LE MATEMATICHE
Vol. L (1995) – Fasc. I, pp. 99-118

DECIDING SET-THEORETIC FORMULAE WITH THE PREDICATE *FINITE* BY A TABLEAU CALCULUS

DOMENICO CANTONE - ROSA RUGGERI CANNATA

In this paper we give a decidable tableau calculus for the unquantified theory *MLSSF*, which involves in addition to the constructs of Multilevel Syllogistic, namely $\in$ (membership), $=$ (equality), $\subseteq$ (set inclusion), $\cup$ (binary union), $\cap$ (binary intersection), and $\setminus$ (set difference), also finite enumerations $\{\cdot, \cdots, \cdot\}$ and the predicate *Finite*.

The notions of $\mathcal{U}$*-hierarchy* and $\mathcal{U}$*-realization* of a graph w.r.t. given $\mathcal{U}$-sets, as well as some of their properties, are discussed and used to show the soundness and the completeness of the tableau calculus presented.

1. Introduction.

In the last few years, the decision problem for various classes of set-theoretic formulae has been studied very actively as part of a project aimed at the design and implementation of a set-theoretically based proof verifier with an *inferential core* comprising, among others, decision procedures for sublanguages of set theory (see [7]). Most theoretical results originated from such research have been collected in [3]. Some of these procedures have already been implemented using *ad hoc* techniques within the system ETNA, a set-theoretically based verification system under development at the University of Catania and New York University (cfr. [4]).

Entrato in Redazione il 27 aprile 1995.

Recently, in [6] we started an investigation aimed at the discovery of decidable tableaux calculi for solving the satisfiability problem for fragments of set theory and related areas.

The advantages of implementing decision procedures by means of decidable tableaux calculi over ad hoc methods are manyfold:

- tableaux naturally maintain information about proof attempts; such information can be used either to reconstruct proofs or, in case of formulae which are not theorems, to construct counter-examples (this can be particularly useful in didactic applications, such as in the use of computers for teaching logic or set theory courses);
- tableaux calculi can easily be extended by new rules, thus allowing, in favourable cases, smooth generalizations to more expressive decidable fragments;
- implementation of decidable tableaux calculi can quite easily be equipped with heuristics and various kinds of control to the user as, for instance, the possibility of temporarily deactivating some of the rules or of introducing new rules.

The search of decidable tableaux calculi continues in this paper, where in particular we will give a decidable tableau calculus for the unquantified theory *MLSSF*, which involves in addition to the constructs of Multilevel Syllogistic, namely $\in$ (membership), $=$ (equality), $\subseteq$ (set inclusion), $\cup$ (binary union), $\cap$ (binary intersection), and $\setminus$ (set difference), also finite enumerations $\{\cdot, \cdots, \cdot\}$ and the predicate *Finite*.

We recall that a decision procedure for *MLSSF* has already been given in [5], though along different lines. Moreover, [1] solves the satisfiability problem for the extension of *MLSSF* with rank and cardinality comparison and [2] solves the decision problem for a restricted quantified theory extending *MLSSF*.

2. Preliminaries.

After introducing the von Neumann standard hierarchy of sets, we define the notions of $\mathcal{U}$*-hierarchy* and of $\mathcal{U}$*-realization* of a graph w.r.t. given $\mathcal{U}$-sets. Finally, we relate these two concepts by proving some properties of realizations with urelements.

Such notions have been first introduced in [6], where it was shown how one can test formulae involving the iterated membership predicate $\in^+$ for the existence of models relative to a $\mathcal{U}$-hierarchy and then transform such models in others not involving urelements.

Here we will slightly modify the definition of realizations in order to take care of the new predicate *Finite*.

2.1. Hierarchies of sets, assignments, and models.

The satisfiability problem dealt with in the present paper refers to the von Neumann standard cumulative hierarchy $\mathcal{V}$ of sets defined by:

$$\begin{aligned} \mathcal{V}_0 &= \emptyset \\ \mathcal{V}_{\alpha+1} &= \mathcal{P}(\mathcal{V}_\alpha)\,, \quad \text{for each ordinal } \alpha, \\ \mathcal{V}_\lambda &= \textstyle\bigcup_{\mu<\lambda} \mathcal{V}_\mu\,, \text{ for each limit ordinal } \lambda, \end{aligned}$$

where $\mathcal{P}(S)$ is the powerset of S.

Let us introduce some basic terminology. The *rank* of a set s, denoted $rk\ s$, is the least ordinal α such that $s \subseteq \mathcal{V}_\alpha$, i.e. $s \in \mathcal{V}_{\alpha+1}$. By ω we denote the least infinite ordinal, namely the set of natural numbers. Sets having finite rank, i.e. belonging to $\mathcal{V}_\omega$, are said *hereditarily finite sets*. For every set s and every ordinal α, we define $s^{(\alpha)}$ by transfinite recursion as follows:

$$\begin{aligned} s^{(0)} &= s \\ s^{(\alpha+1)} &= \{s^{(\alpha)}\}\,, \quad \text{for each ordinal } \alpha, \\ s^{(\lambda)} &= \textstyle\bigcup_{\mu<\lambda} s^{(\mu)}\,, \text{ for each limit ordinal } \lambda. \end{aligned}$$

Then we have easily:

Lemma 2.1. *For any set s and ordinal α, $rk\ s^{(\alpha)} = rk\ s + \alpha$.*

A *(standard) assignment* over a collection of variables V is any map from V into $\mathcal{V}$. An assignment M is said to be *injective* if $Mx \neq My$ for any two distinct variables x, y. A set theoretic formula φ is said to be *satisfiable* by an assignment M over its variables if the formula resulting from φ by substituting in it sets Mx in place of free occurrences of x and by interpreting set theoretic operators and predicates according to their standard meaning is true. An assignment which satisfies a given formula φ is said to be a *model* for φ. A formula φ is said to be *injectively satisfiable* if it has a(n) injective model.

As a technical tool, it will result to be convenient to introduce a collection $\mathcal{U} = \{v_i : i \in I\}$ of special individuals which are not sets, called *urelements*. In such a case, we define the $\mathcal{U}$-hierarchy $\mathcal{V}^{\mathcal{U}}$ as follows:

$$\begin{aligned} \mathcal{V}^{\mathcal{U}}_0 &= \mathcal{U} \\ \mathcal{V}^{\mathcal{U}}_{\alpha+1} &= \mathcal{V}^{\mathcal{U}}_\alpha \cup \mathcal{P}(\mathcal{V}^{\mathcal{U}}_\alpha)\,, \text{ for each ordinal } \alpha, \\ \mathcal{V}^{\mathcal{U}}_\lambda &= \textstyle\bigcup_{\mu<\lambda} \mathcal{V}^{\mathcal{U}}_\mu\,, \quad \text{for each limit ordinal } \lambda. \end{aligned}$$

Notice that the standard von Neumann universe is a $\emptyset$-hierarchy. Elements of a $\mathcal{U}$-hierarchy will be called *$\mathcal{U}$-sets*. In particular, $\mathcal{U}$-sets not belonging to $\mathcal{U}$ will be said to be *proper*. $\mathcal{U}$-assignments, $\mathcal{U}$-models and the concept of (injective) $\mathcal{U}$-satisfiability can be defined in the most natural way.

