Theorem proving in non-standard logics based on the inverse method

Andrei Voronkov[1]

ECRC, Arabellastr.17, 8000 Munich 81, Germany (voronkov@ecrc.de)*

Abstract. We present a general approach to constructing efficient theorem provers which is applicable to classical and some non-classical logics. To define machine-oriented versions of these logics we use Maslov's inverse method. To make the proof-search more efficient one needs to use strategies for reducing search space. Instead of studying concrete systems like the classical predicate calculus we first present an abstract definition of a deductive system and then prove completeness of several strategies for deductive systems. Afterwards we project these theorems from the abstract level of deductive systems to concrete systems, obtaining efficient strategies for them.

1 Introduction

This paper has two main aims. The first is to show how to construct machine-oriented calculi for a wide class of logics based on the ideas of Maslov's inverse method [Mas67, Mas87]. The second is to give a general way to characterize strategies which reduce search space for the machine-oriented calculi.

The title of this paper is somehow misleading because our approach is applicable to classical logic as well. As examples of application of the general techniques from our paper we consider classical logic **Cl** and modal logic **S4**.

In Section 2 we outline the general ideas of the inverse method and construct machine-oriented calculi for classical predicate logic and for **S4**. However the naive construction gives us inefficient calculi which have many redundancies. To make the proof search more efficient one needs to introduce strategies for reducing the search space. As we show, it is possible to introduce strategies which are similar to the subsumption strategy and the tautology deletion strategy of the resolution method, as well as some other strategies.

To this end in Section 3 we define an abstract notion of deductive systems. This notion intuitively captures the general idea of what a theorem proving system is. Then we introduce several strategies for the deductive systems and prove their completeness in the general case.

In Section 4 we show how these abstract definitions can be projected to particular logical systems thus giving efficient machine-oriented versions for them. We also briefly sketch some aspects of the implementation of the strategies in the theorem proving system LISS [Vor90a]. The techniques used here is quite abstract and applicable to many particular systems. So during the presentation we shall sometimes skip from a more concrete level of particular logical systems to a more abstract level of general notions and back.

* On leave from the International Laboratory of Intelligent Systems (SINTEL), 630090, Universitetski Prospect 4, Novosibirsk 90, Russia.

2 Maslov's Inverse Method

Usually proof-search in theorem proving systems is organized either in the form of backward reasoning or in the form of forward reasoning. The advantages and disadvantages of both approaches are well known. Usually backward reasoning is used because it is more goal-directed. Maslov's inverse method uses a version of forward reasoning strategy which is however more goal-oriented than the usual forward reasoning methods. In this section we will informally explain the main idea of the inverse method and apply this method to two particular logical systems: the classical predicate calculus and modal logic **S4**.

The first publications on Maslov's inverse method refer to 1964. Later some of Maslov's paper were published in a more accessible literature (see e.g. [Mas67, Mas69, Mas87]). Lifschitz, Zamov and Mints made another steps towards the explanation of the ideas of the inverse method [Lif86, Zam89, Min90]

The main idea of the inverse method is the following. Suppose that we have a logical system consisting of axioms **Ax** and inference rules **IR**. Suppose then that we want to find a proof of a (goal) formula γ. The (naive) backward procedure starts from γ and tries to apply all possible rules that can be used to prove it, reducing this goal to subgoals. Then this procedure applies to subgoals until (in case of success) we come to axioms.

A normal forward reasoning procedure uses a different approach: starting from axioms it generates new formulas (or sequents) until the goal is reached.

It is well known that the main disadvantage of the forward reasoning that it is not goal-directed. Suppose however that we have a criterion which allows us to select only <u>some relevant</u> initial axioms $\mathbf{Ax}_\gamma$ and inference rules $\mathbf{IR}_\gamma$ which could be used to prove the goal γ. In this case we can <u>specialize</u> the forward procedure to make it a procedure oriented to prove <u>only</u> γ.

It allows one to exploit <u>all</u> properties of a forward reasoning procedure but to make it more goal oriented at the same time. This is the key idea of the inverse method.

One of the problems in applying the inverse method is that of specializing the calculus. For example if we take cut-free sequent calculi for classical predicate logic or for **S4** which are quite friendly for the backward proof search, then the problem is the infiniteness of the sets of axioms and of inference rules.

However it is possible to make the specialized system finite using the notions of unification and the most general unifier. Let us consider as an example the classical predicate calculus **Cl**.

Below $\Gamma, \Delta, \Sigma, \Xi$ are *sets* of formulas. <u>Sequent</u> is an expression of the form $\Gamma \to \Delta$. In all calculi from this paper provable objects are sequents.

System Cl

- Axioms:

$$\varphi \to \varphi$$

- Rules:

$$(\to \wedge) \qquad \frac{\Gamma \to \Delta, \varphi \qquad \Sigma \to \Xi, \psi}{\Gamma, \Sigma \to \Delta, \Xi, \varphi \wedge \psi}$$

$$(\wedge \to) \qquad \frac{\varphi, \Gamma \to \Delta}{\varphi \wedge \psi, \Gamma \to \Delta} \qquad\qquad \frac{\psi, \Gamma \to \Delta}{\varphi \wedge \psi, \Gamma \to \Delta}$$

$$(\to \vee) \quad \frac{\Gamma \to \Delta, \varphi}{\Gamma \to \Delta, \varphi \vee \psi} \qquad \frac{\Gamma \to \Delta, \psi}{\Gamma \to \Delta, \varphi \vee \psi}$$

$$(\vee \to) \quad \frac{\varphi, \Gamma \to \Delta \qquad \psi, \Sigma \to \Xi}{\varphi \vee \psi, \Gamma, \Sigma \to \Delta, \Xi}$$

$$(\to \supset) \quad \frac{\Gamma \to \Delta, \psi}{\Gamma \to \Delta, \varphi \supset \psi} \qquad \frac{\varphi, \Gamma \to \Delta}{\Gamma \to \Delta, \varphi \supset \psi}$$

$$(\supset \to) \quad \frac{\Gamma \to \Delta, \varphi \qquad \psi, \Sigma \to \Xi}{\varphi \supset \psi, \Gamma, \Sigma \to \Delta, \Xi}$$

$$(\neg \to) \quad \frac{\Gamma \to \Delta, \varphi}{\neg\varphi, \Gamma \to \Delta} \qquad (\to \neg) \quad \frac{\varphi, \Gamma \to \Delta}{\Gamma \to \Delta, \neg\varphi}$$

$$(\to \forall) \quad \frac{\Gamma \to \Delta, \varphi}{\Gamma \to \Delta, \forall x \varphi} \qquad (\forall \to) \quad \frac{\varphi[x \leftarrow t], \Gamma \to \Delta}{\forall x \varphi, \Gamma \to \Delta}$$

$$(\to \exists) \quad \frac{\Gamma \to \Delta, \varphi[x \leftarrow t]}{\Gamma \to \Delta, \exists x \varphi} \qquad (\exists \to) \quad \frac{\varphi, \Gamma \to \Delta}{\exists x \varphi, \Gamma \to \Delta}$$

Note: In the rules $(\to \forall)$ and $(\exists \to)$ the variable x has no free occurences in Γ, Δ.

To make the system finite one needs to make finite the set of axioms. To this end we can use the subformula property of the classical predicate calculus: all formulas occuring in a cut-free proof of a formula φ are subformulas of φ. If φ contains quantifiers then the number of its subformulas is infinite, but we can consider only the most general axioms using the notion of unification. To make this idea precise we shall introduce some notions.

The notions of (<u>free</u>) (<u>immediate</u>) <u>subformula</u> are defined as in [Wal90].

