Kanger's choices in automated reasoning

Anatoli Degtyarev and Andrei Voronkov

Computing Science Department
Uppsala University
{anatoli,voronkov}@csd.uu.se

Automated deduction, or automated theorem proving is a branch of science that deals with automatic search for a proof. The contribution of Kanger to automated deduction is well-recognized. His monograph [1957] introduced a calculus **LC**, which was one of the first calculi intended for automated proof-search. His article [1963] was later republished as [Kanger 1983] in the collection of "classical papers on computational logic". Kanger's [1963] (and also [1959]) calculi used some interesting features that have not been noted for a number of years, and the importance of which in the area of automated deduction has been recognized only much later.

Kanger [1963] gives no proofs and uses very succinct presentation. Automated deduction is an area in which very subtle changes in definitions and assertions may lead to inconsistent conclusions. Kanger's [1963] area was theorem proving in sequent calculi with equality and function symbols. Most papers published in this area before 1995 contained serious mistakes, except for Kanger's.

Now, when we are equipped with the impressive amount of techniques developed in this area, we are amazed with the incredible intuition of Kanger that allowed him to choose elegant, interesting (and correct) solutions among many possible choices. This article explains these choices and their place in modern automated deduction.

1 $\models \;\equiv\; \vdash$

The title of this section $\models \;\equiv\; \vdash$ is the logo of the Association for Logic Programming: truth is equivalent to provability. The equivalence of validity and provability for classical logic was proved by Gödel [1930] and is known as Gödel's completeness theorem. The notions of truth and validity in logic are formulated as semantical properties, while the notion of provability is defined in a purely syntactical way, so there seems to be a gap between the two notions.

In 1955–1957 several new proofs of Gödel's completeness theorem appeared [Beth 1955, Hintikka 1955, Schütte 1956, Kanger 1957] in which model theory and proof theory were connected in a very natural manner. They are based on the idea of searching for countermodels of a given formula F by applying a proof-search procedure to F (i.e. trying to establish $\vdash F$).

Kanger proposed to search for a proof in a *sequent calculus* named **LC** [Kanger 1957]. Cut-free sequent calculi for first-order logic have been introduced by Gentzen [1934]. They turned out to be an important tool for investigating basic proof-theoretic problems [e.g. Gentzen 1936, Girard 1987]. It has also been realized that sequent systems give a convenient tool for designing proof-search algorithms by using the rules of a calculus backwards (i.e. from the conclusion to the premise). To prove a sequent $\mathcal{S}$ *"we start from below with $\mathcal{S}$ and proceed upwards from level to level in the tree form. At each level the sequents of the next level above are uniquely and effectively determined — if there is such a level. If there is no such level, this fact is effectively determined, so that the process may brought to an end."* [Kanger 1957, page 31]. Consider some choices that arise when one formalizes sequent calculi.

Choice 1 (structure rules) In the original Gentzen's LK a sequent was an expression $\Gamma \to \Delta$, where Γ, Δ are *sequences* of formulas. Since Γ and Δ play the role of a conjunction and a disjunction, respectively, the logical semantics of a sequent is independent of the order of formulas in Γ, Δ. Neither does it depend on duplicate occurrences of formulas in Γ or Δ. Therefore, Gentzen had to introduce several *structure rules* that allow one to interchange and duplicate formulas in Γ, Δ, and also add new formulas:

$$\frac{\Gamma \to \Delta_1, B, A, \Delta_2}{\Gamma \to \Delta_1, A, B, \Delta_2} \qquad \frac{\Gamma_1, B, A, \Gamma_2 \to \Delta}{\Gamma_1, A, B, \Gamma_2 \to \Delta}$$

$$\frac{\Gamma \to \Delta_1, A, A}{\Gamma \to \Delta, A} \qquad \frac{\Gamma, A, A \to \Delta}{\Gamma, A \to \Delta}$$

$$\frac{\Gamma \to \Delta}{\Gamma \to \Delta, A} \qquad \frac{\Gamma \to \Delta}{\Gamma, A \to \Delta}$$

These rules are called *exchange*, *contraction* and *weakening*. The use of these rules introduced unnecessary technical details in proofs of [Gentzen 1934]. In order to avoid complications, other structures than sequences should be adopted. One obvious choice is the use *sets* instead of sequents. This again makes the formalization of sequent calculi quite complex. Suppose that Γ, Δ are sets and consider the following rule of sequent calculi:

$$\frac{\Gamma \to \Delta \cup \{A\}}{\Gamma \to \Delta \cup \{A \vee B\}} \ (\to \vee)$$

