The Letter before Lambda is Hat: A Reconstruction of Church's Hat Calculus

Akiva Leffert
Cranberry-Melancholy University

SIGBOVIK 2007

Abstract: We present a reconstruction of Alonzo Church's Hat Calculus based on notes discovered under a book shelf. We present evidence that this system was the precursor of the λ-calculus. We then describe the system in full detail. We prove a lack of progress theorem. Finally, we prove an undecidability result by reductio ad absurdem.

Keywords: Computability, Hats

1 Introduction

It is said by those with too much imagination that the λ-calclus sprang fully formed from the head of Alonzo Church and that as he laid the α, β, and η rules on paper, choruses of angels sang Hallelujahs. Those with less imagination and perhaps more wit realize, as Edison said, that genius is hard work and makes one sweaty. Indeed, even the character λ, namesake of said calculus, was not in Church's original work. The usual story is that he borrowed notation from Russell and Whitehead's Principia Mathematica, which used a circumflex over variables to mark abstractions[1]. Church used this early on, for example, writing the identity function as: $\hat{x}.x$. However, due to inferior typesetting technology, the circumflex shifted from above the variable to its left like so: $\hat{}x.x$. This appearance of $\hat{}$ resembles a capital lambda, Λ, which caused some other typesetter, mind no doubt dulled by too much exposure to hot-lead, to use the lower-case λ we know and love. Thus, except for a typographical accident, the λ-calculus would be known as the circumflex-calculus or, more succinctly and colloquially, the hat-calculus.

This sepia-tinged tale of typography makes for a good story to tell little freshlings flush with curiosity about the λ-calculus and the Entscheidungsproblem[3][6].

Harry Bovik, demonstrating his diverse talents, actually made a short film about this[2] which was well received[5]. However, it is wrong in one important detail. Old notes of Church's, recently discovered stuffed under a shelf in the Princeton University Library suggest that this notation was inspired, not by Russell and Whitehead, but by an earlier system which Church sketched out and described in those notes. The account of this discovery can be found in [7]. This system of computation contained more literal hat symbols - see Figure 1. In the remainder of this paper, we describe this system, the Hat Calculus, and sketch a proof of the undecidability of the Down-Feather Problem by reduction from the Halting Problem.

2 Hat Calculus Syntax and Semantics

The complete syntax of the Hat Calculus appears in Figure 2. This system is considerably more, umm, baroque than the λ-calculus. This suggests that Church learned a great deal from the development of this system, abandoning it due to its complexity rather than any inherent computational weakness of the system. Indeed, we later show that this system is Turing-complete.

Definition 2.1 (Up Feather): The symbol is an *up-feather*.

Definition 2.2 (Down Feather): The symbol is a *down-feather*.

Definition 2.3 (Banded Hat): A hat can be combined with a band to create a *banded hat*. For example, a can be combined with a to create . The latter is a banded hat. For the purposes of clarification we may, but probably won't, occasionally refer to hats without bands as *naked hats*.

Definition 2.4 (Banded Feathered Hat): Hats with bands can be combined with feathers to create *banded feathered hats*. For example, is a down feather banded hat as is .

Definition 2.5 (Action Card): All cards in the Hat-Calculus are the same except for the *action cards*: $\boxed{\spadesuit_7}$, the $\boxed{\heartsuit_8}$, the $\boxed{\heartsuit_9}$, $\boxed{\heartsuit_{10}}$.

Definition 2.6 (Inaction Card): A card which is not an action card is an *inaction card*.

Definition 2.7 (n-Carded Banded Hat): Each banded hat is actually an *n-carded banded hat* for some n. An n-carded banded hat is a hat with n cards associated with it. An uncarded hat is just the degenerate case when n is zero. Note that while there are fifty-two (four times thirteen (two times

