm{PL1}}$ for first-order predicate logic $\mathbf{L}_{\mathrm{PL1}}$ introduced in Section 4.2 and the calculus $\mathcal{C}_{\mathrm{K}}$ for the logic $\mathbf{L}_{\mathrm{K}}$ of modalities defined in Section 5.2. The result is a calculus $\mathcal{C}_{(1,2)}$ for first-order modal logic where the modal operators can only occur on the literal level (Example 21). Since, in this case, there is no additional restriction on which $\mathbf{L}_{\mathrm{K}}$-models may be assigned to worlds in $\mathbf{L}_{\mathrm{PL1}}$-models, the function $\mathcal{P}^T(\Pi,\sigma)$ characterising the fibring restriction (Def. 22) is empty for all formula sets Π and labels σ; therefore, the tableau expansion rule that implements the restriction relation is never applied.

Due to space restrictions, we cannot list the tableau expansion and closure rules of the fibred calculus, which can easily be constructed by instantiating the calculi $\mathcal{C}_1$ and $\mathcal{C}_2$ in Definition 24 with $\mathcal{C}_{\mathrm{PL1}}$ resp. $\mathcal{C}_{\mathrm{K}}$. Instead, we prove the formula

$$G = (\forall x)(\Box p(x)) \to [\neg(\exists y)(\Diamond{-}p(y)) \wedge \neg(\exists z)(\Diamond{-}p(z))]$$

to be valid in all models of the logic $\mathbf{L}_{(1,2)} = \mathbf{L}_{(\mathrm{PL1,K})}$, using the fibred calculus $\mathcal{C}_{(1,2)} = \mathcal{C}_{(\mathrm{PL1,K})}$ to construct a closed tableau for $\neg G$.

1 $*{:}\mathsf{T}\ \neg((\forall x)(\Box p(x)) \rightarrow [\neg(\exists y)(\Diamond{-p(y)}) \wedge \neg(\exists z)(\Diamond{-p(z)})])$	
2 $*{:}\mathsf{F}\ (\forall x)(\Box p(x)) \rightarrow [\neg(\exists y)(\Diamond{-p(y)}) \wedge \neg(\exists z)(\Diamond{-p(z)})]$	
3 $*{:}\mathsf{T}\ (\forall x)(\Box p(x))$	
4 $*{:}\mathsf{F}\ \neg(\exists y)(\Diamond{-p(y)}) \wedge \neg(\exists z)(\Diamond{-p(z)})$	
5 $*{:}\mathsf{F}\ \neg(\exists y)(\Diamond{-p(y)})$	**6** $*{:}\mathsf{F}\ \neg(\exists y)(\Diamond{-p(y)})$
7 $*{:}\mathsf{T}\ (\exists y)(\Diamond{-p(y)})$	**15** $*{:}\mathsf{T}\ (\exists y)(\Diamond{-p(y)})$
8 $*{:}\mathsf{T}\ \Diamond{-p(c_1)}$	**16** $*{:}\mathsf{T}\ \Diamond{-p(c_1)}$
9 $(*;1){:}\mathsf{T}\ \Diamond{-p(c_1)}$	**17** $(*;1){:}\mathsf{T}\ \Diamond{-p(c_1)}$
10 $(*;1.1){:}\mathsf{T}\ {-p(c_1)}$	**18** $(*;1.1){:}\mathsf{T}\ {-p(c_1)}$
11 $(*;1.1){:}\mathsf{F}\ p(c_1)$	**19** $(*;1.1){:}\mathsf{F}\ p(c_1)$
12 $*{:}\mathsf{T}\ \Box p(c_1)$	**20** $*{:}\mathsf{T}\ \Box p(c_1)$
13 $(*;1){:}\mathsf{T}\ \Box p(c_1)$	**21** $(*;1){:}\mathsf{T}\ \Box p(c_1)$
14 $(*;1.1){:}\mathsf{T}\ p(c_1)$	**22** $(*;1.1){:}\mathsf{T}\ p(c_1)$
—	—

