PCC 2016 Programme

Mathematical Institute of LMU, Theresienstr. 37–39, 80804 Munich

Thursday May 5: day-long special session dedicated to Ulrich Berger

A027 (Ground floor of the mathematical institute, Block A)

9:00 - 10:00	Opening
	Invited Speaker 1: Dag Normann Revisiting Transfinite Types
10:00 - 10:30	Coffee Break
10:30 - 12:30	Mizuhito Ogawa Decidability by two semi-algorithms
	Dieter Spreen Bitopological spaces and the continuity problem
	Birgit Elbl Decomposing a labelled sequent calculus for the logic of subset spaces
	Fredrik Nordvall Forsberg A Type Theory for Comprehensive Parametric Polymorphism
12:30 - 14:30	Lunch Break (coffee being served at 14h)
14:30 - 16:20	Invited Speaker 2: Paulo Oliva Modified bar recursion - 15 years on
	Thomas Powell The computational content of Zorn's lemma
	Anton Setzer Schemata for Proofs by Coinduction
16:20 - 16:50	Coffee Break
16:50 - 18:10	Invited Speaker 3: Thomas Streicher An effective Spectral Theorem for bounded self adjoint operators
	Lorenz Berger Modelling the human lung
19:30	Dinner at Schneider Bräuhaus, Tal 7, 80331 Munich which is very close to Marienplatz (reservation subject to registration)

Friday May 6

A027 (Ground floor of the math institute, Block A), 9:00–10:00
B349 (3rd floor of the math institute, Block B), 10:00–18:00

9:00 - 10:00	Peter Schuster Eliminating Disjunctions by Disjunction Elimination
	Masahiko Sato Proof theory of the lambda calculus
10:00 - 10:30	Coffee Break
10:30 - 12:30	Thomas Piecha Atomic Systems in Proof-Theoretic Semantics
	Ernst Zimmermann Local confluence in Natural Deduction
	Rene Gazzari Pure Proofs
	Sam Sanders The unreasonable effectiveness of Nonstandard Analysis
12:30 - 14:30	Lunch Break (coffee being served at 14h)
14:30 - 16:30	Alessio Guglielmi Report on a 3-Year Project on Efficient and Natural Proof Systems
	Alessio Santamaria Substitution in Deep Inference via Atomic Flows
	David Sherratt Atomic Lambda Calculus and its connections with Sharing Graphs
	Andrea Aler Tubella Generalising Cut-Elimination through Subatomic Proof Systems
16:30 - 17:00	Coffee Break
17:00 - 18:00	Benjamin Ralph Decomposing First-Order Proofs using Deep Inference
	Lars Kristiansen On Subrecursive Representability of Irrational Numbers

Abstracts of the invited speakers

Dag Normann, University of Oslo, Norway
Revisiting Transfinite Types

In this talk I will reconstruct spaces of countable and uncountable transfinite types, this time using limit spaces. This approach turns out to give a better access to internal concepts of computability for such spaces.

The talk will be a report on ongoing research. This research is rooted in earlier work by Ulrich Berger and by myself, and the revisiting is inspired by recent contributions by Selivanov, Schröder and de Brecht.

Paulo Oliva, Queen Mary University London, UK
Modified bar recursion – 15 years on

I remember as if it was yesterday when during the second year (2001) of my PhD studies we had a visitor giving a talk about "modified bar recursion". I had just learnt about Spector bar recursion from Kohlenbach and was quite intrigued by what kind of different form of bar recursion one could come up with. The speaker was Ulrich Berger, and that talk would shape much of what I have been doing since then. I went on to visit Ulrich in Swansea, and had the honour to co-author two papers [1,2] with him on this new form of bar recursion. In this talk I hope to look back at that work, and much of what followed, in terms of inter-definability results [3] and novel connections to Game Theory via the recent work on selection functions [4,5].

[1] Ulrich Berger and Paulo Oliva, Modified bar recursion, MSCS, 16(2):163-183, 2006

[2] Ulrich Berger and Paulo Oliva, Modified bar recursion and classical dependent choice, LNL, 20:89-107, 2005

[3] Thomas Powell, The equivalence of bar recursion and open recursion, APAL, 165(11):1727-1754, 2014

[4] Martín Escardó and Paulo Oliva, Bar recursion and products of selection functions, JSL, 80(1):1-28, 2015

[5] Martín Escardó and Paulo Oliva, Sequential games and optimal strategies, Proc. of the Royal Society A, 467:1519-1545, 2011

Thomas Streicher, Technical University Darmstadt, Germany
An effective Spectral Theorem for bounded self adjoint operators

Using (sort of) abstract methods from topological domain theory we prove that the spectral theorem for bounded selfadjoint operators is effective in the sense of TTE, i.e., it holds in the Kleene Vesley topos. We also identify the natural topology on the involved spaces induced by their effective representations.

Abstracts of talks

Lars Kristiansen,
On Subrecursive Representability of Irrational Numbers

We consider various ways to represent irrational numbers by subrecursive functions: via Cauchy sequences, Dedekind cuts, trace functions, several variants of sum approximations and continued fractions. Let S be a class of subrecursive functions. The set of irrational numbers that can be obtained with functions from S depends on the representation. We compare the sets obtained by the different representations.

Alessio Guglielmi,
Report on a 3-Year Project on Efficient and Natural Proof Systems

I propose to present generalised normalisation methods and to use example proof systems to illustrate the main points.

Andrea Aler Tubella and Alessio Guglielmi,
Generalising Cut-Elimination through Subatomic Proof Systems

In work presented at PCC 2015, we showed how by considering atoms as self-dual, noncommutative, binary logical relations, we can reduce disparate rules such as introduction, contraction or cut to instances of single linear rule scheme. We show how we exploit this regularity to study cut-elimination. We are able to generalise a cut-elimination procedure to a wide range of systems, and in the process gain valuable insight as to why cut-elimination is such a prevalent phenomenon.

Ernst Zimmermann,
Local confluence in Natural Deduction

The talk tries to show that local confluence is a key concept for the understanding of Natural Deduction. The considerations are restricted to intuitionistic implication, with hints to richer languages and substructural logics. In Natural Deduction we are in a fine position concerning local confluence: reductions of many usual logical connectives a r e locally confluent. Furthermore, certain subclasses of reductions with nice properties can be defined, especially subclasses of reductions, which can be shown to terminate and to preserve local confluence. Due to Newman's lemma this yields confluence of such subclasses of reduction. For confluence of full reductions a commutation property is stated, showing how a reduction commutes with its own subreduction. And termination of full reductions can be shown by using a subreduction cover.

Lorenz Berger,
Modelling the human lung

Creating accurate computational models of the human body is difficult due to many challenges with modelling, validation and the development of correct numerical implementations. We present a computational lung model that tightly couples a poroelastic model of lung tissue to an airway fluid network to model breathing in the lungs. The poroelastic model approximates the porous structure of lung tissue using a continuum model. The naive approach of solving the resulting model is numerically unstable. A stabilized finite element method is presented to discretize the poroelastic equations and overcome these instabilities. To demonstrate the robust coupling between the poroelastic medium and the airway fluid network, numerical simulations on a realistic lung geometry are presented. Remaining challenges on the validation and verification of the model are also discussed.

Sam Sanders,
The unreasonable effectiveness of Nonstandard Analysis

As suggested by the title, the topic of my talk is the vast computational content of classical Nonstandard Analysis. In particular, I will present a template CI which allows one to obtain effective theorems from theorems in 'pure' Nonstandard Analysis, i.e. only involving the nonstandard definitions (of continuity, compactness, Riemann integration, convergence, et cetera). This template CI has been applied successfully to the Big Five systems of Reverse Mathematics, the Reverse Mathematics zoo, and computability theory. The template CI often produces theorems of Bishop's Constructive Analysis.

The framework for the template CI is Nelson's syntactic approach to Nonstandard Analysis, called internal set theory, and its fragments based on Gödel's T as introduced in [1]. Notable results are that applying the template CI to theo- rems involving the nonstandard definitions of respectively continuity, compactness, and open set, the latter definitions are converted into the associated definitions from constructive or computable analysis (resp. continuity with a modulus, totally boundedness, and effectively open set).

Finally, we establish that a theorem of Nonstandard Analysis has the same com- putational content as its 'highly constructive' Herbrandisation. Thus, we establish an 'algorithmic two-way street' between so-called hard and soft analysis, i.e. be- tween the worlds of numerical and qualitative results. However, the study of the Herbrandisations of nonstandard theorems also leads to a new class of function- als (not involving Nonstandard Analysis) with rather strange properties. Chief among these new functionals is the special fan functional which can be computed easily in intuitionistic mathematics, but cannot be computed by the Turing jump functional (2E) or even much stronger comprehension axioms. Similar functionals exist for most theorems from the Reverse Mathematics zoo.

