Cedille2: A proof theoretic redesign of the Calculus of Dependent Lambda Eliminations

by

Andrew Marmaduke

A thesis submitted in partial fulfillment
of the requirements for the Doctor of Philosophy degree
in Computer Science
in the Graduate College
of The University of Iowa

May 2024

Thesis Committee: Aaron Stump, Thesis Supervisor
Cesare Tinelli
J. Garrett Morris
Sriram Pemmaraju
William J. Bowman

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

– Some wise dude

ACKNOWLEDGMENTS

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

ABSTRACT

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

PUBLIC ABSTRACT

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

PREFACE

Chapter 1

INTRODUCTION TO TYPE THEORY

Chapter 2

THEORY DESCRIPTION AND BASIC METATHEORY

$$
\begin{aligned}
t &::= x \mid \mathfrak{b}(\kappa_1, x : t_1, t_2) \mid \mathfrak{c}(\kappa_2, t_1, \ldots, t_{\mathfrak{a}(\kappa_2)}) \\
\kappa_1 &::= \lambda_m \mid \Pi_m \mid \cap \\
\kappa_2 &::= \star \mid \Box \mid \bullet_m \mid \text{pair} \mid \text{proj}_1 \mid \text{proj}_2 \mid \text{eq} \mid \text{refl} \mid J \mid \vartheta \mid \delta \mid \phi \\
m &::= \omega \mid 0 \mid \tau \\
&\mathfrak{a}(\star) = \mathfrak{a}(\Box) = 0 \\
&\mathfrak{a}(\text{proj}_1) = \mathfrak{a}(\text{proj}_2) = \mathfrak{a}(\text{refl}) = \mathfrak{a}(\vartheta) = \mathfrak{a}(\delta) = 1 \\
&\mathfrak{a}(\bullet_m) = 2 \\
&\mathfrak{a}(\text{pair}) = \mathfrak{a}(\text{eq}) = \mathfrak{a}(\varphi) = 3 \\
&\mathfrak{a}(J) = 6
\end{aligned}
$$

$$
\begin{aligned}
\star &:= \mathfrak{c}(\star) & t.1 &:= \mathfrak{c}(\text{proj}_1, t) \\
\Box &:= \mathfrak{c}(\Box) & t.2 &:= \mathfrak{c}(\text{proj}_2, t) \\
\lambda_m\, x{:}t_1.\, t_2 &:= \mathfrak{b}(\lambda_m, x : t_1, t_2) & t_1 =_{t_2} t_3 &:= \mathfrak{c}(\text{eq}, t_1, t_2, t_3) \\
(x : t_1) \to_m t_2 &:= \mathfrak{b}(\Pi_m, x : t_1, t_2) & \text{refl}(t) &:= \mathfrak{c}(\text{refl}, t) \\
(x : t_1) \cap t_2 &:= \mathfrak{b}(\cap, x : t_1, t_2) & \vartheta(t) &:= \mathfrak{c}(\vartheta, t) \\
t_1 \bullet_m t_2 &:= \mathfrak{c}(\bullet_m, t_1, t_2) & \delta(t) &:= \mathfrak{c}(\delta, t) \\
[t_1, t_2, t_3] &:= \mathfrak{c}(\text{pair}, t_1, t_2, t_3) & \varphi(t) &:= \mathfrak{c}(\varphi, t)
\end{aligned}
$$

$$
J(t_1, t_2, t_3, t_4, t_5, t_6) := \mathfrak{c}(J, t_1, t_2, t_3, t_4, t_5, t_6)
$$

Figure 2.1: Generic syntax, there are three constructors, variables, a generic binder, and a generic non-binder. Each are parameterized with a constant tag to specialize to a particular syntactic consruct. The non-binder constructor has a vector of subterms determined by an arity function computed on tags. Standard syntactic constructors are defined in terms of the generic forms.

$$
\begin{aligned}
|x| &= x & |f \bullet_\tau a| &= |f| \bullet_\tau |a| \\
|\star| &= \star & |[t_1, t_2, T]| &= |t_1| \\
|\Box| &= \Box & |t.1| &= |t| \\
|\lambda_0\, x : A.\, t| &= |t| & |t.2| &= |t| \\
|\lambda_\omega\, x : A.\, t| &= \lambda_\omega\, x.\, |t| & |x =_A y| &= |x| =_{|A|} |y| \\
|\lambda_\tau\, x : A.\, t| &= \lambda_\tau\, x : |A|.\, |t| & |\mathrm{refl}(t)| &= \lambda_\omega\, x.\, x \\
|(x : A) \to_m B| &= (x : |A|) \to_m |B| & |J(A, P, x, y, e, w)| &= |e| \bullet_\omega |w| \\
|(x : A) \cap B| &= (x : |A|) \cap |B| & |\vartheta(e)| &= |e| \\
|f \bullet_0 a| &= |f| & |\delta(e)| &= |e| \\
|f \bullet_\omega a| &= |f| \bullet_\omega |a| & |\varphi(a, f, e)| &= |a|
\end{aligned}
$$

