Towards the Automatic Construction of Schematic Proofs

David M. Cerna[1]*and Michael Lettmann[2]†

[1] Reseach Institute for Symbolic Computation
Johannes Kepler University, Linz, Austria
david.cerna@risc.jku.at
[2] Institute of Information Systems
Technische Universität Wien, Vienna, Austria
lettmann@logic.at

Abstract

In recent years *schematic representations* of proofs by induction have been studied for their interesting proof theoretic properties, i.e. allowing extensions of Herbrand's theorem to certain types of inductive proofs. Most of the work concerning these proof theoretic properties presented schematic proofs as sets of proofs connected by *links* together with a global soundness condition. Recently, the $\mathcal{S}i$**LK**-calculus was introduced which provides inferences for expanding the sets of proofs within a schematic proof as well as introducing links without violating the soundness condition. In this work we discuss a simplification of the $\mathcal{S}i$**LK**-calculus which isolates the essential mechanisms and provides a path towards the automated construction of schematic proofs.

1 Introduction

Proof schemata can be thought of as a recursive representation of an infinite sequence of finite proofs. This is in contrast to the formalism of Brotherston and Simpson [1] which concerns infinitely deep proof constructions. Proof schemata comprise of a set of *proof components* which are **G3c**-proofs [11] allowing additional initial sequents referred to as *links* referring to other proof components in the set associated with the given proof schemata. There are rules concerning the linking structure providing a sufficient condition for soundness, though not a necessary condition. The development of a calculus with an implicit soundness condition which is both sufficient and necessary was addressed in [4]. In this work the so called $\mathcal{S}i$**LK**-calculus was developed which allows one to expand the set of proof components and link proofs soundly. The core idea of the $\mathcal{S}i$**LK**-calculus is to prove a pair of sequents where the first sequent in the pair (the base case) defines a goal for the second sequent in the pair (the step case). The goal is defined by the $\circlearrowleft$ rule or "cycling" rule. If one is able to use the rules of the $\mathcal{S}i$**LK**-calculus to match the step case sequent with the goal than one is able to close the pair and has constructed a proof schema component. Interestingly enough the semantic interpretation of the objects

*This research has been partially supported by the Austrian Science Fund (FWF) under the project P 28789-N32.

†Funded by FWF project W1255-N23.

resulting from a $\mathcal{S}i\mathbf{LK}$-derivation perfectly matches the construction provided by Gentzen in [8] thus providing intuition concerning the nature of proof schema.

Unfortunately, the $\mathcal{S}i\mathbf{LK}$-calculus is quite complex and hard to use in practice. However, this mechanized approach to the construction of proof schemata begs the question, can a $\mathcal{S}i\mathbf{LK}$-calculus like approach to proof schema construction be used for inductive theorem proving. In this abstract, we provide an improved $\mathcal{S}i\mathbf{LK}$-calculus based on so called *super sequents*. Like $\mathcal{S}i\mathbf{LK}$-proofs, super sequent proofs can be translated to proof schemata and thus are equivalent to certain types of proof by induction. Also the intuition behind the "cycling" rule is essentially the same. Furthermore, super sequents go beyond the current understanding of proof schema by allowing certain types of mutual recursion. In what follows, we give a short introduction to super sequents and how they can be used for inductive theorem proving.

2 A Brief Introduction to Schematic Languages

In the following sections, **G3c**-calculus refers to the calculus of [11] for first-order logic. For the construction of proof schema, we extend the **G3c**-calculus by introducing two term sorts, links as initial sequents and an equational rule for *recursive defined symbols*. The standard first order individual sort will be denoted by ι, and ω will denote the *numeric sort*, which only contains terms constructed from $\{s(\cdot), 0\}$. Each sort will have its own countably infinite set of variables $\mathcal{V}_\iota$ and $\mathcal{V}_\omega$, respectively. Links can technically be any sequent, and in some sense are a theory extension. In order to guarantee soundness and consistency, we restrict links to proofs which are contained in the proof component set of a proof schema (see [4]).

