TESTING THE CONSISTENCY OF MATHEMATICS

by
Harvey M. Friedman*
Distinguished University Professor of Mathematics, Philosophy, and Computer Science Emeritus
Ohio State University
July 23, 2014
EXTENDED ABSTRACT

*This research was partially supported by the John Templeton Foundation grant ID #36297. The opinions expressed here are those of the author and do not necessarily reflect the views of the John Templeton Foundation.

In [Fr14a] and [Fr14b], we focused on independent statements that are natural, simple, and elegant. Here we fine tune some of these statements into a form that is subject to practical computer investigations that can arguably confirm or definitely refute the consistency of mathematics, as formalized by ZFC, as well as stronger systems involving large cardinal hypotheses. This is a reworking of our initial efforts along these lines from [Fr14a]. Most, but not all, of the relevant definitions from [Fr14a] and [Fr14b] are repeated here.

1. SHAPE OF ALGORITHM FOR SRP

We begin by presenting the general shape of the infinite nondeterministic algorithms associated with SRP.

DEFINITION 1.1. Q is the set of all rationals. Z^+ is the set of all positive integers. N is the set of all nonnegative integers. $Q[0,n] = Q \cap [0,n]$. $x,y \in Q^k$ are order equivalent if and only if for all $1 \leq i,j \leq k$, $x_i < x_j \leftrightarrow y_i < y_j$. $A \subseteq Q[0,n]^k$ is order invariant if and only if for all order equivalent $x,y \in Q[0,n]^k$, $x \in A \leftrightarrow y \in A$. A graph on $Q[0,n]^k$ is a pair $(Q[0,n]^k,E)$, where $E \subseteq Q[0,n]^{2k}$ is the adjacency relation, which is required to be an irreflexive and symmetric binary relation on $Q[0,n]$. An order invariant graph on $Q[0,n]^k$ is a graph on $Q[0,n]^k$ whose adjacency relation is order invariant as a subset of $Q[0,n]^k$. X^* is the set of all nonempty finite sequences from X. If the length of $\beta \in X^*$ is r, then β has r positions numbered 1 through r, with terms $\beta_1,...,\beta_r \in X$.

The parameters of the algorithm consist of

i. $k,n \in Z^+$ and an order invariant graph G on $Q[0,n]^k$.
ii. $\alpha \in Q[0,n]^{k*}$, where $\max(\alpha) = n$.
iii. $f:Q[0,n]^k \to Q[0,n]^{k*}$.
iv. $g:Q[0,n]^{k*} \to Q]0,n]^{k*}$.

NDA(G,α,f,g) runs as follows. (NDA abbreviates "nondeterministic algorithm").

We initialize NDA(G,α,f,g) with α and a pointer to α_1. At any stage, we will have $\beta \in Q[0,n]^{k*}$ and a pointer to some β_i, with some of the β_j's marked (for replacement). At initialization, all of the terms of α are marked. The algorithm proceeds as follows.

1. Replace β_i by some $x \in Q[0,n]^k$, not adjacent to x in G, such that $\max(x) \leq \max(\beta_i)$. Leave x unmarked.
2. Put f(x) at the end of β, and leave all of the terms of f(x) unmarked.
3. Let β' be the resulting element of $Q[0,n]^{k*}$ after instructions 1,2. Place a copy of $g(\beta')$ at the end of β', with all of the terms marked.
4. Advance the pointer to the next term (going forward) that is marked. Go to instruction 1.

It is easy to see that instruction 4 can always be executed, and the pointer always points to a term that is marked. Thus unmarked terms are never modified, and generally speaking, some marked terms are modified, and some marked terms are not modified.

The goal is for the set of all unmarked terms to form a clique in G. If the algorithm is run for infinitely steps (infinitely many applications of 4), then all of the resulting terms will of course be unmarked, and we are simply requiring that the set of all terms form a clique in G. On the other hand, if we are merely running the algorithm for finitely many steps, then some terms will be marked and others unmarked. In fact, there generally will be unmarked terms that are beyond the position of the pointer.

PROP(G,α,f,g). It is possible to execute NDA(G,α,f,g) for infinitely many steps, resulting in an infinite list from $Q[0,n]^k$, such that any two distinct terms are adjacent in G.

PROP(G,α,f,g;t). It is possible to execute NDA(G,α,f,g) with t executions of instruction 4, resulting in a list from $Q[0,n]^k$, such that any two distinct terms are adjacent in G.