Rene Gazzari,
Pure Proofs

We intend to characterize philosophical notions as "to be a pure proof", "a simple proof" or "an elementary proof" from a formal point of view. After developing a good formal definition of those informal notions, we discuss the theorem that every proof may be transformed into a pure proof. We provide a partial solution and illustrate the problem of finishing the proof.

Thomas Piecha,
Atomic Systems in Proof-Theoretic Semantics

In proof-theoretic semantics the validity of atomic formulas, or atoms, is usually defined in terms of their derivability in atomic systems. Such systems can be sets of atoms, figuring as atomic axioms, or sets of atomic rules such as production rules. One can also allow for atomic rules which can discharge atomic assumptions, or even consider higher-level atomic rules which can discharge assumed atomic rules. In proof-theoretic semantics for minimal and intuitionistic logic atomic systems are used as the base case in an inductive definition of validity. We compare two approaches to atomic systems. The first approach is compatible with an interpretation of atomic systems as representations of states of knowledge. We present negative as well as positive completeness results for notions of validity based on this approach. The second approach takes atomic systems to be definitions of atomic formulas. The two views lead to different notions of derivability for atomic formulas, and consequently to different notions of proof-theoretic validity. In the first approach, validity is stable in the sense that for atomic formulas logical consequence and derivability coincide for any given atomic system. In the second approach this is not the case. This indicates that atomic systems as definitions, which determine the meaning of atomic sentences, might not be the proper basis for proof-theoretic validity, or conversely, that standard notions of proof-theoretic validity are not appropriate for definitional rule systems.

Thomas Powell,
The computational content of Zorn's lemma

Zorn's lemma is a well known formulation of the axiom of choice which states that any chain complete partially ordered set has a maximal element. Certain theorems in mathematics can be given a particularly elegant proof using Zorn's lemma - a well-known example of this is the theorem that any ring with unity has a maximal ideal. In this talk I will focus on giving a computational interpretation to Zorn's lemma. More precisely, I will describe a new form of recursion which realizes the functional interpretation of certain restricted instances of Zorn's lemma.

There are two main motivating factors behind this work. The task of making constructive sense of Zorn's lemma is an interesting and challenging proof theoretic problem in its own right. My emphasis here is on providing a natural realizer for the functional interpretation of the lemma which clearly reflects its computational content. This alone is a non-trivial task, as even in the weak cases of Zorn's lemma considered here such a realizer will necessarily be based on an extremely strong form of recursion, undefinable even in Goedel's system T. The second factor is that a direct computational interpretation of Zorn's lemma should enable us to extract intuitive programs from non-constructive proofs which rely on it. This in particular paves the way for a proof theoretic analysis of several important theorems in abstract algebra and well-quasi order theory that make use of choice in this form.

My talk builds on a number of recent studies which examine the constructive meaning of variants of Zorn's lemma, most importantly the work of U. Berger [1], who has given a direct and elegant modified realizability interpretation of a reformulation of the lemma known as open induction. The difference here is that I work in the alternative setting of Goedel's functional interpretation (which requires a different realizing term) and look towards giving a more general interpretation. Moreover, I emphasise the algorithmic behaviour of the realizer, linking it to my own recent research on giving learning-based realizers to the functional interpretation of classical principles [2]. The talk is very much about work in progress, and I aim to emphasise open problems and directions for future research.

[1] U. Berger: A computational interpretation of open induction. Proceedings of LICS 2004.

[2] T. Powell: Goedel's functional interpretation and the concept of learning. To appear in LICS 2016.

Alessio Santamaria,
Substitution in Deep Inference via Atomic Flows

This is a short talk in Alessio Guglielmi's group of talks, describing the problem of substitution of flows into other flows in the effort of studying the concept of substitution of a derivation into the occurrences of a given atom in another derivation.

Birgit Elbl,
Decomposing a labelled sequent calculus for the logic of subset spaces

The logic of subset spaces SSL is a bimodal logic introduced in [1] for formalising reasoning about points and sets. Using the cut-free, one-sided, labelled sequent calculus LSSL-p presented in [2], we prove results concerning the unimodal fragments in a purely syntactic way. Furthermore, we relate the K-L-fragment of the calculus to known systems for S5.

[1] A. Dabrowski, L.S. Moss, R. Parikh. Topological reasoning and the logic of knowledge, Annals of Pure and Applied Logic 78 (1996), pp. 73–110.

[2] B. Elbl. A cut-free sequent calculus for the logic of subset spaces. Submitted.

Anton Setzer,
Schemata for Proofs by Coinduction

Proofs by induction are carried out by following schemata for induction, which makes it easier to carry out such kind of proofs than by using directly the fact that the natural numbers is the least set closed under zero and successor. So for proving $\forall x.\phi(x)$, one doesn't define first $A := \{x \in \mathbb{N} \mid \phi(x)\}$ and show that A is closed under 0 and successor. Instead, one uses the schema of induction. Although using the schema of induction amounts to essentially the same as showing the closure properties of A, using the schema of induction is much easier to use and to teach.

Proofs by coinduction usually follow directly the principle that the coinductively defined set is the largest set satisfying the principles of the coinductively defined set. For instance for carrying out proofs of bisimulation, one usually introduces a relation and shows that it is a bisimulation relation. This makes proofs by coinduction cumbersome and difficult to teach.

In this talk we will introduce schemata for coinduction which are similar to the schemata for induction. The use of the coinduction hypothesis is made easier by defining coinductively defined sets as largest sets allowing observations, rather than as largest sets closed under introduction rules. For instance the set Stream of streams of natural numbers is the largest set allowing observations head : Stream $\rightarrow \mathbb{N}$ and tail : Stream $\rightarrow$ Stream , rather than being the largest set closed under consrm : $\mathbb{N} \rightarrow$ Stream $\rightarrow$ Stream.

Based on this we will first introduce schemata for defining functions by primitive corecursion or guarded recursion. This is dual to the principle of primitive recursion. Then we define schemata for coinductive proofs of equality. Finally we introduce schemata for coinductively defined relations such as bisimulation relations.

We will give examples of how to carry out coinductive proofs on paper. These proofs will make use of the coinduction hypothesis, where restrictions that are dual to those for the use of the induction hypothesis in inductive proofs are used.

Peter Schuster, Davide Rinaldi and Daniel Wessel,
Eliminating Disjunctions by Disjunction Elimination

Completeness theorems, or contrapositive forms of Zorn's Lemma, are often invoked in elementary contexts in which the corresponding syntactical conservation theorems would suffice. We present a pretty universal form of such a syntactical conservation theorem for Horn sequents, using an utterly versatile sandwich criterion given by Scott 1974.

We work with Scott's multi-conclusion entailment relations as extending Tarskian single-conclusion consequence relations. In a nutshell, the extra multi-conclusion axioms can be reduced to rules for the underlying single-conclusion relation that hold in all known mathematical instances. In practice this means to fold up branchings of proof trees by referring to the given mathematical structures.

Applications include the separation and extension theorems known under the names of Krull-Lindenbaum, Artin-Schreier, Szpilrajn and – this is work in progress – Hahn-Banach. Related work has been done on individual instances: in locale theory (Mulvey-Pelletier 1991), dynamical algebra (Coste-Lombardi-Roy 2001, Lombardi (1997-8), formal topology (Cederquist-Coquand-Negri 1998) and proof theory (Coquand-Negri-von Plato 2004). Further motivation for our approach was given by Berger's celebrated paper in LICS 2004.

Dieter Spreen,
Bitopological spaces and the continuity problem

The continuity problem is the question when effective (or Markov computable) maps between effectively given topological spaces are effectively continuous. It will be shown that this is always the case if the the range of the map is effectively bi-regular. As will be shown, such spaces appear quite naturally in the context of the problem.

David Sherratt,
Atomic Lambda Calculus and its connections with Sharing Graphs

Presented is a research idea to connect the atomic lambda calculus and sharing graphs through a formal graphical representation of the atomic lambda calculus, so that we may explore the area between full laziness and optimal reduction.

Benjamin Ralph,
Decomposing First-Order Proofs using Deep Inference

The deep-inference formalism, by allowing for very fine-grained inference steps and freer composition of proofs, has produced important results and innovations in various logics, especially classical propositional logic. A natural progression is to extend these insights to classical first-order logic (FOL) but, although a direct cut-elimination procedure has been provided, there has been no work as of yet that incorporates the many perspectives and techniques developed in the last ten years.