There are two types of defined symbols, *defined function symbols* and *defined predicate symbols* which allow primitive recursive constructions in the language. We assume a set of convergent rewrite rules $\mathcal{E}$ (equational theory) for defined function and predicate symbols. The rules of $\mathcal{E}$ are of the form $\widehat{f}(\bar{t}) = E$, where $\bar{t}$ contains no defined symbols, and either $\widehat{f}$ is a function symbol and E is a term or $\widehat{f}$ is a predicate symbol and E is a formula schema. The rules can be applied in both directions, i.e. the $\mathcal{E}$-rule is reversible. The extension of the **G3c**-calculus incorporating the above constructions is referred to as the **G3cS**-calculus.

A proof schema is a set of pairs of **G3cS**-calculus (a basecase and stepcase proof) connected by links such that replacement of $\mathcal{V}_\omega$ variables by numerals results in a **G3c**-proof after normalization. For more details see one of the following works [2, 3, 4, 5, 7, 9](Instead of the **G3c**-calculus, they use the **LK**-calculus as a basis).

3 A Super Sequent Approach

Super sequents are a variant of the abstraction introduced by [6] for modal logic which the author referred to as *leveled sequents*. A leveled sequent of order n is a sequent containing sequents of order less than n. This allows one to define inference rules specifically for sequents of a particular order. We use the leveled sequent concept to differentiate between sequents which have logical meaning, i.e. *schematic sequents*, and sequents which need to be assumed in order to justify the schematic sequents, what we refer to as *meta sequents*.

Schematic sequents, as in previous work, are pairs of multisets of formula schemata Δ, Π denoted by $\Delta \vdash \Pi$, where formula schemata are first order formulas which may contain defined symbols. To deal with mutual recursion as we do in our even/odd example, both of these concepts can be extended to *hyper schematic (meta) sequents* or a list of schematic (meta)

sequents. Note that hyper sequents differ from leveled sequents in that they do not consider a stratification of the meta-level syntax [10].

Definition 1. *Let* $\Pi \vdash \Delta$ *be a schematic sequent and* $Var_\omega(\Pi \cup \Delta) = \{x_1, \cdots, x_n\}$ *the set of variables over the numeric sort. The sequent* $\forall x_1, \cdots, x_n\,(\Pi \vdash \Delta)$ *is a* schematic meta sequent.

Meta sequents represent possibly valid sequents, i.e. a temporary theory extension. A *super sequent* is a construction of the form $\mathscr{F} \Rrightarrow \mathscr{G}$ where $\mathscr{G}$ is a schematic hyper sequent and $\mathscr{F}$ is a schematic meta hyper sequent. Super sequents are to be interpreted as

$$\mathcal{I}_{\mathcal{S}}(\mathscr{F} \Rrightarrow \mathscr{G}) = \mathcal{I}\left(\bigwedge_{S\in\mathscr{F}} \mathcal{I}_M(S) \;\vdash\; \bigvee_{S\in\mathscr{G}} \mathcal{I}(S)\right) \qquad \mathcal{I}(\Pi \vdash \Delta) = \left(\bigvee_{f\in\Pi} \neg f\right) \vee \left(\bigvee_{f\in\Delta} f\right)$$

$$\mathcal{I}_M(\Pi \vdash \Delta) = \forall x_1, \cdots, x_n\,(\mathcal{I}(\Pi \vdash \Delta)), \text{ for } \mathcal{V}_\omega(\Pi \vdash \Delta) = \{x_1, \cdots, x_n\}.$$

An interesting and important example is the super sequent $\vdash P(m) \Rrightarrow \vdash P(n)$ which when interpreted results in $\mathcal{I}_S\left(\vdash P(m) \Rrightarrow \vdash P(n)\right) = (\neg\forall m P(m)) \vee P(n)$. A formula which is obviously derivable in the **G3c**-calculus.

We will refer to the following calculus as the $\mathcal{S}uper$ **G3c**-calculus being that derivations in the calculus are trees of *super sequents*. When the specific sequent structure is not necessary for understanding we will abbreviate sequents by capital latin characters.