Let Γ be empty and consider four different instantiations for Δ: $\{\}$, $\{A\}$, $\{A \vee B\}$, and $\{A, A \vee B\}$. We obtain the following four instances of this rule:

$$\frac{\rightarrow \{A\}}{\rightarrow \{A \vee B\}}\ (\rightarrow \vee) \qquad \frac{\rightarrow \{A\}}{\rightarrow \{A, A \vee B\}}\ (\rightarrow \vee)$$

$$\frac{\rightarrow \{A, A \vee B\}}{\rightarrow \{A \vee B\}}\ (\rightarrow \vee) \qquad \frac{\rightarrow \{A, A \vee B\}}{\rightarrow \{A, A \vee B\}}\ (\rightarrow \vee)$$

The last one is absurd, among all four instances only the first one is enough to preserve completeness. Therefore, if we choose sets, we have to impose several restrictions on the inference rules. If we prohibit A and $A \vee B$ occur in Δ, we may eventually loose completeness. Even if impose no restrictions we might still be in need of the weakening rule. So what is the right choice for sequents and structure rules in sequent calculi?

Kanger's Choice 1 One distinctive feature of the calculi used in [Kanger 1957, Kanger 1963] is the *full absence of structure rules.* In order to achieve this, sequents are made of *multisets* of formulas and some rules are modified. The use of multisets eliminates the exchange rule. The use of contraction rule is replaced by the explicit duplication of formulas in some (but not all!) rules and changes in some other rules. For example, the $(\rightarrow \exists)$ rule in Kanger's system is

$$\frac{\Gamma \rightarrow \Delta, \exists x\varphi(x), \varphi(t)}{\Gamma \rightarrow \Delta, \exists x\varphi(x)}\ (\rightarrow \exists)$$

(the formula $\exists x\varphi(x)$ is explicitly duplicated), and the rule $(\rightarrow \vee)$ is changed into

$$\frac{\Gamma \rightarrow \Delta, A, B}{\Gamma \rightarrow \Delta, A \vee B}\ (\rightarrow \vee).$$

Finally, to get rid of weakening axioms $\Gamma, A \rightarrow \Delta, A$ are used instead of more traditional $A \rightarrow A$.

Completeness can be proved for virtually any variant of sequent calculi, but even completeness proofs meet small technical problems when it comes to structure rules. The choice made by Kanger to design a system without structure rules at all has now become de facto standard.

Kleene [1952] also described the sequent system G3 with invertible rules, but this property was realized straightforwardly by retaining the principal formula in the premise(s). Later Kleene's G3 was transformed to the system G4 [Kleene 1967][1], which was essentially the system **LC**.

Choice 2 (variants of rules) For some logical connectives, we have a choice among various sequent calculus rules. For example, for the proof of disjunction one can use either the following two rules:

[1]The propositional part of G4 coincided with the propositional rules of Ketonen [1944]

$$\frac{\Gamma \to \Delta, A}{\Gamma \to \Delta, A \vee B} \ (\to \vee) \quad \text{and} \quad \frac{\Gamma \to \Delta, B}{\Gamma \to \Delta, A \vee B} \ (\to \vee)$$

or just one rule

$$\frac{\Gamma \to \Delta, A, B}{\Gamma \to \Delta, A \vee B} \ (\to \vee)$$

The first choice seems to reflect the semantics of disjunction in a more intuitive way. Nevertheless, in Kanger's system the choice of the second rule is made. Why?

Kanger's Choice 2 The main answer is: all inference rules of Kanger's system are *invertible.* A rule is called invertible if the derivability of the conclusion implies the derivability of the premises. For automatic proof-search invertibility of rules is really a remarkable property. It if a sequent $\mathcal{S}$ is unprovable, then *any* derivation tree for $\mathcal{S}$ has a branch containing a countermodel for $\mathcal{S}$. It allowed Kanger to prove completeness *"by means of arguments which are new in some respect and which involve a new turn to the notion of validity"* [Kanger 1957, page 7]. It also allows one to search for a proof in a "don't care" matter: after we have selected a rule to apply, there is no need to undo the selection.

Kleene [1967] notes that the use of his system G4 for proving the completeness theorem "*is quite close to Beth [1955] which gave the present writer the idea for it. In some respect it more resembles Kanger [1957], as the author learned after working it out.*"

2 Use of a sequent calculus as a decision procedure for predicate logic

Kanger was one of the first who used a particular logical calculus as a decision procedure in the backward direction[2]. The point was to guarantee termination of the procedure on target classes of formulas. As examples Kanger considered the class of quantifier-free formulas and the class of $\forall^*\exists^*$ formulas (without functional symbols). Later, Wang [1960] described and implemented a procedure solving this class of formulas, also using backward proof-search in sequent calculi.

The possibility to obtain decision procedures for propositional logics, classical and intuitionistic, using backward proof-search in cut-free Gentzen type calculi was also noted by Kleene [1952]. Later, the use of derivations in machine-oriented calculi to decide some classes of predicate logic has become a generally

[2] Gentzen [1934] was the first to describe a decision procedure for propositional intuitionistic logic as proof-search in his cut-free calculus. The method was based on upwards applications of the rules. Now such approach is known as the *inverse method* [see Mints, Degtyarev, Tammet & Voronkov 1999].

accepted area of research [Maslov 1964, Kallick 1968, Maslov 1968, Joyner jr 1976, Fermüller, Leitsch, Tammet & Zamov 1993, Leitsch, Fermüller & Tammet 1999].

