Welkin
An Information Language

Oscar Bender-Stone

October 2023

Preface

To be written...

Chapters 1 and 2 provide the philosophical basis for Welkin. Chapter 3 contains the Welkin Standard, a formal specification describing high level, algorithmic properties in compliant Welkin interpreters. Chapter 5 explores applying . Chapter 6 draws several conclusions, pointing to the possible future of mathematical document, programming, and the overall organization of ideas.

Chapter 1

Introduction

Startng Outline:

- Describe the ways philosophers and mathematicians have approached organizing the world around them.
  - Look at Cantor, Dedekind, Frege, Russell, Goedel, and up to the present.
  - Look at modern philosophers
- Reflect on two major shortcomings of these approaches: having little metaphysical explanation about *how* things can be combined, and a lack of generality for several concepts
  - No explanation for what *is* a set fundamentally, or, more importantly, how two entities can be combined into one.
    * Include formal and informal definitions
    * Recognize the prior *tool* used to start math in the first place (again, the ability to combine objects or ideas)
    * Explain that, while category theory may make these constructions more precise, it *still relies on predetermined tools*. These tools are physiological in nature, but they are *ultimately metaphysical.*
  - Lack of generality for: structures, the real numbers, logic (particularly paraconsistent logics). Specifically, mention:
    * Cryptomorphisms and Voldemort's Theorem. This is not part of official literature, but has been given in an extensive document. It serves an important philosophcial point (which will be rexammined in Chapter 2)
    * Mention the number of models for the reals, and how it still has not been decided what the official model is. Mention constructive concerns (overarrching with generality). (Cite Lesnik's paper as a step in the right direction, but critique the lack of generality)

    * Mention the role of bifrucation, and that there is opportunity to expand upon it more (into a generalized fuzzy logic)

- Explain the essential goals of Welkin, built upon CFLT
- Describe target audience
  - This document is geared towards philosophically or mathematically inclined persons. The official version is in English, but, in time, translations will be added.
  - In a different document (separate from this repo), there will be a more accessible version as a part of my program on "humanistic logic".

Chapter 2

Background

Possible beginning quote (at least for Western philosophers). "The Fold: Leibniz and the Barouqe"

> the two instances . . . have no windows, . . . for Leibniz, [this! is because the monad's being-for the world is subject to a condition of closure, all compossible monads including a single and same world. For Whitehead, on the contrary, a condition of opening causes all prehension to be already the prehension of another prehension. . . . Prehension is naturally open, open to the world, without having to pass through a window.

2.1 Western and Eastern Thought

- Contrast the distinct philosophies pervasive in Western and Eastern philosophy. Some key points to emphasize:
  - Mechanistic vs spiritual leaning
  - Classical logic vs Different Logics (particularly involving contradictions)
  - Brief mention in other philosophies (possibly ethics or other non-epistemology related fields?)
  -

2.2 Sets: Georg Cantor and Others

- Cover most of Cantor's writings, including "Beiträge zur Begrundung der transfiniten Megenlehre".
  - Touch upon other thinkers that built upon Cantor, including Frege, Dedekind, etc. Dedekind particularly has a section on systems that

he later uses to define naturals (and is extremely relevant to Cantor's definition).

  - Cover the core issues behind Cantor's original definition, including pyschological components. Use modern crtiqiues in current literature. (See references in: "New Essays On Peirce's Mathematical Philosophy")
  - Poin

- Explain the overreaching implications of set theory, as well as its materialistic issues. Briefly explore the possible resolutions to this (e.g., type theory), and demonstrate that they do not sufficiently resolve the materialistic roots.

- Explore the issue of defining an object, claiming a more general theory is possible. Argue that Cantor and others have used foci

- Touch upon the formalist response to set theory, and rebuttal against the claim that formalism alone can be used for mathematical thought.

- Helpful: look at the references in the metamath book

2.3 The Role of Foci

- Explore the issue of defining an object, claiming a more general theory is possible. Argue that Cantor and others have used foci (in a specialized form, collections)

- Justify the existence of different paradoxes the applicability of non-classical logics (particularly paraconsistent logic)

- Explain the shortcomings of foci alone in explaining both new phenomena and generality

2.4 Early Attempts at Continua: Cantor, Dedekind, and Others

- Revisit Cantor, particularly with Cantor's Theorem. Explore the different models of the real numbers.