Figure 2.2: Erasure of syntax, for type-like and kind-like syntax erasure is homomorphic, for term-like syntax erasure reduces to the untyped lambda calculus.

$$
\frac{t_1 \rightsquigarrow_\beta t_1'}{\mathfrak{b}(\kappa, x : t_1, t_2) \rightsquigarrow_\beta \mathfrak{b}(\kappa, x : t_1', t_2)}
\qquad
\frac{t_2 \rightsquigarrow_\beta t_2'}{\mathfrak{b}(\kappa, x : t_1, t_2) \rightsquigarrow_\beta \mathfrak{b}(\kappa, x : t_1, t_2')}
$$

$$
\frac{t_i \rightsquigarrow_\beta t_i' \qquad i \in 1, \ldots, \mathfrak{a}(\kappa)}{\mathfrak{c}(\kappa, t_1, \ldots t_i, \ldots t_{\mathfrak{a}(\kappa)}) \rightsquigarrow_\beta \mathfrak{c}(\kappa, t_1, \ldots t_i', \ldots t_{\mathfrak{a}(\kappa)})}
$$

$$
\begin{aligned}
(\lambda_m\, x : A.\, b) \bullet_m t &\rightsquigarrow_\beta [x := t]b \\
[t_1, t_2, A].1 &\rightsquigarrow_\beta t_1 \\
[t_1, t_2, A].2 &\rightsquigarrow_\beta t_2 \\
J(A, P, x, y, \mathrm{refl}(z), w) &\rightsquigarrow_\beta w \bullet_0 z \\
\vartheta(\mathrm{refl}(t.1)) &\rightsquigarrow_\beta \mathrm{refl}(t) \\
\vartheta(\mathrm{refl}(t.2)) &\rightsquigarrow_\beta \mathrm{refl}(t) \\
\varphi(a, f, e).1 &\rightsquigarrow_\beta a
\end{aligned}
$$

Figure 2.3: Reduction rules for arbitrary syntax.

$$
\begin{aligned}
\mathrm{dom}_\Pi(\omega, K) &= \star & \qquad \mathrm{codom}_\Pi(\omega) &= \star \\
\mathrm{dom}_\Pi(\tau, K) &= K & \qquad \mathrm{codom}_\Pi(\tau) &= \Box \\
\mathrm{dom}_\Pi(0, K) &= K & \qquad \mathrm{codom}_\Pi(0) &= \star
\end{aligned}
$$

Figure 2.4: Domain and codomains for function types. The variable K is either $\star$ or $\Box$.

$$\frac{\vdash \Gamma}{\Gamma \vdash \star \rhd \Box}\ \textsc{Axiom} \qquad \frac{\vdash \Gamma \qquad (x : A) \in \Gamma}{\Gamma \vdash x \rhd A}\ \textsc{Var}$$

$$\frac{\Gamma \vdash t \rhd A \qquad A \rightsquigarrow_\beta^* B}{\Gamma \vdash t \mathrel{\Vert\!\rhd} B}\ \textsc{HdInf} \qquad \frac{\begin{array}{c}\Gamma \vdash t \rhd A \\ \Gamma \vdash B \mathrel{\Vert\!\rhd} K \qquad A \equiv B\end{array}}{\Gamma \vdash t \lhd B}\ \textsc{Chk}$$

$$\frac{}{\vdash \cdot}\ \textsc{CtxEm} \qquad \frac{\begin{array}{c}x \notin \mathrm{FV}(\Gamma) \\ \vdash \Gamma \qquad \Gamma \vdash A \mathrel{\Vert\!\rhd} K\end{array}}{\vdash \Gamma, x : A}\ \textsc{CtxApp}$$

$$\frac{\Gamma \vdash A \mathrel{\Vert\!\rhd} \mathrm{dom}_\Pi(m, K) \qquad \Gamma, x : A \vdash B \mathrel{\Vert\!\rhd} \mathrm{codom}_\Pi(m)}{\Gamma \vdash (x : A) \to_m B \rhd \mathrm{codom}_\Pi(m)}\ \textsc{Pi}$$