In the talk, I will give the outline of a new cut elimination procedure for FOL in deep inference, as well as a decomposition-style presentation of Herbrand's Theorem called a Herbrand Stratification that is proved not as a corollary of cut elimination, but in tandem with it. In doing so, I hope to provide a different and perhaps better perspective on FOL normalisation, Herbrand's Theorem, and their relationship. More concretely, there is good reason to believe that, as in propositional logic, this research can provide us with new results in proof complexity.

Mizuhito Ogawa,
Decidability by two semi-algorithms

Recently, several interesting examples of the decidablity proofs consisting of two semi-algorithms have been shown.

(1) Jerome Leroux, Vector addition system reachability problem, ACM POPL 2011.

(2) Helmut Seidl, Sebastian Maneth, Gregor Kemper, Equivalence of Deterministic Top-Down Tree-to-String Transducers is Decidable, IEEE FOCS 2015.

(3) Yuxi Fu, Checking Equality and Regularity for Normed BPA with Silent Moves, ICALP (2) 2013

Among them (2) and (3) solve long standing open problems, and (1) gives a new simple proof for the decidability of a vector addition system (VAS), which can be tracked back to W.Mayr (1981). This presentation focuses on (1) and (2), which run two semi-algorithms, one tries to say "yes" and another tries to say "no". The semi-algorithms saying "no" fail to say their termination, due to the non-constructive nature of the finite basis property for (1) and the convergence of an ideal sequence in a Noetherian ring for (2).

Fredrik Nordvall Forsberg,
A Type Theory for Comprehensive Parametric Polymorphism

A class of models of System F is presented, where reflexive-graph-category structure for relational parametricity and fibrational models of impredicative polymorphism are combined. To achieve this, we modify the definition of fibrational model of impredicative polymorphism by adding one further ingredient to the structure: comprehension in the sense of Lawvere. Such comprehensive models, once further endowed with reflexive-graph-category structure, enjoy the expected consequences of parametricity. To prove this, we introduce a type theory extending System F with a sound and complete semantics with respect to the class of models considered. Proving the consequences of parametricity within this type theory requires new techniques since equality relations are not available, and standard arguments that exploit equality need to be reworked.

This is joint work with Alex Simpson and Neil Ghani.

Masahiko Sato,
Proof theory of the lambda calculus

We develop a proof theory of the lambda calculus where we study the set of closed lambda terms by inductively defining the set as a free algebra.

The novelty of the approach is that we construct and study lambda calculus without using the notions of variables and alpha-equivalence. In this approach we can study lambda terms as combinators and can have a clean proof of the Church-Rosser Theorem in the Minlog proof assistant.

On Subrecursive Representability of Irrational Numbers

Lars Kristiansen, University of Oslo

We consider various ways to represent irrational numbers by subrecursive functions: via Cauchy sequences, Dedekind cuts, trace functions, several variants of sum approximations and continued fractions. Let $\mathcal{S}$ be a class of subrecursive functions. The set of irrational numbers that can be obtained with functions from $\mathcal{S}$ depends on the representation. We compare the sets obtained by the different representations.

A function $C : \mathbb{N} \to \mathbb{Q}$ is a *Cauchy sequence* for the real number α when $|\alpha - C(n)| < 1/2^n$. A function $D : \mathbb{Q} \to \{0, 1\}$ is a *Dedekind cut* of the real number α when $D(q) = 0$ iff $q < \alpha$. A function $T : \mathbb{Q} \to \mathbb{Q}$ is a *trace function* for the irrational number α when $|\alpha - q| > |\alpha - T(q)|$.

Any irrational number α can be written of the form $\alpha = a + \frac{1}{2^{k_0}} + \frac{1}{2^{k_1}} + \frac{1}{2^{k_2}} + \ldots$ where $k_0, k_1, k_2, \ldots$ is a strictly monotone increasing sequence of natural numbers and a is an integer. Let $A : \mathbb{N} \to \mathbb{N}$ be a strictly monotone function. We will say that A is a *sum approximation from below* of the the real number α if there exists $a \in \mathbb{Z}$ such that $\alpha = a + \sum_{i=0}^{\infty} 1/2^{A(i)+1}$. Any real number can also be written as a difference between an integer and an infinite sum, and we will say that A is a *sum approximation from above* of the the real number α if there exists $a \in \mathbb{Z}$ such that $\alpha = a - \sum_{i=0}^{\infty} 1/2^{A(i)+1}$.

The sum approximations defined above are sum approximations in base 2. We will also consider *general sum approximations* (from above and below). A *general sum approximations* of α is a function that yields the sum approximation of α in any base.

An irrational number α can also be represented by a function $f : \mathbb{N} \to \mathbb{Z}$ where $f(n)$ yields the n^{th} element of the continued fraction $[a_0; a_1, a_2 \ldots]$ of α.

Let $\mathcal{P}_C$, $\mathcal{P}_D$ and $\mathcal{P}_{[]}$ denote the sets of irrationals that are representable, receptively, by primitive recursive Cauchy sequences, primitive recursive Dedekind cuts and primitive recursive continued fractions. Specker [3] proved $\mathcal{P}_D \subset \mathcal{P}_C$, and Lehman [2] proved $\mathcal{P}_{[]} \subset \mathcal{P}_D$ (strict inclusions). We will discuss a number on theorems on how trace functions and (general) sum approximation (from above and below) relate to Cauchy sequences, Dedekind cuts and continued fractions. Most of these theorems can be found in Kristiansen [1].

References

1. L. Kristiansen, *On subrecursive representability of irrational numbers.* Accepted for publication in Computability (the journal of CiE).
2. R. S. Lehman, On Primitive Recursive Real Numbers, *Fundamenta Mathematica* **49**(2) (1961), 105–118.
3. E. Specker, Nicht Konstruktiv Beweisbare Satze Der Analysis, *The Journal of Symbolic Logic* **14**(3) (1949), 145–158.

Report on a 3-Year Project on Efficient and Natural Proof Systems
Alessio Guglielmi (Bath)

I propose to present generalised normalisation methods and to use example proof systems to illustrate the main points.

I will argue that normalisation in Gentzen is a conflation of two mechanisms that operate on two distinct composition methods:

1) sharing/contraction: this is the composition method that, when normalised upon, generates complexity;
2) linear cut: this composition method, when normalised upon, reduces complexity.

As is well known and intuitive, normalising sharing/contraction means, computationally, applying beta reduction to several instances of a variable (at the propositional level) or generating Herbrand expansions (in a predicate calculus).

Since the two mechanisms pull in opposite directions from the point of view of complexity, separating them would allow us to obtain finer computational control. Such separation is impossible in Gentzen formalisms, due to their limitations in proof composition, but it is natural in deep inference.

An additional, natural composition method could be considered in deep inference (and not in Gentzen), namely

3) substitution of proofs inside other proofs.

This is a sound generalisation of Frege substitution, which in turn is conjectured to provide a superpolynomial speed-up in the size of proofs.

I will argue that those three composition methods and their associated normalisation mechanisms can be separated and made independent to a very large extent in deep inference. I will illustrate the impact of these ideas on the design of a proof system addressing some computational requirements. This proof system, called KV, is a multiplicative linear logic with a self-dual non-commutative connective 'seq' and a self-dual modality 'star' providing contraction over seq. Given this characteristics, KV could form the core of a proof system for a Kleene algebra. KV can also be seen as an extension of BV with the star modality. Remarkably, under certain natural and logic-independent constraints, system KV seems to be canonical, in the sense that there is no room to manoeuvre in the design of the rules.

Generalising Cut-Elimination through Subatomic Proof Systems

Andrea Aler Tubella and Alessio Guglielmi

University of Bath

In work presented at PCC 2015 [1], we showed how we can consider atoms as self-dual, noncommutative binary logical relations. We can then define an interpretation map from formulae built in such a 'subatomic' way to ordinary formulae. Through this subatomic representation, we can see *all* the rules needed for a complete system for many propositional logics (including classical and linear ones) as instances of a single linear rule scheme

$$\frac{(A\ \alpha\ C)\ \beta\ (B\ \gamma\ D)}{(A\ \epsilon\ B)\ \zeta\ (C\ \eta\ D)}\quad ,$$

where the Greek letters denote logical connectives subject to certain simple conditions.