3 Proof-search via logical calculi

As soon as the first programs for proving theorems in predicate logic appeared [Prawitz, Prawitz & Voghera 1960, Wang 1960, Gilmore 1960, Davis & Putnam 1960] it has become clear that the main problem consists in instantiating variables in the application of $(\rightarrow \exists)$ and $(\forall \rightarrow)$) rules (also called γ-rules due to [Smullyan 1968, Fitting 1996]).

$$\frac{\Gamma \rightarrow \Delta, \varphi[t/x], \exists x\varphi}{\Gamma \rightarrow \Delta, \exists x\varphi}\ (\rightarrow \exists) \quad \text{and} \quad \frac{\Gamma, \varphi[t/x], \forall x\varphi \rightarrow \Delta}{\Gamma, \forall x\varphi \rightarrow \Delta}\ (\forall \rightarrow)$$

Here sequents are represented by expressions of the form $\Gamma \rightarrow \Delta$, where Γ and Δ are multisets of formulas.

Choice 3 (variable instantiation in γ-rules) How to instantiate variables by terms in γ-rules? The early methods of automated reasoning used the so-called *level saturation*. The set of all variable-free terms was enumerated (usually respecting depending on the term depth) and terms have been substituted one by one in that order. However, it was clear that such a solution is far from the best.

Kanger's Choice 3 The system of Kanger used a new strategy for instantiating variables in the applications of γ-rules. His strategy of instantiating variables is based on two ideas: the use of *free variables* and the *subterm instantiation* explained below.

Free variables used by Kanger have been originally introduced by Prawitz [1960] for logic without equality. The idea was to introduce a new kind of variables, called "dummies", in both [Prawitz 1960] and [Kanger 1963], and to delay instantiation of these variables until necessary information for it has been obtained. Comparing this approach to an earlier work of Beth [1959], Prawitz [1960] noted: *"the solution proposed here is quite different but well-suited for mechanical use"*. This method was independently proposed in Russia by N. Shanin in 1962 [see Maslov 1964] and characterized as the "metavariable method" by Maslov, Mints & Orevkov [1983]. Dummies or metavariables have later been called "free variables" by Fitting [1996].

Information for instantiation in [Kanger 1963] is provided by constructing an uninstantiated proof, and checking from time to time whether one can find values for dummies which make it a valid proof. Kanger [1963] reduces this check to verifying that the top sequents are "directly demonstrable", i.e. can be obtained from axioms by applications of equality rules.

Now we know a generally accepted solution to the problem of variable instantiation. When we apply γ-rules, we temporarily replace the variables by *dummies* or *metavariables*, and delay the instantiation until we try to close one branch of a derivation, i.e. to make it into an axiom. To close the branch one uses the notion of *unification* due to Robinson [1965] that allows to instantiate dummies by *most general* terms. However, at the early stage of research in automated reasoning, this solution has not yet been found.

We shall explain the Kanger's idea of subterm instantiation below, when we consider logic with equality.

4 Logic with equality and variable instantiation in γ-rules

For free-variable sequent-based calculi more practical way of instantiating variables is the introduction of unification in inference rules, that has been considered for calculi without equality in ([Voronkov 1988, Fitting 1996, Voronkov 1992]). For the case with equality the problem of variable instantiation turned out much more difficult[3].

The Kanger's idea of subterm instantiation discussed below was combined with another nice idea of considering a normal form of derivations in a sequent calculus with equality. All equality inferences, i.e. applications of equality rules, have been moved on top of the proof so that to precede all other steps in the proof. Later, the derivations of this form have been named *regular* [e.g. Lifschitz 1967], and used in the generalization of the inverse method to predicate calculus with equality in [Maslov 1971].

A typical branch of a regular derivation in logic with equality has the following form:

$$\begin{array}{rl} \Gamma'' \rightarrow \Delta'', t = t & \\ \vdots & \text{derivation equality rules only} \\ \Gamma' \rightarrow \Delta' & \\ \vdots & \text{derivation with no equality rules} \\ \Gamma \rightarrow \Delta & \end{array}$$

The derivation with no equality rules can use instantiation of variables by dummies. In order to make the derivation a valid one, one should substitute dummies by terms so that the subderivation of $\Gamma' \rightarrow \Delta'$ becomes a valid one. Let us call a *skeleton* of this derivation the subderivation obtained by removing the equality rules and using dummies instead of terms. To establish provability of a sequent, one has to solve the *skeleton instantiation problem*: find a replacement of dummies by terms so that every top sequent of a derivation is provable using only equality rules.