The idea behind a super sequent construction is that the super antecedent represents the necessary theory allowing the super succedent to hold. In the above case, given that $\vdash P(m)$ holds, it is obvious that for any n, $P(n)$ holds. If we are able to construct $P(s(n))$ from $P(n)$ and if $P(0)$ is provable, then the theory is not necessary and can be dropped implying that $P(n)$ is provable by induction alone without any assumptions. The calculus is as follows:

$$\frac{}{\Rrightarrow \Gamma, A \vdash A, \Delta}\, Ax \qquad\qquad \frac{}{\Rrightarrow \Gamma \vdash \top, \Delta}\, Ax$$

We introduce two special parameter types, active and passive for the $\mathcal{S}uper$ **LK**-calculus. Active parameters are introduced by the $\circlearrowright$ rule and will be represented by lowercase latin characters. The intuition behind these variables is that they represent iteration of the induction invariant and thus are treated as a special constant which cannot be quantified. Passive parameters (bold lower case greek letters think eigenvariables) are either introduced by the axiom rule or by the $\Downarrow$ rule. In the following m is the only active parameter, $\boldsymbol{\alpha}$ is a fresh passive parameter, T is a sequent, and $S(0)$ is active parameter free.

$$\frac{\mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(0)}{S(x) \;\big|\; \mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(m) \circ T}\,{}^{1}\circlearrowright \qquad\qquad \frac{S(x) \;\big|\; \mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(s(m))}{\mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(\boldsymbol{\alpha}) \circ T}\,\Downarrow$$

Note that $S(s(m))$ should only contain the active parameter m. Once a sequent succedent of the super sequent is active parameter free it essentially becomes part of the background theory allowing us to use it as an initial sequent. Thus we allow the following rules which mimics *external weakening & contraction*:

$$\frac{\mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(\boldsymbol{\alpha_1}, \ldots, \boldsymbol{\alpha_n})}{\mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(\boldsymbol{\alpha_1}, \ldots, \boldsymbol{\alpha_n}) \;\big|\; S(t_1, \ldots, t_n) \circ T}\, Th$$

$$\frac{\mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(\boldsymbol{\alpha_1}, \cdots, \boldsymbol{\alpha_n}) \circ T \;\big|\; S(t_1, \cdots, t_n)}{\mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(\boldsymbol{\alpha_1}, \cdots, \boldsymbol{\alpha_n}) \circ T}\, Kn$$

$$\frac{\mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(\boldsymbol{\alpha_1}, \cdots, \boldsymbol{\alpha_n}) \;\big|\; Q(\boldsymbol{\beta}_1, \cdots, \boldsymbol{\beta}_n)}{\mathscr{F} \Rrightarrow \mathscr{G} \;\big|\; S(\boldsymbol{\gamma_1}, \cdots, \boldsymbol{\gamma_n}) \vee Q(\boldsymbol{\gamma_1}, \cdots, \boldsymbol{\gamma_n})}\, Gen$$

where $S(\boldsymbol{\alpha_1}, \ldots, \boldsymbol{\alpha_n})$ is active parameter free and t is an arbitrary numeric term.

While these external rules are reasonable given their existence within the standard hyper sequent calculi for modal logics, concepts such as communication do not fit our interpretation.

[1] $(\Gamma_1 \vdash \Delta_1) \circ (\Gamma_2 \vdash \Delta_2) = \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2$

Essentially, when we say $S \mid Q$ we mean both S and Q can be constructed from the given input. The *Gen* captures part of this meaning by introducing a disjunction over a fresh unified eigenvariable.

Unary rules of the first order **G3c**-calculus may be applied to any sequent of the hyper sequent in the succedent of the super sequent. Binary rules of the first order **G3c**-calculus may be applied between two super sequents by choosing one sequent from the hyper sequent in the succedent and join the hypersequents of the antecedent and succedent of the super sequents together. When applying these rules the number of active parameters in the sequents of hyper sequent in the succedent of the super sequents must remain at most one. This can be managed by renaming if necessary.