This ability to see *inside* of atoms gives us a way to reduce disparate rules such as contraction, cut, identity and any logical rule like conjunction-introduction, into a *unique* rule scheme. The fact that so many rules can be made to follow this rule scheme remains however quite surprising. It is an intriguing albeit very useful phenomenon, and we hope the audience to the talk can provide some insight as to the reason behind it. We can exploit having a single rule shape to reason generally on proof systems, allowing us to generalise methods that had to be proven to work for each individual system until now. This approach is particularly fruitful with respect to studying cut-elimination, and in particular it sheds light on *why* cut elimination works on such a wide range of proof systems. More concretely, by studying abstract systems where all the rules follow this scheme we are able to provide a general cut-elimination procedure for many systems without contraction, including all the standard variants of linear logic. Furthermore, by combining it with the decomposition techniques frequently used in deep inference, we can apply this cut-elimination method to an even wider range of systems, including Classical Logic. Looking at cut-elimination in this way helps us understand why it is such a prevalent phenomenon and why many similar cut-elimination arguments work in seemingly very different systems.

From the study of subatomic systems, we are able to give sufficient conditions for a system to enjoy cut-elimination. We provide a procedure, called splitting, that is a generalisation of a common technique employed for cut-elimination in deep inference systems. We show that splitting can be applied in many systems without contraction, such as Linear Logic [3] and including systems with self-dual non-commutative connectives such as BV [4]. The idea behind splitting is very simple, and it is rooted in deep inference methods. In the sequent calculus, formulae have a root connective that allows us to determine which rules are applied immediately above the cut and to follow a classical cut-elimination procedure by studying those rules. In deep inference, rules can be applied anywhere deep in a formula and as such anything can happen above a cut. As a consequence, the splitting method focuses on understanding the behaviour of the context around the cut, and in particular it consists on breaking down a proof in different pieces by following the logical connectives involved in the cut to find their duals. We show that we can then rearrange the different components of the proof to obtain a cut-free proof.

The reach of the splitting technique goes beyond linear systems when combined with another common deep inference method. In many systems, derivations can be arranged into consecutive subderivations made up of only certain rules [2,6]. We call this transformation method decomposition. A shining example of a decomposition theorem in the sequent calculus is Herbrand's Theorem, through which we can decompose a proof into a bottom phase with contraction and quantifier rules and a top phase with propositional rules only. Frequently, like in Classical Logic [5] , we can decompose a proof into a top phase without contractions, and a bottom phase made-up only of contractions. Furthermore, this decomposition is usually obtained by manipulating proofs through local transformations, rather than operating on the proof as a whole. By doing this, we can perform cut-elimination through splitting in the contraction-free top of the proof, and thus we can provide a cut-elimination procedure in two phases for all such decomposable systems.

By breaking down cut-elimination into these two different steps, we are able to observe hidden properties that cannot be explored with classical methods where decomposition and splitting are intertwined. For example, it allows us to study which parts of the cut-elimination process can be done locally, and which necessitate the manipulation of the proof as a whole. Interestingly, we can also observe that the complexity in this cut-elimination procedure stems from the decomposition methods rather than from the elimination of the cuts through splitting.

References

1. A. Aler Tubella and A. Guglielmi. Subatomic proof systems. At http://cs.bath.ac.uk/ag/p/SubAbs2.pdf, 2015.
2. K. Brünnler. Locality for classical logic. *Notre Dame Journal of Formal Logic*, 47(4):557–580, 2006.
3. J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
4. A. Guglielmi. A system of interaction and structure. *ACM Transactions on Computational Logic*, 8(1):1:1–64, 2007.
5. T. Gundersen. *A General View of Normalisation Through Atomic Flows*. PhD thesis, University of Bath, 2009.
6. L. Straßburger. *Linear Logic and Noncommutativity in the Calculus of Structures*. PhD thesis, Technische Universität Dresden, 2003.

This work has been support by EPSRC grant EP/K018868/1 *Efficient and Natural Proof Systems*.

LOCAL CONFLUENCE IN NATURAL DEDUCTION

ERNST ZIMMERMANN

The talk tries to show that local confluence is a natural and essential concept for the understanding of Natural Deduction. The considerations are restricted to intuitionistic implication, with hints to richer languages and substructural logics. In Natural Deduction we are in a fine position concerning local confluence: reductions of many usual logical connectives a r e locally confluent. Furthermore, certain subclasses of reductions with nice properties can be defined, especially subclasses of reductions, which can be shown to terminate and to preserve local confluence. Due to Newman's lemma this yields confluence of such subclasses of reduction, i.e. confluence of subreductions. For confluence of full reductions a commutation property is stated, showing how a reduction commutes in a specific way with its own subreduction. And termination of full reductions can be shown with further properties of subreduction. So, the considerations on Natural Deduction presented in this talk are strongly influenced by concepts of term rewriting.
The talk hopes to clarify and specify ideas exposed in the author's paper *Decomposition of Reduction*, published in: Advances in Natural Deduction, Springer, 2014, Eds. L.C. Pereira, E.H. Haeusler, V. de Paiva.

Boeblingen, Germany
E-mail address: ernst.z@gmx.de

Date: 25th April 2016.

THE UNREASONABLE EFFECTIVENESS OF NONSTANDARD ANALYSIS

SAM SANDERS

Abstract

As suggested by the title, the topic of my talk is the *vast computational content* of *classical* Nonstandard Analysis. In particular, I will present a template $\mathfrak{CI}$ which allows one to obtain *effective* theorems from theorems in 'pure' Nonstandard Analysis, i.e. only involving the *nonstandard* definitions (of continuity, compactness, Riemann integration, convergence, et cetera). This template $\mathfrak{CI}$ has been applied successfully to the Big Five systems of Reverse Mathematics ([7]), the Reverse Mathematics zoo ([3, 6, 8]), and computability theory ([5]). The template $\mathfrak{CI}$ often produces theorems of Bishop's *Constructive Analysis* ([2]).

The framework for the template $\mathfrak{CI}$ is Nelson's syntactic approach to Nonstandard Analysis, called *internal set theory* ([4]), and its fragments based on Gödel's T as introduced in [1]. Notable results are that applying the template $\mathfrak{CI}$ to theorems involving the nonstandard definitions of respectively continuity, compactness, and open set, the latter definitions are converted into the associated definitions from constructive or computable analysis (resp. continuity with a modulus, totally boundedness, and effectively open set).

Finally, we establish that a theorem of Nonstandard Analysis has the *same computational content* as its 'highly constructive' *Herbrandisation*. Thus, we establish an 'algorithmic two-way street' between so-called hard and soft analysis, i.e. between the worlds of numerical and qualitative results. However, the study of the Herbrandisations of nonstandard theorems also leads to a new class of functionals (not involving Nonstandard Analysis) with rather strange properties. Chief among these new functionals is the special fan functional (See [5, §3]) which can be computed easily in intuitionistic mathematics, but cannot be computed by the Turing jump functional ($\exists^2$) or even much stronger comprehension axioms. Similar functionals exist for most theorems from the Reverse Mathematics zoo.

References

[1] Benno van den Berg, Eyvind Briseid, and Pavol Safarik, *A functional interpretation for nonstandard arithmetic*, Ann. Pure Appl. Logic **163** (2012), no. 12, 1962–1994.

[2] Errett Bishop and Douglas S. Bridges, *Constructive analysis*, Grundlehren der Mathematischen Wissenschaften, vol. 279, Springer-Verlag, Berlin, 1985.

[3] Damir D. Dzhafarov, *Reverse Mathematics Zoo*. `http://rmzoo.uconn.edu/`.

[4] Edward Nelson, *Internal set theory: a new approach to nonstandard analysis*, Bull. Amer. Math. Soc. **83** (1977), no. 6, 1165–1198.

[5] Sam Sanders, *The Gandy-Hyland functional and a hitherto unknown computational aspect of Nonstandard Analysis*, Submitted, Available from: `http://arxiv.org/abs/1502.03622` (2015).

Munich Center for Mathematical Philosophy, LMU Munich, Germany & Department of Mathematics, Ghent University, Belgium

E-mail address: `sasander@me.com`.

[6] ———, *The taming of the Reverse Mathematics zoo*, Submitted, `http://arxiv.org/abs/1412.2022` (2015).

[7] ———, *The unreasonable effectiveness of Nonstandard Analysis*, Submitted to APAL special issue of LFCS, `http://arxiv.org/abs/1508.07434` (2016).

[8] ———, *The refining of the taming of the Reverse Mathematics zoo*, To appear in Notre Dame Journal for Formal Logic, `http://arxiv.org/abs/1602.02270` (2016).

René Gazzari, rene.gazzari@uni-tuebingen.de
Department of Computer Science, University of Tübingen, Germany

Pure Proofs

In everyday live of a mathematician you investigate proofs and attribute properties to them as *to be elementary* or *to be pure*. From an intuitive point of view, these notions seem to be unproblematic; the mathematicians understand what they mean. And this is the usual starting point for philosophical problems.