The closed tableau shown on the right is constructed as follows: Tableau formula 1 is put on the tableau initially; then formulae 2–7 are added using the α- and β-rules of $\mathcal{C}_{\mathrm{PL1}}$. The δ-rule of $\mathcal{C}_{\mathrm{PL1}}$ is applied to derive 8 from 7, using the Skolem constant $c_1 = \mathit{sko}((\exists y)(\Diamond{-p(y)}))$. Since 8 is an $\mathbf{L}_{\mathrm{K}}$-formula, the transition rule is applied to add 9 to the branch, which then allows to apply the $\mathbf{L}_{\mathrm{K}}$-expansion rule to derive 10 from 9 (we assume that $\mathit{goedel}(\Diamond{-p(c_1)}) = 1$) and to derive 11 from 10. At this point, the γ-rule of $\mathbf{L}_{\mathrm{PL1}}$ is applied to 3 to derive 12, replacing the universally quantified variable x with the ground term c_1 (which shows that $\mathbf{L}_1$- and $\mathbf{L}_2$-rules can be applied in an arbitrary order). Finally, the transition rule is applied to 12 to derive 13, and the $\mathbf{L}_{\mathrm{K}}$-rule for $\Box$-formulae is applied to derive 14. At this point, the left branch of the tableau is closed by the $\mathbf{L}_{\mathrm{K}}$-closure rule, because it contains the complementary atoms 11 and 14. The right branch is expanded and closed in the same way.

10′ $(*;1.1){:}\mathsf{T}\ {-r(c_1)} \vee {-s(c_1)}$	
14′ $(*;1.1){:}\mathsf{T}\ r(c_1) \wedge s(c_1)$	
23 $(*;1.1;*){:}\mathsf{T}\ r(c_1) \wedge s(c_1)$	
24 $(*;1.1;*){:}\mathsf{T}\ r(c_1)$	
25 $(*;1.1;*){:}\mathsf{T}\ s(c_1)$	
26 $(*;1.1;*){:}\mathsf{T}\ {-r(c_1)} \vee {-s(c_1)}$	
27 $(*;1.1;*){:}\mathsf{T}\ {-r(c_1)}$	**31** $(*;1.1;*){:}\mathsf{T}\ {-s(c_1)}$
28 $(*;1.1;*;1){:}\mathsf{T}\ {-r(c_1)}$	**32** $(*;1.1;*;1){:}\mathsf{T}\ {-s(c_1)}$
29 $(*;1.1;*;1){:}\mathsf{F}\ r(c_1)$	**33** $(*;1.1;*;1){:}\mathsf{F}\ s(c_1)$
30 $(*;1.1;*;1){:}\mathsf{T}\ r(c_1)$	**34** $(*;1.1;*;1){:}\mathsf{T}\ s(c_1)$
—	—

The full power of the fibring method is revealed when the fibring process is iterated to construct a calculus $\mathcal{C}_{[\mathrm{PL1,K}]}$ for the full modal predicate logic $\mathbf{L}_{[\mathrm{PL1,K}]}$; this is possible because the calculi $\mathcal{C}_{(1,2)}, \mathcal{C}_{(1,(2,1))}, \ldots$ are all suitable for fibring. As an example, we use $\mathcal{C}_{[\mathrm{PL1,K}]}$ to prove that the formula is valid in all models of $\mathbf{L}_{[\mathrm{PL1,K}]}$ that is constructed from G replacing the literal $p(x)$ by $r(x) \wedge s(x)$ and replacing the literals $-p(y)$ and $-p(z)$ by $-r(y) \vee -s(y)$ resp. $-r(z) \vee -s(z)$. The construction of the tableau starts as above for G. We only consider the left branch (the right branch can be closed in the same way). Instead of the literals 10 and 14, the branch now contains $10' = (*;1.1){:}\mathsf{T}\ {-r(c_1)} \vee {-s(c_1)}$ and $14' = (*;1.1){:}\mathsf{T}\ r(c_1) \wedge s(c_1)$. The expansion of the branch continues as shown above (to simplify notation, we write $(*;1;*)$ instead of $(*;(1;*))$, etc.). The tableau formula 14′ contains an $\mathbf{L}_{\mathrm{PL1}}$-formula. Therefore, the transition rule is applied, and 23 is derived from 14′; this is the transition rule of the calculus $\mathcal{C}_{(\mathrm{K,PL1})}$ that, during the iteration process, has been fibred with $\mathcal{C}_{\mathrm{PL1}}$ to construct $\mathcal{C}_{(\mathrm{PL1,(K,PL1)})}$. The α-rule of $\mathbf{L}_{\mathrm{PL1}}$ is used to derive 24 and 25 from 23; then, 26 is derived from 10′ by again applying the transition rule, and the β-rule is applied to derive 27 and 31 from 26. The literal $-p(c_1)$ in 27 contains the modal and not the first-oder negation sign. Thus, the transition rule has to be applied again to derive 28, which then allows to derive 29 by applying the rule for modal negation. The atomic tableau formulae 24 and 29 cannot be used to close the branch, because their labels are different. Thus, the transition rule is applied a last time to derive 30 from 24. Then, the branch is closed by 29 and 30.