- Referencing foci, explain how these models do not completley generalize the reals. Eve mention how category theory does not resolve this either (in both a mathematical and philosophical sense).

- Analyze Zeno's paradoxes and Aristotle's original definition of a continuum. Explain set-theoretic continua do not constitute full fleged continua (and are missing key metaphysical properties)

2.5 Continua: Charles Peirce

- Discuss Peirce's role in mathematics, logic, and ontology.
- Briefly mention Pragmatism and Peirce's search for its proof.
- Explain what Perice thought about Cantor's theory, and introduce his concept of multitudes
  - Elaborate on his initial readings of Cantor's theorem and definition of sets
  - Explore Peirce's shortcomings in his definition of collection
  - Mention some of his misconceptions, e.g., his presumption that the Continuum Hypothesis must hold
- Informally justify the mathematical value of Peirce's writings, arguing that more concrete axioms need to be developed from his ideas. In particular, explain why Perice was closer than Cantor to getting to a definitin of a bona fide continuum.

2.6 Folds: Deleuze

Possible quote to include about Deleuze, "The Fold: Leibniz and the Baroque", page 81 (maybe this is better suited for the introduciton?)

> For Leibniz... bifrucations and divergencies of series are genuine borders between incompossible worlds, such that the monads that exist wholly include the compossible world that moves into existence. For Whitehead (and for many modern philosophers)... bifrucations, divergences, incompossibilites, and discord belong to the same motley world that can no longer be included in expressive units, but only made or undone according to prehensive units and variable configurations or changing captures. In a same chaotic world divergent series are endlessly tracing bifurating paths. It is a "chaosmos" of the type found in Joyce, but also in Maurice Leblanc, Borges, or Gomrowicz.

- Describe Deleuze's work in epistemology and relevance throughout modern philosohpy.
- Connect Deleuze to some concepts in Eastern thought, particularly tying his idea of folds with origami. Also mention how he challenged Leibniz's metaphysics.
- Explain how Deleuze's concept of a fold resolves Peirce's previous inquries into collections.

Chapter 3

Continuum Foci Logic and Theory

3.1 Axioms

- Figure out how to write all of the axioms in Lean. (We will want to transition these to Dedkuti, but we want to leverage lean's large math library, for the time being).
  - Lean is based on the Calculus of Inductive Constructions (see ths link for a relative strength of this theory: `https://mathoverflow.net/questions/69229/proof-strength-of-calculus-of-inductive-constructions`)
- However, we want to generalize this to ANY theory... this first requires defining what every
- I am still pondering on a solution. Here is what I am thinking so far: I think we need to recognize how the continuum is secretely hidden in any theory. It is similar to the incompleteness theorems: maybe we can generate a corresponding theorem or property of continua, but we cannot prove it.
- Then, for a proof of the principle, we go back to CFLT. My hope is that, because CFLT is the complete opposite of purely finitistic systems (combinatorial logic or DFAs), we can prove, once and for all, properties about CFLT, under possible metaphysical assumptions. (Much more background needs to be written before I can discuss this in depth, but for now, I believe I must assume that a continuum exists. That may be the ONLY unproven assumption out of everything else, but beyond that, the axioms are purely structural (and may be unneeded).)

Definition 3.1.1. The language of $\mathbf{Fold_0}$ is .

3.2 Fundamental Properties

- Define and prove properties of (generalized) theories
  - Semantic Lifting Lemma
  - Justify how every possible theory is definable in CFLT
  - Prove the consistency, completeness, and uncomputability of CFLT. In particular, show that CFLT can solve *any* possible generalization to the Halting Problem. Heavily connect to Harvey Friedman's unpublished manuscript on Boolean relation theory and incompleteness: `https://bpb-us-w2.wpmucdn.com/u.osu.edu/dist/1/1952/files/2014/01/0EntireBook061311-wh0yjy.pdf`

s

Theorem 3.2.1. Each system of classical logic in **Fold** is consistent.