THEOREM 1. PROP(G,α,f,g) holds if and only if for all t, PROP(G,α,f,g;t) holds.

As expected, the status of PROP(G,α,f,g) depends on G,α,f,g. In the next section, we place a sufficient condition on these four parameters.

2. ALGORITHMS FOR SRP

DEFINITION 1. We use n,m,r,i,j,k, exclusively for positive integers unless indicated otherwise. For $x \in Q^k$, let $\omega(x)$ consist of all $y \in Q^k$ such that for some $i \in Z+$ and $j \in N$, y results from adding j to all coordinates of x that are $\geq$ i. Let $x,y \in Q^k$. $x \equiv y$ if and only if
i. x,y are order equivalent.
ii. if we remove the x_i from x with every $x_j \geq x_i$ lying in Z^+, and remove the y_i from y with every $y_j \geq y_i$ lying in Z^+, then the remnants are identical.

PROPOSITION 2.1. Let G be an order invariant graph on $Q[0,n]^k$ and $\alpha \in Q[0,n]^{k*}$. Assume that for $x \in Q[0,n]^k$, f(x) lists exactly
i. All elements of $\omega(x)$ lying in $Q[0,n]^k$.
ii. All $y \equiv x$ lying in $Q[0,n]^k$.
Let $g(\beta)$, $\beta \in Q[0,n]^{k*}$, list all $y \in Q[0,n]^k$ such that every coordinate of y is a coordinate of some $\beta_i \in Q[0,n]^k$.
Then PROP(G,α,f,g) holds.

PROPOSITION 2.2. Let G,α,f,g be as assumed in Proposition 2.1. For all t, PROP(G,α,f,g;t) holds.

Note that Proposition 2.2 is explicitly Π^0_2. By a direct argument, or by elimination of quantifiers and associated decision procedures, we see that it is Π^0_1, and can be put in explicitly Π^0_1 form by bounding the denominators used.

THEOREM 2.3. Proposition 2.1 is provably equivalent to the consistency of SRP over WKL_0. Proposition 2.2 is provably equivalent to the consistency of SRP over EFA.

The initial idea for confirmation of Con(SRP) is to find a nondeterministic path for PROP(G,α,f,g;t) by an exhaustive search algorithm. According to Theorem 2.3, if the

exhaustive search algorithm fails, then we have obtained an inconsistency in SRP. On the other hand, if the exhaustive search algorithm succeeds, then we have arguably confirmed the consistency of SRP.

Realizing this plan in the practical sense takes a considerable amount of thought.

3. REPLACEMENT STEP

The replacement instruction 1 of the algorithm is the only nondeterministic instruction. Hence in our exhaustive search algorithm, we are going to have to branch on instruction 1. Therefore, we need to minimize the nondeterminism in instruction 1 without affecting the algorithm. So we now require that the replacement instruction 1 be conducted as follows.

Let the state of the algorithm be given by the list of k-tuples $x_1,...,x_r,...,x_s \in Q[0,n]^k$, with the pointer at x_r.

In executing instruction 1, we can of course merely replace x_r by x_r. This may or may not lead to an immediate violation of the clique requirement. However, suppose we wish - or are forced - to replace x_r by some $y \neq x_r$, $y \in Q[0,n]^k$. Here is what we are now required to do.

1. Let A be the least set containing all coordinates of the $x_1,...,x_r$ and $0,...,n$, such that $p \in A \wedge p+1 \leq n \rightarrow p+1 \in A$.
2. Let $A = \{p_1 < ... < p_t\}$. Choose y_1 to be either in A or else halfway between two adjacent elements of A. Let $A_1 = A \cup \{y_1+j: 0 \leq j \leq n \wedge y_1+j \leq n\}$.
3. Choose y_2 in the same way relative to A_1, forming A_2.
4. Continue in this way until we have $y = (y_1,...,y_k) \in Q[0,n]^k$.

THEOREM 3.1. This requirement on the replacement instruction 1 does not affect the operation of the nondeterministic algorithm NDA(G,α,f,g), step by step.

The idea is that any execution of the algorithm is step by step isomorphic to an execution of the algorithm as so modified, in the appropriate sense.

4. PRACTICAL ALGORITHMS

The modification of the replacement step in section 3 definitely takes care of some computational explosion. However, steps 2,3 generally create rather large numbers of k-tuples. We need to greatly reduce this proliferation. It appears that reducing the marked terms is most critical as these are subject to replacement (instruction 1).