When trying to clarify these notions, you recognise how much you rely on (good) intuitions. It seems problematic to provide a clear definition of such notions, a definition which is, in the best case, a formal definition with respect to the formalisations of informal proofs (a derivation in a suitable calculus).

In our talk, we intend to characterise such notions from a formal point of view. Focusing on the notion of *pureness*, we introduce formal concepts allowing to identify the formal counterparts of our intuitive notions. We also sketch how to capture other similar notions. Thereby, we also shed light to the intuitive kernel of such intuitive notions, which may not be given formally.

Once having a good definition of *pure proofs*, we discuss the (still open) problem of the *existence of pure proofs*. Our claim is that every proof may be transformed into a pure proof (according to our definition). We motivate the plausibility of the claim, provide partial solutions and illustrate the problem of finishing the proof.

THE COMPUTATIONAL CONTENT OF ZORN'S LEMMA

Abstract for contributed talk at PCC 2016

THOMAS POWELL

Zorn's lemma is a well known formulation of the axiom of choice which states that any chain complete partially ordered set has a maximal element. Certain theorems in mathematics can be given a particularly elegant proof using Zorn's lemma - a well-known example of this is the theorem that any ring with unity has a maximal ideal. In this talk I will focus on giving a computational interpretation to Zorn's lemma. More precisely, I will describe a new form of recursion which realizes the functional interpretation of certain restricted instances of Zorn's lemma.

There are two main motivating factors behind this work. The task of making constructive sense of Zorn's lemma is an interesting and challenging proof theoretic problem in its own right. My emphasis here is on providing a *natural* realizer for the functional interpretation of the lemma which clearly reflects its computational content. This alone is a non-trivial task, as even in the weak cases of Zorn's lemma considered here such a realizer will necessarily be based on an extremely strong form of recursion, undefinable even in Gödel's system T. The second factor is that a direct computational interpretation of Zorn's lemma should enable us to extract intuitive programs from non-constructive proofs which rely on it. This in particular paves the way for a proof theoretic analysis of several important theorems in abstract algebra and well-quasi order theory that make use of choice in this form.

My talk builds on a number of recent studies which examine the constructive meaning of variants of Zorn's lemma, most importantly the work of U. Berger [1], who has given a direct and elegant modified realizability interpretation of a reformulation of the lemma known as open induction. The difference here is that I work in the alternative setting of Gödel's functional interpretation (which requires a different realizing term) and look towards giving a more general interpretation. Moreover, I emphasise the algorithmic behaviour of the realizer, linking it to my own recent research on giving learning-based realizers to the functional interpretation of classical principles [2]. The talk is very much about work in progress, and I aim to emphasise open problems and directions for future research.

References

[1] U. Berger. A computational interpretation of open induction. In *Proceedings of LICS 2004*, pages 326–334. IEEE Computer Society, 2004.

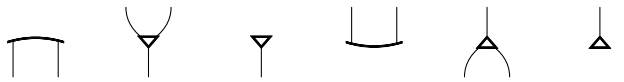
[2] T. Powell. Gödel's functional interpretation and the concept of learning. To appear in Proceedings of LICS 2016.

Substitution in Deep Inference via Atomic Flows

Alessio Santamaria

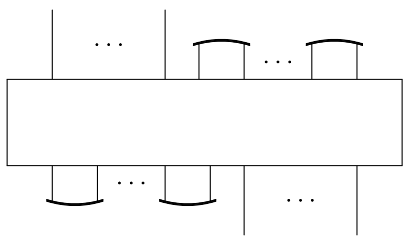
University of Bath

Atomic flows [GG08] are certain kinds of acyclic graphs which capture the structural information of proofs in Deep Inference formalisms [Gug07]. Intuitively, an atomic flow is obtained from a proof retaining the causal dependencies between creation, duplication and destruction of atoms and discarding all informations about logical connectives, units and linear inference rules. More in detail, an atomic flow is obtained from the following generators, modulo some conditions that require acyclicity and the possibility to assign a polarity to every edge in a coherent way:



i↓ c↓ w↓ i↑ c↑ w↑

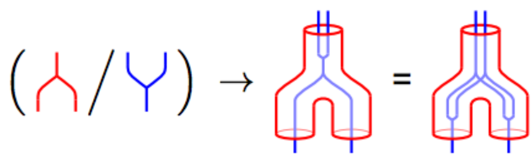
Despite the lack of information that they carry with respect to the original proof, they have been proved to be enough in order to obtain a quasi-polynomial cut elimination for classical logic [Gun09]. It is not difficult to show that any flow can be reduced to a normal form of this shape:



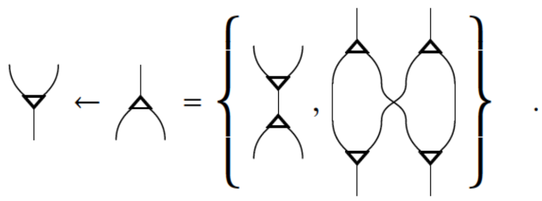
where every i↓ and i↑ is relegated at the top and at the bottom, respectively, of the flow. We can see flows in this form as purely geometric structures and therefore we are interested in studying the algebra of these objects, namely how they can compose. Apart from vertical and horizontal composition, we can consider another operation: *substitution*.

The usual operation of substitution of a formula A into every occurrence of an atom a of another formula B can be generalised to substitution of a derivation φ into the occurrences of an atom a in another derivation ψ. Suppose for example that a appears in both the premiss A and the conclusion B of ψ. Then it makes sense to substitute the premiss C of φ into the occurrences of a in A, the conclusion D of φ into the occurrences of a in B , and to substitute every section of φ into the occurrences of a in the sections of ψ. Here, by *section* of a derivation in deep inference we mean a formula that splits the derivation into two subderivations, such that the formula is the conclusion of one and the premiss of the other one. Naturally, there is more than one way to do this, therefore we would obtain a set of derivations as a result. This means that substitution can be regarded as a compression mechanism: when it is left indicated it has the power to represent in a non-bureaucratic way several derivations.

From the point of view of atomic flows, this operation could be intuitively performed by viewing the edges of the flow $\varPsi$ of ψ as "pipes" that are filled with the flow $\varPhi$ of φ in such a way that the top and bottom edges of $\varPhi$ are stretched until they reach the top and bottom pipes of $\varPsi$, respectively. More precisely, the idea is to associate $\varPhi$ to certain paths in $\varPsi$ that corresponds to a and to subject instances of $\varPhi$ to the same terminations, bifurcations and U-turns of the paths in $\varPsi$ they are substituted into. For example, the substitution of a c↓ inside a c↑ would be



that is,



In order to formalise this intuitive definition, we are searching for an algebraic understanding of atomic flows and their properties, using category theory, that we think might lead to a convincing definition of substitution. In the previous example in particular it seems that we are horizontally composing c↓ and c↑, seen as natural transformations, where the two flows on the right hand of the equation would be the legs of a commutative square.

References

[GG08] Alessio Guglielmi and Tom Gundersen. *Normalisation Control in Deep Inference Via Atomic Flows II*. 2008. **url**: http://cs.bath.ac.uk/ag/p/NormContrDIAtFl2.pdf.

[Gug07] Alessio Guglielmi. "A System of Interaction and Structure". In: *ACM Transactions on Computational Logic* 8.1 (2007), 1:1–64. **doi**: 10.1145/1182613.1182614. **url**: http://cs.bath.ac.uk/ag/p/SystIntStr.pdf.

[Gun09] Tom Gundersen. "A General View of Normalisation Through Atomic Flows". PhD thesis. University of Bath, 2009. **url**: http://tel.archives-ouvertes.fr/docs/00/50/92/41/PDF/thesis.pdf.

Decomposing a labelled sequent calculus for the logic of subset spaces

Birgit Elbl*

The logic of *subset spaces* SSL is a bimodal logic introduced in [2] for formalising reasoning about points and sets. We use $\Box, \Diamond$ for the first set of modalities and $\textsc{k}, \textsc{l}$ for the second. The Hilbert-system for SSL combines S4-axioms for $(\Box, \Diamond)$ with S5-axioms for $(\textsc{k}, \textsc{l})$ and two additional axiom schemata, i.e. *persistence* for literals and the *cross axiom*. The $\Box\Diamond$-fragment of the resulting system is trivialised by these additions, while SSL is conservative over S5 w.r.t. KL-formulas. Using the cut-free, one-sided, labelled sequent calculus **LSSL-p** presented in [3], we can prove these results in a purely syntactic way. In particular, we show that every formula obtained by decorating a tautology with modalities $\Box, \Diamond$ is derivable.

