Re: Cantorian pseudomathematics

Source: http://sci.tech−archive.net/Archive/sci.math/2006−02/msg01305.html

* *From*: Tony Orlow <aeo6@xxxxxxxxxxx>
* *Date*: Wed, 8 Feb 2006 11:09:31 −0500

david petry said:

> Keith Ramsay wrote:
>
> > david petry wrote:
> >
> > |But we can tell, just by looking at X, whether it is a constructively
> > |meaningful statement. Sometimes it is not so clear when the statement
> > |is made in informal natural language, though.
> >
> > I think that when you "just look at" a statement and decide whether it is "constructively meaningful" you do it in a naive way. Going with your gut on these things is a bad idea. You need to spend more time dealing with criteria that are explicit enough that someone can tell whether a statement qualifies or not without being you.
>
> I believe I have stated very precisely what the criteria are. A statement is meaningful iff it makes predictions about the results of computational experiments. A computational experiment can be put in the form "Turing machine T, with input M, will halt within N steps".
>
> I am also claiming that we can build a new formalism for mathematics such that the axioms themselves are computationally meaningful, and such that the laws of logic we use preserve computational meaning. Then every grammatically valid sentence in our formalism will be meaningful.
>
> In general, it is not so easy to take a statement written in the formalism of ZFC (for example), and determine whether it has a meaningful counterpart.

David, it sounds like you want a statement to be a symbolic string that is manipulable through formal rules so that it produces a quantity. When you speak of a Turing machine, that reaffirms that idea. Now, you also seem to be talking about defining a grammar that sets down rules for such statements and their combinations. I think this sounds like a good idea. So, what primitive notions do you need to add to this grammar for that purpose? Have you identified specific problematic constructions that have no computational meaning?

> Errett Bishop's constructive analysis textbook in at least one edition contains a mistake regarding AC. He made a mistake which led him to think it was true. So if there is some way that one can just magically intuit its unacceptability from a constructive point of view by just looking at it, for the first time, it is at least a method that had escaped his attention.

For constructivism, lists are a much more natural structure than sets, and the axiom of choice for lists is trivially true. Indeed, it can be difficult to take concepts from set theory and figure out how they fit into the constructivist view.

Just a comment: I've always felt that Bishop tried too hard to make his constructivism resemble classical mathematics.

> |Proving theorems like AC−>X is not doing constructive mathematics, even
> |in a "broad" sense.
>
> So now you're ready to explain what core, essential aspect of the concept of "constructivity" it violates?

AC is not part of constructive mathematics.

> No. Of
> course not. You derive too much enjoyment from glibly tossing off such claims for stopping to examine the principles in question to be attractive to you.

Glib nonsense.

> > Constructivity is typically explained in terms of the
> > meaning assigned to existential quantifiers. In particular,
> > each existence claim should be accompanied by a
> > "construction", where it is often left rather vague as
> > to what should be counted as a "construction".

> As I have often pointed out, what I'm advocating is not exactly conventional constructivism. There is nothing vague at all about existence in what I am doing.

Well, specifically, you wish an element to be the result of a symbolic manipulation according to specific rules which you consider "computational". Are there any systems of symbolic manipulation which you don't consider computational, or any computations which don't involve symbolic tranformations?

> > One case, however, is more crucial than most people who've
> > heard of constructivism realize, and that's the case of
> > integers (and consequently other finite combinatorial
> > constructions). A constructive proof that an integer exists
> > must (by the nature of the concept of "constructive") give
> > a way of computing it (in decimal form, say).

> But that is already slightly vague. What I am advocating is that we would have to derive from the proof an upper bound on how much computation we need to do in order to compute the integer, in order to say that we have a way to compute it.

What are you computing the integer from? A predecessor? By "how much computation", what do you mean, exactly. How do you measure computation in your scheme. Does that give a size to the set under consideration?

> > A second case
> > that is less crucial but generally fits people's intuitions
> > about what "constructive" should mean is that a construction
> > of a set should give us a way to express it as {x:P(x)} for
> > some explicit property P.

> As I already pointed out, using lists would be preferable to using sets

here.