2.2. Realizations.

Let $G = (N, \widehat{\in})$ be a *directed* acyclic graph and let (V, T) be a partition of N. Also, let (T_{Inf}, T_{Fin}) be a partition of T. We will make use of the following notation: for any $x \in N$, we put

$$G(x) = \{y \in N : y \widehat{\in} x\}.$$

For any given family $\{u_t : t \in T\}$ of non-empty proper $\mathcal{U}$-sets, with $\mathcal{U}$ a collection of urelements, we define the concept of $\mathcal{U}$-realization as follows.

Definition 2.1. *The $\mathcal{U}$-realization of G relative to $\{u_t : t \in T\}$ and to the partitions (V, T) and (T_{Inf}, T_{Fin}), is the $\mathcal{U}$-assignment $R^{\mathcal{U}}$ over $V \cup T$ recursively defined by:*

$$(1) \qquad \begin{array}{lll} R^{\mathcal{U}}x = \{R^{\mathcal{U}}z : z \in V \cup T \text{ and } z \widehat{\in} x\} \cup \bigcup_{t \widehat{\in} x \wedge t \in T_{Inf}} u_t & \text{for } x \text{ in } V; \\ R^{\mathcal{U}}t = \{R^{\mathcal{U}}z : z \in V \cup T \text{ and } z \widehat{\in} t\} \cup \{u_t\} & \text{for } t \text{ in } T. \end{array}$$

Observe that $R^{\mathcal{U}}$ is well defined by (1) above, since the graph G is acyclic. We can define a notion of height, for all $x \in V$, by putting

$$V\text{-}height(x) = \\ = \begin{cases} 0, & \text{if } y \widehat{\notin} x, \text{ for all } y \in V \\ \max\{V\text{-}height(y) : y \in V \wedge y \widehat{\in} x\} + 1, & \text{otherwise.} \end{cases}$$

Next we introduce the concept of V-extensionality.

Definition 2.2. *Let $G = (N, \widehat{\in})$ be a directed graph and let $V \subseteq N$. G is said to be V*-extensional *iff for any two distinct $v, v' \in V$ we have $G(v) \neq G(v')$, i.e. there exists $w \in N$ such that $w \widehat{\in} v \iff w \widehat{\notin} v'$.*

The following lemma states important properties of $\mathcal{U}$-realizations.

Lemma 2.2. *Let $G = (V \cup T, \widehat{\in})$ be a directed graph, with $V \cap T = \emptyset$. Also, let $\{u_t : t \in T\}$ be a family of $\mathcal{U}$-sets. Assume that*

(a) $\widehat{\in}$ is acyclic;

(b) G is V-extensional;

(c) $u_t \neq u_{t'}$ *and* $u_t \cap u_{t'} = \emptyset$*, for all distinct* $t, t' \in T$*;*

(d) $u_t \notin u_{t'}$*, for every* $t, t' \in T$*;*

(e) $u_t \neq R^{\mathcal{U}}v$ *and* $R^{\mathcal{U}}v \notin u_t$*, for all* $t \in T$ *and* $v \in V \cup T$*, where* $R^{\mathcal{U}}$ *is the* $\mathcal{U}$*-realization of* G *relative to* $\{u_t : t \in T\}$ *and to the partitions* (V, T) *and* (T_{Inf}, T_{Fin})*.*

Then

(i) $R^{\mathcal{U}}$ *is injective, i.e.* $R^{\mathcal{U}}v \neq R^{\mathcal{U}}v'$ *for all distinct* $v, v' \in V \cup T$*;*

(ii₁) $R^{\mathcal{U}}v \in R^{\mathcal{U}}v' \iff v \widehat{\in} v'$*, for all* $v, v' \in V \cup T$*;*

(ii₂) $R^{\mathcal{U}}x = R^{\mathcal{U}}y \cup R^{\mathcal{U}}z \iff G(x) = G(y) \cup G(z)$*, for* $x, y, z \in V$*;*

(ii₃) $R^{\mathcal{U}}x = R^{\mathcal{U}}y \cap R^{\mathcal{U}}z \iff G(x) = G(y) \cap G(z)$*, for* $x, y, z \in V$*;*

(ii₄) $R^{\mathcal{U}}x = R^{\mathcal{U}}y \setminus R^{\mathcal{U}}z \iff G(x) = G(y) \setminus G(z)$*, for* $x, y, z \in V$*;*

(ii₅) $R^{\mathcal{U}}x = \{R^{\mathcal{U}}y_1, \ldots, R^{\mathcal{U}}y_k\} \iff G(x) = \{y_1, \ldots, y_k\}$*, for* $x, y_1, \ldots, y_k \in V$*, with* $k \geq 0$*.*

Remark 2.1. Notice that we can always satisfy conditions (c), (d), and (e) by putting $u_t = \{\upsilon_t\}$, where the υ_t's are pairwise distinct urelements, for $t \in T$.

3. A decision procedure for *MLSSF*.

In this section we introduce the unquantified fragment of set theory denoted by *MLSSF* and subsequently we prove its decidability.

The language *MLSSF* contains:

(a) a denumerable infinity of individual variables $x, y, z, \ldots$;
(b) the predicate symbols $\in$, $=$, $\subseteq$, *Finite*;
(c) the operators $\cap$, $\cup$, $\setminus$, $\{\cdot\}$;
(d) the constant $\emptyset$ (empty set);
(e) parenthesis (to form compound terms);
(f) the logical connectives $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ (to form compound formulae).

It can easily be seen that for decidability purposes we can limit ourselves to considering only conjunctions of literals each of which has one of the following types (where the constant $\emptyset$ may occur).

$$\begin{array}{lllll} & x \in y, & x \notin y, & z = x \cup y, & z = x \cap y, \\ (2) & & & & \\ & z = x \setminus y, & x = \{y_1, \ldots, y_k\}, & Finite(x), & \neg Finite(x). \end{array}$$

We will call such formulae *normalized conjunctions of MLSSF*.

We characterize the meaning of the constant $\emptyset$ by assuming that any normalized conjunction of *MLSSF* contains the literal $\emptyset = \emptyset \setminus \emptyset$ and by treating then $\emptyset$ as another variable.

The following theorem expresses the injective satisfiability of *MLSSF* normalized conjunctions in terms of certain combinatorial conditions on realizations. Such characterization will then be turned into a decision procedure by further imposing an upper bound on the cardinality of the set of nodes T.