9 Conclusion

We have presented a uniform method for constructing a sound and complete tableau calculus for a fibred logic from calculi for its component logics. Conditions have been identified that tableau calculi have to satisfy to be suitable for fibring; the conditions are neither too weak nor too strong. Since tableau calculi are already known for most "basic" logics, it is possible to construct calculi for all "complex" logics that can be constructed by fibring basic logics. The main advantages of a uniform framework for fibring calculi are:

To construct a calculus for the combination $\mathbf{L}_{[1,2]}$ of two particular logics, no knowledge is needed about the interaction between calculi for $\mathbf{L}_1$ and $\mathbf{L}_2$. Thus, a calculus for the combination $\mathbf{L}_{[1,2]}$ can be obtained quickly and easily.

Soundness and completeness of the fibred calculus does not have to be proven; it follows from Theorem 25 if the fibred calculi are suitable for fibring.

A calculus $\mathcal{C}_1$ for a logic $\mathbf{L}_1$ can be fibred with a calculus $\mathcal{C}_2$ for a "sub-logic" $\mathbf{L}_2$ of $\mathbf{L}_1$ (for example, propositional logic is a sub-logic if predicate logic); although $\mathcal{C}_1$ can handle the whole logic $\mathbf{L}_1$, the calculus $\mathcal{C}_2$ may be more efficient for formulae from $\mathbf{L}_2$ such that the fibred calculus $\mathcal{C}_{(1,2)}$ is more efficient than $\mathcal{C}_1$. This can be seen as a generalisation of the theory reasoning method.

Acknowledgement. We thank Guido Governatori and two anonymous referees for useful comments on an earlier version of this paper.

References

1. W. Ahrendt and B. Beckert. An improved δ-rule for ground first-order tableaux. Unpublished draft available from the authors, 1997.
2. B. Beckert and D. Gabbay. A general framework for fibring semantic tableaux. Unpublished draft available from the authors, 1997.
3. B. Beckert, R. Hähnle, and P. H. Schmitt. The even more liberalized δ-rule in free variable semantic tableaux. In *Proceedings of KGC*, LNCS 713. Springer, 1993.
4. M. D'Agostino and D. Gabbay. Fibred tableaux for multi-implication logics. In *Proceedings of TABLEAUX*, LNCS 1071. Springer, 1996.
5. M. D'Agostino, D. Gabbay, R. Hähnle, and J. Posegga, editors. *Handbook of Tableau Methods*. Kluwer, Dordrecht, 1998. To appear.
6. D. Gabbay. Fibred semantics and the weaving of logics. Part 1: Modal and intuitionistic logics. *Journal of Symbolic Logic*, 61:1057–1120, 1996.
7. D. Gabbay. An overview of fibred semantics and the combination of logics. In *Proceedings of FroCoS*. Kluwer, Dordrecht, 1996.
8. D. Gabbay. *Fibring Logic*. Oxford University Press, 1998. Forthcoming.
9. D. Gabbay and G. Governatori. Fibred modal tableaux. Draft, 1997.