[3]A significant step in the spirit of subterm instantiation was made by Shostak [1978] who proved that for any set M of ground clauses satisfiability of M in the first-order logic with equality is equivalent to Boolean satisfiability of M together with all ground instances of equality axioms obtained by substituting terms from M for variables.

Choice 4 (skeleton instantiation) Can we find any reasonable strategy of skeleton instantiation? For several years, a number of techniques have been developed with the aim of solving skeleton instantiation. The first time this problem has been described as *simultaneous rigid E-unification* in [Gallier, Raatz & Snyder 1987]. Several faulty algorithms solving simultaneous rigid E-unification have been proposed in e.g. [Gallier, Narendran, Plaisted & Snyder 1988, Gallier, Narendran, Raatz & Snyder 1992, Goubault 1994], until the problem was proved undecidable in [Degtyarev & Voronkov 1996*b*]. Later, it was shown that even very small fragments of the problem are still undecidable [Plaisted 1995, Veanes 1997*a*, Veanes 1997*b*, Degtyarev, Gurevich, Narendran, Veanes & Voronkov 1998].

The exact connections of simultaneous rigid E-unification problem with other decision problems, including skeleton instantiation are described in [Degtyarev, Gurevich & Voronkov 1996, Voronkov 1998*b*].

In view of the undecidability, there is no computational way of solving skeleton instantiation. The unification-based methods to handle variable instantiation were found only recently [Degtyarev & Voronkov 1996*a*, Degtyarev & Voronkov 1998*b*]. One naive solution would be to blindly substitute variables by all possible terms, thus obtaining at least a semi-decision procedure. All other proposals published before 1996 were essentially incomplete (or had mistakes in the completeness proofs), except for Kanger's.

Kanger's Choice 4 Kanger [1963] proposed a method for instantiating free variables that can be characterized as *subterm instantiation*. This method has been referred to by the name of *minus-normalization* in [Matulis 1962, Norgela 1974, Maslov & Mints 1983].

According to this method, instantiation of variables in the backwards application of γ-rule is made only by ground terms, i.e. by terms without variables, explicitly occurring in its conclusion. This method is complete for first-order classical logic and incomplete for intuitionistic logic. Unfortunately even in simplest cases, minus-normalization can require a huge number of instantiation. Some results on subterm instantiation are proved by Norgela [1974].

For logic without equality and function symbols a similar restriction on γ-rules was used in proof procedures developed in [Quine 1955, Hintikka 1955, Beth 1955, Schütte 1956].

Kanger's intuition was that instantiations admitted by subterm instantiation contains the set that could be generated by applying the most general unification technique to proof search in sequent calculi. Consider the following example. Suppose we want to find a proof of the formula $\exists x(P(f(x),x) \supset P(x,g(c)))$. We can start from computation of the most general unifier of literals $P(f(x),x)$ and $P(y,g(c))$, because they have opposite occurrences (variables in the second literal are renamed). This computation consists of two steps. Firstly, we replace x by $g(c)$, and secondly, y by $f(g(c))$. The same steps would be done by applying Kanger's subterm instantiation.

$$\begin{array}{c} \ldots P(f(g(c)),g(c))\ldots \rightarrow \ldots P(f(g(c)),g(c))\ldots \\ \vdots \\ \rightarrow \ldots P(f(g(c)),g(c)) \supset P(g(c),g(c)), P(f(f(g(c))),f(g(c))) \supset P(f(g(c)),g(c))\ldots \\ \vdots \\ \dfrac{\dfrac{\rightarrow P(f(c),c) \supset P(c,g(c)), P(f(g(c)),g(c)) \supset P(g(c),g(c))\ldots}{\rightarrow P(f(c),c) \supset P(c,g(c)), \exists x(P(f(x),x) \supset P(x,g(c)))}}{\rightarrow \exists x(P(f(x),x) \supset P(x,g(c)))} \end{array}$$

Let us now consider the connection of (simultaneous) rigid E-unification with Kanger's subterm instantiation. Rigid E-unification can be formulated as follows. Given a (finite) set of equations $E = \{s_1 = t_1, \ldots, s_n = t_n\}$ and the equation $s = t$, does there exist a substitution θ such that $\vdash \forall(s_1\theta = t_1\theta \wedge \ldots \wedge s_n\theta = t_n\theta \supset s\theta = t\theta)$ [4].

A rigid E-unification problem $(E, s = t)$ corresponds to the sequent $E \rightarrow s = t$. A simultaneous rigid E-unification problem corresponds to a finite number of such sequents situated in the leaves of a sequent derivation. Undecidability of simultaneous rigid E-unification was proved in 1995 [see Degtyarev & Voronkov 1996*b*], its comprehensive investigation can be found in [Degtyarev et al. 1996, Voronkov 1998*b*, Voronkov 1998*a*]. After that Degtyarev & Voronkov [1998*b*] proposed a complete proof procedure for sequent-type calculi with equality based on incomplete but terminating procedure for rigid E-unification. We can say that the subterm instantiation of Kanger was the first incomplete but terminating algorithm for simultaneous rigid E-unification giving a complete proof search procedure. Kanger's method cannot be called a truly free-variable method, but it avoids exhaustive search in the set of all terms. In addition, Kanger's method is complete unlike several other methods proposed between 1987 and 1994. A detailed survey of proof-search methods in sequent based calculi with equality can been found in [Degtyarev & Voronkov 1998*a*, Degtyarev & Voronkov 1999].