Theorem 3.1. *Let φ be a normalized conjunction of MLSSF with variables V. Then φ is injectively satisfiable in the standard universe of von Neumann (with no urelements) if and only if there exist:*

- *a finite set of nodes T disjoint from V,*
- *a partition (T_{Inf}, T_{Fin}) of T, and*
- *a directed acyclic graph $G = (N, \widehat{\in})$, with $N = V \cup T$,*

such that if $R^{\mathcal{U}}$ is a $\mathcal{U}$-realization of the graph G relative to given pairwise distinct singletons of urelements and to the partitions (V, T) and (T_{Inf}, T_{Fin}), then the following conditions are satisfied:

(a) $R^{\mathcal{U}}$ is an injective $\mathcal{U}$-model of all literals in φ not involving the predicate Finite;
(b) if Finite(x) occurs in φ then $t \widehat{\notin} x$, for all $t \in T_{inf}$.
(c) if $\neg$Finite(x) occurs in φ then $t \widehat{\in} x$, for some $t \in T_{Inf}$.

Proof. We first prove that the conditions of the theorem are sufficient. Thus, let φ be a normalized conjunction of literals of the above types involving the variables $V = \{x_1, \ldots, x_m\}$. Assume that we have

- a set $T = \{t_1, \ldots, t_r\}$ of nodes disjoint from V with a partition (T_{Inf}, T_{Fin}), and
- a directed acyclic graph $G = (N, \widehat{\in})$, with $N = V \cup T$,

such that conditions (a), (b), and (c) of the theorem are satisfied, with respect to a $\mathcal{U}$-realization $R^{\mathcal{U}}$ relative to given pairwise distinct singletons u_t of urelements, for $t \in T$.

For each $t_i \in T$, we define a set $\overline{u}_{t_i}$ (with no urelements) by putting

$$\overline{u}_{t_i} = \{\{i, \emptyset^{(\omega+j)}\} : j < \omega\}. \tag{3}$$

Since $\widehat{\in}$ is acyclic, we can give the following recursive definition

$$\begin{aligned} Mx &= \{Mz : z \in V \cup T \text{ and } z \widehat{\in} x\} \cup \textstyle\bigcup_{t \widehat{\in} x \wedge t \in T_{Inf}} \overline{u}_t && \text{for } x \text{ in } V; \\ Mt &= \{Mz : z \in V \cup T \text{ and } z \widehat{\in} t\} \cup \{\overline{u}_t\} && \text{for } t \text{ in } T. \end{aligned} \tag{4}$$

Notice that the realization M is an assignment in the standard universe of von Neumann with no urelements. In order to apply Lemma 2.2 to the realization M, we need to verify the following items:

(A) G is V-extensional;

(B) $\overline{u}_{t_i} \neq \overline{u}_{t_j}$, for all $i \neq j$, $i, j \in \{1, \ldots r\}$;
(C) $\overline{u}_t \notin \overline{u}_{t'}$, for every $t, t' \in T$;
(D) $\overline{u}_t \cap \overline{u}_{t'} = \emptyset$, for all distinct $t, t' \in T$;
(E) $\overline{u}_t \neq Mv$, for all $t \in T$ and $v \in V \cup T$;
(F) $Mv \notin \overline{u}_t$, for every $t \in T$ and $v \in V \cup T$.

Concerning (A), since by assumption $R^{\mathcal{U}}$ is injective, then for all distinct $v, v' \in V$ we have $R^{\mathcal{U}}v \neq R^{\mathcal{U}}v'$, from which $G(v) \neq G(v')$, proving the V-extensionality of G.
Concerning (B) and (D), observe that, by construction, for each $i = 1, \ldots, r$, the set $\overline{u}_{t_i}$ contains only elements of type $\{i, \emptyset^{(\omega+k)}\}$ and also that if $\{i, \emptyset^{(\omega+j)}\} = \{i', \emptyset^{(\omega+j')}\}$ then $i = i'$.
Concerning (C), it is enough to observe that all sets $\overline{u}_{t_i}$ have the same rank $\omega \cdot 2$.

Finally, in order to prove that (E) and (F) hold, it suffices to verify that for all $v \in V \cup T$ either $rk\ Mv > \omega \cdot 2$ or $rk\ Mv < \omega$. To this end, notice that if $Mt \in {}^{+}\{Mv\}$, for some t in T, then obviously $rk\ Mv > \omega \cdot 2$. On the other hand, if $Mt \notin {}^{+}\{Mv\}$, for all t in T, then by induction on *V-height*(v) it can easily be shown that $rk\ Mv < \omega$.

Thus, from Lemma 2.2, M is a model for all literals of φ not involving the predicate symbol *Finite*.
It remains to show that M also models correctly all literals in φ of type $Finite(x)$ and $\neg Finite(x)$. If $Finite(x)$ occurs in φ, then by (b) the set $\bigcup_{t \widehat{\in} x \wedge t \in T_{Inf}} \overline{u}_t$ is empty, so that Mx is finite. If $\neg Finite(x)$ occurs in φ then by (c) we have that there exists $t \in T_{Inf}$ such that $t \widehat{\in} x$. Since by definition $\overline{u}_t$ is infinite, it follows that Mx is infinite too.

This completes the sufficiency part of the proof of Theorem 3.1.

Next we prove that the conditions stated in Theorem 3.1 are also necessary. We will need the following definition and lemma.

Definition 3.1. *Given an injective assignment M over a collection of variables N, the* membership graph of M relative to N *is the graph $G_M = (N, \widehat{\in}_M)$, where $\widehat{\in}_M$ is the relation over N defined by*

$$w \widehat{\in}_M v \iff Mw \in Mv, \quad \textit{for all } v, w \in N.$$

Lemma 3.1. *Let M be an injective assignment over a collection N of variables and let $G_M = (N, \widehat{\in}_M)$ be its membership graph relative to N. Then*

(a) $\widehat{\in}_M$ is acyclic;
(b) if $Mx = My \cup Mz$ then $G(x) = G(y) \cup G(z)$, for $x, y, z \in V$;
(c) if $Mx = My \cap Mz$ then $G(x) = G(y) \cap G(z)$, for $x, y, z \in V$;

(d) *if* $Mx = My \setminus Mz$ *then* $G(x) = G(y) \setminus G(z)$, *for* $x, y, z \in V$;
(e) *if* $Mx = \{My_1, \dots, My_k\}$ *then* $G(x) = \{y_1, \dots, y_k\}$, *for* $x, y_1, \dots, y_k \in N$, *with* $k \geq 0$.

Proof. (a) is an immediate consequence of the definition of $\widehat{\in}_M$ and the acyclicity of the membership relation $\in$. Properties (b), (c), (d), and (e) follow easily from the definition of $\widehat{\in}_M$. □

Now let us assume that the normalized *MLSSF* conjunction φ is satisfied by a given injective model M and let $V = \{x_1, \dots, x_m\}$ be the collection of variables occurring in φ.