We use a schedule in advance of just how many of the terms generated by g at a given stage are actually going to be entered. The k-tuples to be entered are then chosen randomly (perhaps with biases) using a fixed pseudo random process. Obviously this affects the operation of the nondeterministic algorithm.

We now address the choice of the order invariant graph on $Q[0,n]^k$. Note that G can be uniquely given by a list γ of k-tuples from {1,...,2k} whose

*) set of terms forms an initial segment of 1,...,2k, where the first half is lexicographically earlier than the second half.

Two vertices x,y are taken to be adjacent if and only if (x,y) or (y,x) is order equivalent to some 2k-tuple on γ.

γ should be generated by a pseudo random process relative to the size of γ (perhaps with biases). It is not clear what the size should be. It would seem that the size shouldn't be too big or too small. My general feeling is that just about any size not too big or small will result in suitably complicated usable graphs for our purposes - provided k is fairly small.

α should also be generated by a pseudo random process relative to the chosen size of α (perhaps with biases). Because of the preferred status of 0,1,...,n, we think that some terms of α should have coordinates solely from 0,...,n. Others should be random (with minimized denominators), with a bias toward 0,...,n.

5. INITIAL EXPERIMENT

Consider the following initial experimental suggestion. To begin with we suggest setting k = 4 and n = 8. Although this is not at all crucial for initial experimentation, we believe, but are not claiming, that Propositions 2.1 and 2.2 go well beyond ZFC already for k = 4 and n = 8.

Set γ to be a pseudo randomly chosen list from *) of section 4 of length the square root of the size of *). This defines the order invariant graph G on $Q[0,n]^k$ as in the paragraph after *).

Set α = (1,2,3,4), (8,7,6,5), (9/2,0,15/2,5).

Use the full f throughout, as the values of f are not going to be marked for replacement.

On every application of g, pick just 3 terms chosen by a pseudo random process, with some bias towards coordinates from 0,...,8. Set t = 12.

It is plausible but by no means certain that the corresponding exhaustive search tree for this is tractable. Recall that we are requiring here that the first 12 marked terms be replaced by terms forming a clique in G. Obviously, when the clique condition is violated, the tree gets aborted. There should be lots of aborted vertices in the search tree, greatly controlling its size.

If the search tree actually turns out to be rather limited because we are drowning in violations of the clique requirement, then there are many ways to go for a larger exhaustive search tree. The most obvious is to simply raise t as much as possible, while still conforming to the above plan.

The idea is to have a highly intense nondeterministic process which challenges but does not go beyond the practical. We believe that there are many interesting implementation issues that greatly effect what is practical here.

6. REFUTATION OR CONFIRMATION?

If exhaustive search does not produce a path through the nondeterministic process, then we have an inconsistency in SRP - and in fact fragments of SRP. On the other hand, if we find a certificate, then we have arguably confirmed the consistency of SRP. At the very least, this represents a kind of evidence for the consistency of at least fragments of SRP going beyond ZFC that is *very different* from any of the usual cited evidence - e.g., intellectual coherence, experience, and so forth.

Specifically, note that we humans believe in advance that these exhaustive searches must succeed - at least if we accept the widely believed consistency of SRP. But without using consistency of SRP (or a substantial fragment thereof going well beyond ZFC), it does not seem reasonable that we would know how to prove in ZFC that there is such a certificate without using the computer. Furthermore, even if we believe Con(SRP) and consequently that there is a certificate, this does not seem to allow us to construct an actual certificate without using the computer.

See [Fr14a] for an earlier version of this material, and some related discussion raising some additional points.

Such investigations can also be applied starting with the much stronger statement in [Fr14b] corresponding to the HUGE cardinal hierarchy.

REFERENCES

[Fr14a] Invariant Maximality and Incompleteness, 37 pages, April 30, 2014, to appear. Supersedes January 8, 2014 version. https://u.osu.edu/friedman.8/foundational-adventures/downloadable-manuscripts/

[Fr14b]. Order Invariant Relations and Incompleteness, 6 pages, July 19, 2014. Extended abstract. This has the currently preferred Pi01 incompleteness from SRP and from HUGE, and finite incompleteness from SRP. Too early to remove earlier versions. https://u.osu.edu/friedman.8/foundational-adventures/downloadable-manuscripts/