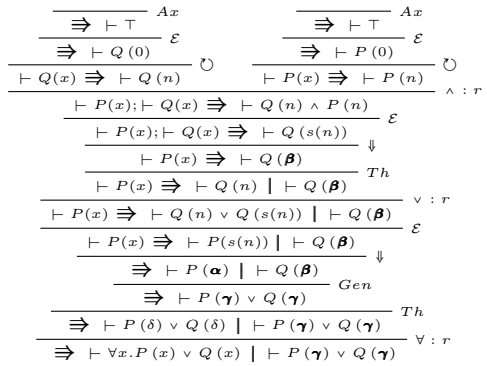
4 Automating Abstractions

Consider the equational theory $\mathcal{E} = \{P(s(n)) = Q(n) \vee Q(s(n)), P(0) = \top, Q(s(n)) = P(n) \wedge Q(n), Q(0) = \top\}$. We prove $\forall x.P(x) \vee Q(x)$. In order to illustrate this, we try to prove the example by starting at the bottom:

$$\dfrac{\dfrac{\dfrac{???}{\Rrightarrow \vdash P(\gamma), Q(\gamma)}}{\Rrightarrow \vdash P(\gamma) \vee Q(\gamma)}}{\Rrightarrow \vdash \forall x.P(x) \vee Q(x)}$$

Notice that in this case, we are not able to apply any rule of the **G3c**-calculus to the sequent $\Rrightarrow \vdash P(\gamma), Q(\gamma)$. But there might be a subterm t of the numeric sort s.t. replacing this term by 0 makes $\vdash P(\gamma) \vee Q(\gamma)$ provable. In order to find such a term, we consider the theory of both P and Q and check whether we can substitute a term of the sequent such that it is provable and can be the auxiliary sequent of a cycling rule, note that replacement by $s(n)$ and replacement by 0 works for both predicate symbols. Thus, each predicate has a proper auxiliary sequent for the cycling rule and now we need to consider the various possible proof constructions which lead to an application of the cycling inference rule. There are two possibilities which we present below:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Rrightarrow \vdash \top}\,Ax}{\Rrightarrow \vdash P(0)}\,\mathcal{E}}{\vdash P(x) \Rrightarrow \vdash Q(n), P(n)}\,\circlearrowright \qquad \dfrac{\dfrac{\dfrac{}{\Rrightarrow \vdash \top}\,Ax}{\Rrightarrow \vdash Q(0)}\,\mathcal{E}}{\vdash Q(x) \Rrightarrow \vdash Q(n), Q(n)}\,\circlearrowright}{\vdash P(x); \vdash Q(x) \Rrightarrow \vdash Q(n), P(n) \wedge Q(n)}\,\wedge : r}{\vdash P(x); \vdash Q(x) \Rrightarrow \vdash Q(n), Q(s(n))}\,\mathcal{E}}{\vdash P(x); \vdash Q(x) \Rrightarrow \vdash Q(n) \vee Q(s(n))}\,\vee : r}{\vdash P(x); \vdash Q(x) \Rrightarrow \vdash P(s(n))}\,\mathcal{E}}{\vdash Q(x) \Rrightarrow \vdash P(\boldsymbol{\alpha})}\,\Downarrow \qquad \dfrac{\dfrac{\dfrac{}{\Rrightarrow \vdash \top}\,Ax}{\Rrightarrow \vdash Q(0)}\,\mathcal{E}}{\vdash Q(x) \Rrightarrow \vdash Q(n)}\,\circlearrowright}{\vdash Q(x) \Rrightarrow \vdash P(\boldsymbol{\alpha}) \mid \vdash P(n)}\,Th}{\vdash Q(x) \Rrightarrow \vdash P(\boldsymbol{\alpha}) \mid \vdash P(n) \wedge Q(n)}\,\wedge : r}{\vdash Q(x) \Rrightarrow \vdash P(\boldsymbol{\alpha}) \mid \vdash Q(s(n))}\,\mathcal{E}}{\Rrightarrow \vdash P(\boldsymbol{\alpha}) \mid \vdash Q(\boldsymbol{\beta})}\,\Downarrow}{\Rrightarrow \vdash P(\boldsymbol{\gamma}) \vee Q(\boldsymbol{\gamma})}\,Gen}{\Rrightarrow \vdash P(\delta) \vee Q(\delta) \mid \vdash P(\boldsymbol{\gamma}) \vee Q(\boldsymbol{\gamma})}\,Th}{\Rrightarrow \vdash \forall x.P(x) \vee Q(x) \mid \vdash P(\boldsymbol{\gamma}) \vee Q(\boldsymbol{\gamma})}\,\forall : r$$