We begin by defining M also over a new collection T of variables.

In what follows, we say that a set Σ *distinguishes* a collection S if $\Sigma \cap s \neq \Sigma \cap s'$, for any two distinct $s, s' \in S$. Consider the following procedure.

```
Procedure 1
    Σ1 := ∅;
    Done := ∅;
    for v ∈ V do
        if Σ1 does not distinguish {Mv' : v' ∈ Done}∪{Mv} then
            let v' ∈ Done such that Mv' ∩ Σ1 = Mv ∩ Σ1;
            pick d ∈ (Mv' \ Mv) ∪ (Mv \ Mv');
            Σ1 := Σ1 ∪ {d};
            Done := Done ∪ {v};
        end if;
    end for;
    return Σ1;
end procedure 1
```

Let Σ_1 be the set returned by Procedure 1. By induction on the number of iterations of the for-loop, it can easily be shown that, at termination, the set Σ_1 distinguishes $\{Mv : v \in V\}$, i.e. $((Mv \setminus Mv') \cup (Mv' \setminus Mv)) \cap \Sigma_1 \neq \emptyset$, for all distinct $v, v' \in V$. Moreover, $|\Sigma_1| \leq |V| - 1$.

Let $\{\sigma_1, \dots, \sigma_k\} = \Sigma_1 \setminus \{Mv : v \in V\}$.

Next, consider the following procedure.

```
Procedure 2
    Σ2 := ∅;
    for w ∈ V do
        if Mw is infinite then
            pick σ ∈ Mw\(Σ1 ∪ ⋃{Mv : v ∈ V and Mv is finite});
            Σ2 := Σ2 ∪ {σ};
        end if;
    end for;
    return Σ2;
end procedure 2
```

Let Σ_2 be the set returned by Procedure 2. Obviously, $\Sigma_1 \cap \Sigma_2 = \emptyset$ and $|\Sigma_2| \leq |V|$. Moreover, if $\neg Finite(x)$ occurs in φ then $Mx \cap \Sigma_2 \neq \emptyset$ and if $Finite(x)$ occurs in φ then $Mx \cap \Sigma_2 = \emptyset$.

Let $\{\sigma_{k+1}, \dots, \sigma_r\} = \Sigma_2$. In correspondence with each σ_i, for $i = 1, \dots, r$, we introduce a new variable t_i and extend M over the newly introduced variables by putting

$$Mt_i = \sigma_i, \quad \text{for} \quad i = 1, \dots, r.$$

Put also $T = \{t_1, \dots, t_r\}$, $T_{Fin} = \{t_1, \dots, t_k\}$, and $T_{Inf} = \{t_{k+1}, \dots, t_r\}$.

Let now $G_M = (V \cup T, \widehat{\in}_M)$ be the membership graph of M relative to $V \cup T$. Plainly, by construction, conditions (b) and (c) of Theorem 3.1 are satisfied. Moreover, by Lemma 3.1 (a), G_M is acyclic.

Let $R^{\mathcal{U}}$ be a $\mathcal{U}$-realization of the graph G_M relative to given pairwise distinct singletons of urelements and to the partitions (V, T) and (T_{Inf}, T_{Fin}). In order to complete the proof of the theorem we need to verify that $R^{\mathcal{U}}$ is an injective $\mathcal{U}$-model of all literals in φ not involving the predicate symbol *Finite*. We will accomplish this by verifying the hypotheses of Lemma 2.2.

We already have that $\widehat{\in}_M$ is acyclic. Let us show that G_M is V-extensional. If $v, v' \in V$ are two distinct variables, then, by the construction carried out in Procedure 1, there exists $w \in V \cup T$ such that $Mw \in ((Mv \setminus Mv') \cup (Mv' \setminus Mv))$. Hence $w \widehat{\in}_M v \iff w \widehat{\notin}_M v'$, proving the V-extensionality of G_M. The remaining conditions of Lemma 2.2 follow immediately from the particular choice of the $\mathcal{U}$-sets involved in the definition of $R^{\mathcal{U}}$, as already observed in Remark 2.1. Hence $R^{\mathcal{U}}$ is an injective $\mathcal{U}$-assignment. Moreover, since M satisfies φ, then Lemma 3.1 and (ii$_1$)–(ii$_5$) of Lemma 2.2 yield that $R^{\mathcal{U}}$ satisfies all literals of φ not involving the predicate symbol *Finite*.

This completes the proof of Theorem 3.1. □

As a by-product of the preceding proof, we also have that the set T satisfies the following cardinality relation

$$|T| < 2|V|.$$

Thus we can state the following

Corollary 3.1. *Theorem 3.1 continues to hold even if the additional restriction* $|T| < 2|V|$ *is required.*

The preceding corollary implies trivially the following theorem.

Theorem 3.2. *The satisfiability problem for normalized conjunctions of MLSSF is NP-complete.*

Notice that Theorem 3.1 can be easily generalized to handle the ordinary satisfiability problem also for *extended normalized conjunctions* of *MLSSF* which, in addition to clauses listed in (2), can involve clauses of types $x = y$ and $x \neq y$ too. This is stated in the following theorem, whose proof is immediate.

Theorem 3.3. *Let φ be an extended normalized conjunction of MLSSF. Then φ is satisfiable if and only if there exists an equivalence relation $\sim$ over the variables of φ such that*

- *$\sim$ agrees with clauses in φ of types $x = y$ and $x \neq y$, i.e. if $x = y$ (resp. $x \neq y$) is in φ then $x \sim y$ (resp. $x \not\sim y$);*
- *$\widetilde{\varphi}$ is injectively satisfiable, where $\widetilde{\varphi}$ denotes the result of identifying in φ equivalent variables.*

By combining Theorems 3.2 and 3.3, we obtain

Corollary 3.2. *The satisfiability problem for extended normalized conjunctions of MLSSF is NP-complete.*

4. Hintikka sets for *MLSSF*.

Let H be a set of literals of the following types:

$$(5)\quad \begin{array}{lllll} z = x \cup y, & z = x \cap y, & z = x \setminus y, & x = \{y_1, \dots, y_k\}, & x = y, \\ x \neq y, & x \in y, & x \notin y, & \mathit{Finite}(x), & \neg \mathit{Finite}(x) \end{array}$$

(extended *MLSSF* literals) containing the literal $\emptyset = \emptyset \setminus \emptyset$, let (V, T) be a partition of the variables occurring in H such that $\emptyset \in V$, and let (T_{Inf}, T_{Fin}) be a partition of T.