A list being just a set with order, right?

> So for an axiom system we have two natural criteria for constructivity, known as the numeric existence property and the set existence property. It has the numeric existence property if there is a procedure for converting a proof of a statement of the form (En)Q(n) where n ranges over the integers into a proof of Q(n) for a specific n given in decimal form. It has the set existence property if there is a procedure for converting a proof of a statement of the form (ES)Q(S) where S ranges over sets (of some kind) into a proof of a statement of the form Q({x:R(x)}) for some predicate R.
>
> There are two fundamental axioms famous for creating problems with the numeric and set existence principles. Most mathematically trained people are aware of proofs using the axiom of choice of the existence of things like a Vitali set (with one representative from each member of R/Q) where one doesn't expect to be able to give an explicit example. This is associated with ZFC's failure of the set existence principle. Now a lot of people don't realize this can be patched over by going to ZF+V=L, where V=L implies AC. But set theorists usually seem to prefer axioms that contradict V=L so it doesn't make much difference. The second and more serious issue is with the way the law of excluded middle destroys the numeric existence principle.
>
> Not every constructivist is "happy" with every formal system satisfying the numeric and set existence properties. I don't know of any additional requirement, however, for which there is a consensus that it is a necessity for a system to be a "constructive" system, and its proofs "constructive" proofs. Among people who want proofs to be constructive, I suppose there's often interest in their being predicative as well. It strikes me as somewhat arbitrary and severe to say that this is some kind of absolute requirement.

I believe that you are approaching constructivism from the point of view that Cantorian set theory is "the big picture", and constructive mathematics is merely a mildly interesting small picture which must fit

> somewhere within the big picture. That, I suspect, is the obstacle to
> communication we seem to be having.

Actually, I think Keith is addressing a core issue in this area of foundations. Set theory has this primitive "is an element", but it seems to me that one can start with "has a property", and define "is an element" and "is equal" based on the concept of properties. If one defines sets in terms of objects that share a property, and then goes on to define the properties in terms of a grammar with some primitive types as values, then perhaps that grammatical definition would satisfy what you are trying to achieve. This is what I am currently considering.

>> Then there's the issue of "computational meaning".
>>
>> Some famous judge said of pornography that he couldn't define it precisely but could recognize it when he saw it. If one wants to be able to say something more satisfying than that about mathematics that is "obscenely" lacking in computational meaning, then one has further work to do.

> Whatever.

>> You seem not to believe me when I say it, but there's a simple, straightforward way of applying the "mathematical theories must make predictions" rule that tells us to dump the same two axioms, and nothing else. Take "computational predictions" to mean only the most obvious kind of thing, namely, statements of the form (n)P(n) where n ranges over the integers and P is a primitive recursive predicate. Then say that an axiom system should be no more complicated than necessary to entail all conclusions of that form that it does.

> So you want to eliminate statements like "for all n, A(n) exists", where 'A' is the Ackerman function. Why would you do such an arbitrary thing?

>> By the work of Goedel on AC, we can show that removing it has no effect on which arithmetic statements we can prove,

> > let alone the kind of computational prediction I'm
> > considering here. Likewise there are some basic results on
> > the law of excluded middle that show it also doesn't give
> > us any more "predictions" than we had to begin with. I don't
> > know whether it counts as a coincidence or not, but these
> > are the same two axioms as were nonconstructive in the
> > obvious way.
>
> Again, you're starting with classical set theory, and working "down" to
> constructivism. That's bound to lead to confusion.

It seems like Keith is pointing out that AC and LEM are not necessary, which should be good news. Of course, he is putting things in terms of standard theory, but I think he's doing a good job of describing what differences there are in basic assumptions between the two schools of thought, no?