1. φ is an immediate (and a free immediate) subformula of $\neg\varphi$, both φ and ψ are immediate (and free immediate) subformulas of $\varphi \wedge \psi, \varphi \supset \psi$ and $\varphi \vee \psi$.
2. For any term t, the formula $\varphi[x \leftarrow t]$ is an immediate subformula and φ is the free immediate subformula of $\forall x \varphi$ and $\exists x \varphi$.
3. If φ is a (free) immediate subformula of ψ or identical to ψ, then φ is a (free) subformula of ψ.
4. If φ is a (free) subformula of ψ and ψ is a (free) subformula of χ then φ is a (free) subformula of ψ.

   For modal logics we add to this definition the following:
5. φ is the free immediate subformula of the formulas $\Box\varphi$ and $\Diamond\varphi$.

We make two agreements to be held throughout the paper:

From now on we assume that γ is a fixed ground formula to be proved.
In what follows "subformula" means "subformula with the fixed occurrence". (In other words, we consider different occurences of the same subformula as different subformulas.)

The notions of negative and positive occurrence of a subformula to a formula are defined as usual. To stress that a subformula φ occurs in the formula γ positively (negatively) we shall use the notation φ^+ (φ^-). The notion of a (most general) unifier is defined as usual. Let E_1, E_2 be terms or formulas, θ_1, θ_2 substitutions. Then the pair $\langle\theta_1, \theta_2\rangle$ is called r-unifier of E_1, E_2 iff $E_1\theta_1$ is identical to $E_2\theta_2$. Such a unifier is a most general r-unifier if for any r-unifier $\langle\theta_1', \theta_2'\rangle$ of E_1, E_2 there is a substitution θ such that $\theta_i'\theta$ is equal to θ_i, $i = 1, 2$.

Now we are ready to define the specialized system $\mathbf{Cl}_\gamma$. The set of sequents of $\mathbf{Cl}_\gamma$ consists of sequents of the form $\varphi_1, \ldots, \varphi_n \to \psi_1, \ldots, \psi_m$ such that φ_i are positive and ψ_j are negative subformulas of γ. It is easy to see that any subformula of γ can be obtained from a free subformula by applying some substitution. For this reason in the formulation of $\mathbf{Cl}_\gamma$ below we shall represent formulas of $\mathbf{Cl}_\gamma$ in the form $\varphi\theta$ where φ is a free subformula of γ, θ a substitution[2].

System $\mathbf{Cl}_\gamma$.

In all the rules given below φ, ψ are free subformulas of γ.

- Axioms:

  $$\varphi_1\theta_1 \to \varphi_2\theta_2$$

  where φ_1 is a negative and φ_2 a positive free subformulas of γ, $\langle\theta_1, \theta_2\rangle$ the most general r-unifier of φ_1, φ_2.[3]
- One-premise rules[4]:

  $$(\wedge \to)\quad \frac{\varphi[\bar{x}\leftarrow\bar{t}], \Gamma \to \Delta}{(\varphi\wedge\psi)[\bar{x}\leftarrow\bar{t}], \Gamma \to \Delta} \qquad \frac{\psi[\bar{y}\leftarrow\bar{s}], \Gamma \to \Delta}{(\varphi\wedge\psi)[\bar{y}\leftarrow\bar{s}], \Gamma \to \Delta}$$

  $$(\to \vee)\quad \frac{\Gamma \to \Delta, \varphi[\bar{x}\leftarrow\bar{t}]}{\Gamma \to \Delta, (\varphi\vee\psi)[\bar{x}\leftarrow\bar{t}]} \qquad \frac{\Gamma \to \Delta, \psi[\bar{y}\leftarrow\bar{s}]}{\Gamma \to \Delta, (\varphi\vee\psi)[\bar{y}\leftarrow\bar{s}]}$$

  $$(\to \supset)\quad \frac{\Gamma \to \Delta, \psi[\bar{y}\leftarrow\bar{s}]}{\Gamma \to \Delta, (\varphi\supset\psi)[\bar{y}\leftarrow\bar{s}]} \qquad \frac{\varphi[\bar{x}\leftarrow\bar{t}], \Gamma \to \Delta}{\Gamma \to \Delta, (\varphi\supset\psi)[\bar{x}\leftarrow\bar{t}]}$$

  $$(\neg \to)\quad \frac{\Gamma \to \Delta, \varphi[\bar{x}\leftarrow\bar{t}]}{\neg\varphi[\bar{x}\leftarrow\bar{t}], \Gamma \to \Delta} \qquad (\to \neg)\quad \frac{\varphi[\bar{x}\leftarrow\bar{t}], \Gamma \to \Delta}{\Gamma \to \Delta, \neg\varphi[\bar{x}\leftarrow\bar{t}]}$$

  Note: in all one-premise rules φ, ψ are free subformulas of γ, $\bar{x}$ are all free variables of φ, $\bar{y}$ are all variables of ψ.
- Quantifier rules:

  $$(\to \forall)\quad \frac{\Gamma \to \Delta, \varphi[\bar{x}\leftarrow\bar{t}, y\leftarrow v]}{\Gamma \to \Delta, \forall y\varphi[\bar{x}\leftarrow\bar{t}]} \qquad (\forall \to)\quad \frac{\varphi[\bar{x}\leftarrow\bar{t}, y\leftarrow s], \Gamma \to \Delta}{\forall y\varphi[\bar{x}\leftarrow\bar{t}], \Gamma \to \Delta}$$

  $$(\to \exists)\quad \frac{\Gamma \to \Delta, \varphi[\bar{x}\leftarrow\bar{t}, y\leftarrow s]}{\Gamma \to \Delta, \exists y\varphi[\bar{x}\leftarrow\bar{t}]} \qquad (\exists \to)\quad \frac{\varphi[\bar{x}\leftarrow\bar{t}, y\leftarrow v], \Gamma \to \Delta}{\exists y\varphi[\bar{x}\leftarrow\bar{t}], \Gamma \to \Delta}$$

  Note: In all quantifier rules φ, ψ are free subformulas of γ, $\bar{x}, y$ are all free variables of φ and the variable v has no free occurences in $\Gamma, \Delta, \varphi[\bar{x}\leftarrow\bar{t}]$.

---

[2] Another reason for such representation is that it is actually used in the implementation [Vor90a].

[3] It is sufficient to allow only atomic formulae in the axioms.

[4] We shall freely use the overbar notion for the tuples — whenever $\bar{x}$ is met in the text it denotes the tuple $x_1, \ldots, x_n$.

- Two-premise-rules:

$$(\to\wedge)\quad \frac{\Gamma\to\Delta,\varphi[\bar{x}{\leftarrow}\bar{t}]\qquad \Sigma\to\Xi,\psi[\bar{x}{\leftarrow}\bar{s}]}{\Gamma\theta,\Sigma\theta\to\Delta\theta,\Xi\theta,(\varphi\wedge\psi)[\bar{x}{\leftarrow}\bar{t}\theta]}$$

$$(\vee\to)\quad \frac{\varphi[\bar{x}{\leftarrow}\bar{t}],\Gamma\to\Delta\qquad \psi[\bar{x}{\leftarrow}\bar{s}],\Sigma\to\Xi}{(\varphi\vee\psi)[\bar{x}{\leftarrow}\bar{t}\theta],\Gamma\theta,\Sigma\theta\to\Delta\theta,\Xi\theta}$$

$$(\supset\to)\quad \frac{\Gamma\to\Delta,\varphi[\bar{x}{\leftarrow}\bar{t}]\qquad \psi[\bar{x}{\leftarrow}\bar{s}],\Sigma\to\Xi}{(\varphi\supset\psi)[\bar{x}{\leftarrow}\bar{t}\theta],\Gamma\theta,\Sigma\theta\to\Delta\theta,\Xi\theta}$$

  Note: in all two-premise rules φ,ψ are free subformulas of γ, $\bar{x}$ are all free variables of φ,ψ, θ is a most general unifier of $\bar{s}$ and $\bar{t}$.
- Factoring rules:

$$(f\to)\quad \frac{\varphi[\bar{x}{\leftarrow}\bar{t}],\varphi[\bar{x}{\leftarrow}\bar{s}],\Gamma\to\Delta}{\varphi[\bar{x}{\leftarrow}\bar{t}\theta],\Gamma\theta\to\Delta\theta}\qquad (\to f)\quad \frac{\Gamma\to\Delta,\varphi[\bar{x}{\leftarrow}\bar{t}],\varphi[\bar{x}{\leftarrow}\bar{s}]}{\Gamma\theta\to\Delta\theta,\varphi[\bar{x}{\leftarrow}\bar{t}\theta]}$$

  Note: in the factoring rules φ is a free subformula of γ, $\bar{x}$ are all free variables of φ, θ is a most general unifier of $\bar{s}$ and $\bar{t}$.