Assume that $V \cup T$ is linearly ordered by $\prec$ in such a way that

$$(6)\qquad v \prec t \ \text{ for all } v \in V\,,\ t \in T \ \text{ and } \ \emptyset = \min_{\prec} V\,.$$

Let $\sim_H$ be the minimal equivalence relation among variables induced by the literals of type $x = y$ in H. For each variable x occurring in H put

$$repr_H(x) = \min\{y : y \sim_H x\}\,,$$

where the minimum is taken with respect to the given ordering $\prec$ [1]. Extend homomorphically the map $repr_H$ to formulae, sets of formulae, and sets of variables, and put accordingly

$$\widetilde{H} = repr_H(H), \quad \widetilde{V} = repr_H(V), \quad \widetilde{T} = repr_H(T).$$

Also, put

$$\widetilde{T}' = \widetilde{T} \setminus \widetilde{V}, \quad \widetilde{T}'_{Fin} = \widetilde{T}_{Fin} \setminus \widetilde{V}, \quad \widetilde{T}'_{Inf} = \widetilde{T}_{Inf} \setminus \widetilde{V}.$$

Notice that if a $\sim_H$-class contains any variable from V, its representative will certainly be an element in V.

We now associate a graph $G_H = (\widetilde{V} \cup \widetilde{T}', \widehat{\in}_H)$ to H as follows.

Definition 4.1. *Let H, (V, T), $\widetilde{H}$, $\widetilde{V}$, $\widetilde{T}$ be as above. The* canonical graph *of H relative to (V, T) is the directed graph $G_H = (\widetilde{V} \cup \widetilde{T}', \widehat{\in}_H)$, where $x \widehat{\in}_H y$ holds whenever the literal $x \in y$ is in $\widetilde{H}$.*

Next we define the notions of *V-based MLSSF* set and of *MLSSF*-Hintikka set relative to a given partition (V, T) of variables.

Definition 4.2. *A set H of extended MLSSF literals is said to be V-*based, *where V is a set of variables such that $\emptyset \in V$, if all variables occurring in literals of type $x = y \cup y'$, $x = y \cap y'$, $x = y \setminus y'$, $y = \{x_1, \ldots, x_k\}$ (with $k \geq 1$), $\neg Finite(x)$ present in H are in V.*

Definition 4.3. *Let H be a set of extended MLSSF literals of type (5), let (V, T) be a partition of the variables in H, and let (T_{Inf}, T_{Fin}) be the partition of T defined by*

$$T_{Fin} = \{t \in T : \text{ the literals } t \in v \text{ and } Finite(v) \text{ are in } H, \text{ for some } v \in V\},$$

$$T_{Inf} = T \setminus T_{Fin}.$$

*Then H is said to be an MLSSF-*Hintikka set relative to *(V, T) if the following* saturation rules *are satisfied:*

S1. if $z = y \cup y'$ and $x \in y$ are in H, then $x \in z$ is in H;
S2. if $z = y \cup y'$ and $x \in y'$ are in H, then $x \in z$ is in H;
S3. if $z = y \cup y'$ and $x \in z$ are in H, then either $x \in y$ or $x \in y'$ is in H;

(1) In what follows, given a partition of type (V, T), we will always assume that $\emptyset \in V$ and that there is also an ordering relation $\prec$ satisfying condition (6), even if this is not explicitly stated.

S4. *if* $z = y \cap y'$ *and* $x \in z$ *are in* H, *then* $x \in y$ *and* $x \in y'$ *are in* H;
S5. *if* $z = y \cap y'$, $x \in y$, *and* $x \in y'$ *are in* H, *then* $x \in z$ *is in* H;
S6. *if* $z = y \setminus y'$, *and* $x \in z$ *are in* H, *then* $x \in y$ *and* $x \notin y'$ *are in* H;
S7. *if* $z = y \setminus y'$, $x \in y$, *and* $x \notin y'$ *are in* H, *then* $x \in z$ *is in* H;
S8. *if* $y = \{x_1, \ldots, x_k\}$ *is in* H *for some* $k \geq 1$, *then* $x_i \in y$ *is in* H *for all* $i \in \{1, \ldots, k\}$;
S9. *if* $y = \{x_1, \ldots, x_k\}$ *and* $z \in y$ *are in* H *for some* $k \geq 1$, *then* $z = x_i$ *is in* H *for some* $i \in \{1, \ldots, k\}$;
S10. *if* $x = y$ *and* φ *are in* H, *then* φ^y_x *and* φ^x_y *are in* H (2);
S11. *for each unordered pair of distinct variables* x, y *in* $V \cup T$, *either* $x = y$ *is in* H, *or* $x \neq y$ *is in* H;
S12. *for each unordered pair of distinct variables* x, y, *with* x *in* $V \cup T$ *and* y *in* T, *either* $x \in y$ *is in* H, *or* $x \notin y$ *is in* H;
S13. *for each* $x \neq y$ *in* H, *with* x, y *in* V, *either* $w \in x$ *and* $w \notin y$ *are in* H, *or* $w \notin x$ *and* $w \in y$ *are in* H, *for some variable* w *in* $V \cup T$;
S14. *if* $\neg Finite(z)$ *is in* H, *then there exists* $w_z \in \widetilde{T}'_{Inf}$ *such that* $w_z \in z$ *is in* H.

Remark. By saturation rule S10, for every *MLSSF*-Hintikka set H we have always $\widetilde{H} \subseteq H$.

Lemma 4.1. *Let* H *be an MLSSF-Hintikka set relative to* (V, T) *and let* G_H *be its corresponding canonical graph. We have:*

(a) *if no membership chain* $x \in x_1 \in \cdots \in x_n \in x$ *(for any* $n \geq 0$*) is in* $\widetilde{H}$, *then* G_H *is acyclic;*
(b) *if* $\widetilde{H}$ *does not contain any pair of complementary literals then* G_H *is* $\widetilde{V}$*-extensional.*

Proof. Concerning (a), clearly to any cycle $x \mathrel{\widehat{\in}}_H x_1 \cdots \mathrel{\widehat{\in}}_H x_n \mathrel{\widehat{\in}}_H x$ in G_H there corresponds a cycle $x \in x_1 \in \cdots \in x_n \in x$ of membership relations in $\widetilde{H}$.

Concerning (b), assume that $\widetilde{H}$ does not contain pairs of complementary literals. Let x, y be two distinct variables in $\widetilde{V}$. By saturation rule S11, $x \neq y$ or $y \neq x$ occurs in $\widetilde{H}$, so that by saturation rule S13 there exists a variable $w \in \widetilde{V} \cup \widetilde{T}'$ such that $w \in x$ and $w \notin y$ are in $\widetilde{H}$ or $w \notin x$ and $w \in y$ are in $\widetilde{H}$. Observe that, by hypothesis, if $w \notin y$ is in $\widetilde{H}$, then $w \in y$ cannot be in $\widetilde{H}$, so that $w \mathrel{\widehat{\notin}}_H y$. Thus $w \mathrel{\widehat{\in}}_H x$ if and only if $w \mathrel{\widehat{\notin}}_H y$, proving the $\widetilde{V}$-extensionality of G_H. □

Next we introduce the concept of *consistent* *MLSSF*-Hintikka set.