$$\frac{\Gamma \vdash A \mathrel{\Vert\!\rhd} \mathrm{dom}_\Pi(m, K) \qquad \Gamma, x : A \vdash t \rhd B \qquad x \notin \mathrm{FV}(|t|) \text{ if } m = 0}{\Gamma \vdash \lambda_m\, x{:}A.\, t : (x : A) \to_m B}\ \textsc{Lam}$$

$$\frac{\Gamma \vdash f \mathrel{\Vert\!\rhd} (x : A) \to_m B \qquad \Gamma \vdash a \lhd A}{\Gamma \vdash f \bullet_m a \rhd [x := a]B}\ \textsc{App}$$

Figure 2.5: Inference rules for function types, including erased functions. The variable K is either $\star$ or $\Box$.

$$\frac{\Gamma \vdash A \Rrightarrow \star \qquad \Gamma, x : A \vdash B \Rrightarrow \star}{\Gamma \vdash (x : A) \cap B \rhd \star}\ \textsc{Int} \qquad \frac{\begin{array}{c}\Gamma \vdash T \Rrightarrow (x : A) \to_\tau B \\ \Gamma \vdash t \lhd A \\ \Gamma \vdash s \lhd [x := t]B \qquad |t| =_\beta |s|\end{array}}{\Gamma \vdash [t, s; T] \rhd (x : A) \cap B}\ \textsc{Pair}$$

$$\frac{\Gamma \vdash t \Rrightarrow (x : A) \cap B}{\Gamma \vdash t.1 \rhd A}\ \textsc{Fst} \qquad \frac{\Gamma \vdash t \Rrightarrow (x : A) \cap B}{\Gamma \vdash t.2 \rhd [x := t.1]B}\ \textsc{Snd}$$

Figure 2.6: Inference rules for intersection types.

$$\frac{\begin{array}{c}\Gamma \vdash A \Rrightarrow \star \\ \Gamma \vdash a \lhd A \qquad \Gamma \vdash b \lhd A\end{array}}{\Gamma \vdash a =_A b \rhd \star}\ \textsc{Eq} \qquad \frac{\Gamma \vdash t \rhd A}{\Gamma \vdash \text{refl}(t) \rhd t =_A t}\ \textsc{Refl}$$

$$\frac{\begin{array}{c}\Gamma \vdash A \Rrightarrow \star \\ \Gamma \vdash P \lhd (x\ y : A) \to_\tau (e : x =_A y) \to_\tau \star \qquad \Gamma \vdash x \lhd A \qquad \Gamma \vdash y \lhd A \\ \Gamma \vdash e \lhd x =_A y \qquad \Gamma \vdash w \lhd (a : A) \to_0 P \bullet_\tau a \bullet_\tau a \bullet_\tau \text{refl}(a)\end{array}}{\Gamma \vdash J(A, P, x, y, e, w) \rhd P \bullet_\tau x \bullet_\tau y \bullet_\tau e}\ \text{J}$$

$$\frac{\Gamma \vdash e \Rrightarrow a.i =_T b.j \qquad \Gamma \vdash a \Rrightarrow (x : A) \cap B \qquad \Gamma \vdash b \lhd (x : A) \cap B}{\Gamma \vdash \vartheta(e) \rhd a =_{(x:A)\cap B} b}\ \textsc{Prm}$$

$$\frac{\begin{array}{c}\Gamma \vdash f \Rrightarrow (a : A) \to_\omega (x : A) \cap B \\ \Gamma \vdash a \lhd A \qquad \Gamma \vdash e \lhd (a : A) \to_\omega a =_A (f \bullet_\omega a).1 \qquad \text{FV}(|e|) = \varnothing\end{array}}{\Gamma \vdash \varphi(a, f, e) \rhd (x : A) \cap B}\ \textsc{Cast}$$

$$\frac{\Gamma \vdash e \lhd \text{ctt} =_{\text{cBool}} \text{cff}}{\Gamma \vdash \delta(e) \rhd (X : \star) \to_0 X}\ \textsc{Sep}$$

Figure 2.7: Inference rules for equality types where cBool $:= (X : \star) \to_0 (x : X) \to_\omega (y : X) \to_\omega X$; ctt $:= \lambda_0\, X : \star.\ \lambda_\omega\, x : X.\ \lambda_\omega\, y : X.\ x$; and cff $:= \lambda_0\, X : \star.\ \lambda_\omega\, x : X.\ \lambda_\omega\, y : X.\ y$. Also, $i, j \in \{1, 2\}$

Chapter 3

PROOF NORMALIZATION AND RELATIONSHIP TO SYSTEM $\mathbf{F}^{\omega}$

Chapter 4

CONSISTENCY AND RELATIONSHIP TO CDLE

Chapter 5

OBJECT NORMALIZATION

A φ_i-proof is a proof that allows i nested φ syntactic constructs. For example, a φ_0-proof allows no φ subterms, a φ_1-proof allows φ subterms but no nested φ subterms, and a φ_2-proof allows φ_1 subterms. Defined inductively, a φ_0-proof is a proof with no φ syntactic constructs and a φ_{i+1}-proof is a proof with φ_i-proof subterms.

