Final Exam

15-317/657 Constructive Logic

André Platzer

May 8, 2020

Name: ______________________________

Andrew ID: ______________________________

Instructions

- **For fairness reasons all answers must be typed on a computer in text editors/text-processing (e.g. LaTeX) and submitted as PDF. Besides PDF viewers, no other software is allowed and no handwritten answers/scans are accepted. You can use scratch paper but not hand it in.**
- **Only** the following resources can be used during this exam:
  1. 15317 lecture and recitation notes
  2. editors or text-processing software
  3. **private** Piazza posts or email with course staff

  All other communications with anyone about the exam or this course during the exam period constitute an academic integrity violation.
- You have 24 hours from when the exam was available to complete it.
- There are 4 problems on 5 pages.
- **Submit** on GradeScope → Final → Submit assignment

	Max	Score
Proof Terms	90	
Propositional Theorem Proving	80	
Prolog Principles	50	
Linear Logic Cuts	80	
Total:	300	

1 Proof Terms (90 points)

This question studies proof terms of natural deduction. Recall that a proof term is called *abnormal* if it can be reduced by some local reduction of proof terms. Otherwise *normal*/irreducible.

10 **Task 1** Give a **normal** proof term for $\big((A \supset C) \wedge (B \supset C)\big) \supset \big((A \vee B) \supset (C \vee C)\big)$ or explain why that is impossible.

10 **Task 2** Give an **abnormal** proof term for $(A \supset (B \wedge C)) \supset \big(A \supset C\big)$ or explain why that is impossible.

10 **Task 3** Give a **normal** proof term justifying $A \supset ((A \vee B) \supset A)$ or explain why that is impossible.

10 **Task 4** Give an **abnormal** proof term justifying $A \supset ((A \vee B) \supset A)$ or explain why that is impossible.

10 **Task 5** Give an **abnormal** proof term justifying $(A \vee B) \supset A$ or explain why that is impossible.

20 **Task 6** Briefly **explain** whether there is a true proposition A of intuitionistic propositional logic for which there is no proof term M such that $M : A$ proves.

20 **Task 7** Briefly **explain** whether there is a true proposition A of intuitionistic propositional logic for which there is no **abnormal** proof term M such that $M : A$ proves.

2 Propositional Theorem Proving (80 points)

The contraction-free sequent calculus $\longrightarrow$ is *sound* and *complete* w.r.t. $\Longrightarrow$ and *terminates*: all its premises are strictly smaller in a well-founded ordering. Each of the following tasks drops one rule from our original contraction-free sequent calculus and replaces it with another. **Explain** whether these properties still hold when replacing *only* the indicated rule and **mark (s)** for **s**ound wrt. $\Longrightarrow$, **(u)** for **u**nsound, **(c)** for **c**omplete wrt. $\Longrightarrow$, **(i)** for **i**ncomplete, **(t)** for **t**erminating, **(n)** for **n**onterminating. If they fail, show an example demonstrating the failure. To get you started here's a simple example: Replacing rule $\wedge R$ by rule $P0$ would make it

$$\frac{\Gamma \longrightarrow A \quad \Gamma \longrightarrow B}{\Gamma \longrightarrow A \wedge B}\ \wedge R \qquad \frac{\Gamma \longrightarrow A}{\Gamma \longrightarrow A \wedge B}\ P0$$

(u) because $\longrightarrow \top \wedge \bot$ proves by $P0 + \top R$ but is (constructively) false as it implies $\bot$ by $\wedge L$.
(c) every sequent provable by $\wedge R$ is provable by $P0$, which has a subset of the premises of $\wedge R$.
(t) the same ordering shows termination because $P0$ produces a subset of the premises of $\wedge R$.

20 **Task 1** Explain what happens when we only replace rule $\vee{\supset}L$ by rule $P1$:

$$\frac{\Gamma, A_1 \supset B, A_2 \supset B \longrightarrow C}{\Gamma, (A_1 \vee A_2) \supset B \longrightarrow C}\ \vee{\supset}L \qquad \frac{\Gamma, A_1 \supset B \longrightarrow C}{\Gamma, (A_1 \vee A_2) \supset B \longrightarrow C}\ P1$$

20 **Task 2** Explain what happens when we only replace rule $\vee R_2$ by rule $P2$:

$$\frac{\Gamma \longrightarrow B}{\Gamma \longrightarrow A \vee B}\ \vee R_2 \qquad \frac{\Gamma \longrightarrow B \vee A}{\Gamma \longrightarrow A \vee B}\ P2$$

20 **Task 3** Explain what happens when we only replace rule $\bot{\supset}L$ by rule $P3$:

$$\frac{\Gamma \longrightarrow C}{\Gamma, \bot \supset B \longrightarrow C}\ \bot{\supset}L \qquad \frac{\Gamma, \top \supset B \longrightarrow C}{\Gamma, \bot \supset B \longrightarrow C}\ P3$$

20 **Task 4** Explain what happens when we only replace rule $P{\supset}L$ by rule $P4$:

$$\frac{P \in \Gamma \quad \Gamma, B \longrightarrow C}{\Gamma, P \supset B \longrightarrow C}\ P{\supset}L \qquad \frac{\Gamma \longrightarrow P \quad \Gamma, B \longrightarrow C}{\Gamma, P \supset B \longrightarrow C}\ P4$$