5 More on equational logic

Handling equality in automated theorem proving is one of the central topics in automated reasoning. Kanger has anticipated many tendencies used in the modern methods of reasoning with equality.

A sequent-based proof procedure for logic with functional symbols and equality was described in [Kanger 1959] and [Kanger 1963]. It was the first extensive analysis of a sequent-style system for equational logic. A more easily accessible publication is [Kanger 1983].

[4] The word "rigid" is introduced to distinguish rigid E-unification from E-unification: given a (finite) set of equations $E = \{s_1 = t_1, \ldots, s_n = t_n\}$ and the equation $s = t$, find a substitution θ such that $\forall(s_1 = t_1), \ldots, \forall(s_n = t_n) \vdash s\theta = t\theta$.

Choice 5 (equality rules in sequent calculi) Essentially all formalizations of sequent calculi with equality use variants of substitution of equals by equals:

$$\frac{s = t, \Gamma_t^s \to \Delta_t^s}{s = t, \Gamma \to \Delta}$$

The number of variations is huge. One can allow to replace one occurrence of s by t or several occurrences, put restrictions on the form of Γ and Δ, replace s by t only in Δ, only replace s by t but not t by s for some terms s and t etc.

Kanger's Choice 5 As for the formalization of equality rules, the contribution of Kanger to automated deduction is really remarkable. We shall consider four novelties introduced by Kanger and discuss their connections with the state-of-the-art techniques in automated reasoning with equality. One novelty introduce by Kanger on the application of equality rules has been mentioned above: to use regular derivations. The other three are discussed below.

Two other novelties introduced by Kanger are the use of **simultaneous replacement** of *all* occurrences of the same subterm (we consider *simultaneous* and *replacement* as two novelties):

$$\frac{s = t, \Gamma_t^s \to \Delta_t^s}{s = t, \Gamma \to \Delta}$$

where Γ_t^s denotes the result of the simultaneous replacement of *all* occurrences of the term s by t in Γ.

It has appeared that this rule is enough to establish that a sequent is "directly demonstrable". In other words, this rule decides the *uniform word problem*: whether $s_1 = t_1, \ldots, s_n = t_n \vdash s = t$, where s_i, t_i, s, t are ground terms. This problem is also equivalent to the decidability problem of the quantifier-free theory of equality. Its decidability has been proved in [Ackermann 1954] but there no practical algorithm had been proposed. Now the uniform word problem is solved by the so called congruence closure algorithm [Shostak 1978, Nelson & Oppen 1980]. The termination of Kanger's method of establishing of direct demonstrability is based on very important restriction put on the above equality rule which we consider to be the fourth novelty. Kanger only allowed its **non-increasing applications** of equality rules, i.e. those in which the depth of t is not greater than the depth of s (we do not distinguish here the formula $s = t$ from $t = s$ and consider non-increasing constraint in the above rule as orientation of equality).

From the viewpoint of the current knowledge of the area Kanger's rule is interesting in the following. Firstly, it can be considered as the rule of *demodulation*, or *simplification*, introduced to automated deduction by [Wos, Robinson, Carson & Shalla 1967] as an heuristic tool for discarding apparently irrelevant clauses. It is essential that Γ and Δ are *replaced* but not retained, and this calculus does not contain the contraction rule [5]. It has long ago been acknowl-

[5]in opposite to Lifschitz [1967] and Orevkov [1969] who considered more extravagant ways of orientation of equations but in calculi with contraction.

edged that simplification and other techniques for elimination of redundancy are indispensable for an acceptable behavior of any practical theorem prover [see Bachmair & Ganzinger 1999]. However, the first complete procedure for solving the word problem (with free variables) combining orientation of equations with simplification technique was represented only much later in the famous paper of Knuth & Bendix [1970].

Secondly, Kanger's rule is close to the rule of *simultaneous paramodulation* of Benanav [1990]. Simultaneous paramodulation is a refinement of *paramodulation* — the main rule for handling equality introduced by [Robinson & Wos 1969]. Paramodulation is defined on clauses with free variables, and uses most general unifiers. Completeness of paramodulation was an open problem for many years. A great obstacle was that paramodulation does not have the *lifting property*. It means that in the case of paramodulation standard techniques of proving completeness could not be applied. This technique is to first prove the existence of a ground derivation and then to *lift* it to the non-ground case. It is remarkable that the system with simultaneous paramodulation has the lifting property, and it was sufficient to prove completeness of simultaneous paramodulation only on the ground level. In the same way Kanger's rule could also be lifted from the ground to the non-ground level with would guarantee completeness.

