A Matrix Characterization for $\mathcal{MELL}$

Heiko Mantel[1] Christoph Kreitz[2]

[1] *German Research Center for Artificial Intelligence (DFKI)*
Stuhlsatzenhausweg 3, 66123 Saarbrücken, Germany
mantel@dfki.de

[2] *Department of Computer Science*
Cornell-University, Ithaca, NY 14853-7501
kreitz@cs.cornell.edu

Abstract. We present a matrix characterization of logical validity in the multiplicative fragment of linear logic with exponentials. In the process we elaborate a methodology for proving matrix characterizations correct and complete. Our characterization provides a foundation for matrix-based proof search procedures for $\mathcal{MELL}$ as well as for procedures which translate machine-found proofs back into the usual sequent calculus.

1 Introduction

Linear logic [12] has become known as a very expressive formalism for reasoning about action and change. During its rather rapid development linear logic has found applications in logic programming [14, 19], modeling concurrent computation [11], planning [18], and other areas. Its expressiveness, however, results in a high complexity. Propositional linear logic is undecidable. The multiplicative fragment ($\mathcal{MLL}$) is already $\mathcal{NP}$-complete [16]. The complexity of the multiplicative exponential fragment ($\mathcal{MELL}$) is still unknown. Consequently, proof search in linear logic is difficult to automate. Girard's sequent calculus [12], although covering all of linear logic, contains too many redundancies to be useful for efficient proof search. Attempts to remove permutabilities from sequent proofs [1, 10] and to add proof strategies [23] have provided significant improvements. But because of the use of sequent calculi some redundancies remain. Proof nets [7], on the other hand, can handle only a fragment of the logic.

Matrix characterizations of logical validity, originally developed as foundation of the *connection method* for classical logic [2, 3, 5], avoid many kinds of redundancies contained in sequent calculi and yield a compact representation of the search space. They have been extended successfully to intuitionistic and modal logics [24] and serve as a basis for a uniform proof search method [20] and a method for translating matrix proofs back into sequent proofs [21, 22]. Resource management similar to multiplicative linear logic is addressed by the *linear connection method* [4]. Fronhöfer [8] gives a matrix characterization of $\mathcal{MLL}$ that captures some aspects of weakening and contraction but does not appear to generalize any further. In [15] we have developed a matrix characterization for $\mathcal{MLL}$ and extended the uniform proof search and translation procedures accordingly.

In this paper we present a matrix characterization for the full multiplicative exponential fragment including the constants **1** and $\bot$. This characterization uses Andreoli's focusing principle [1] as one of its major design steps and does

J. Dix, F. L. Del Cerro, U. Furbach eds., 6^{th} *European Workshop on Logics in Artificial Intelligence, European Workshop, JELIA '98*, LNAI 1489, pp. 169–183, ©Springer Verlag, 1998.

not appear to share the limitations of the previous approaches. Our approach also includes a methodology for developing such a characterization. By introducing a series of intermediate calculi the development of a matrix characterization for $\mathcal{MELL}$ becomes manageable. Each newly introduced calculus adds one compactification principle and is proven correct and complete with respect to the previous one. We expect that this methodology will generalize to further fragments of linear logic as well as to a wide spectrum of other non-classical logics.

We first create a compact representation Σ_1' of Girard's sequent calculus [12] by adopting Smullyan's tableaux notation to $\mathcal{MELL}$ (Section 2). By introducing the notion of multiplicities, i.e. an eager handling of contraction and a lazy handling of weakening, we arrive at a *dyadic* calculus Σ_2' which we then refine to a *triadic* calculus Σ_3' by removing redundancies which are due to permutabilities (Section 3). In Section 4 we develop a calculus Σ_{pos} which operates on *positions* in a formula tree instead of on the subformulas themselves. In order to express the peculiarities of some connectives we insert *special positions* into the formula tree. Finally, in Section 5, we arrive at the matrix characterization, technically the most demanding but also the most compact of all calculi. Proofs are only sketched briefly. Details can be found in the first author's technical report [17].

2 Multiplicative Exponential Linear Logic

Linear logic [12] treats formulas like resources that disappear after their use unless explicitly marked as reusable. Technically, it can be seen as the outcome of removing the rules for contraction and weakening from the classical sequent calculus and re-introducing them in a controlled manner. Linear negation $^\perp$ is involutive like classical negation. The two traditions for writing the sequent rule for conjunction result in two different conjunctions $\otimes$ and & and two different disjunctions $⅋$ and $\oplus$. The constant `true` splits up into $\mathbf{1}$ and $\top$ and `false` into $\perp$ and $\mathbf{0}$. The unary connectives ? and ! mark formulas for a controlled application of weakening and contraction. Quantifiers $\forall$ and $\exists$ are added as usual.

Linear logic can be divided into the multiplicative, additive, and exponential fragment. While in the multiplicative fragment resources are used exactly once, resource sharing is enforced in the additive fragment. Exponentials mark formulas as reusable. All fragments exist on their own right and can be combined freely. The full power of linear logic comes from combining all of them.

Throughout this article we will focus on multiplicative exponential linear logic ($\mathcal{MELL}$), the combination of the multiplicative and exponential fragments, leaving the additive fragment and the quantifiers out of consideration. $^\perp$, $\otimes$, $⅋$, $\multimap$, $\mathbf{1}$, $\perp$, !, and ? are the connectives of $\mathcal{MELL}$. Linear negation $^\perp$ expresses the difference between resources that are to be used up and resources to be produced. $F^\perp$ means that the resource F must be produced. Having a resource $F_1 \otimes F_2$ means having F_1 as well as F_2. $F_1 \multimap F_2$ allows the construction of F_2 from F_1. $F_1 ⅋ F_2$ is equivalent to $F_1^\perp \multimap F_2$ and to $F_2^\perp \multimap F_1$. Having a resource $\mathbf{1}$ has no impact while nothing can be constructed when $\perp$ is used up. A resource $!F$ acts like a machine which produces any number of copies of F. During the construction of $!F$ only such machines can be used. ? is the dual to !.

$$\frac{}{\langle A,+\rangle,\langle A,-\rangle}\ axiom \qquad \frac{}{\tau}\ \tau \qquad \frac{\Upsilon}{\Upsilon,\omega}\ \omega \qquad \frac{\Upsilon, succ_1(o)}{\Upsilon, o}\ o$$

$$\frac{\Upsilon, succ_1(\alpha), succ_2(\alpha)}{\Upsilon,\alpha}\ \alpha \qquad \frac{\Upsilon_1, succ_1(\beta) \quad \Upsilon_2, succ_2(\beta)}{\Upsilon_1,\Upsilon_2,\beta}\ \beta$$

$$\frac{\Upsilon, succ_1(\nu)}{\Upsilon,\nu}\ \nu \qquad \frac{\boldsymbol{\nu}, succ_1(\pi)}{\boldsymbol{\nu},\pi}\ \pi \qquad \frac{\Upsilon}{\Upsilon,\nu}\ w \qquad \frac{\Upsilon,\nu,\nu}{\Upsilon,\nu}\ c$$

Table 1. Sequent calculus Σ_1' for $\mathcal{MELL}$ in uniform notation

The validity of a linear logic formula can be proven syntactically by using a sequent calculus. For multi-sets Γ and Δ of formulas $\Gamma \longrightarrow \Delta$ is called a *sequent*. It can be understood as the specification of a transformation which constructs Δ from Γ. The formulas in Γ are connected implicitly by $\otimes$ while the formulas in Δ are connected implicitly by $\parr$.

By adopting Smullyan's uniform notation to $\mathcal{MELL}$ we receive a compact representation of sequent calculi, which simplifies proofs about their properties. A *signed formula* $\varphi = \langle F,k\rangle$ denotes an occurrence of F in Δ or Γ. Depending on the *label* F and its *polarity* $k \in \{+,-\}$, a signed formula will receive a *type* α, β, ν, π, o, τ, ω, or *lit* according to the tables below. The functions $succ_1$ and $succ_2$ return the major signed subformulas of a signed formula. Note that during the decomposition of a formula the polarity switches only for $^\perp$ and $\multimap$. We use type symbols as meta-variables for signed formulas of the respective type, e.g. α stands for a signed formula of type α.

lit	$\langle A,-\rangle$	$\langle A,+\rangle$
τ	$\langle\perp,-\rangle$	$\langle\mathbf{1},+\rangle$
ω	$\langle\mathbf{1},-\rangle$	$\langle\perp,+\rangle$

α	$\langle F_1\otimes F_2,-\rangle$	$\langle F_1\parr F_2,+\rangle$	$\langle F_1\multimap F_2,+\rangle$
$succ_1(\alpha)$	$\langle F_1,-\rangle$	$\langle F_1,+\rangle$	$\langle F_1,-\rangle$
$succ_2(\alpha)$	$\langle F_2,-\rangle$	$\langle F_2,+\rangle$	$\langle F_2,+\rangle$

β	$\langle F_1\otimes F_2,+\rangle$	$\langle F_1\parr F_2,-\rangle$	$\langle F_1\multimap F_2,-\rangle$
$succ_1(\beta)$	$\langle F_1,+\rangle$	$\langle F_1,-\rangle$	$\langle F_1,+\rangle$
$succ_2(\beta)$	$\langle F_2,+\rangle$	$\langle F_2,-\rangle$	$\langle F_2,-\rangle$

o	$\langle F^\perp,-\rangle$	$\langle F^\perp,+\rangle$
$succ_1(o)$	$\langle F,+\rangle$	$\langle F,-\rangle$

ν	$\langle !F,-\rangle$	$\langle ?F,+\rangle$
$succ_1(\nu)$	$\langle F,-\rangle$	$\langle F,+\rangle$

π	$\langle ?F,-\rangle$	$\langle !F,+\rangle$
$succ_1(\pi)$	$\langle F,-\rangle$	$\langle F,+\rangle$

A sequent calculus Σ_1' based on this uniform notation is depicted in table 1. In a rule the sequents above the line are the *premises* and the one below the *conclusion.* A *principal formula* is a formula that occurs in the conclusion but not in any premise. Formulas that occur in a premise but not in the conclusion are called *active.* All other formulas compose the *context.* Σ_1' is shown correct and complete wrt. Girard's original sequent calculus [12] by a straightforward induction over the structure of proofs.

In *analytic proof search* one starts with the sequent to be proven and reduces it by application of rules until the *axiom*-rule or the τ-rule can be applied. There are several choice points within this process. First, a principal formula must be chosen. Unless the principal formula has type ν, this choice determines which rule must be applied. If a β-rule is applied the context of the sequent must be partitioned onto the premises (*context splitting*). Several solutions have been proposed in order to optimize these choices [1, 10, 23, 6, 13]. Additional difficulties arise from the rules *axiom*, τ, and π. The rules *axiom* and τ require an empty context which expresses that all formulas must be used up in a proof. The π rule requires that all formulas in the context are of type ν. Though the connectives of linear logic make proof search more difficult they also give rise to new possibilities. Some applications for linear logic programming are illustrated in [19].

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{a_{000}, a_{0100}}\,axiom}{a_{000}, o_{010}}\,o}{a_{000}, \nu_{01}}\,\nu \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{a_{0010}, a'_{0100}}\,axiom}{a_{0010}, o'_{010}}\,o}{a_{0010}, \nu'_{01}}\,\nu}{\pi_{001}, \nu'_{01}}\,\pi}{\dfrac{\beta_{00}, \nu_{01}, \nu'_{01}}{\dfrac{\beta_{00}, \nu_{01}}{\alpha_0}\,\alpha}\,c}\,\beta
$$

$lab(\varphi')$	φ'
$(A \otimes !A) ⅋ ?(A^\perp)$	α_0
$A \otimes !A$	β_{00}
A	a_{000}
$!A$	π_{001}
A	a_{0010}
$?(A^\perp)$	ν_{01}, ν'_{01}
$A^\perp$	o_{010}, o'_{010}
A	a_{0100}, a'_{0100}

Fig. 1. Example Σ'_1-proof.

Example 1. Figure 1 presents a Σ'_1-proof of $\varphi = \langle (A \otimes !A) ⅋ ?(A^\perp), + \rangle$. We abbreviate subformulas φ' of φ by position markers as shown in the table on the right. The proof requires that the contraction rule c is applied before the β-rule.

3 N-adic Sequent Calculi for $\mathcal{MELL}$

In this section we define two intermediate sequent calculi which are closely related to Andreoli's dyadic calculus Σ_2 and triadic calculus Σ_3 [1] but differ in the way structural rules are handled. While Andreoli uses a lazy strategy for both contraction and weakening, our calculi Σ'_2 and Σ'_3, which are *not* intended for proof search, are based on an eager strategy for contraction. Eager contraction corresponds to the concept of multiplicities in matrix characterizations [24].

3.1 Dyadic Calculus

$$
\dfrac{}{\Theta : \langle A, + \rangle, \langle A, - \rangle}\,axiom \qquad \dfrac{}{\Theta : \tau}\,\tau \qquad \dfrac{\Theta : \Upsilon, succ_1(o)}{\Theta : \Upsilon, o}\,o \qquad \dfrac{\Theta : \Upsilon}{\Theta : \Upsilon, \omega}\,\omega
$$

$$
\dfrac{\Theta : \Upsilon, succ_1(\alpha), succ_2(\alpha)}{\Theta : \Upsilon, \alpha}\,\alpha \qquad \dfrac{\Theta_1 : \Upsilon_1, succ_1(\beta) \quad \Theta_2 : \Upsilon_2, succ_2(\beta)}{\Theta_1, \Theta_2 : \Upsilon_1, \Upsilon_2, \beta}\,\beta
$$

$$
\dfrac{\Theta, succ_1(\nu)^{\mu(\nu)} : \Upsilon}{\Theta : \Upsilon, \nu}\,\nu \qquad \dfrac{\Theta : succ_1(\pi)}{\Theta : \pi}\,\pi \qquad \dfrac{\Theta : \Upsilon, \varphi}{\Theta, \varphi : \Upsilon}\,focus
$$

Table 2. Dyadic sequent calculus Σ'_2 for $\mathcal{MELL}$ in uniform notation

In sequent proofs there are two possible notions of occurrence of a formula φ: an occurrence of φ as subformula in some formula tree or its occurrences within a derivation. The difference between these two becomes only apparent when contraction is applied. In $\mathcal{MELL}$ only formulas of type ν are *generic*, i.e. may be contracted. Since we are aiming at a matrix characterization, we apply contraction in an eager way. For this purpose we introduce a function μ which determines the *multiplicity* of an occurrence of a formula, i.e. the number of copies of that occurrence in a proof.[1] Let Θ and Υ be multi-sets of signed formulas. A *dyadic sequent* $S = \Theta : \Upsilon$ has two zones which are separated by a colon. Θ is called the *unbounded zone* and Υ the *bounded zone* of S.

The sequent calculus Σ'_2 for dyadic sequents depicted in table 2 employs eager contraction. Derivations of a dyadic sequent S are defined with respect to

[1] Wallen's multiplicities for modal and intuitionistic logics are based on occurrences within a *formula tree* [24]. Our notion respects the resource sensitivity of linear logic.

a fixed multiplicity function μ. This function is important whenever the ν-rule is applied. Additionally, Σ_2' uses lazy weakening. There is no explicit weakening rule but weakening is done implicitly in the rules *axiom* and τ. The π rule requires that all context formulas are in the unbounded zone instead of requiring them to have a specific type as in Σ_1'. A special rule *focus* moves a formula from the unbounded into the bounded zone. A signed formula φ is *derivable* in Σ_2' if $\cdot : \varphi$ is derivable for some multiplicity, where $\cdot$ denotes the empty multi-set.

