Logics of Emergent Computation

Meredith L. Patterson
May 30, 2018
Outline

- Background
- Consistency and Paraconsistency
- Kinds of Paraconsistent Logics
- Avenues of Exploration
History: George Boole

- **Boolean algebra**
  - A set
  - Two binary operators
  - One unary operator
- Basis for algebraic interpretations of logic
History: L. E. J. Brouwer and Arend Heyting

- Classical logic: “What assertions are true?”
- Intuitionistic logic: “What can we construct? Are we justified in doing so?”
- Classical logic: $A \land \neg A$ is always false, $A \lor \neg A$ is always true
- Intuitionistic logic: $A \land \neg A$ is still false, but $A \lor \neg A$ is not necessarily true!
- Classical negation: $\neg B$ means “B is false”
- Intuitionistic negation: $\neg B$ means “there exists a counterexample for B”
History: The Curry-Howard Correspondence

- *Types* correspond to *axiom schemas*
- K combinator: $A \to B \to A$ (compare $P \supset Q \supset P$)
- S combinator: $(A \to B \to C) \to (A \to B) \to A \to C$
  (compare $(P \supset Q \supset R) \supset (P \supset Q) \supset P \supset R$)
- Proving a proposition just means constructing a term with the type that corresponds to the proposition
- Different type theories from different logics:
  - Propositional $\to$ function types (simply typed lambda calculus)
  - Predicate $\to$ dependent types
  - Second-order $\to$ parametric polymorphism
History: Per Martin-Löf and Intuitionistic Type Theory

- Dependent functions -> universal quantifier
- Dependent pairs -> existential quantifier
- BHK interpretation: what constitutes a proof of an assertion?
- Negation \( \neg = \) “\( p \) implies a proposition that can’t be fulfilled”
- MLTT gives rise to mechanised proof assistance
Snapshots from the Langsec Workshop: Bogk and Schöpl (2014)

- “Can we write a verified PDF parser in Coq?”
- Well ... kinda.
- A subset terminates. The entire thing doesn’t.
  - “And I’d have gotten away with it, too, if it weren’t for you crazy length fields!”
- Counterexample == nonterminating program on every PDF interpreter!
  - “The illusion that your program is manipulating its data is powerful, but it is an illusion. The data is controlling your program.”
Snapshots from the Langsec Workshop: Vanegue (2015)

- How do we ask the right questions?
- Well, it helps if you have language to ask them in
- So let’s come up with a semantics for heap allocation
- “All problems in computer science can be solved by another layer of indirection.”
What We Talk About When We Talk About Consistency

\[ A \land \neg A \rightarrow B \]
(traditional notation)

\[ A, \neg A \vdash B \]
(turnstile notation)
What We Talk About When We Talk About Paraconsistency

Control Flow: CL, CIL, SL, BIL, LL
Concurrency: CIL, SL, BIL, LL
Codata Types: SL, BIL, LL
Bi-intuitionistic Logic

- Int and co-Int living under the same roof
- Both implication and co-implication
- Both classical negation and paraconsistent negation
Linear Logic

- *Substructural* logics: remove one or more structural rules
- Forbids contraction and weakening
- Instead of implication/coimplication, new operators!
  - Multiplicative conjunction, $\otimes$ (“times”) - simultaneous resource use, consumer-directed
  - Additive disjunction, $\oplus$ (“plus”) - alternative resource use, producer-directed
  - Additive conjunction, $\&$ (“with”) - alternative resource use, consumer-directed
  - Multiplicative disjunction, $\oplus$ (“par”) - simultaneous resource use, producer-directed
  - $!$ (“of course!”)
  - $?$ (“why not?”)
- Undecidable :( 
  - But *affine logic* (which adds global weakening) is ok!
  - Ohai Rust
Subtractive Logic

- Subtraction as dual of implication
- A C-H correspondence has been demonstrated (Tristan Crolard)
  - I don’t entirely understand it yet
  - But, it’s an extension of the lambda-calculus, like the other correspondences
- What kinds of computation do these new calculi allow us to model more effectively?
Now What?

- Get our bearings. Exploits as dual of intended computation -> dualised logics for reasoning about interaction
  - Not all interaction is exploitative. But introduce interaction, and you introduce that possibility.
- Which maps best fit the territory?
- Category-theoretic interpretations
- Mechanise all the logics