References

Ackermann, W. [1954], *Solvable Cases of the Decision Problem*, Horth-Holland.

Bachmair, L. & Ganzinger, H. [1999], A theory of resolution, *in* A. Robinson & A. Voronkov, eds, 'Handbook of Automated Reasoning', Elsevier Science and MIT Press. To appear.

Benanav, D. [1990], Simultaneous paramodulation, *in* M. Stickel, ed., 'Proc. 10th Int. Conf. on Automated Deduction', Vol. 449 of *Lecture Notes in Artificial Intelligence*, pp. 442–455.

Beth, E. [1955], 'Semantic entailment and formal derivability', *Mededelingen der Koninklijke Nederlandse Akademie van Wetenschappen, Afd. Letterkunde, Nieuwe Reeks* **18**(13).

Beth, E. [1959], *The Foundations of Mathematics*, North-Holland Pub. Co., Amsterdam.

Davis, M. & Putnam, H. [1960], 'A computing procedure for quantification theory', *Journal of the Association for Computing Machinery* **7**(3). Reprinted as [Davis & Putnam 1983].

Davis, M. & Putnam, H. [1983], A computing procedure for quantification theory, *in* J. Siekmann & G. Wrightson, eds, 'Automation of Reasoning. Classical Papers on Computational Logic', Vol. 1, Springer Verlag, pp. 125–150. Originally appeared as [Davis & Putnam 1960].

Degtyarev, A., Gurevich, Y., Narendran, P., Veanes, M. & Voronkov, A. [1998], The decidability of simultaneous rigid E-unification with one variable, *in* T. Nipkow, ed., 'Rewriting Techniques and Applications, RTA'98', Vol. 1379 of *Lecture Notes in Computer Science*, Springer Verlag, pp. 181–195.

Degtyarev, A., Gurevich, Y. & Voronkov, A. [1996], Herbrand's theorem and equational reasoning: Problems and solutions, *in* 'Bulletin of the European Association for Theoretical Computer Science', Vol. 60, pp. 78–95. The "Logic in Computer Science" column.

Degtyarev, A. & Voronkov, A. [1996*a*], Equality elimination for the tableau method, *in* J. Calmet & C. Limongelli, eds, 'Design and Implementation of Symbolic Computation Systems. International Symposium, DISCO'96', Vol. 1128 of *Lecture Notes in Computer Science*, Karlsruhe, Germany, pp. 46–60.

Degtyarev, A. & Voronkov, A. [1996*b*], 'The undecidability of simultaneous rigid E-unification', *Theoretical Computer Science* **166**(1–2), 291–300.

Degtyarev, A. & Voronkov, A. [1998*a*], Equality reasoning in sequent-based calculi, Technical Report MPI-I-98-2-011, Max-Planck Institut für Informatik, Saarbrücken.

Degtyarev, A. & Voronkov, A. [1998*b*], 'What you always wanted to know about rigid E-unification', *Journal of Automated Reasoning* **20**(1), 47–80.

Degtyarev, A. & Voronkov, A. [1999], Equality reasoning in sequent-based calculi, *in* A. Robinson & A. Voronkov, eds, 'Handbook of Automated Reasoning', Elsevier Science and MIT Press. To appear.

Fermüller, C., Leitsch, A., Tammet, T. & Zamov, N. [1993], *Resolution Methods for the Decision Problem*, Vol. 679 of *Lecture Notes in Computer Science*, Springer Verlag.

Fitting, M. [1996], *First Order Logic and Automated Theorem Proving*, 2nd edn, Springer Verlag, New York. 1st edition appeared in 1990.

Gallier, J., Narendran, P., Plaisted, D. & Snyder, W. [1988], Rigid E-unification is NP-complete, *in* 'Proc. IEEE Conference on Logic in Computer Science (LICS)', IEEE Computer Society Press, pp. 338–346.

Gallier, J., Narendran, P., Raatz, S. & Snyder, W. [1992], 'Theorem proving using equational matings and rigid E-unification', *Journal of the Association for Computing Machinery* **39**(2), 377–429.

Gallier, J., Raatz, S. & Snyder, W. [1987], Theorem proving using rigid E-unification: Equational matings, *in* 'Proc. IEEE Conference on Logic in Computer Science (LICS)', IEEE Computer Society Press, pp. 338–346.

Gentzen, G. [1934], 'Untersuchungen über das logische Schließen', *Mathematical Zeitschrift* **39**, 176–210, 405–431. Translated as [Gentzen 1969].

Gentzen, G. [1936], 'Die Wiederspruchsfreiheit der reinen Zahlentheorie', *Mathematische Annalen* **112**, 493–565.