Theorem 1 (Completeness). *Let $\mathcal{P}_1$ be a Σ_1'–proof for $S_1 = \Upsilon, \boldsymbol{\nu}_{\overline{c}}$ where $\boldsymbol{\nu}_{\overline{c}}$ consists of all signed formulas of type ν in S_1 to which the contraction rule is not applied in $\mathcal{P}_1$. Then there is a multiplicity μ_2 such that the dyadic sequent $S_2 = succ_1(\boldsymbol{\nu}_{\overline{c}}) : \Upsilon$ can be derived in Σ_2'.*[2]

Theorem 2 (Correctness). *Let $\mathcal{P}_2$ be a Σ_2'–proof for $S_2 = \Theta_1^+, \Theta_2^- : \Upsilon$ with multiplicity μ_2 where Θ_1^+ and Θ_2^- contain only positive and negative signed formulas, respectively. Then there exists a Σ_1'–proof $\mathcal{P}_1$ for the unary sequent $S_1 = ?\Theta_1^+, !\Theta_2^-, \Upsilon$.*

3.2 Triadic Calculus

$$\frac{}{\Theta : \langle A,+\rangle, \langle A,-\rangle \Uparrow \cdot}\ axiom \qquad \frac{}{\Theta : \tau \Uparrow \cdot}\ \tau \qquad \frac{\Theta : \Upsilon \Uparrow \Xi}{\Theta : \Upsilon \Uparrow \Xi, \omega}\ \omega$$

$$\frac{\Theta : \Upsilon \Downarrow succ_1(o)}{\Theta : \Upsilon \Downarrow o}\ o\Downarrow \qquad \frac{\Theta : \Upsilon \Uparrow \Xi, succ_1(o)}{\Theta : \Upsilon \Uparrow \Xi, o}\ o\Uparrow$$

$$\frac{\Theta : \Upsilon \Uparrow \Xi, succ_1(\alpha), succ_2(\alpha)}{\Theta : \Upsilon \Uparrow \Xi, \alpha}\ \alpha \qquad \frac{\Theta_1 : \Upsilon_1 \Downarrow succ_1(\beta) \quad \Theta_2 : \Upsilon_2 \Downarrow succ_2(\beta)}{\Theta_1, \Theta_2 : \Upsilon_1, \Upsilon_2 \Downarrow \beta}\ \beta$$

$$\frac{\Theta, succ_1(\nu)^{\mu(\nu)} : \Upsilon \Uparrow \Xi}{\Theta : \Upsilon \Uparrow \Xi, \nu}\ \nu \qquad \frac{\Theta : \cdot \Uparrow succ_1(\pi)}{\Theta : \cdot \Downarrow \pi}\ \pi$$

$$\frac{\Theta : \Upsilon \Downarrow \varphi}{\Theta, \varphi : \Upsilon \Uparrow \cdot}\ focus_1 \qquad \frac{\Theta : \Upsilon \Downarrow \varphi}{\Theta : \Upsilon, \varphi \Uparrow \cdot}\ focus_2 \qquad \frac{\Theta : \Upsilon, \varphi \Uparrow \Xi}{\Theta : \Upsilon \Uparrow \Xi, \varphi}\ defocus \qquad \frac{\Theta : \Upsilon \Uparrow \varphi}{\Theta : \Upsilon \Downarrow \varphi}\ switch$$

In *focus*$_2$ φ must not be of type *lit* or τ. In *defocus* φ must be of type *lit*, τ, β, or π. In *switch* φ must be of type *lit*, τ, ω, α, or ν.

Table 3. Triadic sequent calculus Σ_3' for $\mathcal{MELL}$ in uniform notation

During proof search in sequent calculi the order of some rule applications may be permuted. For linear logic, the permutabilities and non-permutabilities of sequent rules have been investigated in [1, 10]. Andreoli's *focusing principle* [1] allows to fix the order of permutable rules without losing completeness. A distinctive feature of this principle is that the reduction ordering is determined for *layers of formulas* rather than for individual formulas. Let φ be a signed formula, Ξ be a sequence and Θ and Υ be multi-sets of signed formulas. A *triadic sequent* $S = \Theta : \Upsilon \Downarrow \varphi$ or $S = \Theta : \Upsilon \Uparrow \Xi$ has three zones. Θ is called the *unbounded zone*, Υ the *bounded zone*, and φ or Ξ the *focused* zone. The sequent is either in *synchronous* ($\Downarrow$) or in *asynchronous* mode ($\Uparrow$).

[2] For convenience we extend functions and connectives to multi-sets of signed formulas. $succ_1(\boldsymbol{\nu}_{\overline{c}})$ abbreviates $\{succ_1(\nu) \mid \nu \in \boldsymbol{\nu}_{\overline{c}}\}$, $?\Theta$ denotes $\{\langle ?F, k\rangle \mid \langle F, k\rangle \in \Theta\}$, etc.

The sequent calculus Σ_3' for triadic sequents depicted in table 3 employs the focusing principle. Derivations are defined with respect to a fixed multiplicity function μ. The multiplicity is important whenever the rule ν is applied. A signed formula φ is derivable in Σ_3' if the sequent $\cdot : \varphi \Uparrow \cdot$ is derivable for some μ.

In Σ_3' there are two focusing rules, *focus*$_1$ and *focus*$_2$, which move a signed formula into the focus. Both rules switch the sequent into synchronous mode. Depending on the structure of the formula this enforces a sequence of rules to be applied next. Since selection of these rules is deterministic, the permutabilities of rule applications are removed from the search space. The matrix characterization developed in this article exploits this focusing principle. However, it yields a representation with even less redundancies than a calculus like Σ_3' can.

Theorem 3 (Completeness). *Let $\mathcal{P}_2$ be a Σ_2'–proof for $S_2 = \Theta : \Upsilon$ with multiplicity μ_2 and let Υ' be a linearization of Υ. Then there is a multiplicity μ_3 such that the triadic sequent $S_3 = \Theta : \cdot \Uparrow \Upsilon'$ is derivable in Σ_3'.*

Theorem 4 (Correctness). *Let $\mathcal{P}_3$ be a Σ_3'–proof for $S_3 = \Theta : \Upsilon \Uparrow \Xi$ (or $\Theta : \Upsilon \Downarrow \varphi$) with multiplicity μ_3. Then there is a Σ_2'–proof $\mathcal{P}_2$ for the dyadic sequent $S_2 = \Theta : \Upsilon, \Xi'$ (or $\Theta : \Upsilon, \varphi$) for some multiplicity μ_2, where Ξ' is the multi-set that contains the same signed formulas as the sequence Ξ.*

4 A Position Calculus for $\mathcal{MELL}$

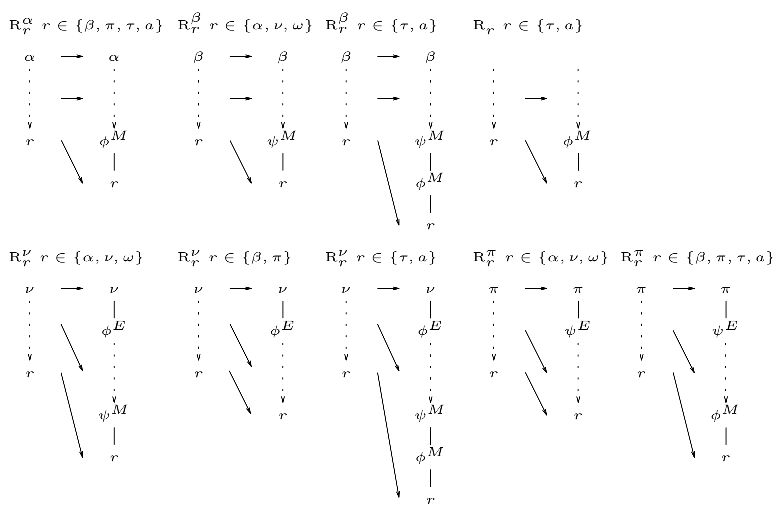


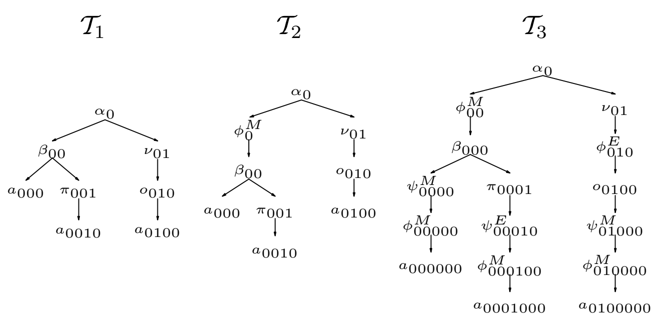
Fig. 2. Rules for inserting special positions into basic position trees

In the previous section we have used established techniques for removing redundancies in the search space of a specific sequent proof. In order to reason about logical validity as such, the non-classical aspects of sequent proofs must be expressed in a more compact way. Wallen [24] uses prefixes of special positions for modal logics and intuitionistic logic. We adapt this approach to $\mathcal{MELL}$ and introduce position calculi as intermediate step in the development of matrix characterizations. We capture the difference between an occurrence of a formula within a proof or as subformula by basic positions and positions.