Notice that we could also replace $\boldsymbol{\gamma}$ in $\Rrightarrow \vdash P(\boldsymbol{\gamma}) \vee Q(\boldsymbol{\gamma})$ with 0. This would lead to the super sequent

$$\vdash P(x) \vee Q(x) \Rrightarrow \vdash P(s(n)) \vee Q(s(n))$$

but the idea of the current calculus is to reduce the inductive statement as much as possible. This probably makes the proof search simpler. The interested reader may try to prove the super sequent above. Moreover, note that the presented proofs create as a by-product the theory extension $P(\boldsymbol{\gamma}) \vee Q(\boldsymbol{\gamma})$.

Of course, this is a trivial example, but even in more complex frameworks, we only have to compare the defined terms and formulas of the equational theory with the terms and formulas of the sequent. Checking the provability of the basecase is a normal proof search in the **G3c**-calculus. As soon as we try to prove the stepcase, we have to allow additional inductions, i.e. additional applications of such replacements. In general, if we want to prove

$$\mathscr{F} \Rrightarrow \mathscr{G} \mid S(t)$$

we can exchange t with 0 and check whether $S(0)$ is provable. If this is provable, we proceed with

$$\forall x S(x) \mid \mathscr{F} \Rrightarrow \mathscr{G} \mid S(s(n))$$

In order to find reasonable applications of $\circlearrowright$ inferences we keep track of the various defined symbols and check for possible term instantiations allowing the construction of an auxiliary sequent of the cycling rule. We plan to continue these investigations in future work.

References

[1] James Brotherston. Cyclic proofs for first-order logic with inductive definitions. In *Tableaux'05*, volume 3702 of *Lecture Notes in Comp. Sci.*, pages 78–92. 2005.

[2] David M. Cerna. *Advances in schematic cut elimination.* PhD thesis, Technical University of Vienna, 2015. http://media.obvsg.at/p-AC12246421-2001.

[3] David M. Cerna and Alexander Leitsch. Schematic cut elimination and the ordered pigeonhole principle. In *Automated Reasoning - 8th International Joint Conference, IJCAR, 2016, Coimbra, Portugal, June 27 - July 2, 2016, Proceedings*, pages 241–256, 2016.

[4] David M. Cerna and Michael Lettmann. Integrating a global induction mechanism into a sequent calculus. In *TABLEAUX'17*, Lecture Notes in Comp. Sci., Sept. 2017.

[5] David M. Cerna and Michael Lettmann. Towards a clausal analysis of proof schemata. In *SYNASC'17*, IEEE Xplorer. IEEE, Sept. 2017.

[6] Kosta Dosen. Sequent-systems for modal logic. *J. Symb. Log.*, 50(1):149–168, 1985.

[7] Cvetan Dunchev, Alexander Leitsch, Mikheil Rukhaia, and Daniel Weller. Cut-elimination and proof schemata. In *Logic, Language, and Computation*, pages 117–136, 2013.

[8] Gerhard Gentzen. Fusion of several complete inductions. In M.E. Szabo, editor, *The Collected Papers of Gerhard Gentzen*, volume 55 of *Studies in Logic and the Foundations of Mathematics*, pages 309 – 311. Elsevier, 1969.

[9] Alexander Leitsch, Nicolas Peltier, and Daniel Weller. CERES for first-order schemata. *J. Log. Comput.*, 27(7):1897–1954, 2017.

[10] G.E. Mints. Some calculi of modal logic. *journal Proc. Steklov Inst. Math.*, pages 97–124, 1968.

[11] Anne S. Troelstra and Helmut Schwichtenberg. *Basic Proof Theory*, volume 43 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, The Pitt Building, Trumpington Street, Cambridge, United Kingdom, second edition, 1996.