Factoring rules are needed for capturing contraction rules. Without them the system is incomplete. The techniques for proving completeness may be found in [Vor90b]:

Theorem 1. *Let γ be a ground formula. Then the sequent $\to\gamma$ is provable in* **Cl** *iff it is provable in* $\mathbf{Cl}_\gamma$*.* □

The main difference between the system $\mathbf{Cl}_\gamma$ and the original system **Cl** is that instead of arbitrary substitutions the most general unifiers are used. This makes the system finite in the following sense:

1. The set of axioms of $\mathbf{Cl}_\gamma$ is finite up to renaming variables.
2. For each sequent (pair of sequents) of $\mathbf{Cl}_\gamma$ there is only a finite number of rules applicable to this sequent (these pair of sequents). For each applied rule the conclusion is unique up to renaming variables.

However in spite of the finiteness the system $\mathbf{Cl}_\gamma$ has a lot of redundancies which will be studied later.

Another example of the specialization of a system is that of **S4**. The set of sequents of **S4** is defined similar to **Cl**. The axioms and rules of **S4** are all axioms and rules of **Cl** plus the rules

$$(\to\Box)\quad \frac{\Box\Gamma\to\Diamond\Delta,\varphi}{\Box\Gamma\to\Diamond\Delta,\Box\varphi}\qquad (\Box\to)\quad \frac{\varphi,\Gamma\to\Delta}{\Box\varphi,\Gamma\to\Delta}$$

$$(\to\Diamond)\quad \frac{\Gamma\to\Delta,\varphi}{\Gamma\to\Delta,\Diamond\varphi}\qquad (\Diamond\to)\quad \frac{\varphi,\Box\Gamma\to\Diamond\Delta}{\Diamond\varphi,\Box\Gamma\to\Diamond\Delta}$$

where $\Box\Gamma=\{\Box\psi\mid\psi\in\Gamma\}$ and $\Diamond\Delta=\{\Diamond\psi\mid\psi\in\Delta\}$. The specialized system $\mathbf{S4}_\gamma$ has the same rules as $\mathbf{Cl}_\gamma$ plus the above four rules restricted to the sequents of $\mathbf{S4}_\gamma$.

The technique of specializing described above is applicable at least to the following calculi:

1. the classical predicate calculus;
2. the intuitionistic predicate calculus;
3. several logics without the contraction rule: the direct logic of Ketonen and Weychrauch [KW84], Grishin's logic without the contraction rule [Gri81], Mey's logic without the contraction rule [Mey89], linear logic [GLT89];
4. Some modal logics: **S4**, **K**.

It can be also extended to some other logics using the technique of hiding special axioms into the unification algorithm [OH91] but we shall not dwell on it here.

Not all the logics are so easy to handle as **S4** or **Cl**. To illustrate rules which are more difficult to specialize we give the example of the rule $(\vee \rightarrow)$ of the intuitionistic sequent calculus **Int**. The difference between **Int** and **Cl** is that there can be at most one formula in the right part (succedent) of a sequent. This seemingly inessential difference produces the rule

$$\frac{\varphi[\bar{x}\leftarrow\bar{s}_1], \Gamma \rightarrow \chi[\bar{x}\leftarrow\bar{t}_1] \qquad \psi[\bar{x}\leftarrow\bar{s}_2], \Delta \rightarrow \chi[\bar{x}\leftarrow\bar{t}_2]}{\varphi\vee\psi[\bar{x}\leftarrow\bar{s}_1\theta], \Gamma\theta, \Delta\theta \rightarrow \chi[\bar{x}\leftarrow\bar{t}_1\theta]}$$

where θ is a most general unifier of $\bar{s}_1, \bar{t}_1$ with $\bar{s}_2, \bar{t}_2$. As one can see, we have to unify not only $\bar{s}_1$ with $\bar{s}_2$, as in the case of classical logic, but also $\bar{t}_1$ with $\bar{t}_2$.

3 Deductive Systems and Strategies

The next problem with an efficient implementation of the inverse method is to introduce strategies reducing the search space. But instead of looking for the strategies for concrete systems we shall introduce a generalization called deductive systems. Intuitively, deductive system is a set of axioms and inference rules operating on a set of clauses[5]. Some clauses are distinguished and called goals. Below we shall formulate a notion of subsumption ordering for the deductive systems which is later used for proving completeness of several strategies for the deductive systems in general.

In the next section we show how to develop strategies for classical logic and for **S4** based on the results from this section. Classical logic and **S4** were chosen only as examples. We do not directly consider here other interesting logical systems like the intuitionistic predicate calculus due to the lack of space. However the results of this section are applicable at least to all logics mentioned in Section 2.

Now we shall proceed to the formal definitions. This part of paper is rather technical but the intuition behind definitions is usually clear and in most of cases we omit explanations.

A triple $\langle \mathbf{C}, \mathbf{G}, \mathbf{R} \rangle$ is called a deductive system iff

1. $\mathbf{C}$ is a set;
2. $\mathbf{G} \subseteq \mathbf{C}$;
3. $\mathbf{R} \subseteq \mathbf{C}^\star \times \mathbf{C}$.

where $\mathbf{C}^\star$ denotes the set of all finite subsets of $\mathbf{C}$.

In what follows the elements of $\mathbf{C}$ will be called clauses, $\mathbf{G}$ is the subset of goal clauses, $\mathbf{R}$ the the set of inference rules of the deductive system. The rule $\langle \{c_1, \ldots, c_n\}, c \rangle$ will usually be written in the form $c_1, \ldots, c_n / c$.

Axiom is an inference rule with $n = 0$. We impose one restriction to deductive systems: if $c_1, \ldots, c_n / c \in \mathbf{R}$ then for every $i \in \{1, \ldots, n\}$, we have $c_i \notin \mathbf{G}$.[6]

[5] Clauses should not necessarily be as in the resolution method. They may be e.g. sequents.
[6] This condition is not essential, but it simplifies many proofs.

The notion of provability is defined as usual. A clause c is provable from the clauses $c_0, \ldots, c_m$ iff there exists a sequence $d_0, \ldots, d_n$ of clauses such that

1. $d_n = c$;
2. for every $i \in \{0, \ldots, n\}$ either $d_i \in \{c_0, \ldots, c_m\}$ or there is a rule $\bar{e}/d_i \in \mathbf{R}$ such that $\bar{e} \subseteq \{d_0, \ldots, d_{i-1}\}$.

Such a sequence $d_0, \ldots, d_n$ is called a proof of c from $c_0, \ldots, c_m$. The number $n+1$ is the length of the proof $d_0, \ldots, d_n$. We shall also say that c is provable from $c_1, \ldots, c_m$ in $n+1$ steps.