Gentzen, G. [1969], Investigations into logical deduction, *in* M. Szabo, ed., 'The Collected Papers of Gerhard Gentzen', North Holland, Amsterdam, pp. 68–131. Originally appeared as [Gentzen 1934].

Gilmore, P. [1960], 'A proof method for quantification theory: its justification and realization', *IBM J. of Research and Development* **4**, 28–35. Reprinted as [Gilmore 1983].

Gilmore, P. [1983], A proof method for quantification theory: its justification and realization, *in* J. Siekmann & G. Wrightson, eds, 'Automation of Reasoning. Classical Papers on Computational Logic', Vol. 1, Springer Verlag, pp. 151–158. Originally published as [Gilmore 1960].

Girard, J.-Y. [1987], *Proof Theory and Logical Complexity*, Studies in Proof Theory, Bibliopolis, Napoly.

Gödel, K. [1930], 'Die vollständingkeit der axiome des logischen funktionenkalküls', *Monatshefte für Mathematik and Physik* **37**, 349–360.

Goubault, J. [1994], Rigid $\vec{E}$-unifiability is DEXPTIME-complete, *in* 'Proc. IEEE Conference on Logic in Computer Science (LICS)', IEEE Computer Society Press.

Hintikka, K. [1955], 'Form and content in quantification theory', *Acta Philosophica Fennica* **8**, 7–55.

Joyner jr, W. [1976], 'Resolution strategies as decision procedures', *Journal of the Association for Computing Machinery* **23**, 398–417.

Kallick, B. [1968], A decision procedure based on the resolution method, *in* 'IFIP'68', North Holland, pp. 365–377.

Kanger, S. [1957], *Provability in Logic*, Vol. 1 of *Studies in Philosophy*, Almqvist and Wicksell, Stockholm.

Kanger, S. [1959], *Handbook i logic*, Stockholm.

Kanger, S. [1963], A simplified proof method for elementary logic, *in* P. Braffort & D. Hirschberg, eds, 'Computer Programming and Formal Systems', North Holland, pp. 87–94. Reprinted as [Kanger 1983].

Kanger, S. [1983], A simplified proof method for elementary logic, *in* J. Siekmann & G. Wrightson, eds, 'Automation of Reasoning. Classical Papers on Computational Logic', Vol. 1, Springer Verlag, pp. 364–371. Originally published as [Kanger 1983].

Ketonen, O. [1944], ‘Untersuchungen zum Prädikatenkalkül’, *Annales Academia Scientiarum Fennicae* **23**. Ser. A, I Mathematixa-physica.

Kleene, S. [1952], *Introduction to Metamathematics*, Van Nostrand P.C., Amsterdam.

Kleene, S. [1967], *Mathematical Logic*, John Wiley and Sons.

Knuth, D. & Bendix, P. [1970], Simple word problems in universal algebras, *in* J. Leech, ed., ‘Computational Problems in Abstract Algebra’, Pergamon Press, Oxford, pp. 263–297.

Leitsch, A., Fermüller, C. & Tammet, T. [1999], Resolution decision procedures, *in* A. Robinson & A. Voronkov, eds, ‘Handbook of Automated Reasoning’, Elsevier Science and MIT Press. To appear.

Lifschitz, V. [1967], ‘A normal form of derivations in predicate calculus with equality and function symbols (in Russian)’, *Zapiski Nauchnyh Seminarov LOMI* **4**, 58–64. English Translation in: Seminars in Mathematics: Steklov Math. Inst. 4, Consultants Bureau, NY-London, 1969.

Maslov, S. [1964], ‘The inverse method of establishing deducibility in the classical predicate calculus’, *Soviet Mathematical Doklady* **5**, 1420–1424.

Maslov, S. [1968], The inverse method of establishing deducibility of logical calculi (in Russian), *in* ‘Collected Works of MIAN’, Vol. 98, Nauka, Moscow, pp. 26–87.

Maslov, S. [1971], ‘The generalization of the inverse method to predicate calculus with equality (in Russian)’, *Zapiski Nauchnyh Seminarov LOMI* **20**, 80–96. English translation in: Journal of Soviet Mathematics 1, no. 1.

Maslov, S. & Mints, G. [1983], The proof-search theory and the inverse method (in Russian), *in* M. G., ed., ‘Mathematical Logic and Automatic Theorem Proving’, Nauka, Moscow, pp. 291–314.

Maslov, S., Mints, G. & Orevkov, V. [1983], Mechanical proof-search and the theory of logical deduction in the USSR, *in* J.Siekmann & G.Wrightson, eds, ‘Automation of Reasoning (Classical papers on Computational Logic)’, Vol. 1, Springer Verlag, pp. 29–38.

Matulis, V. [1962], ‘Two variants of classical predicate calculus without structure rules (in Russian)’, *Soviet Mathematical Doklady* **147**(5), 1029–1031.

Mints, G., Degtyarev, A., Tammet, T. & Voronkov, A. [1999], The inverse method, *in* A. Robinson & A. Voronkov, eds, ‘Handbook of Automated Reasoning’, Elsevier Science and MIT Press. To appear.