3.3 Key Implications

Chapter 4

The Welkin Standard

4.1 Preliminaries

4.1.1 Character Encodings

In a formalist fashion, our BNF variant leaves encodings as an generalized notion. Let Char be an arbitrary, finite set. An **encoding** is a mapping $E : \mathbb{N} \to B$. We denote CHAR = E(CHAR) and CHAR* as the Kleene-closure of CHAR. Moreover there is a distinguished subimage NUMBER $\subset$ $chars$. Character encodings may be given as finite tables. Several major encodings are formally defined in the following sources.

- ASCII []
- UTF-8 []
- UTF-16

4.1.2 EBNF Variant

Our variant of EBNF uses the following notation:

- Rules are optionally surroudned in angular brackets $< term >$;
- ::= denotes rule assignment. Any rule ending in $CHAR*$ or $NUMBER*$ means the rule is a member of $CHAR*$, $NUMBER*$,;
- | denotes alternation;
- term$*$ means 0 or more instances and is a shorthand for the rule $terms$ ::= $termterms|term|\varepsilon$;
- term? means at most one instance and is a shorthand for the rule $terms$ ::= $term|\varepsilon$;

- Single quotes denote a string in CHAR*. In thesse strings are double or single quotes in CHAR*, the notation [‘] or [’] is used, resectively, to avoid confusion.

4.2 The Welkin Language

There are two fundamental variants of Welkin:

- Base Welkin, a language mirroring the core data structures;
- Standard Welkin, which extends Base Welkin with attributes. These attibutes can alter or customize the interpreter’s behavior.

Both can be parsed with LALR parsers and fundamentally have the same semantics.

Syntax

```
terms           ::=  term*
term            ::=  graph | connection | ident | string | num
graph           ::=  <graph ident>?  ‘{’ terms ‘}’
<graph ident>   ::=  ident | string
connection      ::=  term connector term
connector       ::=  edge
                  |  <left arrow>
                  |  <right arrow>
edge            ::=  ‘-’ term ‘-’
<left arrow>    ::=  ‘-’ term ‘->’
<right arrow>   ::=  ‘<-’ term ‘-’
ident           ::=  CHAR*
num             ::=  NUMBER*
string          ::=  [‘‘] CHAR [’’] | [‘] CHAR [’]
```

Semantics

We first form a **Welkin Information Graph,** from which we form the final stored data.

Definition 4.2.1. A **Welkin Information Graph** is a tuple $G = (V, E, A)$, where

- V is a set of **vertices** or **units,**
- $E \subseteq V^2$ is a set of **directed edges,**,
- $A : \mathrm{CHAR} \to V \cup E$ is an **alias function.**

The

Base Welkin is given by the grammar in Figure ??. Note that if NUMBER = $\emptyset$, any instances of the rule num must be excluded from the parser. This grammar defines the text-based interface to the Welkin Information Graph (see Subsection 2.? for more details). Every .welkin file has a corresponding configuration. It may either be put as a comment or (preferred) written in a separate file, which, by default, is called config.welkin. It has the following format:

- Encoding
  - Options: ascii, utf-8, utf-16, other
  - In the case of other: we need to specify how to define an encoding. (We need a light-weight API for implementations)
- Grammar
  - Strength: bounded (only finitely nested graphs with a given nesting limit, no recursion), no-self (arbitrary nesting limit, but no recursion), full (recursion allowed)
  - Customized: use a builtin template or custom welkin file. These can be used to change any part of the grammar, including adding keywords, the symbols used, adding new symbols, etc. Essentially, this will be a way to built new grammars from the original specification; we will need a separate parser for this (i.e., a parser of BNF/Welkin accepted notation).
- (Optional) Language
  - Defaults to English. Can be written in the writer's desired language (as long as it has been configured in Encoding above)

Alternatively, a file with a different name may be used; see the section "Graph Application Behavior" for more details. For subsequent chapter, we refer to the file above as **-config.**

4.3 General Application Behavior

4.3.1 Customization

All Welkin files are infinitely customizable via the welkin config file, which is written in standard welkin.

```
term -> { graph connection ident string}
graph -> {{ident {}}->'{'--term--'}'}
connection -> {term--connector--term}
connector -> {edge arrow}
edge -> '--'
```

```
arrow -> ‘->’
ident -> CHAR*
string -> ‘‘’’ CHAR* ‘‘’’ | ‘’̀ CHAR* ‘’́
```

4.4 Core Algorithms

4.4.1 Graph Encoding

Chapter 5

Expanding Complexity