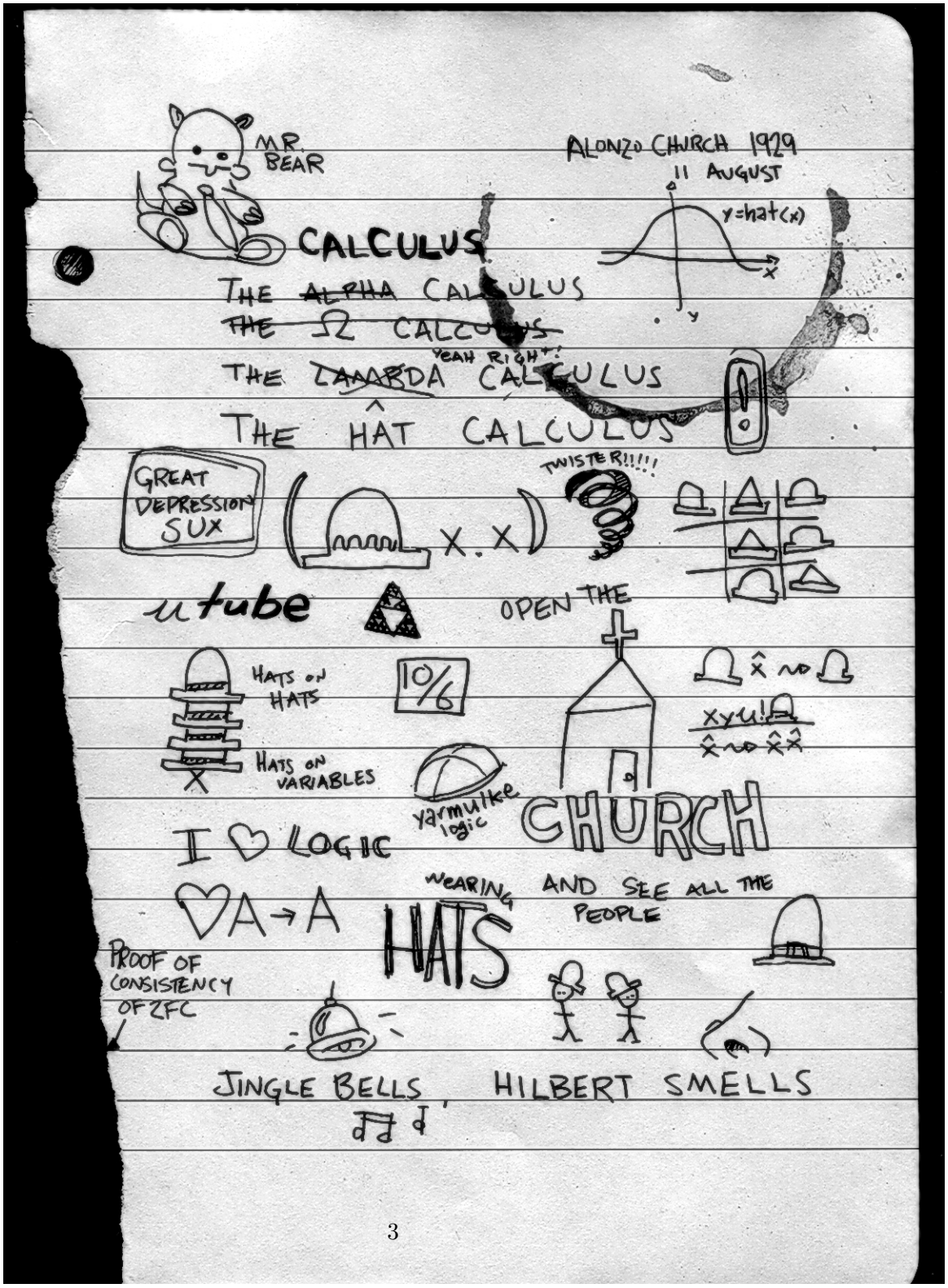


Figure 1: Excerpt From The Lost Notebook of Alonzo Church

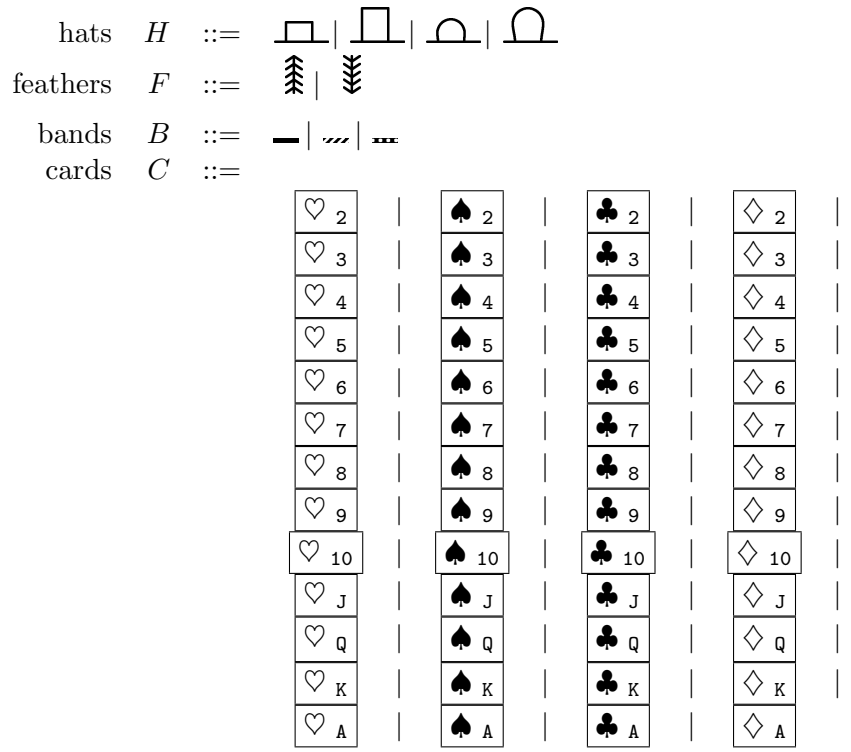
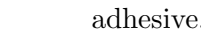
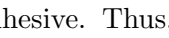