Nelson, G. & Oppen, D. [1980], ‘Fast decision procedures based on congruence closure’, *Journal of the Association for Computing Machinery* **27**(2), 356–364.

Norgela, S. [1974], On the size of derivations under minus-normalization (in Russian), *in* V. Smirnov, ed., 'The Theory of Logical Inference', Institute of Philosophy, Moscow.

Orevkov, V. [1969], 'On nonlengthening applications of equality rules (in Russian)', *Zapiski Nauchnyh Seminarov LOMI* **16**, 152–156. English Translation in: Seminars in Mathematics: Steklov Math. Inst. 16, Consultants Bureau, NY-London, 1971, p.77-79.

Plaisted, D. [1995], Special cases and substitutes for rigid E-unification, Technical Report MPI-I-95-2-010, Max-Planck-Institut für Informatik.

Prawitz, D. [1960], 'An improved proof procedure', *Theoria* **26**, 102–139. Reprinted as [Prawitz 1983].

Prawitz, D. [1983], An improved proof procedure, *in* J. Siekmann & G. Wrightson, eds, 'Automation of Reasoning. Classical Papers on Computational Logic', Vol. 1, Springer Verlag, pp. 162–201. Originally appeared as [Prawitz 1960].

Prawitz, D., Prawitz, H. & Voghera, N. [1960], 'A mechanical proof procedure and its realization in an electronic computer', *Journal of the Association for Computing Machinery* **7**(2). Reprinted as [Prawitz, Prawitz & Voghera 1983].

Prawitz, D., Prawitz, H. & Voghera, N. [1983], A mechanical proof procedure and its realization in an electronic computer, *in* J. Siekmann & G. Wrightson, eds, 'Automation of Reasoning. Classical Papers on Computational Logic', Vol. 1, Springer Verlag, pp. 162–201. Originally appeared as [Prawitz et al. 1960].

Quine, W. [1955], 'A proof procedure for quantification theory', *Journal of Symbolic Logic* **20**, 141–149.

Robinson, G. & Wos, L. [1969], Paramodulation and theorem-proving in first order theories with equality, *in* B. Meltzer & D. Michie, eds, 'Machine Intelligence', Vol. 4, Edinburgh University Press, pp. 135–150.

Robinson, J. [1965], 'A machine-oriented logic based on the resolution principle', *Journal of the Association for Computing Machinery* **12**(1), 23–41.

Schütte, K. [1956], 'Ein system des verknüpfenden schliessens', *Archiv für Mathematische Logik und Grundlagenforshung* **2**, 34–67.

Shostak, R. [1978], 'An algorithm for reasoning about equality', *Communications of the ACM* **21**, 583–585.

Smullyan, R. [1968], *First-Order Logic*, Springer Verlag.

Veanes, M. [1997*a*], On Simultaneous Rigid E-Unification, PhD thesis, Uppsala University.

Veanes, M. [1997*b*], The undecidability of simultaneous rigid E-unification with two variables, *in* G. Gottlob, A. Leitsch & D. Mundici, eds, 'Computational Logic and Proof Theory. 5th Kurt Gödel Colloquium, KGC'97', Vol. 1289 of *Lecture Notes in Computer Science*, Vienna, Austria, pp. 305–318.

Voronkov, A. [1988], A proof search method for first order logic, *in* 'Preliminary Proceedings of COLOG-88', Vol. 2, Tallinn, pp. 104–118.

Voronkov, A. [1992], Theorem proving in non-standard logics based on the inverse method, *in* D. Kapur, ed., '11th International Conference on Automated Deduction', Vol. 607 of *Lecture Notes in Artificial Intelligence*, Springer Verlag, Saratoga Springs, NY, USA, pp. 648–662.

Voronkov, A. [1998*a*], Herbrand's theorem, automated reasoning and semantic tableaux, *in* 'Proc. IEEE Conference on Logic in Computer Science (LICS)', IEEE Computer Society Press, pp. 252–263.

Voronkov, A. [1998*b*], Simultaneous rigid E-unification and other decision problems related to Herbrand's theorem, UPMAIL Technical Report 152, Uppsala University, Computing Science Department. To appear in Theoretical Computer Science.

Wang, H. [1960], 'Towards mechanical mathematics', *IBM J. of Research and Development* **4**, 2–22. Reprinted as [Wang 1983].

Wang, H. [1983], Towards mechanical mathematics, *in* J. Siekmann & G. Wrightson, eds, 'Automation of Reasoning. Classical Papers on Computational Logic', Vol. 1, Springer Verlag, pp. 244–264. Originally published as [Wang 1960].

Wos, L., Robinson, G., Carson, D. & Shalla, L. [1967], 'The concept of demodulation in theorem proving', *Journal of the Association for Computing Machinery* **14**, 698–709.