---

(2) φ^y_x denotes the result of substituting in φ all occurrences of y by x.

Definition 4.4. *An MLSSF-Hintikka set H relative to* (V, T) *is said to be* consistent *if the following consistency rules hold:*

C1. H contains no complementary literals X and $\neg X$*, and*

C2. H contains no semantic contradictions of type $x \neq x$*, or* $x \in x_1 \in \cdots \in x_n \in x$*, for* $n \geq 0$.

We now have the following theorem.

Theorem 4.1. *Every consistent MLSSF-Hintikka set H relative to a partition* (V, T) *such that* $\widetilde{H}$ *is* $\widetilde{V}$*-based is satisfiable.*

Proof. Let H be as in the hypothesis and let $G_H = (\widetilde{V} \cup \widetilde{T}', \widehat{\in}_H)$ be its canonical graph. Notice that from Lemma 4.1, G_H is acyclic and $\widetilde{V}$-extensional.

We prove that $\widetilde{H}$ is injectively satisfiable in the standard universe of von Neumann with no urelements by applying Theorem 3.1. So let $R^{\mathcal{U}}$ be the $\mathcal{U}$-realization of G_H relative to $(\widetilde{V}, \widetilde{T}')$, to the partition $(\widetilde{T}'_{Inf}, \widetilde{T}'_{Fin})$ of T introduced in Definition 4.3, and to given pairwise distinct singletons of urelements.

We need to show that $R^{\mathcal{U}}$ injectively satisfies all literals in H not involving the predicate *Finite* and that conditions (b) and (c) of Theorem 3.1 are satisfied. By Lemma 2.2(i), $R^{\mathcal{U}}$ is injective. Moreover, from saturation rules S1 – S14, we have easily:

- $G_H(x) = G_H(y) \cup G_H(z)$, if $x = y \cup z$ occurs in $\widetilde{H}$;
- $G_H(x) = G_H(y) \cap G_H(z)$, if $x = y \cap z$ occurs in $\widetilde{H}$;
- $G_H(x) = G_H(y) \setminus G_H(z)$, if $x = y \setminus z$ occurs in $\widetilde{H}$;
- $G_H(x) = \{y_1, \ldots, y_k\}$, if $x = \{y_1, \ldots, y_k\}$ occurs in $\widetilde{H}$.

Therefore, from (ii_2)–(ii_5) of Lemma 2.2 and the $\widetilde{V}$-basedness of $\widetilde{H}$, $R^{\mathcal{U}}$ satisfies all clauses in $\widetilde{H}$ of type $x = y \cup z$, $x = y \cap z$, $x = y \setminus z$, and $x = \{y_1, \ldots, y_k\}$. Moreover, from (ii_1) of Lemma 2.2 it follows that also clauses in $\widetilde{H}$ of type $x \in y$, $x \notin y$, are satisfied by $R^{\mathcal{U}}$.

Finally, by consistency of H and by saturation rule S14, if $\neg Finite(z)$ is in H, then $w_z \widehat{\in}_H z$, for some $w_z \in \widetilde{T}'_{Inf}$. Moreover, by the definition of T_{Inf} and T_{Fin}, if $Finite(x)$ is in H, then $t \widehat{\notin}_H x$, for all $t \in \widetilde{T}'_{Inf}$, proving respectively conditions (b) and (c) of Theorem 3.1. Thus, from the same theorem it follows that there exists an injective model M of $\widetilde{H}$ in the standard universe of von Neumann. This model can obviously be extended to a model M' of H by putting

$$Mx = M(repr_H(x)), \quad \text{for } x \in V \cup T,$$

thus completing the proof of our theorem. □

5. A decidable tableau calculus for *MLSSF*.

In this section we provide a tableau calculus for collections of *MLSSF* literals in the style of Smullyan's analytic tableaux (cfr. [8]).

Based on the decision test for *MLSSF* given in the preceding sections and on an effective way to construct *complete* tableaux, it will turn out that the tableau calculus for *MLSSF* is decidable and can be used as a system which provides both counter-examples and proofs.

The rules of our tableau calculus for *MLSSF* are those listed in Table 1.

We begin with some definitions and terminology.

Definition 5.1. *Let $\mathcal{S}$ be a finite collection of MLSSF literals and let $V_{\mathcal{S}}$ be the set of variables occurring in $\mathcal{S}$. An* initial *MLSSF*-tableau *for $\mathcal{S}$ is a one-branch tree whose nodes are labeled by the literals in $\mathcal{S}$.*

An MLSSF-tableau *for $\mathcal{S}$ is a tableau labeled with MLSSF literals which can be constructed from the initial tableau for $\mathcal{S}$ by a finite number of applications of the rules (T1) – (T15) of Table 1, where rule (T13) can be applied only to literals of type $x \neq y$, with $x, y \in V_{\mathcal{S}}$, and rule (T14) only to variables in $V_{\mathcal{S}}$.*

Definition 5.2. *Let $\mathcal{T}$ be an MLSSF-tableau for a given finite collection $\mathcal{S}$ of MLSSF literals, and let $V_{\mathcal{S}}$ be the set of variables occurring in $\mathcal{S}$.*

A branch ϑ of $\mathcal{T}$ is said to be

- strict, *if no rule is used more than once on ϑ with the same literals;*
- $V_{\mathcal{S}}$-saturated, *if the collection H_{ϑ} of literals in ϑ forms an MLSSF-Hintikka set relative to a partition $(V_{\mathcal{S}}, T)$, for some set T of variables;*
- $V_{\mathcal{S}}$-closed, *if either it contains a set of literals of the form $x \in x_1 \in \cdots \in x_n \in x$, for some variables $x, x_1, \ldots, x_n$, with $n \geq 0$, or it contains a pair of complementary literals X, $\neg X$, or it contains a literal of type $x \neq x$, for some variable x;*
- $V_{\mathcal{S}}$-complete, *if either it is* $V_{\mathcal{S}}$-saturated, *or it is* $V_{\mathcal{S}}$-closed;
- satisfiable, *if there exists a model for the literals occurring on ϑ.*

$\mathcal{T}$ is said to be

- strict, *or* $V_{\mathcal{S}}$-saturated, *or* $V_{\mathcal{S}}$-closed, *or* $V_{\mathcal{S}}$-complete, *if such are all its branches;*
- satisfiable, *if one of its branches is satisfiable.*

In order to prove the soundness and completeness of our tableau calculus, we need to show that there is an effective procedure which generates a strict $V_{\mathcal{S}}$-complete tableau for any finite collection $\mathcal{S}$ of *MLSSF* literals.