Figure 2: Hat-Calculus Syntax

two times thirteen)) cards, only four of the cards are distinguished by the semantics of the language, the action cards. We suspect that Church was perhaps not at his best when designing this system.

Definition 2.8 (Final State): A hat-calculus expression is considered final if all of the feathers are down feathers.

The process of computation is the process of attaching bands to hats, feathers and cards to banded-hats, and stacking hats on other hats. As in the lamba-calculus, juxtoposition is application, in this case, application of adhesive. Thus,  steps to  . Unfortunately, this convenient combination notation doesn't work as well as we add cards and feathers to hats. Thus, we use the notation $B(H, F, [C_1, \ldots, C_n])$ to represent a completely applied n-carded banded hat. If we wished to combine an n-carded banded hat with another card, say, the King of Hearts, $\boxed{\heartsuit_{\mathtt{K}}}$, we would write this like so: $B(H, F, [C_1, \ldots, C_n])\boxed{\heartsuit_{\mathtt{K}}}$. This would step to $B(H, F, [C_1, \ldots, C_n, \boxed{\heartsuit_{\mathtt{K}}}])$. Note that combining action cards has a different effect discussed later.

Hats can be stacked. If two hats are juxtaposed we combine them into a stack. Stacks of hats can also be stacked in this manner. It is not possible to combine a stack of hats with a single hat in this manner. The application of a band to a stack of hats has the effect of applying that band to all of the hats. If the band of a hat is replaced it loses all of its cards and feathers.

2.1 Action Cards

Combining the $\boxed{\spadesuit_{7}}$ card with a hat or stack of hats causes all of the feathers to flip - i.e. all down feathers become up feathers and vice versa.

Combining the $\boxed{\heartsuit_{8}}$ card with a stack of hats removes all of the hats from the top and bottom until a hat is reached with a down-feather.

Combining the $\boxed{\heartsuit_{9}}$ with a stack of hats duplicates the stack.

The $\boxed{\heartsuit_{10}}$ is the *ungluer*. It pulls all of the feathers, bands, and cards off of a hat. It also creates a new action card at the end of the expression. Which particular action card is chosen is non-deterministic.

3 Results

Definition 3.1 (Stuck): An expression of the Hat-Calculus is *stuck* if it cannot step and is not a final state.

Theorem 3.1 (Lack of Progress): There exists a stuck state. Proof: The expression  cannot step, but is not final.

Definition 3.2 (Down-Feather Problem): The down-feather problem asks whether a given Hat-Calculus expression will reduce to a stuck state.

Theorem 3.2 (Universality): We're pretty sure it's Turing complete. It's got a queue or something.

Theorem 3.3 (Undecidability): The Down-Feather problem is undecidable. Like we said, it's probably Turing complete.

4 Conclusion

The Hat-Calculus was developed by Alonzo Church before the λ-calculus. It is a Turing-complete language of computation with a rather ungainly syntax. Actually, it's unarguably nonsense[4]. Fortunately, Church later developed the λ-calculus, which isn't crap (we hope).

References

[1] Henk Barendregt. The impact of the lambda calculus on logic and computer science. *Bulletin of Symbolic Logic*, 3(3):181–215, 1997.

[2] Harry Bovik. Lambda-calculus: The feature film extravangza. Feature Film.

[3] Alonzo Church. A note on the Entscheidungsproblem. *Journal of Symbolic Logic*, 1:40–41, 1936.

[4] Jean-Yves Girard. Locus solum: From the rules of logic to the logic of rules. *Mathematical. Structures in Comp. Sci.*, 11(3):301–506, 2001.

[5] Fred Hacker. Harry Bovik shouldn't be let near a camera. Letter to the Editor.

[6] Richard Karp. The Entscheidungsproblem is probably NP-complete. Private communication in an elevator.

[7] William Lovas and Tom Murphy VII. The hidden finds of janitorial work. *Proceedings of Found Stuff Symposium*, 9(1):1299–1578, 2005.