A clause c is provable (in n steps) iff it is provable (in n steps) from the empty set of clauses. A set of clauses is successful (in n steps) iff some goal clause $g \in \mathbf{G}$ is provable from this set (in n steps).

The following definition is related to the subsumption strategy of resolution (as well as that of the other methods defined below). Let $\prec$ be some partial ordering on $\mathbf{C}$. It is called a subsumption ordering iff the following two condition hold:

1. Let $R \in \mathbf{R}$ be an inference rule of the form $c_1, \ldots, c_n/c$ and $c_1 \prec c'_1, \ldots, c_n \prec c'_n$. Then either $c \prec c'_i$ for some $i \in \{1, \ldots, n\}$, or there is a rule $R' \in \mathbf{R}$ such that R' takes the form $\bar{e}/c'$ with $e \subseteq \{c'_1, \ldots, c'_n\}$ and $c \prec c'$;
2. If $c \in \mathbf{G}$ and $c \prec c'$ then $c' \in \mathbf{G}$.[7]

Let $\prec$ be a binary relation on clauses. We expand this relation to the sets of clauses in the following way: for $\bar{c}, \bar{c}'$ sets of clauses $\bar{c} \prec \bar{c}'$ iff for every $c \in \bar{c}$ there is a $c' \in \bar{c}'$ such that $c \prec c'$.

It is interesting that some known strategies of the resolution method are in a special way based on the notion of subsumption. It will be clear from the definitions and theorems given below.

In the following we shall use the following notation for provability: $c \vdash^n_{\mathbf{D}} c$ means that c is provable from $\bar{c}$ in n steps in the deductive system $\mathbf{D}$. We usually omit subscripts $\mathbf{D}$ and n when they are clear from the context or unimportant.

Theorem 2. *Let $\bar{c}$ be a successful set of clauses and $\bar{c} \prec \bar{c}'$. Then $\bar{c}'$ is successful. Moreover if for a goal clause g, $\bar{c} \vdash^n g$ then there is a goal clause g' such that $\bar{c} \vdash^m g'$ for some $m \leq n$.*

Proof. Let $d_0, \ldots, d_n$ be a proof of some goal clause c. We construct step by step sequences of clauses $\bar{e}_0, \ldots, \bar{e}_n$ such that for every $i \in \{0, \ldots, n\}$, $\{d_0, \ldots, d_i\} \prec \bar{e}_i$. Let $\bar{e}_{-1}$ be the empty sequence. We suppose that the sequence $\bar{e}_{i-1}$ is already constructed. Consider d_i. If $d_i \in \mathbf{C}$ then there exists a $d \in \bar{c}'$ such that $d_i \prec d$. In this case we let $\bar{e}_i = \bar{e}_{i-1}, d$. Otherwise by the definition of a proof there is $R \in \mathbf{R}$ of the form $d_j, \ldots, d_k/d_i$ with $j, \ldots, k < i$. Let $d'_j, \ldots, d'_k$ be elements of $\bar{e}_{i-1}$ with $d_j \prec d'_j, \ldots, d_k \prec d'_k$. Since $\prec$ is a subsumption ordering then either $d_i \prec d'_l$ for some $l \in \{j, \ldots, k\}$ or there is a rule $R' \in \mathbf{R}$ of the form $\bar{d}/d'_i$ such that $\bar{d} \subseteq \{d'_j, \ldots, d'_k\}$ and $d_i \prec d'_i$. In the former case we let $\bar{e} = \bar{e}_{i-1}$, in the latter $\bar{e}_i = \bar{e}_{i-1}, d'_i$.

It is easy to verify that $\{d_0, \ldots, d_i\} \prec \bar{e}_i$ and that $\bar{e}_i$ is a proof. Consider $\bar{e}_n$. Since $\{d_0, \ldots, d_n\} \prec \bar{e}_n$, then there is $e \in \bar{e}_n$ such that $d_n \prec e$. By the definition of subsumption orderings $e \in \mathbf{G}$. Deleting from $\bar{e}_n$ the part following e we obtain the needed proof of the goal clause e from $\bar{c}'$. □

[7] If $\prec$ is a subsumption ordering and $c \prec d$, then we say that d subsumes c. Sometimes in the literature the notation for "d subsumes c" is $d \leq c$, but we prefer the other way, since subsumption intuitively means that the subsuming clause is better than the subsumed one.

From this theorem the correctness and completeness of the subsumption strategies follows: one can delete from a set of clauses a clause subsumed by another clause from this set:

Theorem 3. *Let $\prec$ be a subsumption ordering, $c_0 \prec c_1$. Then $\{c_0, \ldots, c_n\}$ is successful iff $\{c_1, \ldots, c_n\}$ is successful.*

Proof. Immediately follows from Theorem 2 and from the fact that $\{c_0, \ldots, c_n\} \prec \{c_1, \ldots, c_n\}$ and $\{c_1, \ldots, c_n\} \prec \{c_0, \ldots, c_n\}$. □

Below we introduce some relationships between deductive systems called homomorphisms. If there is a homomorphism from a deductive system to another deductive system then to know whether some set of clauses is successful in the first system it is sufficient to know whether the image of this set under homomorphism is successful in the second system. If the second system is in some way simpler then the first one, it is reasonable to try to construct such a homomorphism. There may be examples of homomorphisms in logical calculi. For example, different refinements of the same calculus can be homomorphic.

This definition of homomorphism is somehow restrictive and we then introduce a modification of the notion of homomorphism called $\prec$-homomorphism, which is based on some subsumption ordering. $\prec$-homomorphisms have the same property as the usual ones: a set of clauses is successful iff the image of this set under homomorphism is successful. In Theorem 6 we investigate some particular example when such a homomorphism exists. Surprisingly this theorem is applicable to many logics. For example, in [Vor90b] we in fact introduced the $\prec$-homomorphic image of a variant of the classical predicate calculus. Examples of application of Theorem 6 are given in the next section.

Let $\mathbf{D}_0 = \langle \mathbf{C}_0, \mathbf{G}_0, \mathbf{R}_0 \rangle$ and $\mathbf{D}_1 = \langle \mathbf{C}_1, \mathbf{G}_1, \mathbf{R}_1 \rangle$ be two deductive systems. The provability in $\mathbf{D}_0$ and $\mathbf{D}_1$ will be denoted by $\vdash_0$ and $\vdash_1$ respectively. A homomorphism from $\mathbf{D}_0$ to $\mathbf{D}_1$ is any mapping $f : \mathbf{C}_0 \to \mathbf{C}_1$ such that

1. If $\bar{c} \vdash_0 d$ then $f(\bar{c}) \vdash_1 f(d)$;
2. If $f(\bar{c}) \vdash_1 f(d)$ then there exists $e \in \mathbf{C}_0$ with $f(e) = f(d)$ such that $\bar{c} \vdash_0 e$.
3. $\bar{c}$ is successful in $\mathbf{D}_0$ iff $f(\bar{c})$ is successful in $\mathbf{D}_1$.

Theorem 4. *Let $\mathbf{D}_0 = \langle \mathbf{C}_0, \mathbf{G}_0, \mathbf{R}_0 \rangle$ and $\mathbf{D}_1 = \langle \mathbf{C}_1, \mathbf{G}_1, \mathbf{R}_1 \rangle$ be two deductive systems. Let $f : \mathbf{C}_0 \to \mathbf{C}_1$ be such that $\mathbf{G}_1 \subseteq f(\mathbf{C}_0)$. Then f is a homomorphism iff the following conditions hold:*

1. *If $\bar{c}/d \in \mathbf{R}_0$, then $f(\bar{c}) \vdash_1 f(d)$;*
2. *If $f(\bar{c})/f(d) \in \mathbf{R}_1$ then for some e with $f(e) = f(d)$, $\bar{c} \vdash_0 e$.*
3. *(a) If $g \in \mathbf{G}_0$ then $\{f(g)\}$ is successful in $\mathbf{D}_1$.*
   (b) If $f(c) \in \mathbf{G}_1$ then $\{c\}$ is successful in $\mathbf{D}_0$.