3 Prolog Principles (50 points)

This question studies symbolic computation in Prolog with polynomials in one variable (written x). Polynomials are represented as a list of integer coefficients, e.g.:

```
[5,6,7,8]   represents the polynomial   5 + 6*x + 7*x^2 + 8*x^3
```

In this question you will define predicates `padd/3`, `pscale/3`, `pmul/3` to compute the representation of polynomials representing polynomial addition, scaling, and multiplication, respectively. For example, the following queries are expected to succeed:

```
padd([1,2,3],[5,6],[6,8,3]),pscale(3,[1,2],[3,6]),pmul([1,2,3],[5,7],[5,17,29,21]).
```

Modes describe the intended ways of using a predicate. Mode `+pol` indicates an input argument that needs to be provided satisfying `pol/1`. Mode `-pol` indicates an output argument satisfying `pol/1` that will be computed by the predicate when all inputs are provided, where:

```
pol([A|As]) :- integer(A), pol(As).
pol([]).
```

10 **Task 1** Write a Prolog program `padd(+pol,+pol,-pol)` that takes two `pol` representations as inputs in the first and second arguments and produces a `pol` representation of their sum as the output in the third argument.

10 **Task 2** Write a Prolog program `pscale(+integer,+pol,-pol)` that takes an integer as input in the first argument, a `pol` representation as input in the second argument and produces a `pol` representation of the second argument multiplied/scaled by the first argument as the output in the third argument.

30 **Task 3** Write a Prolog program `pmul(+pol,+pol,-pol)` that takes two `pol` representations as inputs in the first and second arguments and produces a `pol` representation of the product of the input polynomials as the output in the third argument.

4 Linear Logic Cuts (80 points)

This question studies cuts in linear logic. We simply write $\Delta, A \Vdash C$ for $\Delta, A\ \mathit{res} \Vdash C\ \mathit{true}$. Recall that the *linear* cut theorem for linear logic constructs a deduction $\mathcal{F}$ from deductions $\mathcal{D}$ and $\mathcal{E}$ and (just like the ordinary cut theorem for intuitionistic logic) is also proved by induction on the structure of the formula A as well as the deductions $\mathcal{D}$ and $\mathcal{E}$.

Theorem 1 (Linear cut) *If* $\overset{\mathcal{D}}{\Delta \Vdash A}$ *and* $\overset{\mathcal{E}}{\Delta', A \Vdash C}$ *then* $\overset{\mathcal{F}}{\Delta, \Delta' \Vdash C}$.

[20] **Task 1** Provide and briefly explain a counterexample justifying from its resource semantics why the *ordinary* structural cut theorem of intuitionistic logic does *not* hold for linear logic:

If $\Delta \Vdash A$ and $\Delta, A \Vdash C$ then $\Delta \Vdash C$

[20] **Task 2** Commodore Horgiatiki performed one case of linear cut elimination. But he is missing some parts and is unsure whether he got a correct proof. Fill in **all** missing arguments and justifications and steps so that you obtain a complete proof. If there are any errors or missing justifications in Horgiatiki's proof, clearly mark and explain in one line. Unnecessary steps are not necessarily incorrect but still need a justification of their (in)correctness.

$$\mathcal{D} = \dfrac{\overset{\mathcal{D}_1}{\Delta \Vdash A_1} \quad \overset{\mathcal{D}_2}{\Delta \Vdash A_2}}{\Delta \Vdash A_1 \,\&\, A_2}\,\&R \quad \text{and} \quad \mathcal{E} = \dfrac{\overset{\mathcal{E}_1}{\Delta', A_1 \Vdash C}}{\Delta', A_1 \,\&\, A_2 \Vdash C}\,\&L_1$$

$\Delta \Vdash A_1$	1 By ______________________
$\Delta \Vdash A_2$	2 By ______________________
$\Delta', A_1 \Vdash C$	3 By ______________________
$\Delta', A_2 \Vdash C$	4 By ______________________
$\Delta, \Delta' \Vdash C$	5 By ______________________

[20] **Task 3** Prove the case of the *linear* cut theorem where $\mathcal{D}$ ends with $\multimap R$ and $\mathcal{E}$ ends with $\multimap L$:

$$\mathcal{D} = \dfrac{\overset{\mathcal{D}_1}{\Delta, A_1 \Vdash A_2}}{\Delta, \Vdash A_1 \multimap A_2}\,\multimap R \quad \text{and} \quad \mathcal{E} = \dfrac{\overset{\mathcal{E}_1}{\Delta'_1 \Vdash A_1} \quad \overset{\mathcal{E}_2}{\Delta'_2, A_2 \Vdash C}}{\Delta'_1, \Delta'_2, A_1 \multimap A_2 \Vdash C}\,\multimap L$$

[20] **Task 4** When replacing $\multimap$ by $\supset$ and $\Vdash$ by $\Longrightarrow$ does a proof of Task 3 justify the case of cut formula $A_1 \supset A_2$ as principal formula of the ordinary cut theorem for intuitionistic logic? Explain.