Basic Position Trees Let V_b be an arbitrary set of *basic positions*. The *basic position tree* for a signed formula φ is a directed tree $\mathcal{T}_b = (V_b, E_b)$ which originates from a formula tree by inserting additional nodes.[3] These nodes are inserted by applying rewrite rules from figure 2 until none of them is applicable. The functions $succ_1$ and $succ_2$ are redefined for basic position trees such that $succ_1(bp)$ is the left and $succ_2(bp)$ the right successor of the node *bp*. For a basic position $bp \in V_b$, the corresponding formula, main connective, signed formula, polarity, and type can be retrieved by the functions *lab*, *con*, *sform*, *pol*, and *Ptype*, respectively. For the inserted nodes we introduce new *special types* ϕ^M, ψ^M, ϕ^E, and ψ^E which are assigned according to the rewrite rules. For an inserted basic position of type ϕ^M, ψ^M, or ϕ^E the value of the functions *lab*, *con*, and *pol* equals the value of the successor node and for type ψ^E it is the value of the predecessor node. We use type symbols as meta-variables for basic positions of the respective type and a as a meta-variable for basic positions of type *lit*.

A rewrite rule R can be applied to a tree $\mathcal{T}$ if its left hand side matches a subtree $\mathcal{T}'$ of $\mathcal{T}$. In this case $\mathcal{T}'$ is rewritten according to the pattern on the right hand side of R. The dotted lines in the patterns match arbitrary subtrees that contain only nodes of type o. A special case are the rules R_a and R_τ. They can only be applied if there are just positions of type o between the root and the leaf. The other rewrite rules separate layers of subformulas within a formula tree: $\mathrm{R}^{t_1}_{t_2}$ inserts special positions wherever a subformula of type t_1 has a subformula of type t_2. The rewrite system is confluent and noetherian.

Example 2. We illustrate the application of the rule $\mathrm{R}^{\alpha}_{\beta}$. Below, we have depicted the formula tree $\mathcal{T}_1$ for $\langle (A \otimes !A) \parr ?(A^{\perp}), + \rangle$. β_{00} is a successor of α_0 with no nodes in between. The subtree consisting of α_0, β_{00}, and the edge between them matches the left hand side of $\mathrm{R}^{\alpha}_{\beta}$. The tree is rewritten to $\mathcal{T}_2$ and can further be rewritten by applying R^{β}_{a}, R^{π}_{a}, and R^{ν}_{a}. The resulting basic position tree is $\mathcal{T}_3$.



p	$con(p)$
α_0	$\parr$
ϕ^M_{00}, β_{000}	$\otimes$
$\pi_{0001}, \psi^E_{00010}$	!
ν_{01}	?
ϕ^E_{010}	$\perp$
o_{0100}	$\perp$
ψ^M_{01000}	$\perp$

Special positions represent the possible behavior of a layer within a derivation. Positions of type ψ^M and ψ^E are called *constants* while those of type ϕ^M and ϕ^E are *variables*. Inserted variables express that the corresponding formula may be part of the bounded (ϕ^M) or unbounded (ϕ^E) zone. During a sequence of

[3] We denote basic positions by strings over $\{0, 1\}$. 0 is the root position of a tree. Extending a tree by 0 or 1 yields the basic position for the left or right successor node.

proof rule applications that ends with a constant position, a part of the context may have to be of a specific type. The requirements of the π–rule, for instance, are expressed by a ψ^E position.

Position Trees, Position Forests, and Position Sequents A *position tree* for a signed formula wrt. a multiplicity μ is constructed from a basic position tree by subsequently replacing subtrees of positions ν with $\mu(\nu) = n$ with n copies of that tree starting from the root. New positions are assigned to the copies.[4]

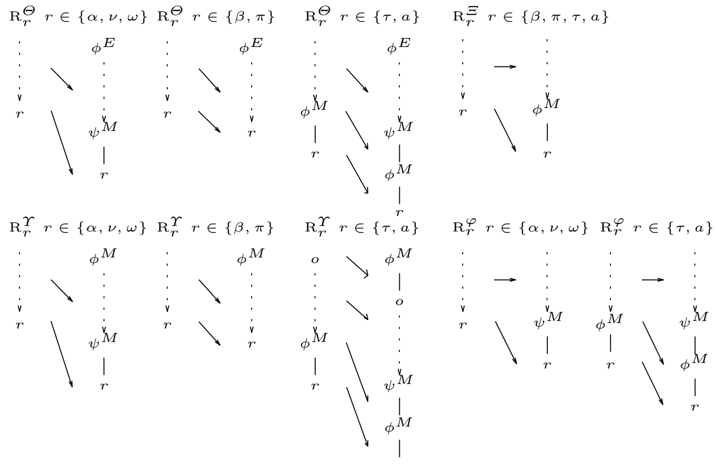
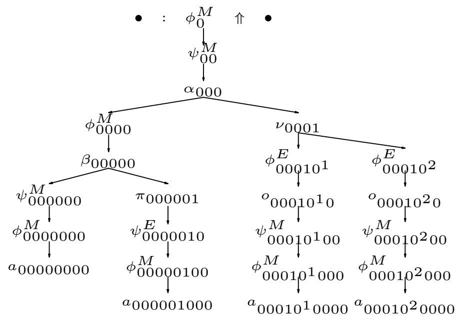


Fig. 3. Rules for inserting special positions into basic position forests

As formulas can be represented as formula trees, sequents can be represented as *sequent forests*, i.e. collections of formula trees that are divided into different zones. A *basic position forest* for a triadic sequent is a collection of basic position trees which consists of three zones (unbounded, bounded, and focused) and has a mode ($\Downarrow$ or $\Uparrow$). The trees are modified by the rewrite rules in figure 3, which modify trees only at their roots. The exponent of a rewrite rule defines the zone in which it can be applied. $\mathrm{R}_?^{\Theta}, \mathrm{R}_?^{\Upsilon}, \mathrm{R}_?^{\Xi}$, and $\mathrm{R}_?^{\varphi}$ can be applied to trees in the unbounded zone, bounded zone, focused zone (mode $\Uparrow$) and focused zone (mode $\Downarrow$), respectively. A *position forest* is a collection of position trees which is divided into three zones, has a mode, and which is constructed from a basic position forest together with a multiplicity. We use *position sequent* as well as *position matrix* as a synonym for position tree. We denote trees by their roots if the rest of the tree is obvious from the context. Note, that by the definition of the rewrite rules a root in the unbounded zone is always of type ϕ^E. A root in the bounded zone is of type ϕ^M. A root in the focused zone in mode $\Uparrow$ has a type in $\{o, \omega, \alpha, \nu, \phi^M\}$ and a root in mode $\Downarrow$ has a type from $\{o, \beta, \pi, \psi^M, \psi^E\}$.

Example 3. Position sequents can be represented graphically. Let ϕ_0^M be the root of a position tree which corresponds to $\langle (A\otimes !A) \parr ?(A^{\perp}), +\rangle$ with multiplicity $\mu(\nu_{0001})$=2. The position sequent for $\cdot : \phi_0^M \Uparrow \cdot$ is depicted in figure 4.

[4] We denote positions by strings over $\{0, 1, 0^m\}_{m\in\mathrm{I\!N}}$. Positions in different copies of a subtree are distinguished by their exponents.



p	$\mathrm{pre}_{\mathcal{M}}(p)$
$a_{00000000}$	$\phi_0^M\,\psi_{00}^M\,\phi_{0000}^M\,\psi_{000000}^M\,\phi_{0000000}^M$
$a_{000001000}$	$\phi_0^M\,\psi_{00}^M\,\phi_{0000}^M\,\psi_{0000010}^E\,\phi_{00000100}^M$
$a_{00010^1 000}$	$\phi_0^M\,\psi_{00}^M\,\phi_{00010^1}^E\,\psi_{00010^1 00}^M\,\phi_{00010^1 000}^M$
$a_{00010^2 000}$	$\phi_0^M\,\psi_{00}^M\,\phi_{00010^2}^E\,\psi_{00010^2 00}^M\,\phi_{00010^2 000}^M$

Prefixes of leaves (see section 5)

Fig. 4. Position sequent for the sequent $\cdot : \phi_0^M \Uparrow \cdot$

Position Calculus A sequent calculus for position sequents is depicted in table 4. Derivations of a position sequent S are defined with respect to a fixed multiplicity function μ for S. The position calculus makes apparent that, as pointed out earlier, inserted positions express how the context is affected in certain rules. The rewrite rules in figure 3 guarantee that the reduction of a position sequent by a rule results again in a position sequent. In a proof, the mode of a position sequent can be switched either by ψ^M or ψ^E. There is no *defocus* rule as in Σ_3' nor does the π rule cause a switch. Defocusing is done by the rules ϕ^M and ν. For all other rules there is a corresponding rule in Σ_3'.

$$\frac{}{\Phi^E : \phi_1^M, \phi_2^M \Uparrow \cdot}\ \mathit{axiom} \qquad \text{where } Ptype(succ_1(\phi_1^M)) = Ptype(succ_1(\phi_2^M)) = lit,\ lab(\phi_1^M) = lab(\phi_2^M), \text{ and } pol(\phi_1^M) \neq pol(\phi_2^M)$$

$$\frac{\Phi^E : \Phi^M \Downarrow succ_1(o)}{\Phi^E : \Phi^M \Downarrow o}\ o\Downarrow \qquad \frac{\Phi^E : \Phi^M \Uparrow \Xi, succ_1(o)}{\Phi^E : \Phi^M \Uparrow \Xi, o}\ o\Uparrow$$

$$\frac{\Phi^E : \Phi^M \Uparrow \Xi}{\Phi^E : \Phi^M \Uparrow \Xi, \omega}\ \omega \qquad \frac{}{\Phi^E : \phi_1^M \Uparrow \cdot}\ \tau \quad \text{where } Ptype(succ_1(\phi_1^M)) = \tau$$

$$\frac{\Phi^E : \Phi^M \Uparrow \Xi, succ_1(\alpha), succ_2(\alpha)}{\Phi^E : \Phi^M \Uparrow \Xi, \alpha}\ \alpha \qquad \frac{\Phi_1^E : \Phi_1^M \Downarrow succ_1(\beta) \quad \Phi_2^E : \Phi_2^M \Downarrow succ_2(\beta)}{\Phi_1^E, \Phi_2^E : \Phi_1^M, \Phi_2^M \Downarrow \beta}\ \beta$$

$$\frac{\Phi^E : \Phi^M \Downarrow succ_1(\pi)}{\Phi^E : \Phi^M \Downarrow \pi}\ \pi \qquad \frac{\Phi^E, succ_1^1(\nu), \ldots, succ_1^{\mu(\nu)}(\nu) : \Phi^M \Uparrow \Xi}{\Phi^E : \Phi^M \Uparrow \Xi, \nu}\ \nu$$

$$\frac{\Phi^E : \Phi^M \Downarrow succ_1(\phi_1^E)}{\Phi^E, \phi_1^E : \Phi^M \Uparrow \cdot}\ focus_1 \qquad \frac{\Phi^E : \Phi^M \Downarrow succ_1(\phi_1^M)}{\Phi^E : \Phi^M, \phi_1^M \Uparrow \cdot}\ focus_2 \quad \text{where } Ptype(succ_1(\phi_1^M)) \notin \{lit, \tau\}$$

$$\frac{\Phi^E : \Phi^M, \phi_1^M \Uparrow \Xi}{\Phi^E : \Phi^M \Uparrow \Xi, \phi_1^M}\ \phi^M \qquad \frac{\Phi^E : \Phi^M \Uparrow succ_1(\psi_1^M)}{\Phi^E : \Phi^M \Downarrow \psi_1^M}\ \psi^M \qquad \frac{\Phi^E : \cdot \Uparrow succ_1(\psi_1^E)}{\Phi^E : \cdot \Downarrow \psi_1^E}\ \psi^E$$

Table 4. Position calculus Σ_{pos} for $\mathcal{MELL}$ in uniform notation

Theorem 5 (Completeness). *Let $\mathcal{P}_3$ be a Σ_3'–proof for $S_3 = \Theta : \Upsilon \Uparrow \Xi$ (or $S_3 = \Theta : \Upsilon \Downarrow \varphi$) with multiplicity μ_3. Then there is a multiplicity $\widetilde{\mu_3}$ such that the corresponding position sequent $\widetilde{S_3}$ is derivable in Σ_{pos}.*

Theorem 6 (Correctness). *Let $\mathcal{P}$ be a Σ_{pos}–proof for $S = \Phi^E : \Phi^M \Uparrow \Xi$ (or $S = \Phi^E : \Phi^M \Downarrow \varphi$) with multiplicity μ. Then there is a multiplicity $\widetilde{\mu}$ such that the corresponding triadic sequent $\widetilde{S} = \mathrm{sform}(\Phi^E) : \mathrm{sform}(\Phi^M) \Uparrow \mathrm{sform}(\Xi)$ (or $\widetilde{S} = \mathrm{sform}(\Phi^E) : \mathrm{sform}(\Phi^M) \Downarrow \mathrm{sform}(\varphi)$) is derivable in Σ_3'.*

5 The Matrix Characterization for $\mathcal{MELL}$

On the basis of the position calculus we can now develop a compact characterization of logical validity. For this purpose we extend the concept of position forests by important technical notions, state the requirements for validity in $\mathcal{MELL}$, and prove them to be sufficient and complete. We then summarize all the insights gained in this paper in a single characterization theorem.

Fundamental Concepts. A matrix characterization of logical validity of a formula φ is expressed in terms of properties of certain sets of subformulas of φ. We use positions to achieve a compact representation of φ and its subformulas.

A *matrix* $\mathcal{M}$ is a position forest, constructed from a basic position forest and a multiplicity μ. The set of positions in a matrix $\mathcal{M}$ is denoted by $Pos(\mathcal{M})$. The set of *axiom positions* $AxPos(\mathcal{M})$ contains all positions with principal type τ or *lit*. The set of *weakening positions* $WeakPos(\mathcal{M})$ contains all positions with principal type ω and all positions of type ν with $\mu(\nu) = 0$. The set of *leaf positions* is defined as $LeafPos(\mathcal{M}) = AxPos(\mathcal{M}) \cup WeakPos(\mathcal{M})$. $\boldsymbol{\beta}(\mathcal{M})$ is the set of all positions in $Pos(\mathcal{M})$ of type β. The sets $\Phi^M(\mathcal{M})$, $\Psi^M(\mathcal{M})$, $\Phi^E(\mathcal{M})$, and $\Psi^E(\mathcal{M})$ are defined accordingly. The set of *special positions* is defined as $SpecPos(\mathcal{M}) = \Phi^M(\mathcal{M}) \cup \Psi^M(\mathcal{M}) \cup \Phi^E(\mathcal{M}) \cup \Psi^E(\mathcal{M})$.

A *weakening map* for $\mathcal{M}$ is a subset of $\Phi^E(\mathcal{M}) \cup WeakPos(\mathcal{M})$. This novel concept is required because of the restricted application of weakening in $\mathcal{MELL}$.

A *path* is a set of positions. The set $Paths(\mathcal{T})$ of *paths through a position tree* $\mathcal{T}$ is defined recursively by

- $P = \{0\}$, the set containing the root of $\mathcal{T}$, is a path through $\mathcal{T}$.
- If $P \cup \{p\}$ is a path through $\mathcal{T}$ then the following are paths

$$\begin{array}{ll} P \cup \{succ_1(p), succ_2(p)\} & \text{if } Ptype(p) = \alpha \\ P \cup \{succ_1(p)\} \text{ and } P \cup \{succ_2(p)\} & \text{if } Ptype(p) = \beta \\ P \cup \{succ_1(p)\} & \text{if } Ptype(p) \in \{o, \pi, \phi^M, \psi^M, \phi^E, \psi^E\} \\ P \cup \bigcup_{i \leq \mu(p)} \{succ_1^i(p)\} & \text{if } Ptype(p) = \nu, \text{ and } \mu(\nu) > 0 \end{array}$$

The *set of paths through a set of position trees* $\mathcal{F}_s$ is defined recursively by

$$\begin{array}{ll} Paths(\emptyset) & = \emptyset, \\ Paths(\{\mathcal{T}\}) & = Paths(\mathcal{T}), \text{ and} \\ Paths(\{\mathcal{T}\} \cup \mathcal{F}_s') & = \{P_1 \cup P_2 \mid P_1 \in Paths(\mathcal{T}), P_2 \in Paths(\mathcal{F}_s')\}. \end{array}$$

The *set of paths through a matrix* is defined by

$$\begin{array}{l} Paths(\Phi^E : \Phi^M \Downarrow \varphi) = Paths(\Phi^E \cup \Phi^M \cup \{\varphi\}) \text{ and} \\ Paths(\Phi^E : \Phi^M \Uparrow \Xi) = Paths(\Phi^E \cup \Phi^M \cup \Xi) \quad . \end{array}$$

A *path of leaves* through $\mathcal{M}$ is a subset of $LeafPos(\mathcal{M})$. $LPaths(\mathcal{M})$ denotes the *set of all paths of leaves through* $\mathcal{M}$. Since leaf positions are not decomposed in the definition of paths, a path of leaves contains only irreducible positions.

A *connection* in a matrix $\mathcal{M}$ is a subset of $AxPos(\mathcal{M})$. It is either a two-element set $\{p_1, p_2\}$ where p_1 and p_2 are positions with different polarity, $lab(p_1) = lab(p_2)$, and $Ptype(p_1) = lit = Ptype(p_2)$, or a one-element set $\{p_1\}$ with $Ptype(p_1) = \tau$. *A connection C is on a path P* if $C \subseteq P$.

The *prefix* $pre_{\mathcal{M}}(p)$ of a position $p \in Pos(\mathcal{M})$ is defined by

– $pre_{\mathcal{M}}(0) \quad = \begin{cases} r_0 & \text{if } r = Ptype(0) \in \{\phi^M, \psi^M, \phi^E, \psi^E\} \\ \varepsilon & \text{otherwise} \end{cases}$

– $pre_{\mathcal{M}}(p'i) \ = \begin{cases} pre_{\mathcal{M}}(p')r_{p'i} & \text{if } r = Ptype(p'i) \in \{\phi^M, \psi^M, \phi^E, \psi^E\} \\ pre_{\mathcal{M}}(p') & \text{otherwise} \end{cases}$

If $p_1 \ll p_2$, i.e. p_1 is a predecessor of p_2 in the position tree, then $pre_{\mathcal{M}}(p_1)$ is an initial substring of $pre_{\mathcal{M}}(p_2)$. We denote this by $pre_{\mathcal{M}}(p_1) \angle pre_{\mathcal{M}}(p_2)$. The prefixes of leaves of the example position sequent are displayed in figure 4.

A *multiplicative prefix substitution* is a mapping $\sigma_M : \Phi^M \to (\Phi^M \cup \Psi^M)^*$. An *exponential prefix substitution* is a mapping $\sigma_E : \Phi^E \to (\Phi^M \cup \Psi^M \cup \Phi^E \cup \Psi^E)^*$. A *multiplicative exponential prefix substitution* is a mapping $\sigma : (\Phi^M \cup \Phi^E) \to (\Phi^M \cup \Psi^M \cup \Phi^E \cup \Psi^E)^*$ which maps elements from Φ^M to strings from $(\Phi^M \cup \Psi^M)^*$ only. Substitutions are extended homomorphically to strings and are assumed to be computed by unification.

Complementarity. Matrix characterizations for classical [5] and non-classical [24] logics are based on a notion of *complementarity*. Essentially this means that every path through a matrix must contain a unifiable connection. These requirements also hold for linear logic but have to be extended by a few additional properties. We shall specify all these requirements now.

In the following we always assume $\mathcal{M}$ to be a matrix, $\mathcal{C}$ and $\mathcal{W}$ to be a set of connections and a weakening map for $\mathcal{M}$, and σ to be a prefix substitution.

- The *spanning property* is the most fundamental requirement. Each path of leaves must contain a connection. A set of connections $\mathcal{C}$ *spans* a matrix $\mathcal{M}$ iff for every path $P \in LPaths(\mathcal{M})$ there is a connection $C \in \mathcal{C}$ with $C \subseteq P$.
- The *unifiability property* states that connected leaves must be made identical wrt. their prefixes. Furthermore, because of the restricted application of weakening in $\mathcal{MELL}$, that each position in a weakening map must be related to a connection. $\langle \mathcal{C}, \mathcal{W} \rangle$ is *unifiable* if there exists a prefix substitution σ such that (1) $\sigma(pre_{\mathcal{M}}(p_1)) = \sigma(pre_{\mathcal{M}}(p_2))$ for all $C \in \mathcal{C}$ and $p_1, p_2 \in C$ and (2) for all $wp \in \mathcal{W}$ there is some $C = \{p, \dots\} \in \mathcal{C}$ with $\sigma(pre_{\mathcal{M}}(wp)) \angle \sigma(pre_{\mathcal{M}}(p))$. In this case σ is called a *unifier* for $\langle \mathcal{C}, \mathcal{W} \rangle$.
  A unifier σ of $\langle \mathcal{C}, \mathcal{W} \rangle$ can always be modified to a unifier σ' such that $\sigma'(pre_{\mathcal{M}}(\phi^E)) = \sigma'(pre_{\mathcal{M}}(p))$ holds for all $\phi^E \in \mathcal{W}$ and some connection $C = \{p, \dots\} \in \mathcal{C}$. A *grounded substitution* σ' is constructed from a substitution σ by removing all variable positions from values. If σ is a unifier for $\langle \mathcal{C}, \mathcal{W} \rangle$ then the corresponding grounded substitution σ' is a unifier as well.
- The *linearity property* expresses that no resource is to be used twice, i.e. contraction is restricted. A resource cannot be connected more than once and cannot be connected at all if that part of the formula is weakened. $\langle \mathcal{C}, \mathcal{W} \rangle$ is *linear* for $\mathcal{M}$ iff (1) any two connections $C_1 \neq C_2 \in \mathcal{C}$ are disjoint and (2) no predecessor ϕ^E of a position p in a connection $C \in \mathcal{C}$ belongs to the set $\mathcal{W}$.
- The *relevance property* requires that each resource must be used at least once. A resource is used if it is connected or weakened. $\langle \mathcal{C}, \mathcal{W} \rangle$ has the *relevance* property for $\mathcal{M}$ if for $p \in LeafPos(\mathcal{M})$ either (1) $p \in C$ for some $C \in \mathcal{C}$, or (2) $p \in \mathcal{W}$, or (3) some predecessor ϕ^E of p belongs to $\mathcal{W}$.

– The *cardinality property* expresses that the number of branches in a sequent proof is adequate. It substitutes the *minimality property* in [8], which would require a complicated test for $\mathcal{MELL}$. A pair $\langle\mathcal{C},\mathcal{W}\rangle$ has the *cardinality property* for $\mathcal{M}$ if $|\mathcal{C}| + \sum_{\phi^E\in\mathcal{W}} \boldsymbol{\beta}(\phi^E) = \boldsymbol{\beta}(\mathcal{M}) + 1$.

Definition 1. *A matrix $\mathcal{M}$ is* complementary *iff there are a set of connections $\mathcal{C}$, a weakening map $\mathcal{W}$ and a prefix substitution σ such that (1) $\mathcal{C}$ spans $\mathcal{M}$, (2) σ is a unifier for $\langle\mathcal{C},\mathcal{W}\rangle$, and (3) $\langle\mathcal{C},\mathcal{W}\rangle$ is linear for $\mathcal{M}$ and has the relevance and cardinality properties. We also say that $\mathcal{M}$* is complementary for *$\mathcal{C}$, $\mathcal{W}$, and σ.*

The complementarity of a matrix ensures the existence of a corresponding Σ_{pos}-proof. Each requirement captures an essential aspect of such a proof. Thus, there are relations between basic concepts in matrix proofs and Σ_{pos}-proofs. Paths are related to sequents. A connection on a path expresses the potential to close a Σ_{pos}-branch by an application of *axiom* or τ which involves the connected positions. A weakening map $\mathcal{W}$ contains all positions which are explicitly weakened by the rules ω and ν (for $\mu(\nu) = 0$) or implicitly weakened in *axiom* and τ. The unifiability of prefixes guarantees that connected positions can move into the same Σ_{pos} branch and that positions in $\mathcal{W}$ can be weakened in some branch. Linearity and relevance resemble the lack of contraction and weakening for arbitrary formulas, while cardinality expresses the absence of the rule of mingle, i.e. a proof can only branch at the reduction of β-type positions.

Since complementarity captures the essential aspects of Σ_{pos}-proofs but no unimportant details the search space is once more compactified. Problems like e.g. context splitting at the reduction of β-type positions simply do not occur.

Soundness. The unifiability property requires that each position in a weakening map $\mathcal{W}$ is related to some connection $C \in \mathcal{C}$. Let $AssSet(C)$ be the union of C and the set of all positions in $\mathcal{W}$ which are related to C.

A matrix $\mathcal{M}$ can be seen as a collection of trees, i.e. a forest. Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be position trees in $\mathcal{M}$. We add for each connection C edges which link all positions in $AssSet(C)$. In order to identify the connected components of the resulting graph we define a relation $AssRel$ by

$$AssRel(\mathcal{T}_1,\mathcal{T}_2) \;\; \textit{iff} \;\; \exists C \in \mathcal{C}.\exists p_1 \in Pos(\mathcal{T}_1), p_2 \in Pos(\mathcal{T}_2).p_1, p_2 \in AssSet(C)$$

Let $\sim$ be the reflexive transitive closure of $AssRel$. A *connected component* in $\mathcal{M}$ is a set of position trees which is an equivalence class of $\sim$.

We define a function $FCons$ which reduces a set of connections $\mathcal{C}$ to those connections C whose elements are contained within a certain position forest $\mathcal{F}$. We define $FCons(\mathcal{F},\mathcal{C}) = \{C \in \mathcal{C} \mid \forall p \in C.p \in Pos(\mathcal{F})\}$.

Theorem 7. *Let $\langle\mathcal{C},\mathcal{W}\rangle$ be linear and relevant for the matrix $\mathcal{M}$ and σ be a unifier for $\langle\mathcal{C},\mathcal{W}\rangle$. Then $|\boldsymbol{\beta}(\mathcal{F})| < |FCons(\mathcal{F},\mathcal{C})| + \sum_{\phi^E\in\mathcal{W}} |\boldsymbol{\beta}(\phi^E)|$ holds for any connected component $\mathcal{F}$ in $\mathcal{M}$.*

Theorem 7 is proven by induction over the size of $\boldsymbol{\beta}(\mathcal{F})$. In the induction step one first shows that there is always a node of type β, the separator of a β–chain, which can be removed such that the preconditions of the theorem hold. The induction hypothesis can be applied after removing that node.

Corollary 1. *Let $\langle \mathcal{C}, \mathcal{W} \rangle$ be linear for the matrix $\mathcal{M}$ and have the relevance and cardinality properties for $\mathcal{M}$. Let σ be a unifier for $\langle \mathcal{C}, \mathcal{W} \rangle$. Then there is exactly one connected component in $\mathcal{M}$.*

Lemma 1. *Let $\langle \mathcal{C}, \mathcal{W} \rangle$ be linear for the matrix $\mathcal{M}$ and have the relevance and cardinality properties for $\mathcal{M} = \Phi^E : \Phi^M \Downarrow \psi_1^E$. Let σ be a grounded prefix substitution which unifies $\langle \mathcal{C}, \mathcal{W} \rangle$. Then for any $p \in LeafPos(\mathcal{M})$ there is a string s such that $\sigma(pre_{\mathcal{M}}(p)) = \psi_1^E.s$ holds.*

Corollary 1 and lemma 1 show how the prefix substitution guarantees a proper context management. Due to the definition of prefix substitutions the multi-set Φ^M in lemma 1 must be empty. This ensures that the rule ψ^E is applicable in the constructed position calculus proof.

Theorem 8 (Correctness). *If a matrix $\mathcal{M}$ is complementary for a set $\mathcal{C}$ of connections, a weakening map $\mathcal{W}$, and a substitution σ then there is a position calculus proof $\mathcal{P}$ for $\mathcal{M}$.*

Proof. Define the weight *of a matrix $\mathcal{M}$ by weighing the number of positions in $\mathcal{M}$ and use well-founded induction wrt. the weight of matrices. In the induction step we perform a complete case analysis of the structure of $\mathcal{M}$. The difficult cases where β or ψ^E must be applied is shown with the help of corollary 1 and lemma 1. The remaining proof is tedious but straightforward. Details can be found in [17].*

Completeness. For a Σ_{pos}-proof $\mathcal{P}$ of a matrix $\mathcal{M}$ we construct a set $ConSet(\mathcal{P})$ of connections, a weakening map $WeakMap(\mathcal{P})$, and a relation $\sqsubset_{\mathcal{P}} \subseteq SpecPos(\mathcal{M})^2$. The connections in $ConSet(\mathcal{P})$ are constructed from applications of *axiom* and τ in $\mathcal{P}$. If *axiom* is applied on $\Phi^E : \phi_1^M, \phi_2^M \Uparrow \cdot$ then $\{succ_1(\phi_1^M), succ_1(\phi_2^M)\}$ is in $ConSet(\mathcal{P})$. If τ is applied on a sequent $\Phi^E : \phi_1^M \Uparrow \cdot$ then $\{succ_1(\phi_1^M)\}$ is in $ConSet(\mathcal{P})$. $WeakMap(\mathcal{P})$ contains those elements of $WeakPos(\mathcal{M})$ which are explicitly weakened by ω or ν and those elements from $\Phi^E(\mathcal{M})$ which are implicitly weakened in *axiom* or τ. $\sqsubset_{\mathcal{P}}$ resembles the order in which special positions are reduced. We write $p \sqsubset_{\mathcal{P}} p'$ if p is reduced before p', i.e. the reduction occurs closer to the root of the proof tree $\mathcal{P}$. For any proof $\mathcal{P}$, $\sqsubset_{\mathcal{P}}$ is irreflexive, antisymmetric, and transitive, thus an ordering and $\ll$ (for $\mathcal{M}$) is a subordering of $\sqsubset_{\mathcal{P}}$.

Lemma 2. *Let $\mathcal{M}$ be a matrix, $\mathcal{P}$ be a position calculus proof for $\mathcal{M}$, and $p_1, p_2 \in (\Psi^M(\mathcal{M}) \cup \Psi^E(\mathcal{M}))$ be positions in $\mathcal{M}$ with $p_1 \neq p_2$. If there is a position $p \in SpecPos(\mathcal{M})$ with $p_1 \sqsubset_{\mathcal{P}} p$ and $p_2 \sqsubset_{\mathcal{P}} p$ then either $p_1 \sqsubset_{\mathcal{P}} p_2$ or $p_2 \sqsubset_{\mathcal{P}} p_1$ holds.*

We finally construct the substitution $\sigma_{\mathcal{P}}$ from $\sqsubset_{\mathcal{P}}$. Let $\phi \in (\Phi^M(\mathcal{M}) \cup \Phi^E(\mathcal{M}))$ be a variable special position in $\mathcal{M}$. We define $\sigma_{\mathcal{P}}(\phi) = Z = \psi_1 \dots \psi_n$ where the string $Z \in (\Psi^M(\mathcal{M}) \cup \Psi^E(\mathcal{M}))^*$ has the following properties.

- *sortedness:* $\psi_i \sqsubset_{\mathcal{P}} \psi_{i+1}$ holds for all $i \in \{1, \dots, n-1\}$.
- *prior reduction:* $\psi_i \sqsubset_{\mathcal{P}} \phi$ holds for all $i \in \{1, \dots, n\}$.
- *exclusivity:* $p \ll \phi \Rightarrow p \sqsubset_{\mathcal{P}} \psi_1$ holds for all $p \in SpecPos(\mathcal{M})$.
- *maximality:* For any $\psi \in \Psi^M(\mathcal{M}) \cup \Psi^E(\mathcal{M})$ not in Z with $\psi \sqsubset_{\mathcal{P}} \phi$ exists a $p \ll \phi$ with $\psi \sqsubset_{\mathcal{P}} p$.

According to the structure of Σ_{pos}-proofs, $\sigma_{\mathcal{P}}$ really is a prefix substitution. The construction guarantees that for any $\phi^M \in \Phi^M$ holds $\sigma_{\mathcal{P}} \in (\Phi^M(\mathcal{M}) \cup \Psi^M(\mathcal{M}))^*$.

Theorem 9 (Completeness). *Let $\mathcal{P}$ be a position calculus proof for a matrix $\mathcal{M}$. Then $\mathcal{M}$ is complementary for ConSet($\mathcal{P}$), WeakMap($\mathcal{P}$), and $\sigma_{\mathcal{P}}$.*
Proof. Each of the following properties is proven by induction over the structure of $\mathcal{P}$. (1) *ConSet*($\mathcal{P}$) spans $\mathcal{M}$. (2) $\sigma_{\mathcal{P}}$ is a *unifier* for $\langle \mathit{ConSet}(\mathcal{P}), \mathit{WeakMap}(\mathcal{P})\rangle$. (3) $\langle \mathit{ConSet}(\mathcal{P}), \mathit{WeakMap}(\mathcal{P})\rangle$ is linear, has the relevance property for $\mathcal{M}$, and has the cardinality property for $\mathcal{M}$. The individual proofs are lengthy because induction over $\mathcal{P}$ and case analysis is necessary. Details can be found in [17].

The Characterization. The characterization theorem proven in this section is the foundation for matrix based proof search methods. It yields a compactified representation of the search space which can be exploited by proof search methods in the same way as for other logics [20]. The method has been extended uniformly to multiplicative linear logic, as shown in [15]. Along the same lines an extension to $\mathcal{MELL}$ is possible.

Theorem 10 (Characterization Theorem). *A formula φ is valid in $\mathcal{MELL}$ if and only if the corresponding matrix is complementary for some multiplicity.*
Proof. Correctness follows from theorems 8, 6, 4, 2, and the correctness of Σ_1'. Completeness follows from theorems 9, 5, 3, 1, and the completeness of Σ_1'.

Example 4. Let $\mathcal{M}$ be the matrix for $\varphi = \langle (A \otimes !A) \invamp ?(A^\perp), + \rangle$ from figure 4. We choose $\mathcal{C} = \{\{a_{00000000}, a_{00010^20000}\}, \{a_{000001000}, a_{00010^10000}\}\}$, $\mathcal{W} = \emptyset$, and

$$\sigma = \{\phi^M_{0000}\backslash\varepsilon,\ \phi^M_{0000000}\backslash\psi^M_{00010^200},\ \phi^M_{00000100}\backslash\psi^M_{00010^100},\ \phi^E_{00010^1}\backslash\psi^E_{0000010},\ \phi^M_{00010^1000}\backslash\varepsilon,\ \phi^E_{00010^2}\backslash\psi^M_{000000},\ \phi^M_{00010^2000}\backslash\varepsilon\}\ .$$

Then $\mathcal{M}$ is complementary for $\mathcal{C}$, $\mathcal{W}$, and σ. Consequently φ is valid in $\mathcal{MELL}$.

6 Conclusion

We have presented a matrix characterization of logical validity for the full multiplicative exponential fragment of linear logic ($\mathcal{MELL}$). It extends our characterization for $\mathcal{MLL}$ [15] by the exponentials ? and ! and the multiplicative constants **1** and $\perp$. Our extension, as pointed out in [8], is by no means trivial and goes beyond all existing matrix characterizations for fragments of linear logic.

In the process we have also outlined a methodology for developing matrix characterizations from sequent calculi and for proving them correct and complete. It introduces a series of intermediate calculi, which step-wisely remove redundancies from sequent proofs while capturing their essential parts, and arrives at a matrix characterization as the most compact representation for proof search.

If applied to modal or intuitionistic logics, this methodology would essentially lead to Wallen's matrix characterization [24]. In order to capture the resource sensitivity of linear logic, however, we have introduced several refinements. The notion of multiplicities is based on positions instead of basic positions. Different types of special positions are used. The novel concept of weakening maps makes us able to deal with the aspects of resource management.

Fronhöfer has developed matrix characterizations for various variations of the multiplicative fragment [8]. Compared to his work for linear logic our characterization captures additionally the multiplicative constants and the controlled application of weakening and contraction. In fact, we are confident that our methodology will extend to further fragments of linear logic as well as to other resource sensitive logics, such as affine or relevant logics.

In the future we plan to extend our characterization to quantifiers, which again is a non-trivial problem although much is known about them in other logics. Furthermore, the development of *matrix systems* as a general theory of matrix characterizations has become possible. These systems would include a uniform framework for defining notions of complementarity and a methodology for supporting the proof of characterization theorems. Matrix systems might also enable us to integrate induction into connection-based theorem proving.

Matrix characterizations are known to be a foundation for efficient proof search procedures for classical, modal and intuitionistic logics [20] and $\mathcal{MLL}$ [15]. We expect that these proof procedures can now be extended to $\mathcal{MELL}$ and a wide spectrum of other logics, as soon as our methodology has led us to a matrix characterization for them.

Acknowledgements. We would like to thank Serge Autexier for his patience and his valuable comments while we were discussing the details of this work.

References

1. J.-M. Andreoli. Logic programming with focusing proofs in linear logic. *Journal of Logic and Computation*, 2(3):297–347, 1992.
2. P. Andrews. Theorem-Proving via General Matings. *Jour. of the ACM* 28 (2), 193–214, 1981.
3. W. Bibel. On matrices with connections. *Jour. of the ACM* 28, 633–645, 1981.
4. W. Bibel. A deductive solution for plan generation. *New Generation Computing* 4:115–132, 1986.
5. W. Bibel. *Automated Theorem Proving.* Vieweg, 1987.
6. I. Cervesato, J.S. Hodas, F. Pfenning. Efficient resource management for linear logic proof search. In *Extensions of Logic Programming*, LNAI 1050, pages 67–81. Springer, 1996.
7. V. Danos & L. Regnier. The structure of the multiplicatives. *Arch. Math. Logic* 28:181–203, 1989.
8. B. Fronhöfer. *The action-as-implication paradigm.* CS Press, 1996.
9. D. Galmiche. Connection methods in linear logic fragments and proof nets. Technical report, CADE–13 workshop on proof search in type-theoretic languages, 1996.
10. D. Galmiche & G. Perrier. On proof normalization in linear logic. *TCS*, 135:67–110, 1994.
11. V. Gehlot and C. Gunter. Normal process representatives. *Sixth Annual Symposium on Logic in Computer Science*, pages 200–207, 1991.
12. J.-Y. Girard. Linear logic. *TCS*, 50:1–102, 1987.
13. J. Harland and D. Pym. Resource-Distribution via Boolean Constraints. *14^{th} Conference on Automated Deduction*, LNCS 1249, pp. 222–236. Springer, 1997.
14. J.S. Hodas & D. Miller. Logic programming in a fragment of linear logic. *Journal of Information and Computation*, 110(2):327–365, 1994.
15. C. Kreitz, H. Mantel, J. Otten, S. Schmitt. Connection-Based Proof Construction in Linear Logic. *14^{th} Conference on Automated Deduction*, LNCS 1249, pp. 207–221. Springer, 1997.
16. P. Lincoln and T. Winkler. Constant-only multiplicative linear logic is NP-complete. *TCS*, 135:155-169, 1994.
17. H. Mantel. Developing a Matrix Characterization for $\mathcal{MELL}$. Technical Report, DFKI Saarbrücken, 1998. http://www.dfki.de/vse/staff/mantel/Papers/98tr-dev-mc-mell.ps.gz
18. M. Masseron, C. Tollu, J. Vauzeilles. Generating plans in linear logic. In *Foundations of Software Technology and Theoretical Computer Science*, LNCS, Springer,1991.
19. D. Miller. FORUM: A Multiple-Conclusion Specification Logic. *TCS*, 165(1):201-232, 1996.
20. J. Otten & C. Kreitz. A Uniform Proof Procedure for Classical and Non-classical Logics. *KI-96: Advances in Artificial Intelligence*, LNAI 1137, pp. 307–319. Springer, 1996.
21. S. Schmitt, C. Kreitz. Converting non-classical matrix proofs into sequent-style systems. *CADE–13*, LNAI 1104, pp. 418–432, Springer, 1996.
22. S. Schmitt, C. Kreitz. A uniform procedure for converting non-classical matrix proofs into sequent-style systems. *Journal of Information and Computation*, submitted.
23. T. Tammet. Proof strategies in linear logic. *Jour. of Automated Reasoning*, 12:273–304, 1994.
24. L. Wallen. *Automated deduction in nonclassical logic.* MIT Press, 1990.