The definition of a shortening homomorphism is similar to the definition of a homomorphism. A mapping f is a shortening homomorphism iff:

1. If $\bar{c} \vdash_0^n d$ then $f(\bar{c}) \vdash_1^m f(d)$ for some $m \leq n$.
2. If $f(\bar{c}) \vdash_1 f(d)$ then there exists $e \in \mathbf{C}_0$ with $f(e) = f(d)$ such that $\bar{c} \vdash_0 e$.
3. $\bar{c}$ is successful in $\mathbf{D}_0$ iff $f(\bar{c})$ is successful in $\mathbf{D}_1$, and if for a goal clause g we have $\bar{c} \vdash_0^n g$ then there is a goal clause $g' \in \mathbf{G}_1$ such that $f(\bar{c}) \vdash_1^m g'$ for some $m \leq n$.

As we noted above the notion of homomorphism is somewhat restricted and below we introduce a similar notion of $\prec$-homomorphism.

Let $\prec_0$ and $\prec_1$ be subsumption orderings for $\mathbf{D}_0$ and $\mathbf{D}_1$ respectively. A $\langle\prec_0,\prec_1\rangle$-homomorphism (or in the following simply $\prec$-homomorphism) from $\mathbf{D}_0$ to $\mathbf{D}_1$ is any mapping $f:\mathbf{C}_0\to\mathbf{C}_1$ such that

1. If $\bar{c}\vdash_0 d$ then $f(\bar{c})\vdash_1 e$ for some $e\in\mathbf{C}_1$ such that $f(d)\prec_1 e$.
2. If $f(\bar{c})\vdash_1 f(d)$ then there exist $e,h\in\mathbf{C}_0$ such that $f(e)=f(d)$, $e\prec_0 h$, $\bar{c}\vdash_0 h$.
3. $\bar{c}$ is successful in $\mathbf{D}_0$ iff $f(\bar{c})$ is successful in $\mathbf{D}_1$.

There are characterizations of shortening homomorphisms and $\prec$-homomorphisms similar to that of Theorem 4, which are omitted here.

A shortening $\prec$-homomorphism from $\mathbf{D}_0$ to $\mathbf{D}_1$ is any mapping $f:\mathbf{C}_0\to\mathbf{C}_1$ such that

1. If $\bar{c}\vdash_0^n d$ then $f(\bar{c})\vdash_1^m e$ for some $m\leq n$ and $e\in\mathbf{C}_1$ such that $f(d)\prec_1 e$.
2. If $f(\bar{c})\vdash_1 f(d)$ then there exist $e,h\in\mathbf{C}_0$ such that $f(e)=f(d)$, $e\prec_0 h$, $\bar{c}\vdash_0 h$.
3. $\bar{c}$ is successful in $\mathbf{D}_0$ iff $f(\bar{c})$ is successful in $\mathbf{D}_1$, and if for a goal clause $g\in\mathbf{G}_0$ we have $\bar{c}\vdash_0^n$ then there is a goal clause $g'\in\mathbf{G}_1$ such that $f(\bar{c})\vdash_1^m g$ for some $m\geq n$.

Theorem 5. *Let $\mathbf{D}_0=\langle\mathbf{C}_0,\mathbf{G}_0,\mathbf{R}_0\rangle$ and $\mathbf{D}_1=\langle\mathbf{C}_1,\mathbf{G}_1,\mathbf{R}_1\rangle$ be two deductive systems. Let $f:\mathbf{C}_0\to\mathbf{C}_1$ is such that $\mathbf{G}_1\subseteq f(\mathbf{C}_0)$. Then f is a shortening $\prec$-homomorphism iff the following conditions hold:*

1. *If $\bar{c}/d\in\mathbf{R}_0$ then either $\{f(d)\}\prec_1 f(\bar{c})$ or for some $e\in\mathbf{C}_1$ such that $f(d)\prec_1 e$ we have $f(\bar{c})/e\in\mathbf{R}_1$.*
2. *If $f(\bar{c})/f(d)\in\mathbf{R}_1$ then for some $e,h\in\mathbf{C}_0$ we have $f(e)=f(d)$, $e\prec_0 h$, $\bar{c}\vdash_0 h$.*
3. *(a) If $g\in\mathbf{G}_0$ then $f(g)\in\mathbf{G}_1$.*
   (b) If $f(c)\in\mathbf{G}_1$ then $\{c\}$ is successful in $\mathbf{D}_0$.

Note. The notions of a homomorphism and a shortening homomorphism are instances of the notions of a $\prec$-homomorphism and a shortening $\prec$-homomorphism when $\prec_0$ and $\prec_1$ stand for the equality.

Here we will show some general construction of $\prec$-homomorphisms. Let $\mathbf{D}_0=\langle\mathbf{C}_0,\mathbf{G}_0,\mathbf{R}_0\rangle$ be a deductive system. We construct its homomorphic image $\mathbf{D}_0=\langle\mathbf{C}_0,\mathbf{G}_0,\mathbf{R}_0\rangle$ in the following way. Let $f:\mathbf{C}_0\to\mathbf{C}_1$ be any function giving by any $c_0\in\mathbf{C}_0$ some $c_1\in\mathbf{C}_1$ such that $c\vdash c_1$ and $c\prec c_1$. We define $\mathbf{G}_1$ as $\mathbf{G}$ and $\mathbf{C}_1$ as $f(\mathbf{C})$. Define $\mathbf{R}_1$ in the following way. Let $c_1,\ldots,c_n/c\in\mathbf{R}$, $d_1=f(c_1),\ldots,d_n=f(c_n)$ and d be a clause from $\mathbf{C}_1$ with $f(c)\vdash d$. Then $d_1,\ldots,d_n/d\in\mathbf{R}_1$ and all rules from $\mathbf{R}_1$ are obtained in this way. Finally let $\mathbf{D}_1=\langle\mathbf{C}_1,\mathbf{G}_1,\mathbf{R}_1\rangle$.

Note. The construction of $\mathbf{D}_1$ from $\mathbf{D}_0$ is the most simple in the case when there is exactly one clause from $\mathbf{C}_1$ deducible from the given clause. This is true for all particular deductive systems considered in the next section.

Finally we let $\prec_0=\prec_1=\prec$. For this construction $\mathbf{D}_1=\langle\mathbf{C}_1,\mathbf{G}_1,\mathbf{R}_1\rangle$ the following theorem holds.

Theorem 6. *The function f is a shortening $\prec$-homomorphism from $\mathbf{D}_0=\langle\mathbf{C}_0,\mathbf{G}_0,\mathbf{R}_0\rangle$ to $\mathbf{D}_1$.*

Proof. We apply Theorem 5 and prove 1–3b.

1. Obvious.

2. Let $f(\bar{c})/(d)\in\mathbf{R}_1$. Then there is a rule $\bar{a}/e\in\mathbf{R}_0$ such that $f(\bar{a}) = f(\bar{c})$ and $f(e) = f(d)$. For a set of clauses $\bar{b}$ we denote by $\bar{c}\vdash\bar{b}$ the fact that for every $b\in\bar{b}$ $\bar{c}\vdash b$. Since $\bar{a}\prec f(\bar{c})$ and $a\vdash_0 e$ then there is e_1 such that $e\prec e_1$ and $f(\bar{c})\vdash e_1$. Since $e\prec e_1$ and $e\vdash_0 f(d)$ then there is a clause h such that $f(d)\prec h$ and $f(\bar{c})\vdash_0 h$. Applying $\bar{c}\vdash_0 f(\bar{c})$ and $e\prec f(d)$ we obtain $\bar{c}\vdash_0 h$ and $e\prec h$.
3. (a) It is easy to see that if $g\in\mathbf{G}_0$ then $f(g) = g$.
   (b) Trivially follows from $\mathbf{G}_1 = \mathbf{G}_0$. □

How to apply Theorem 6? For this we have to find a function f with the above properties. Theorem 6 does not help very much in the case of resolution. The only non-trivial examples of clauses c, d with $c\vdash d$ and $c\prec d$ in the resolution system are subsuming factors. Fortunately in the logical systems obtained by the inverse method the examples are numerous.

A clause c is called maximal iff there is no clause d with $c\vdash d$ and $c\prec d$. If there is a set of clauses which are obviously non-maximal, then one can try to refine the deductive system so as not to consider these clauses, constructing the homomorphic image. It could be especially efficient if one can discover such sets of clauses during preprocessing. In the subsequent sections we show that examples of non-maximal clauses are numerous in $\mathbf{Cl}_\gamma$ and $\mathbf{Mod}_\gamma$. The strength of the notion of non-maximal clauses is also demonstrated by the example from Section 5.

It is generally known that we may not to generate tautologies in the resolution method. We can say that the set of all tautologies is useless in resolution systems. Below we shall formulate a rather general theorem which gives some sufficient conditions for a set of clauses to be useless. It is interesting that in particular this theorem applies directly to the tautologies in the resolution method[8].

A set of clauses U is called useless iff for any clauses $c_1,\ldots,c_n\in\mathbf{C}$, if there exists a proof of a goal $g\in\mathbf{G}$ from $c_1,\ldots,c_n$ then there exists a proof of some $g_1\in\mathbf{G}$ from $c_1,\ldots,c_n$ in which no clause from U occurs. It is easy to note that the set of all tautologies of resolution (i.e. clauses containing a contrary pair of literals) is useless. The following definition and Theorem 7 below allow one to prove uselessness of a set of clauses.

Let $\mathbf{D} = \langle\mathbf{C},\mathbf{G},\mathbf{R}\rangle$ be a deductive system and $\prec$ a subsumption ordering on $\mathbf{D}$. A set of clauses $U\subseteq\mathbf{C}$ is called $\prec$-decreasing iff

1. $U\bigcap\mathbf{G} = \emptyset$;
2. For any inference rule $\bar{c}/c_0$ such that $\bar{c}\bigcap U \neq \emptyset$, either $c_0\in U$ or $\{c_0\}\prec\bar{c}$.

Theorem 7. *If U is $\prec$-decreasing then it is useless.*

Proof. Let U be a $\prec$-decreasing set of clauses. We call a proof in $\mathbf{D}$ clear iff no clause from U occurs in this proof. First we prove the following statement:

> Let the clause c is provable from $c_1,\ldots,c_n$. Then either there is a clause $d\in U$ with $c\prec d$ or there is a clear proof from $c_1,\ldots,c_n$ of a clause d such that $c\prec d$.

We use induction on the length l of the proof of c from $c_1,\ldots,c_n$.

1. $l = 1$. In this case the proof consists of one clause c. If $c\in U$ then the statement is trivially satisfied. If $c\notin U$ then the proof is clear.

[8] We had discovered some useless classes of sequents during the first implementation of LISS [Vor90a] in 1987. Only later we had found out that the useless classes of sequents relate to the notion of subsumption in the same way as tautologies to subsumption in the resolution method.

2. $l \neq 1$. The proof takes the form $d_1, \ldots, d_l$ where $d_1 = c$. If $c \in \{c_1, \ldots, c_n\}$ then we prove as in case 1. Let $c \notin \{c_1, \ldots, c_n\}$. Then c is obtained by applying a rule $e_1, \ldots, e_k / c$ where $\{e_1, \ldots, e_k\} \subseteq \{d_1, \ldots, d_{l-1}\}$.By inductive hypothesis $e_1, \ldots, e_k$ satisfy the statement. Consider two cases.
   (a) For all e_i there exists a clear proof of some f_i such that $e_i \prec f_i$. By the definition of subsumption orderings either $c \prec e_i$ for some i or there is a rule $f_1, \ldots, f_m / f \in \mathbf{R}$ with $c \prec f$. In the former case take f as d, in the latter take e_i as d.
   (b) There is some e_i, say e_1, and $f \in U$ such that $e_i \prec f$. Thus $\{e_1, \ldots, e_k\} \prec \{f, e_2, \ldots, e_k\}$. By the definition of a subsumption ordering either $c \prec \{f, e_2, \ldots, e_k\}$, from which the theorem trivially follows, or there exists a rule $\bar{a}/h$ with $\bar{a} \subseteq \{f, e_2, \ldots, e_k\}$ and $c \prec h$. Let $f \in \bar{e}$. Since $f \in U$ and U is $\prec$-decreasing, then either $h \in U$ or $h \prec \{f, e_2, \ldots, e_k\}$. In both cases c satisfies the statement. Now let $\bar{a} \subseteq \{e_2, \ldots, e_k\}$. If all $e_2, \ldots, e_k$ have clear proofs, then the statement of the theorem is satisfied, otherwise we continue as before for $e_1, \ldots, e_k$.

To complete the proof we note that if a goal clause g is provable from $c_1, \ldots, c_n$ then either $g \prec u$ for some $u \in U$ (which is impossible since $U \bigcap \mathbf{G} = \emptyset$) or there is a clear proof of some g_1 from $c_1, \ldots, c_n$ such that $g \prec g_1$ (and hence $g_1 \in \mathbf{G}$). □

4 Applications to Particular Logics

In this section we apply the general techniques from the previous section to refine the specialized systems $\mathbf{Cl}_\gamma$ and $\mathbf{S4}_\gamma$ from Section 2. In both cases we shall denote the corresponding deductive systems by the same notations.

We start from $\mathbf{Cl}_\gamma$. Let $\mathbf{Cl}_\gamma$ be the following deductive system $\langle \mathbf{C}, \mathbf{G}, \mathbf{R} \rangle$:

1. The set $\mathbf{C}$ of clauses is the set of all sequents of $\mathbf{Cl}_\gamma$;
2. The set $\mathbf{R}$ of inference rules is that of the axioms and inference rules of $\mathbf{Cl}_\gamma$;
3. $G = \{\rightarrow \gamma\}$.

Now we define a binary relation $\prec_{\mathrm{Cl}}$ on $\mathbf{C}$ in the following way: $c \prec_{\mathrm{Cl}} d$ iff there is a substitution θ such that for every formula $\psi \in d\theta$ there exists a formula $\varphi \in c$ such that φ is a free subformula of ψ.

Theorem 8. *The relation $\prec_{\mathrm{Cl}}$ is a subsumption ordering.*

Note. There is a close relation between proofs in Gentzen-type systems and resolution proofs pointed out by Maslov for classical logic [Mas69] and considered in more details by Mints [Min90]. In principle we could easily modify the subsumption strategy of resolution to formulate a similar strategy for the Gentzen-type predicate calculus. But in this case we obtain a weaker notion of subsumption. This weaker relation $\prec$ may be obtained from our definition of the ordering $\prec_{\mathrm{Cl}}$ by replacing the words "is free subformula" to the words "is identical to". It is weaker e.g. because under our subsumption ordering a sequent consisting of a free subformula φ of γ can be subsumed by any free subformula ψ of γ such that φ is a subformula of ψ. But under the usual definition of subsumption φ cannot be subsumed by any non-goal sequent different from φ.

Now we apply Theorem 6 for the case of $\mathbf{Cl}_\gamma$.

Theorem 9. *A sequent $c \in \mathbf{C}$ is not maximal if there is a formula $\varphi \in c$ such that φ is an immediate subformula of a subformula of γ of one of the following forms: $(\varphi_1 \vee \varphi_2)^+$, $(\varphi_1 \supset \varphi_2)^+$, $\neg\varphi$, $(\varphi_1 \wedge \varphi_2)^-$, $(\exists x \varphi_1)^+$, $(\forall x \varphi_1)^-$.*

The immediate application of this theorem is that we can exclude these non-maximal formulas from the considerations. It is implemented in the system LISS [Vor90a] in the following way: on the preprocessing phase we construct the calculus in which the formulas which are not maximal according to Theorem 9 may not occur.

Now consider Theorem 7 applied to $\mathbf{Cl}_\gamma$. Let the set $U_{\mathrm{Cl}} = U_1 \bigcup U_2$ where

$U_1 = \{c \mid \exists \varphi_1, \varphi_2 \in c$ such that there is a free subformula $(\psi_1 \wedge \psi_2)^+$ (or $(\psi_1 \vee \psi_2)^-$, or $(\psi_1 \supset \psi_2)^-$), $\bar{x}$ are all free variables of ψ_1, ψ_2, and there exists a tuple of terms $\bar{t}$ such that φ_1 is a subformula of $\psi_1[\bar{x} \leftarrow \bar{t}]$, φ_2 is a subformula of $\psi_2[\bar{x} \leftarrow \bar{t}]\}$;

$U_2 = \{c \mid \forall \varphi \in c$, φ is an immediate subformula of $(\forall x \varphi_1)^+$ or $(\exists x \varphi_1)^-$.

Theorem 10. *The set U_{Cl} is $\prec$-decreasing.*

Note. There is an algorithm to verify whether $c \in U_{\mathrm{Cl}}$ which is not much more complicated than the algorithm verifying whether a clause in the resolution method is a tautology.

The strategy of deleting the sequents belonging to U_{Cl} from the search space proved to be very strong on many tests. The reason for it is that usually (as well as for the resolution tautologies) many rules are applicable to clauses from U_{Cl} and that sometimes clauses from U_{Cl} are hard to subsume.

Now we shall consider modal logic **S4**.

If φ is a subformula of ψ then <u>path from φ to ψ</u> is the sequence of formulae $\chi_1, \ldots, \chi_n$ such that $\varphi = \chi_1$, $\psi = \chi_n$ and for every $i \in \{1, \ldots, n-1\}$ χ_i is an immediate subformula of χ_{i+1}.

The specialized system $\mathbf{Mod}_\gamma$ is defined like $\mathbf{Cl}_\gamma$. To define a subsumption ordering for $\mathbf{Mod}_\gamma$ let us first introduce a relation $\leq_{\mathrm{Mod}}$ between subformulas of γ. Let φ, ψ be subformulas of γ, φ is a subformula of ψ and $\pi = (\varphi_0, \ldots, \varphi_n)$ is the path from φ to ψ. Then $\varphi \leq_{\mathrm{Mod}} \psi$ iff

1. φ is a free subformula of ψ;
2. ψ takes the form $(\Box \psi_1)^-$ or $(\Diamond \psi_1)^+$ or there is no subpath in π of the form $(\Box \chi_1)^-, \chi_2$ or $(\Diamond \chi_1)^+, \chi_2$.

Now we let $c \prec_{\mathrm{Mod}} d$ iff there is a substitution θ such that for every formula $\psi \in d\theta$ there exists a formula $\varphi \in c$ such that $\varphi \leq_{\mathrm{Mod}} \psi$.

Theorem 11. *The relation $\prec_{\mathrm{Mod}}$ is a subsumption ordering.*

The following theorem gives us examples of non-maximal sequents for $\mathbf{Mod}_\gamma$:

Theorem 12. *A sequent $c \in \mathbf{C}$ is not maximal if there is a formula $\varphi \in c$ such that one of the following conditions hold:*

1. *φ is an immediate subformula of one of the following subformulas of γ: $(\varphi_1 \vee \varphi_2)^+$, $(\varphi_1 \supset \varphi_2)^+$, $\neg \varphi$, $(\varphi_1 \wedge \varphi_2)^-$, $(\exists x \varphi_1)^+$, $(\forall x \varphi_1)^-$ and φ does not take the form $(\Box \chi)^-$ or $(\Diamond \chi)^+$;*
2. *φ is the immediate subformula of a subformula of γ of the form $(\Box \chi)^-$ or $(\Diamond \chi)^+$.*

Now let $U_{\mathrm{Mod}} = \{c \mid \exists \varphi_1, \varphi_2 \in c$ such that there is a free subformula of γ of the form $(\psi_1 \wedge \psi_2)^+$ (or $(\psi_1 \vee \psi_2)^-$, or $(\psi_1 \supset \psi_2)^-$), and there is a substitution θ such that $\varphi_1 \leq_{\mathrm{Mod}} \psi_1 \theta$, $\varphi_2 \leq_{\mathrm{Mod}} \psi_2 \theta\}$;

Theorem 13. *The set U_{Mod} is $\prec$-decreasing (in $\mathbf{Mod}_\gamma$).*

Note. In the case of **S4** it is possible to find some other natural $\prec$-decreasing sets. For example, we can add to U the set $\{c \mid \exists\varphi, \psi \in c$ such that φ is the immediate subformula of a subformula of γ of the form $\forall x\varphi$, ψ is the immediate subformula of $\Box\psi$ and x has a free occurrence to $\psi\}$.

There are other applications of the techniques introduced in this paper. For example in the case of logics without the contraction rule the notion of $\prec$-decreasing set of clauses allows one to prove decidability of some of the systems mentioned in the introduction:

Theorem 14. *Let* **D** *be a deductiove system based on any of the following logics:*

1. *Direct logic of Ketonen and Weyhrauch [KW84];*
2. *The logic of Grishin [Gri81];*
3. *The logic of Mey [Mey89];*
4. *The linear fragment of linear logic [GLT89].*

Let U be the set

$\{c \mid$ *There are $\varphi_1, \varphi_2 \in c$ and free subformulae ψ_1, ψ_2 of γ such that*
(a) ψ_1 is a subformula of ψ_2;
(b) $\varphi_1, \varphi_2 \in c$ have different occurences in c $\}$.

Then

1. *U is $\prec$-decreasing;*
2. *For every $\bar{c} \in \mathbf{C}$, the set $\{d \mid \bar{c} \vdash d$ and $d \notin U\}$ is finite.*

Corollary 15. *All logics from Theorem 14 are decidable.*

5 An example

All the techniques described here are implemented in the theorem proving system LISS [Vor90a]. It may seem from the definition that the algorithms for implementing the strategies are very complicated. However in practice they are not so different from those of resolution. The algorithms for applying inference rules of the specialized systems, for checking subsumption and for $\prec$-decreasing sets are very similar to the application of the resolution rule, checking for subsumption and checking for tautologies in the resolution method. All the work connected with the non-maximal clauses is done during the preprocessing phase.