> > The rest of the axioms in ZFC do give new predictions. The
> > axiom of replacement is often written in a form implying
> > the law of excluded middle, but it can be written not to.
> > Without AC or LEM this leaves us with a system called IZF
> > that satisfies the numeric and set existence properties.
> > I guess it's not very well liked for practical use, but in
> > a theoretical way it nicely matches ZFC. One can prove the
> > existence of the same recursive functions in each for
> > instance.
> >
> > IZF can prove the consistency of IZF with the axiom of
> > replacement removed, which IZF with the axiom of replacement
> > removed can't do (by Goedel's second incompleteness theorem).
> > Since a consistency statement is equivalent to one of the
> > form (n)P(n) this counts as some kind of additional
> > prediction.
> >
> > IZF with the axiom of replacment removed can prove the
> > consistency of IZF with the power set axiom and the axiom
> > of replacement both removed.
> >
> > IZF with both the power set axiom and the axiom of replacement
> > removed can prove the consistency of IZF with the axiom of
> > infinity removed. If the axiom of infinity is replaced with
> > its negation, we get a system essentially the same as elementary
> > arithmetic. There's a model of it consisting only of the sets
> > that can be built up in finitely many steps from the empty set.
> >
> > The only way to get the full "computational content" is to

> > keep all the axioms in IZF. Presumably there's some way to simplify the system by replacing axioms, but none of the axioms can be just tossed out.
>
> I'm surprised the mob hasn't bumped you off yet. Clearly, you know too much.

Keith apparently knows a lot. I found this explanation very interesting ans sensible, though I am not sure about proofs of consistency. Thanks, Keith.

> I've pointed this out to you before that there is something magical (deceptive) about consistency proofs. If you believe that the theorems of some formalism are in fact truths, then you must believe (at a minimum) that the formalism is consistent. So if you use a strong formal system to "prove" the consistency of a weaker system, you are already assuming that the stronger system, and hence the weaker system, is consistent. So you are "proving" nothing, except to true believers who already believe the conclusion.
>
> > The criteria I described above give us a principled way to designate certain kinds of mathematics as nonconstructive.
> >
> > Is there a principled way to designate some of the mathematics that counts as constructive by those criteria, and say that, nevertheless, it lacks computational meaning?
>
> Again, you are implicitly making the claim that in order to talk about constructive mathematics, we must be able to talk about classical mathematics. There's something fundamentally dishonest about that; you are stacking the deck in your favor.
>
> > As far as I can tell, you just go on your own gut reaction, without any principle that someone not sharing your intuitions could apply in a consistent way. It's hard to see how one could in a principled way dispose of all the "Cantorian" mathematics you have such a problem with
>
> Here's what I am claiming: we can build up constructive mathematics independently of classical mathematics. Then, we can notice that classical mathematics implies the existence of a universe much "bigger"

than the universe of constructive mathematics, and hence there is
necessarily something about classical mathematics that is fantasy,
where "fantasy" is defined to be anything beyond the world we can
observe (i.e. the constructive universe).

There is no unprincipled gut reaction involved.

Hmmmm..... How do you know that what you view as non−constructive has just not had its construction discovered/defined yet? How will you know when all possible constructions have been exhausted? Doesn't that depend largely on how you define the grammar for the statements you allow? In a sense, this seems to be pushing the problem back a level, but not necessarily solving the ultimate problem of consistency, but I am no expert.

|I know we've discussed this a few times before. The result that is
not
|disputed by either constructivists or classical mathematicians is that
|if we are given a well defined list of well defined real numbers (so
|that every digit of every number can be computed), then the diagonal
|method gives us a new number not on that list. The classical
|mathematicians claim, essentially, that the argument is still valid
|when the list and the numbers on the list may not be well defined.

You're equivocating between computable, which is a property of certain objects, and "definedness" which is really a property of the description of the object that needs to be satisfied before one can ask whether the object described by the description has some property or not.

You describe constructivists as if they automatically assume that reals are computable. They don't. The argument makes no use of any such assumption.

Constructively, it makes no sense to say something exists unless it can be computed. At least, that's what *this* constructivist says.

And by "computed", you mean that it can be produced as a state reached by Turing machine processing a symbolic input string, right? But then, doesn't its computability depend on the prior definition of a state in the Turing machine which represents the constructed value? I think what you need to do, to define what you mean by "computationally meaningful", is design your Turing machine and thus the proposed grammar that you envision as restricting the kinds of statements that are allowed to define set membership. Does that make any sense?

--
Smiles,

Tony
.