Review of Using Automata to Prove Mathematical Theorems

Felix Karg

12. Januar 2018

Introduction

Sound plan. I'm looking forward to it. Maybe an aactual short introduction to Lagrange's theorem for binary squares would have been better, but I do not know how exactly that would be feasible.

Lagrange's Theorem for Binary Squares

Sloppy english, as every now and then. A single read-over would certainly have found most of the ... grammatically weird sentences, assumably being leftovers from unfinished rewritings. I would have wanted to have additional information on Lagrange's Theorem for Binary Squares, especially as to why it is particularly interisting that this case is true, and not for it to appear simply as something someone has recently found just another way of proving this. Why is it this relevant, what are maybe some of the implications? Maybe this is just a follow-up, but for me it's not exactly surprising that it holds, as is. Either way, I would have wanted to have more background-information on this Lemma, as to why it actually is relevant or supposed to be surprising. I can understand the goal of wanting to provide a general example of how to utilize automaton for proving number-theoretic statements, but it's not the goal to prove something in a way just for it to be proving in this way, as in, it's not supposed to end in itself (dt: dem Selbstzweck dienen), right?

Using automata for proving: General approach

In the beginning, it got explained in quite a detailed manner, still, as of now I do not think to understand exactly what the approach is, or how I would/should approach it. Especially the following: The Language of the automaton is what? The functions

$r : M \mapsto \Sigma^*$? How will r be represented then? Exactly what is the goal? And why whyis it useful for us to show that A is universal, as you put it? What does it help us to know that our automaton will just accept every input? Would that be even valid versions of r? Why? Additionally, but this is only something minor, why does $L(A) \subseteq r(M)$ hold by definition?

It would be nice to know not only that you intend to contact the authors, but that you actually did it already and are waiting wor an answer. Otherwise you might still not have contacted them when the talk starts and that would be a pity, since the initial process of contacting them would not take more than 30minutes at most, I'd assume.

After reading the specialization much of my confusion remains. I understand the general procedure, here, but we aren't even creating an automaton that's going to accept all the inputs later on, ... In the talk this can hopefully be explained better. In step 3, we create an automaton accepting all integers with length ≥ 18 is that not already using the theorem, not proving it?

Construction of Automata for the Main Lemma

I think it's a good approach to the talk.

Folded Representation of Integers

Maybe make more explicitly clear that we need this for upholding our constraints for the automaton as input.

Adding up

I can nuderstand this quite well, but I think some people might be confused for a short time regarding us constructing two automatons now, up until it is explicitly statet. Though I would probably leave it at that. Other than that, it might be useful to have more of a visualisation for the adding and the ruqired transition function etc. Maybe show an example-computation, and an example-Automaton (even a part of it) might be really helping in understanding.

Computing the Proof Using The Ultimate Framework

Could you actually show the Code, and (if available) the verification?

General discussion

You could get into Computer-Assisted Theorem-Provers here, explain what they do (did they use one here?) and roughly how they work (if you have the time, otherwise just explain how they did it).

Discussion of Didactic Questions

I've read this part pretty much in the beginning, and other than that, my comments are on the specific locations, if there are some.