$$(T1)\quad \frac{\begin{array}{c} z = y \cup y' \\ x \in y \end{array}}{x \in z} \qquad (T2)\quad \frac{\begin{array}{c} z = y \cup y' \\ x \in y' \end{array}}{x \in z} \qquad (T3)\quad \frac{\begin{array}{c} z = y \cup y' \\ x \in y \end{array}}{x \in y \mid x \in y'}$$

$$(T4)\quad \frac{\begin{array}{c} z = y \cap y' \\ x \in z \end{array}}{\begin{array}{c} x \in y \\ x \in y' \end{array}} \qquad (T5)\quad \frac{\begin{array}{c} z = y \cap y' \\ x \in y \\ x \in y' \end{array}}{x \in z} \qquad (T6)\quad \frac{\begin{array}{c} z = y \setminus y' \\ x \in z \end{array}}{\begin{array}{c} x \in y \\ x \notin y' \end{array}}$$

$$(T7)\quad \frac{\begin{array}{c} z = y \setminus y' \\ x \in y \\ x \notin y' \end{array}}{x \in z} \qquad (T8)\quad \frac{y = \{x_1, \dots, x_k\}}{\begin{array}{c} x_1 \in y \\ \vdots \\ x_k \in y \end{array}} \qquad (T9)\quad \frac{\begin{array}{c} y = \{x_1, \dots, x_k\} \\ z \in y \end{array}}{z = x_1 \mid \cdots \mid z = x_k}$$

$$(T10)\quad \frac{\begin{array}{c} x = y \\ \varphi \end{array}}{\begin{array}{c} \varphi^y_x \\ \varphi^x_y \end{array}} \qquad (T11)\quad \frac{}{x = y \mid x \neq y} \qquad (T12)\quad \frac{}{x \in y \mid x \notin y}$$

$$(T13)\quad \frac{x \neq y}{\begin{array}{c} w \in x \\ w \notin y \end{array} \;\Bigg|\; \begin{array}{c} w \notin x \\ w \in y \end{array}} \qquad (T14)\quad \frac{}{Finite(x) \mid \neg Finite(x)} \qquad (T15)\quad \frac{\begin{array}{c} \neg Finite(z) \\ Finite(y_1) \\ \vdots \\ Finite(y_k) \end{array}}{\begin{array}{c} w_z \in z \\ w_z \notin y_1 \\ \vdots \\ w_z \notin y_k \\ w_z \neq x_1 \\ \vdots \\ w_z \neq x_m \end{array}}$$

Table 1: Tableaux rules for *MLSSF*

Lemma 5.1. *For any finite collection $\mathcal{S}$ of MLSSF literals, the following procedure Complete$(\cdot)$ constructs a strict $V_\mathcal{S}$-complete tableau for $\mathcal{S}$, when called with input $\mathcal{S}$.*

```
Procedure Complete(S);
      Comment: S is a finite collection of MLSSF literals.
      V_S := set of variables occurring in S;
      T := initial tableau for S;
      PartialSaturation (T, T);
      while there exists a non-closed branch τ in T
            such that ¬Finite(z) occurs in τ, for some variable z ∈ V_S,
            and on which rule (T15) has not yet been applied to the literal ¬Finite(z) do
                  pick such a branch τ and variable z;
                  ApplyRuleT15(τ, z);
end procedure.
```

```
Procedure PartialSaturation(T, ϑ);
      apply strictly and in all possible ways rules (T1) – (T14) on all (new) branches
      of T which extend ϑ, applying rule (T13) only to literals of type x ≠ y,
      with x, y ∈ V_S and rule (T14) only to variables in V_S;
end procedure.
```

```
Procedure ApplyRuleT15(τ, z);
      Comment: τ is a non-closed branch, z is a variable in V_S,
                  and ¬Finite(z) occurs in τ.
      add to the branch τ the following formulae
      w_z ∈ z, w_z ∉ y_1, ..., w_z ∉ y_k, w_z ≠ x_1, ..., w_z ≠ x_m,
      where {y_1, ..., y_k} = {v ∈ V_S : Finite(v) is in τ} and {x_1, ..., x_m} = V_S;
      PartialSaturation(T, τ);
end procedure.
```

Proof. Let $V_\mathcal{S}$ be the set of variables occurring in $\mathcal{S}$. It is enough to verify that the call *Complete*$(\mathcal{S})$ to the above procedure builds a strict $V_\mathcal{S}$-complete tableau $\mathcal{T}_\mathcal{S}$ for $\mathcal{S}$ in a finite number of steps.

Concerning termination, one needs to observe that the branching factor of $\mathcal{T}_\mathcal{S}$ is bounded by $|V_\mathcal{S}|$ and that each branch cannot involve more than $\mathcal{O}(|V_\mathcal{S}|)$ newly introduced variables.

To show $V_\mathcal{S}$-completeness, let ϑ be a branch of $\mathcal{T}_\mathcal{S}$ and let H_ϑ be the collection of literals on ϑ. Assume that ϑ is not $V_\mathcal{S}$-closed. Then we need to prove that ϑ is $V_\mathcal{S}$-saturated.

Saturation w.r.t. rules S1 – S13 follows from the repeated calls to procedure $PartialSaturation(\cdot,\cdot)$, either from $Complete(\cdot)$ or from $ApplyRuleT15(\cdot,\cdot)$.

To show also saturation w.r.t. rule S14, let z be a variable in V_S for which the literal $\neg Finite(z)$ belongs to H_ϑ. Then in the *while-loop* of procedure $Complete(\cdot)$, a call is made to $ApplyRuleT15(\vartheta', z)$, for some initial segment ϑ' of ϑ. Such a call will introduce in ϑ the following literals

(7) $$w_z \in z\,,$$

(8) $$w_z \notin y_1, \ldots, w_z \notin y_k\,,$$

(9) $$w_z \neq x_1, \ldots, w_z \neq x_m\,,$$

where $y_1, \ldots, y_k$ are all the variables y in V_S for which $Finite(y)$ is in ϑ' (and therefore in ϑ) and $x_1, \ldots, x_m$ are all the variables of V_S.

Without loss of generality, we can assume that $w_z = repr_{\widetilde{H_\vartheta}}(w_z)$. The inequalities (9) imply at once $w_z \notin V_S$. Thus, to show that $w_z \in \widetilde{T}'_{Inf}$, namely that $w_z \in \widetilde{T}_{Inf} \setminus \widetilde{V}_S$, it is enough to prove that $w_z \in T_{Inf}$. But if $w_z \in T_{Fin}$, then the literals $w_z \in v$ and $Finite(v)$ would be in H_ϑ, for some $v \in V_S$, which is impossible because of the literals (8).