Consider a simple example from the paper of Mints [Min90]. Let γ be the following formula to be proved in **S4**:

$$\gamma = \Box\neg\Box\neg(\Box a \vee \Box\neg\Box a)$$

It has the following free subformulae:

$$\begin{array}{llll}
\varphi_1^+ = \neg\Box\neg(\Box a \vee \Box\neg\Box a) & & \varphi_6^+ = a \\
\varphi_2^- = \Box\neg(\Box a \vee \Box\neg\Box a) & & \varphi_7^+ = \Box\neg\Box a \\
\varphi_3^- = \neg(\Box a \vee \Box\neg\Box a) & & \varphi_8^+ = \neg\Box a \\
\varphi_4^+ = \Box a \vee \Box\neg\Box a & & \varphi_9^- = \Box a \\
\varphi_5^+ = \Box a & & \varphi_{10}^- = a
\end{array}$$

According to Theorem 12 any sequent containing one of the formulae $\varphi_3, \varphi_4, \varphi_5, \varphi_7, \varphi_{10}$ is not maximal, so we may omit them. Thus refined system $\mathbf{Mod}_\gamma$ have only one axiom $\varphi_9 \to \varphi_6$ and five inference rules:

$$(r1)\quad \frac{\Box\Gamma \to \Diamond\Delta, \varphi_1}{\Box\Gamma \to \Diamond\Delta, \gamma} \qquad (r2)\quad \frac{\varphi_2, \Gamma \to \Delta}{\Gamma \to \Delta, \varphi_1} \qquad (r3)\quad \frac{\Box\Gamma \to \Diamond\Delta, \varphi_6}{\varphi_2, \Box\Gamma \to \Diamond\Delta}$$

$$(r4)\quad \frac{\Box\Gamma \to \Diamond\Delta, \varphi_8}{\varphi_2, \Box\Gamma \to \Diamond\Delta} \qquad (r5)\quad \frac{\varphi_9, \Gamma \to \Delta}{\Gamma \to \Delta, \varphi_8}$$

The proof consists of 5 steps:

1. $\varphi_9 \to \varphi_6$	[axiom]	4. $\varphi_2 \to$	[rule $(r4)$ from 3]
2. $\varphi_2, \varphi_9 \to$	[rule $(r3)$ from 1]	5. $\to \varphi_1$	[rule $(r2)$ from 4]
3. $\varphi_2 \to \varphi_8$	[rule $(r5)$ from 2]	6. $\to \gamma$	[rule $(r1)$ from 5]

The proof search in our system was deterministic. The only generated clauses that were not immediately subsumed are exactly the ones shown in the proof. The maximal number of clauses kept in the memory during the proof search is 2, because clause 1 is subsumed by clause 2 and clause 4 subsumes all previously generated clauses. In the prover implemented by Basylev, which is based on the presentation from [Min90] there are 9 initial clauses, the number of clauses actually used in the proof is 15 and the total number of generated clauses is 50 (the statistics is taken from [Min90]).

6 Related Work

In our opinion the notion of strategy for automated deduction is of a great importance and needs more research. At the time we know only a few papers which consider the relation between strategies for different proof systems.

In the paper of Maslov [Mas69] the relation between strategies of resolution and those of inverse method are considered. In [AEH90] some strategies of resolution were generalized to modal logics. However in both papers the definitions are much more concrete than in our paper. We do not know any other paper in which definitions are as abstract and applicable to different kinds of logics as in our paper.

The possibility of applying the inverse method to non-classical logics as well was noted originally by Maslov. However this application was not intensively studied. A general method of theorem proving in non-classical logics was studied in [Wal90]. He uses the connection method [Bib82] instead of the inverse method. Some techniques used in the proofs in [Wal90] are very similar to our techniques. But in our opinion the inverse method applied to non-classical logics is much simpler than the connection method. The efficiency of the two methods is difficult to compare. To the best of our knowledge no strategies for the connection method (like our subsumption strategy) have been developed.

A very similar treatment of the inverse method had been proposed by Mints [Min90]. He views the inverse method as a transformation of the formula to be proved to a set of clauses in a way similar to ours. The main difference is that we make more efforts to the classification of rules,while the clauses are very simple. Mints has more different types of clauses and less different types of rules. In our opinion, the stress on rules is more efficient, because the rules can be *compiled*. Mints does not consider strategies like ours.

Proof search in the intuitionistic sequent calculus was recently considered by Shankar [Sha92]. His system is based on a variant of tableau method. To decrease the search space he considers impermutabilities [Kle52] among the rules of intuitionistic predicate logic. We achieve almost the same effect by using subsumption and the notion of non-maximal clauses. It is also possible to use the information about permutabilities in our approach. It is interesting to see if this information can improve performance. Shankar uses top-down linear strategy of derivation.

Our techniques concerned with with non-maximal clauses can also be used in Shankar's system. Intuitively, if a clause is not maximal, then there is a rule of the sequent calculus, to which one can commit, not considering other possible rules. What is important, it is possible to find a lot of non-maximal clauses during preprocessing, thus compiling more efficient rules. In Shankar's system it could possibly be achieved by a kind of abstract interpretation of the set of his Prolog-like rules.

Acknowledgements

Norbert Eisinger had made many comments which helped to improve this paper. Mark Wallace had corrected my English in the preliminary version of the paper.

References

[AEH90] I. Auffray, P. Enjalbert, and J-J. Hebrard. Strategies for modal resolution: Results and problems. *Journal of Automated Reasoning*, 6:1–38, 1990.

[Bib82] W. Bibel. *Automated theorem proving.* Vieweg Verlag, 1982.

[GLT89] J-Y. Girard, Y. Lafont, and P. Taylor. *Proofs and Types.* Cambridge University Press, 1989.

[Gri81] V.N. Grishin. Predicate and set-theoretic calculi based on logic without the contraction rule (in Russian). *Izvestiya Akad. Nauk SSSR*, 45(1):47–68, 1981.

[KW84] J. Ketonen and R. Weyhrauch. A decidable fragment of predicate calculus. *Theoretical Computer Science*, 32(3):297–309, 1984.

[Kle52] S.C. Kleene. Permutability of inferences in Gentzen's calculi LK and LJ. *Memoirs of the American Mathematical Society*, 10, 1952.

[Lif86] V. Lifschitz. What is the inverse method? *Journal of Automated Reasoning*, 5(1):1–24, 1986.

[Mas67] S.Yu. Maslov. An inverse method for establishing deducibility of nonprenex formulas of the predicate calculus. In J.Siekmann and G.Wrightson, editors, *Automation of Reasoning (Classical papers on Computational Logic)*, volume 2, pages 48–54. Springer Verlag, 1983.

[Mas69] S.Yu. Maslov. Relationship between tactics of the inverse method and the resolution method. In J.Siekmann and G.Wrightson, editors, *Automation of Reasoning (Classical papers on Computational Logic)*, volume 2, pages 264–272. Springer Verlag, 1983.

[Mas87] S.Yu. Maslov. *Theory of Deductive Systems and its Applications.* MIT Press, 1987.

[Mey89] D. Mey. *Private communications.*

[Min90] G. Mints. Gentzen-type systems and resolution rules. Part I. Propositional logic. In P. Martin-Löf and G. Mints, editors, *COLOG-88*, volume 417 of *Lecture Notes in Computer Science*, pages 198–231. Springer Verlag, 1990.

[OH91] H.-Yü. Ohlbach and A. Herzig. Parameter structures for parameterized modal operators. In *Proc. IJCAI'91*, pages 512–517, 1991.

[Sha92] N. Shankar. Proof search in the intuitionistic sequent calculus. In *12th International Conference on Automated Deduction*, Lecture Notes in Computer Science, 1992.

[Vor90a] A.A. Voronkov. LISS - the logic inference search system. In Mark Stickel, editor, *Proc. Int. Conf. on Automated Deduction*, volume 449 of *Lecture Notes in Computer Science*, pages 677–678, Kaiserslautern, Germany, 1990. Springer Verlag.

[Vor90b] A.A. Voronkov. A proof-search method for the first order logic. In P.Martin-Lof and G.Mintz, editors, *COLOG'88*, volume 417 of *Lecture Notes in Computer Science*, pages 327–340. Springer Verlag, 1990.

[Wal90] L.A. Wallen. *Automated Deduction in Nonclassical Logics.* MIT Press, 1990.

[Zam89] N.K. Zamov. Maslov's inverse method and decidable classes. *Annals of Pure and Applied Logic*, 42(2):165–194, 1989.