Obviously, a cut-free proof of a KL-formula contains no application of $\Box$- or $\Diamond$-rules. We relate the KL-part of the calculus in two ways to calculi for S5. In contrast to [5, 1], **LSSL-p** makes use of compound expressions for worlds and a “is-world”-predicate. This way, the internalised semantics is close to the original subset space logic, and the frame rules can be kept simple (without eigenvariable conditions), but due to these modifications of the general method, the KL-fragment of **LSSL-p** is not the S5-system in [5]. However, it can be obtained by adding some further restrictions to the rules of the system constructed following the strategy in [5] for the S5-frames with the full relation for accessibility. Furthermore, the role of labels in the KL-fragment is so simple that we can readily shift to a variant using lists or multisets of sequents instead. The system obtained this way can be regarded as a one-sided version of Poggiolesi’s calculus in [4]. Thus, we found another way to relate the KL-fragment to S5.

References

1. D. Garg, V. Genovese and S. Negri, *Countermodels from sequent calculi in multi-modal logics*, in: *Proceedings of the 2012 27th Annual IEEE/ACM Symposium on Logic in Computer Science*, LICS ’12 (2012), pp. 315–324.
2. A. Dabrowski, L.S. Moss and R. Parikh, *Topological reasoning and the logic of knowledge*, Annals of Pure and Applied Logic **78** (1996), pp. 73–110.
3. B. Elbl, *A cut-free sequent calculus for the logic of subset spaces*, submitted.
4. F. Poggiolesi, *A cut-free simple sequent calculus for modal logic S5*, The Review of Symbolic Logic **1** (2008), pp. 3–15.
5. S. Negri, *Proof analysis in modal logic*, Journal of Philosophical Logic **34** (2005), pp. 507–544.

*Institut für Theoretische Informatik, Mathematik und Operations Research, Fakultät für Informatik, UniBw München, 85577 Neubiberg, Germany.
Email: Birgit.Elbl@unibw.de

Schemata for Proofs by Coinduction

Anton Setzer[1]

Dept. of Computer Science, Swansea University, Singleton Park, Swansea SA2 8PP
a.g.setzer@swan.ac.uk

Proofs by induction are carried out by following schemata for induction, which makes it easier to carry out such kind of proofs than by using directly the fact that the natural numbers is the least set closed under zero and successor. So for proving $\forall x.\varphi(x)$, one doesn't define first $A := \{x \in \mathbb{N} \mid \varphi(x)\}$ and show that A is closed under 0 and successor. Instead, one uses the schema of induction. Although using the schema of induction amounts to essentially the same as showing the closure properties of A, using the schema of induction is much easier to use and to teach.

Proofs by coinduction usually follow directly the principle that the coinductively defined set is the largest set satisfying the principles of the coinductively defined set. For instance for carrying out proofs of bisimulation, one usually introduces a relation and shows that it is a bisimulation relation. This makes proofs by coinduction cumbersome and difficult to teach.

In this talk we will introduce schemata for coinduction which are similar to the schemata for induction. The use of the coinduction hypothesis is made easier by defining coinductively defined sets as largest sets allowing observations, rather than as largest sets closed under introduction rules. For instance the set Stream of streams of natural numbers is the largest set allowing observations $\mathrm{head} : \mathrm{Stream} \to \mathbb{N}$ and $\mathrm{tail} : \mathrm{Stream} \to \mathrm{Stream}$, rather than being the largest set closed under $\mathrm{cons} : \mathbb{N} \to \mathrm{Stream} \to \mathrm{Stream}$.

Based on this we will first introduce schemata for defining functions by primitive corecursion or guarded recursion. This is dual to the principle of primitive recursion. Then we define schemata for coinductive proofs of equality. Finally we introduce schemata for coinductively defined relations such as bisimulation relations.

We will give examples of how to carry out coinductive proofs on paper. These proofs will make use of the coinduction hypothesis, where restrictions that are dual to those for the use of the induction hypothesis in inductive proofs are used.

The general theory of schemata for coinductive proofs can be found in our article [1].

References

[1] Anton Setzer. How to reason coinductively informally. Accepted for publication in Reinhard Kahle, Thomas Strahm, Thomas Studer (Eds.): Festschrift on occasion of Gerhard Jäger's 60th birthday, Springer., 20 November 2015.

Atomic Lambda Calculus and its connections with Sharing Graphs

David R. Sherratt

The purpose of sharing in the λ-calculus is to have better control over duplication. Sharing is the use of a single representation for multiple instances of a common subterm. During evaluation, instead of duplicating a term, we can share it. This allows us to evaluate all the copies of the subterm simultaneously, by evaluating their shared representation [5]. Sharing is implemented using various techniques, including `let`-expressions in term calculi. In the atomic λ-calculus [2], sharing is expressed as $u[y_1, ..., y_n \leftarrow t]$ where t is being shared in u, and the variables y_1 through y_n in u represent the shared instances of t.

With sharing, duplication can be delayed until the point where the instances of a shared function are applied to distinct arguments. Since a function is destroyed upon evaluation, at this point duplication becomes necessary. The question is then how much do we duplicate: do we need to copy the whole term or can parts remain shared? With simple sharing mechanisms such as the above, the maximum that can remain shared upon duplication are the *maximal free subexpressions* (MFS) of a term (the largest subterms that, if they contain a variable x, they also contain its binder λx). A formalism that duplicates all but the maximal free subexpressions of a term is called *fully lazy*.

Atomic λ-calculus achieves full laziness by effectively integrating the duplication process with MFS-extraction. MFS-extraction is obtained through the duplication process which is atomic. At any point, a shared term is either a MFS where it is permuted so it is not duplicated but shared, or it is not a MFS in which case we begin duplicating.

Atomic λ-calculus extends λ-calculus with explicit sharing and a *distributor* construct [2]. The introduction of the distributor allows us to duplicate terms atomically and refine the computation of terms into smaller steps. The distributor works directly on the λ-abstractions $\lambda x.t$. The calculus uses the distributor to duplicate the body t n times into the tuple $\langle t_1, .., t_n\rangle$ while maintaining one copy of the constructor to obtain $\lambda x.\langle t_1, .., t_n\rangle$, and then to distribute to obtain n copies of $\lambda x.t$. Atomic λ-calculus duplicates terms via sharing reductions:

$$\lambda x.t \rightsquigarrow \lambda x.\langle y_1, ..., y_n\rangle[y_1, ..., y_n \leftarrow t] \rightsquigarrow^* \lambda x.\langle t_1, ..., t_n\rangle \rightsquigarrow \lambda x.t, ..., \lambda x.t$$

The calculus is a Curry-Howard interpretation of a deep inference [1] proof system for intuitionistic logic. The *distribution rule* enables the atomicity property in deep inference: it allows a contraction on an implication to be reduced locally i.e. without duplicating the whole subproof. This property is what allows the introduction of the distributor in term calculus, a computational interpretation of the distribution rule (a combination of the medial rule and the co-contraction rule).

$$\frac{A \to (B \wedge C)}{(A \to B) \wedge (A \to C)}\ d \qquad\qquad \frac{A \to B}{(A \to B) \wedge (A \to b)}\ c \sim \frac{\mathrm{A} \to \dfrac{B}{B \wedge B}\ c}{(A \to B) \wedge (A \to B)}\ d$$

A natural intuitive graphical interpretation of atomic λ-calculus is used for illustration purposes, but they have interesting similarities with sharing graphs [3]. Sharing graphs implement λ-expression reduction that avoid any copying that could later cause duplication of work, an *optimal* [4] algorithm for λ-calculus reduction. This is achieved through graph reduction techniques. However, there are significant differences between the two. The calculus is fully lazy but not optimal like sharing graphs, however the calculus has global typing unlike sharing graphs which have only local typing.

The aims of my research is to make precise the connection between the atomic λ-calculus and sharing graphs. We wish to make the intuitive graphs used to illustrate atomic λ-calculus more precise, formally connecting the atomic λ-calculus and sharing graphs. To achieve this one would have to refine the control mechanisms for the atomic λ-calculus i.e. explore efficient control mechanisms for a graph-based version of the atomic λ-calculus. This will allow us to compare more directly the control mechanisms of the calculus and sharing graphs: to characterize the distinction and similarities with the formal graphical illustration of atomic λ-calculus and sharing graphs, and to explore the area between full laziness and optimality.

References

[1] Alessio Guglielmi, Tom Gundersen, and Michel Parigot. *A proof calculus which reduces syntactic bureaucracy.* (2010): 135-150.