Finally, notice that strictness is enforced at each step. □

In view of the preceding result, the following theorem entails the soundness and completeness of our tableau calculus.

Theorem 5.1. (Soundness and completness). *Let S be a finite collection of MLSSF literals involving the variables V_S and let $\mathcal{T}$ be a V_S-complete tableau for S, relative to the partition (V_S, T), for some set T of new variables. Then S is satisfiable if and only if $\mathcal{T}$ is not V_S-closed.*

Proof. Plainly, rules (T1) – (T14) of Table 1 are sound, as can easily be verified. Concerning rule (T15), notice that if M is any model satisying the literals

$$\neg Finite(z),\ Finite(y_1),\ \cdots,\ Finite(y_k)\,,$$

then obviously

$$Mz \setminus ((My_1 \cup \cdots \cup My_k) \cup \{Mx_1, \ldots, Mx_m\}) \neq \emptyset\,,$$

where $\{x_1, \ldots, x_m\} = V_S$. Therefore M can be extended over a new variable w_z in such a way as to satisfy also the literals

$$w_z \in z,\ w_z \notin y_1,\ \cdots,\ w_z \notin y_k,\ w_z \neq x_1,\ \cdots,\!:\ w_z \neq x_m\,.$$

Assume now that $\mathcal{S}$ is satisfiable. The soundness of rules (T1) – (T15) yields that the tableau $\mathcal{T}$ is satisfiable. Let ϑ be a satisfiable branch of $\mathcal{T}$. Then, obviously, ϑ (and in turn $\mathcal{T}$) is not $V_{\mathcal{S}}$-closed.

On the other hand, if $\mathcal{T}$ is not $V_{\mathcal{S}}$-closed, then it has a non-$V_{\mathcal{S}}$-closed branch ϑ. It is easy to see that the collection H_ϑ of the literals occurring on ϑ forms a consistent *MLSSF*-Hintikka set relative to the partition $(V_{\mathcal{S}}, T)$, for a set T of new variables. Since H_ϑ is $\widetilde{V}$-based, from Theorem 4.1 it follows that H_ϑ is satisfiable and, *a fortiori*, $\mathcal{S}$ is satisfiable too. □

Lemma 5.1 and Theorem 5.1 yield readily that by coupling the procedure *Complete*$(\cdot)$ with a closure test, one obtains a decision procedure for normalized extended conjunctions of *MLSSF*. But, needless to say, this approach constitutes just the skeleton of a decision procedure, which can be enriched with strategies and heuristics to achieve efficiency improvements.

6. Some additional rules for *MLSSF*.

A heuristic which allows to shorten proofs consists in extending our initial tableau rules (T1) – (T15) with new, more powerful ones, which allow to concentrate in a single step multiple applications of the initial rules. This is exemplified by the new rules in Table 2, which allow to avoid some unnecessary splittings.

It is an easy matter to check that the new rules (T16) – (T22) are consequences of our initial rules (T1) – (T15).

7. Conclusions.

We have presented a decision procedure for the extension of *MLS* with finite enumerations and the predicate *Finite*. We adapted such a decision procedure in the form of a decidable tableau calculus which we proved to be both sound and complete. A simple heuristic consisting in extending our tableau calculus with new rules able to avoid some unnecessary splittings has been presented.

Future work will include the implementation of the decision procedure for *MLSSF* in the form of a tableau calculus within the system ETNA. We also plan to develop and implement decidable tableaux calculi for other extensions of *MLS* involving various combinations of the following set constructs: *Pow* (powerset), *Un* (unionset), $\times$ (cartesian product), η (choice function), $\bigcap$ (unary intersection), etc. (cfr.[3]). Additionally, we intend to look for suitable heuristics and to investigate the possibility to enrich (semi-)automatically a given tableau

$$\text{(T16)}\ \frac{}{Finite(\emptyset)} \qquad \text{(T17)}\ \frac{z = y \cup y' \quad Finite(z)}{Finite(y) \quad Finite(y')} \qquad \text{(T18)}\ \frac{y = \{x_1, \dots, x_k\}}{Finite(y)}$$

$$\text{(T19)}\ \frac{z = y \cup y' \quad Finite(y) \quad Finite(y')}{Finite(z)} \qquad \text{(T20)}\ \frac{z = y \cap y' \quad Finite(y)}{Finite(z)} \qquad \text{(T21)}\ \frac{z = y \cap y' \quad Finite(y')}{Finite(z)}$$

$$\text{(T22)}\ \frac{z = y \setminus y' \quad Finite(y)}{Finite(z)}$$

Table 2: Extra rules for *MLSSF*

calculus for a fragment of set theory with new useful rules which allow to avoid unnecessary splittings, a sort of a completion strategy in the Knuth-Bendix style.

Acknowledgments. The first author acknowledges partial support by C.N.R. of Italy under Grants n. 94.00411.CT12 (Research Project "Sistemi Interattivi per il Ragionamento Automatico"), n. 95.00411.CT12 (Research Project S.E.T.A.), and by MURST 40% project *Calcolo algebrico e simbolico.*

The second author also acknowledges fellowships by C.N.R. under Grants n. 201.01.112 and 201.12.612.

REFERENCES

[1] D. Cantone - V. Cutello, *A decision procedure for set-theoretic formulae involving rank and cardinality comparison,* Proceedings of the 3rd Italian Conference on Theoretical Computer Science (A. Bertoni, C. Böhm, P. Miglioli, Eds.), World Scientific, pp. 150–163, 1989.

[2] D. Cantone - V. Cutello - A. Policriti, *Decidability results for classes of purely universal formulae and quantifiers elimination in set theory,* Le Matematiche, 43

(1988), pp. 305–336.

[3] D. Cantone - A. Ferro - E.G. Omodeo, *Computable set theory,* International Series of Monographs on Computer Science, N. 6, Clarendon Press, Oxford, 1989.

[4] D. Cantone - A. Ferro - D. Sanfilippo, *ETNA:* E*xtensible* T*heorem prover in* NA*tural deduction,* Manuscript, November 1992.

[5] D. Cantone - S. Ghelfo - E.G. Omodeo, *The automation of syllogistic. I. Syllogistic normal forms*, J. Symbolic Computation, 6 (1) (1988), pp. 83–98.

[6] D. Cantone - R. Ruggeri Cannata, *A decidable tableau calculus for a fragment of set theory with iterated membership,* To appear in Journal of Automated Reasoning, (1995).

[7] D. Cantone - J.T. Schwartz, *A set-theoretically based proof verifier,* Manuscript. 1988.

[8] W.K. Smullyan, *First-Order Logic,* Ergebnisse der Mathematik und ihrer Grenzgebiete, Band 43, Springer-Verlag, Berlin, 1968.

Dipartimento di Matematica,
Università di Catania,
Viale A. Doria 6,
95125 Catania (ITALY)