For any φ_i-proof p there is a strictification $s(p)$ that is a φ_0-proof in Figure 5.1.

Lemma 1 (Strictification Preserves Inference)**.** *Given* $\Gamma \vdash t \rhd A$ *then* $\Gamma \vdash s(t) \rhd A$

Proof. By induction on the typing rule, the φ rule is the only one of interest:

Case:
$$\frac{\overset{\mathcal{D}_1}{\Gamma \vdash f \mathrel{|\!\!\rhd} (a : A) \to_\omega (x : A) \cap B} \quad \overset{\mathcal{D}_2}{\Gamma \vdash a \lhd A} \qquad \overset{\mathcal{D}_3}{\Gamma \vdash e \lhd (a : A) \to_\omega a =_A (f \bullet_\omega a).1} \qquad \overset{\mathcal{D}_4}{\mathrm{FV}(|e|) = \varnothing}}{\Gamma \vdash \varphi(a, f, e) \rhd (x : A) \cap B}$$

Need to show that $\Gamma \vdash s(\varphi(a, f, e)) \rhd (x : A) \cap B$ which reduces to: $\Gamma \vdash s(f) \bullet_\omega s(a) \rhd (x : A) \cap B$. By the IH we know that $s(f)$ infers the same function type, and that $s(a)$ infers the same argument type, therefore the application rule concludes the proof.

□

Lemma 2 (Strict Proofs are Normalizing)**.** *Given* $\Gamma \vdash t \rhd A$ *then* $s(t)$ *is strongly normalizing*

Proof. Direct consequence of strong normalization of proofs □

$$
\begin{array}{rclrcl}
s(x) & = & x & s([s,t,T]) & = & [s(s),s(t),s(T)] \\
s(\star) & = & \star & s(t.1) & = & s(t).1 \\
s(\Box) & = & \Box & s(t.2) & = & s(t).2 \\
s(\lambda_m\, x\!:\!A.\, t) & = & \lambda_m\, x\!:\!s(A).\, s(t) & s(x =_A y) & = & s(x) =_{s(A)} s(y) \\
s((x : A) \to_m B) & = & (x : s(A)) \to_m s(B) & s(\mathrm{refl}(t)) & = & \mathrm{refl}(s(t)) \\
s((x : A) \cap B) & = & (x : s(A)) \cap s(B) & s(\vartheta(e)) & = & \vartheta(s(e)) \\
s(f \bullet_m a) & = & s(f) \bullet_m s(a) & s(\delta(e)) & = & \delta(s(e))
\end{array}
$$

$$
\begin{aligned}
s(J(A,P,x,y,r,w)) &= J(s(A),s(P),s(x),s(y),s(r),s(w)) \\
s(\varphi(a,f,e)) &= s(f) \bullet_\omega s(a)
\end{aligned}
$$

Figure 5.1: Strictification of a proof.

Lemma 3 (Strict Objects are Normalizing)**.** *Given* $\Gamma \vdash t \rhd A$ *then* $|s(t)|$ *is strongly normalizing*

Proof. Proof Idea:

Proof reduction tracks object reduction in the absence of φ constructs. Thus, the normalization of a proof provides an upper-bound on the number of reductions an object can take to reach a normal form. $\Box$

A proof, $\Gamma \vdash t_1 \rhd A$, is contextually equivalent to another proof, $\Gamma \vdash t_2 \rhd A$, if there is no context with hole of type A whose object reduction diverges for t_1 but not t_2. In other words, if a context can be constructed that distinguishes the terms based on their object reduction.

Lemma 4. *A* φ_1*-proof,* p*, is contextually equivalent to its strictification,* $s(p)$

Proof. Proof by induction on the typing rule for p, focus on the application rule:

Case: $$\frac{\overset{\mathcal{D}_1}{\Gamma \vdash f \mathrel{\|\!\rhd} (x : A) \to_m B} \qquad \overset{\mathcal{D}_2}{\Gamma \vdash a \lhd A}}{\Gamma \vdash f \bullet_m a \rhd [x := a]B}$$

In particular, we care about when $f = \varphi(v, b, e).2$ and $m = \omega$. Note that the first projection has a proof-reduction that yields a which makes it unproblematic.