[2] Tom Gundersen, Willem Heijltjes, and Michel Parigot. *Atomic lambda calculus: A typed lambda-calculus with explicit sharing.* Proceedings of the 2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science. IEEE Computer Society, 2013.

[3] John Lamping. *An algorithm for optimal lambda calculus reduction.* Proceedings of the 17th ACM SIGPLAN-SIGACT symposium on Principles of programming languages. ACM, 1989.

[4] Jean-Jacques Lévy. *Optimal reductions in the lambda-calculus.* To HB Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism (1980): 159-191.

[5] Christopher Peter Wadsworth. *Semantics and Pragmatics of the Lambda-Calculus.* Diss. University of Oxford, 1971.

Decomposing First-Order Proofs using Deep Inference

Benjamin Ralph

University of Bath

The deep-inference formalism, by allowing for very fine-grained inference steps and freer composition of proofs, has produced important results and innovations in various logics, especially classical propositional logic. A natural progression is to extend these insights to classical first-order logic (FOL) but, although a direct cut-elimination procedure has been provided [2], there has been no work as of yet that incorporates the many perspectives and techniques developed in the last ten years.

In the talk, I will give the outline of a new cut elimination procedure for FOL in deep inference, as well as a decomposition-style presentation of Herbrand's Theorem called a *Herbrand Stratification* that is proved not as a corollary of cut elimination, but in tandem with it. In doing so, I hope to provide a different and perhaps better perspective on FOL normalisation, Herbrand's Theorem, and their relationship. More concretely, there is good reason to believe that, as in propositional logic [1], this research can provide us with new results in proof complexity.

Deep Inference Deep inference differs from the sequent calculus in that composition of proofs is allowed with the same connectives that are used for the composition of formulae [5]. Thus in classical propositional logic, two proofs $\begin{array}{c} A \\ \phi \| \\ B \end{array}$ and $\begin{array}{c} C \\ \psi \| \\ D \end{array}$ can be composed not only with conjunction, as is possible in the sequent calculus, but also with disjunction:

$$\begin{array}{c} A \\ \phi \| \\ B \end{array} \wedge \begin{array}{c} C \\ \psi \| \\ D \end{array} = \begin{array}{c} A \wedge C \\ \phi \wedge \psi \| \\ B \wedge D \end{array} , \qquad \begin{array}{c} A \\ \phi \| \\ B \end{array} \vee \begin{array}{c} C \\ \psi \| \\ D \end{array} = \begin{array}{c} A \vee C \\ \phi \vee \psi \| \\ B \vee D \end{array}$$

This freedom of composition has enabled many proof-theoretic innovations: the reduction of cut to atomic form by a local procedure of polynomial-time complexity [3], and the development of a quasi-polynomial cut elimination procedure for propositional logic using a geometric invariant of proofs known as the *atomic flow* [5]. In FOL, we also allow quantifiers to be applied to proofs, not only formulae:

$$\exists x \left[\begin{array}{c} A \\ \phi \| \\ B \end{array} \right] = \begin{array}{c} \exists x A \\ \exists x \phi \| \\ \exists x B \end{array} , \qquad \forall x \left[\begin{array}{c} A \\ \phi \| \\ B \end{array} \right] = \begin{array}{c} \forall x A \\ \forall x \phi \| \\ \forall x B \end{array}$$

Normalisation in Deep Inference. Recently, study of normalisation in deep-inference proof systems has led to the perspective that the process is a conflation of two mechanisms that operate on two distinct composition methods: contraction and a linear cut. When normalised, the first of these mechanisms increases complexity, whereas the second reduces it. Thus, two-stage cut elimination procedures for proof systems are being developed: those which first *decompose* a proof into a suitable form before linear cut elimination then is performed.

Herbrand's Theorem as Decomposition I will show how for FOL, a certain presentation of Herbrand's theorem I call a *Herbrand Stratification* effectively carries out the decomposition phase of normalisation. This inverts the more common idea of using cut elimination to prove Herbrand's Theorem [4], and fits with the complexity narrative: proving Herbrand's Theorem constructively requires increasing the size of a proof, possibly greatly [6].

The proof of Herbrand's Theorem that I will present, which draws inspiration from normalisation techniques developed for propositional logic that use the atomic flow, proceeds by combining existential instantiation and contraction into a single rule, called a *Herbrand Expander*:

$$\mathsf{cont}\,\frac{\exists x A \vee \mathsf{n}{\downarrow}\dfrac{A[a_1/x]}{\exists x A} \vee \ldots \vee \mathsf{n}{\downarrow}\dfrac{A[a_n/x]}{\exists x A}}{\exists x A} \longrightarrow \mathsf{h}{\downarrow}\,\frac{\exists x A \vee A[a_i/x]_1^n}{\exists x A}$$

and pushing these rules to the bottom of the proof. The result, what I call a *Herbrand Stratification* of a proof, is a version of Herbrand's Theorem for deep inference and allows for propositional linear cut elimination methods to be used to complete normalisation.

$$\text{Herbrand Stratification}: \begin{array}{c} \phi \| \\ A \end{array} \longrightarrow \begin{array}{l} \| \text{Propositional Rules} \\ H(A) \\ \| \text{Herbrand Expanders} \\ A \end{array}$$

References

1. P. Bruscoli, A. Guglielmi, T. Gundersen, and M. Parigot. A quasipolynomial cut-elimination procedure in deep inference via atomic flows and threshold formulae. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 136–153. Springer, 2010.
2. K. Brünnler. Cut elimination inside a deep inference system for classical predicate logic. *Studia Logica*, 82(1):51–71, 2006.
3. K. Brünnler and A. F. Tiu. A local system for classical logic. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 347–361. Springer, 2001.
4. S. R. Buss. On Herbrand's theorem. In *Logic and Computational Complexity*, pages 195–209. Springer, 1995.
5. A. Guglielmi, T. Gundersen, and M. Parigot. A proof calculus which reduces syntactic bureaucracy. *Leibniz International Proceedings in Informatics (LIPIcs)*, 6:16, 2010.
6. R. Statman. Lower bounds on Herbrand's theorem. *Proceedings of the American Mathematical Society*, pages 104–107, 1979.

A Type Theory for Comprehensive Parametric Polymorphism

Fredrik Nordvall Forsberg

April 27, 2016

A class of models of System F is presented, where reflexive-graph-category structure for relational parametricity [DR04] and fibrational models of impredicative polymorphism [See87] are combined. To achieve this, we modify the definition of fibrational model of impredicative polymorphism by adding one further ingredient to the structure: comprehension in the sense of Lawvere [Law70]. Such comprehensive models, once further endowed with reflexive-graph-category structure, enjoy the expected consequences of parametricity. To prove this, we introduce a type theory extending System F with a sound and complete semantics with respect to the class of models considered. Proving the consequences of parametricity within this type theory requires new techniques since equality relations are not available, and standard arguments that exploit equality need to be reworked.

This is joint work with Alex Simpson and Neil Ghani, based on a paper recently published at FoSSaCS [GNFS16].