We know that $s(v) = v$ because f is a φ_1-proof. Let v_n be the normal form of v and note that $|v_n|$ is also normal. Likewise, we have e_n and $|e_n|$ normal.

Suppose there is a context $C[\cdot]$ where $|p|$ diverges but $|s(p)|$ normalizes. (Note that the opposite assumption is impossible). If $|v_n|$ is a variable, then reduction in $|p|$ is blocked (contradiction). Otherwise $|v_n| = \lambda\, x.\, x\ t_1\ \cdots\ t_n$ where t_i are normal.

Now it must be the case that $|e \bullet_\omega v| = |e_n| \bullet_\omega |v_n|$ is normalizing. Thus, we have a refl proof that $v_n = (f \bullet_\omega v_n).1$. (Note, this proof *must* be refl because $\mathrm{FV}(|e|) = \varnothing$). But, this implies convertibility, thus $|v_n| =_\beta |f| \bullet_\omega |v_n|$, but this must mean more concretely that $|f| \bullet_\omega |v_n| \rightsquigarrow_\beta |v_n|$. Yet $|f| \bullet_\omega |v_n| \bullet_\omega a$ is strongly normalizing because it is $s(p)$. Therefore, p in this case is strongly normalizing which refutes the assumption yielding a contradiction.

□

Lemma 5. *If t_1 is strongly normalizing and contextually equivalent to t_2 then t_2 is strongly normalizing*

Proof. Immediate by the definition of contextual equivalence. □

Theorem 1. *A φ_i-proof p is strongly normalizing for all i*

Proof. By induction on i.

Case: $i = 0$

Immediate because $s(p) = p$ and strict proofs are strongly normalizing.

Inductive Case:

Suppose that φ_i-proof is strongly normalizing. Goal: show that φ_{i+1}-proof is strongly normalizing.

□

Chapter 6

CEDILLE2: SYSTEM IMPLEMENTATION

Chapter 7

CEDILLE2: INTERNALLY DERIVABLE CONCEPTS

Chapter 8

CONCLUSION AND FUTURE WORK

Appendix A

PROOFS OF CHAPTER 1

Appendix B

UNDECIDED

Hello!

BIBLIOGRAPHY

[1] Kenneth J. Arrow. “A Difficulty in the Concept of Social Welfare”. In: *J Polit Econ* 58.4 (Aug. 1950), pp. 328–346. DOI: `10.1086/256963`. URL: `https://doi.org/10.1086%2F256963`.

[2] Jon Kleinberg. “An Impossibility Theorem for Clustering”. In: *Proceedings of the 15th International Conference on Neural Information Processing Systems*. NIPS’02. Cambridge, MA, USA: MIT Press, 2002, pp. 463–470.

[3] Daniel Müllner. *Modern hierarchical, agglomerative clustering algorithms*. 2011. arXiv: `1109.2378 [stat.ML]`.

[4] R. Sibson. “SLINK: An optimally efficient algorithm for the single-link cluster method”. In: *Comput. J.* 16.1 (Jan. 1973), pp. 30–34. DOI: `10.1093/comjnl/16.1.30`. URL: `https://doi.org/10.1093%2Fcomjnl%2F16.1.30`.

[5] Gurjeet Singh, Facundo Memoli, and Gunnar Carlsson. “Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition”. In: *Eurographics Symposium on Point-Based Graphics*. Ed. by M. Botsch et al. The Eurographics Association, 2007. ISBN: 978-3-905673-51-7. DOI: `10.2312/SPBG/SPBG07/091-100`.

[6] Guillaume Tauzin et al. *giotto-tda: A Topological Data Analysis Toolkit for Machine Learning and Data Exploration*. 2020. arXiv: `2004.02551 [cs.LG]`.

[7] Hendrik van Veen et al. “Kepler Mapper: A flexible Python implementation of the Mapper algorithm.” In: *J. Open Source Softw.* 4.42 (Oct. 2019), p. 1315. DOI: `10.21105/joss.01315`. URL: `https://doi.org/10.21105%2Fjoss.01315`.

[8] Youjia Zhou et al. *Mapper Interactive: A Scalable, Extendable, and Interactive Toolbox for the Visual Exploration of High-Dimensional Data*. 2021. arXiv: `2011.03209 [cs.CG]`.