References

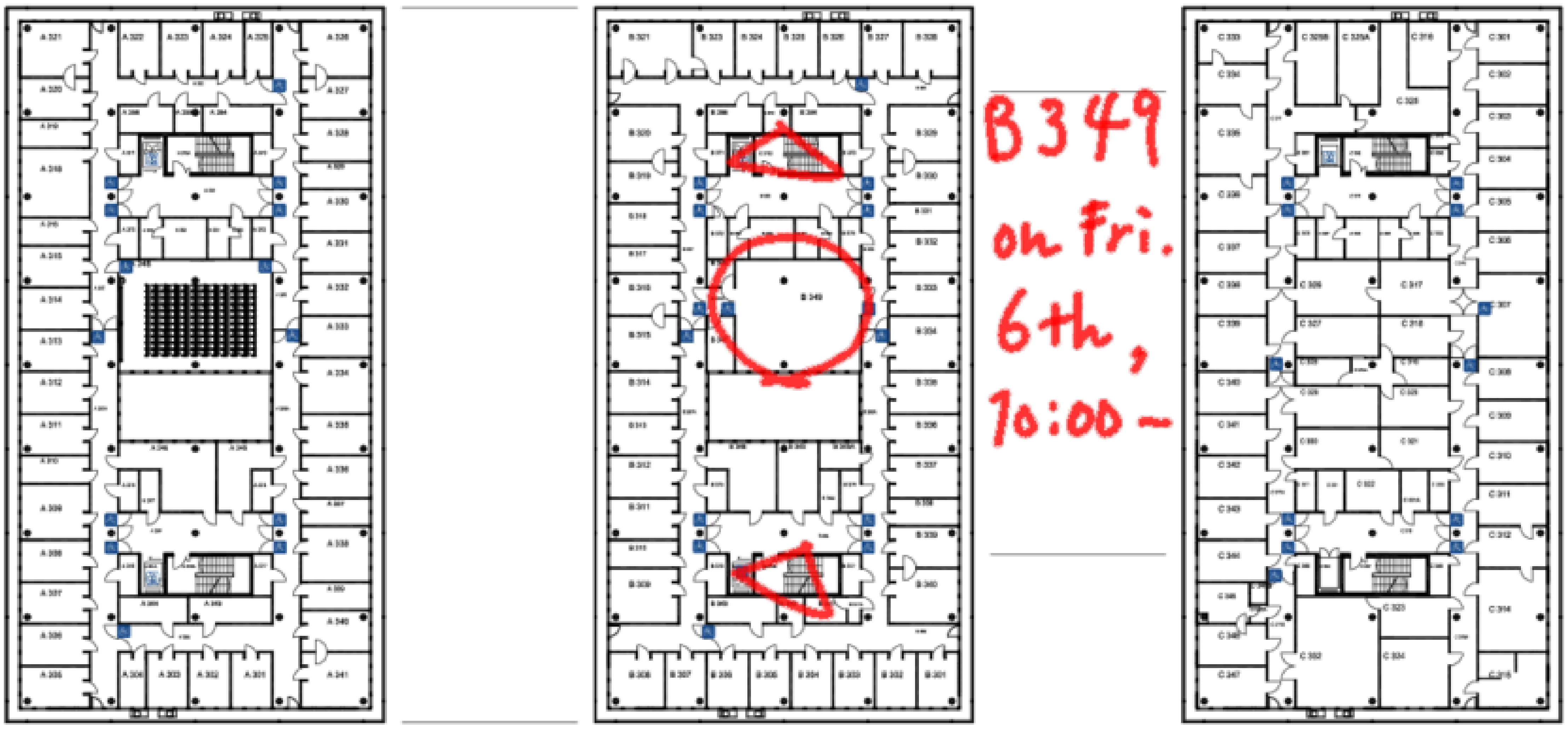
[DR04] Brian Dunphy and Uday Reddy. Parametric limits. In *LICS 2004*, pages 242–251, 2004.

[GNFS16] Neil Ghani, Fredrik Nordvall Forsberg, and Alex Simpson. Comprehensive parametric polymorphism: categorical models and type theory. In B. Jacobs and C. Löding, editors, *FoSSaCS 2016*, volume 9634 of *LNCS*, pages 3–19. Springer, 2016.

[Law70] F William Lawvere. Equality in hyperdoctrines and comprehension schema as an adjoint functor. *Applications of Categorical Algebra*, 17:1–14, 1970.

[See87] Robert A.G Seely. Categorical semantics for higher order polymorphic lambda calculus. *Journal of Symbolic Logic*, pages 969–989, 1987.

3rd Floor

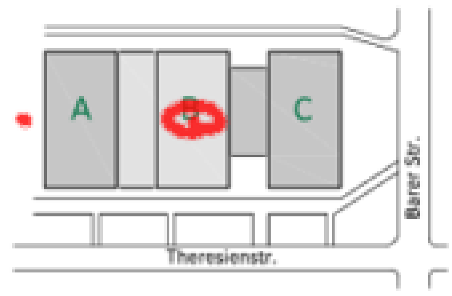


B349 on Fri. 6th, 10:00~

use stairs or elevators marked by ◁ on Fri.

- Rampe voll zugänglich
- Rampe einge. zugänglich
- Gebäude voll zugänglich
- Gebäude einge. zugänglich
- Personenaufzug voll zugänglich
- Personenaufzug einge. zugänglich
- WC voll zugänglich
- WC eingeschränkt zugänglich
- Audio Anlage für Hörbehinderten

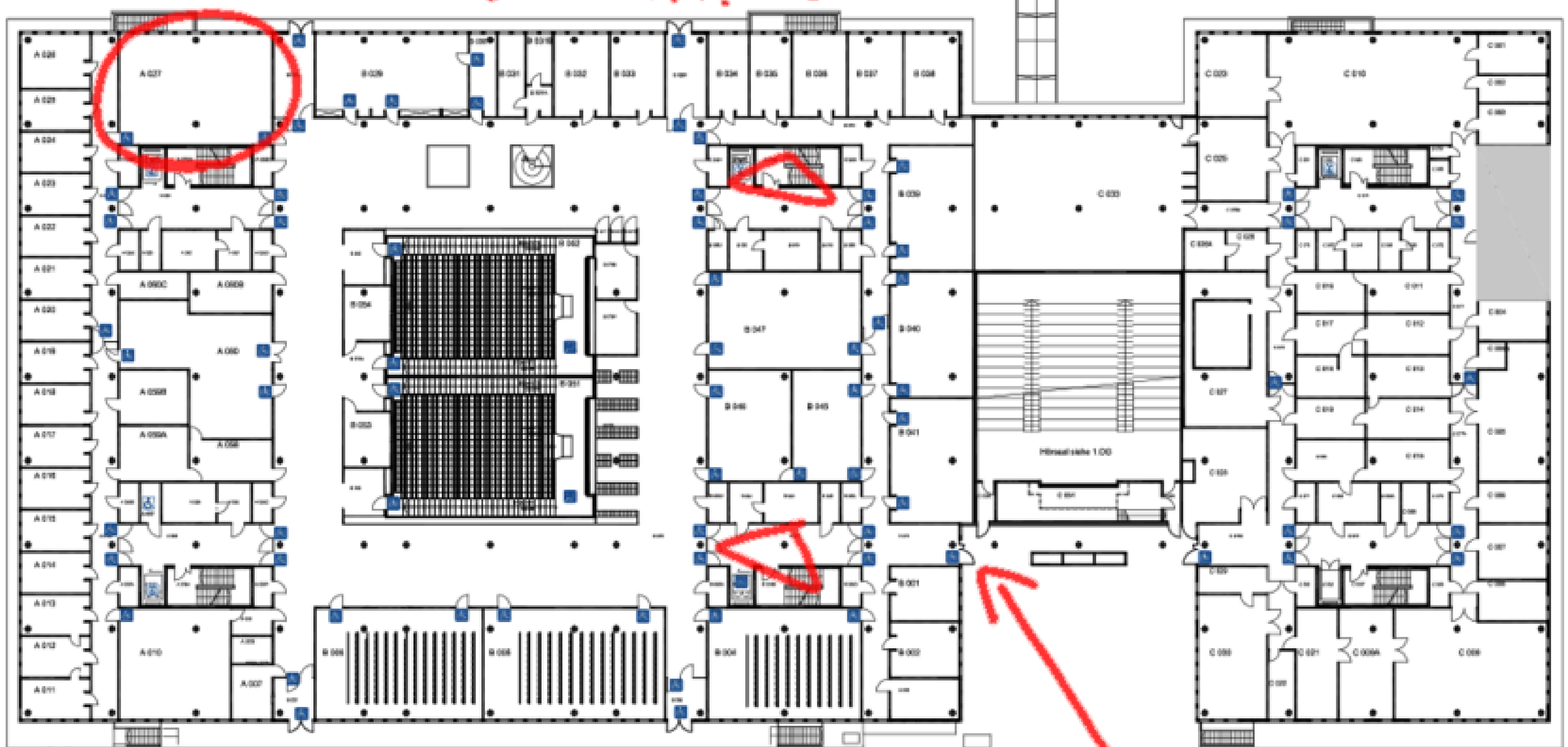
Ergänzende Informationen unter dem Link
www.lmu.de/barrierefrei



LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

A027 on Thu. 5th 9:00~18:10 & on Fri. 6th 9:00~10:00

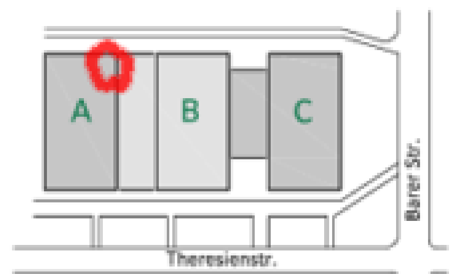
Ground Floor



Entrance on Thu.

- Rampe voll zugänglich
- Rampe einge. zugänglich
- Gebäude voll zugänglich
- Gebäude einge. zugänglich
- Personenaufzug voll zugänglich
- Personenaufzug einge. zugänglich
- WC voll zugänglich
- WC eingeschränkt zugänglich
- Audio Anlage für Hörbehinderten

Ergänzende Informationen unter dem Link
www.lmu.de/barrierefrei



LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN