Challenging Epistemology: Interactive Proofs and Zero Knowledge

Justin Bledin*

September 18, 2008

Abstract: This essay explores what, if anything, research on interactive zero knowledge proofs has to teach philosophers about the epistemology of mathematics and theoretical computer science. Though such proof systems initially appear 'revolutionary' and are a nonstandard conception of 'proof,' I will argue that they do not have much philosophical import. Possible lessons from this work for the epistemology of mathematics—our models of mathematical proof should incorporate interaction, our theories of mathematical evidence must account for probabilistic evidence, our valuation of a mathematical proof should focus solely on its persuasive power—are either misguided or old hat. And while the differences between interactive and mathematical proofs suggest a need to develop a separate epistemology of theoretical computer science (or at least complexity theory) that differs from our theory of mathematical knowledge, a casual look at the actual practice of complexity theory indicates that such a distinct epistemology might not be necessary.

Dimitri the Cat is showing Fievel the Mouse his new maze.[1] The maze has two openings, one on the North side and one on the South side.

Dimitri: 'Fievel, my little mouse, I bet you cannot traverse my maze. If you can, there is much cheese for you. If not, a delectable mouse for me!'

Fievel: 'But Dimitri, you are a cunning cat. How do I know there is a way through your maze at all? Prove this to me and I will accept your challenge.'

Dimitri: 'You rascal! To give you a proof, I would have to show you a path through the maze. Then what fun would we have?'

Fievel: 'Not necessarily. You can just drop me at some random spot inside the maze. I will then randomly pick either the North or South opening and

*The Group in Logic and the Methodology of Science. University of California, Berkeley. jbledin@berkeley.edu.

[1]This example is adapted from one in Goldreich [2008a], p. 372, involving Odysseus and the Labyrinth of Aeaea.

you will randomly lead me there. If we repeat this enough times, I'll become convinced that a path between the two openings exists.'

Dimitri: 'Hmmm, and you still won't know a path between the openings, just a collection of random walks to them. Brilliant Fievel! Let's get started then.'

1 Introduction

This essay is a case study involving two related gems of theoretical computer science: *interactive proofs* and *zero knowledge* protocols. Introduced by Goldwasser, Micali and Rackoff [1985], interactive proofs are dynamic communications between a prover and verifier where a sequence of messages is exchanged back and forth as the prover attempts to convince the verifier of the truth of this or that mathematical fact. In the limiting case where the prover provides the verifier with no additional knowledge beyond the truth of the considered claim, the proof is zero knowledge. The very notion of an 'interactive zero knowledge proof' may initially seem paradoxical: how can one prove anything without yielding such additional knowledge? But imagine a situation where you (the prover) wish to convince a friend (the verifier) that you are *not* blind. You place a red and a yellow ball in a box and ask your friend to blindfold herself. You then tell your friend to take the balls, shuffle them behind her back, and show you one of the balls while you stand in front of her. If the ball is red, you shout out 'purple.' If the ball is yellow, you shout out 'green.' You then tell your friend to shuffle the balls behind her back again, but keeping track of them this time so that she remembers which ball was previously shown, and show you another one. Again, you shout out 'purple' if you see the red ball and 'green' otherwise. If you repeat this enough times, your friend will become convinced that you are able to distinguish between the red and yellow balls by sight, though she still will not know the respective colors of each of the balls in her hands.

The opening dialogue between Dimitri and Fievel suggests another informal example of an interactive proof which I will discuss in more detail in the next section. In fact, given plausible assumptions, zero knowledge interactive proofs exist for a wide range of mathematical claims concerning graph colorability, the satisfiability of a Boolean formula, and so on. This applicability has excited some theoretical computer scientists who herald zero knowledge interactive proofs as *revolutions* in our understanding of proof. In a [2001] survey paper, for example, Oded Goldreich and Avi Wigderson boldly write: "Combining randomness and interaction lead [theoretical computer science] to create and successfully investigate fascinating concepts such as interactive proofs, zero-knowledge proofs and Probabilistic Checkable Proofs (PCP). Each of these concepts introduces a deep and fruitful revolution in the understanding of the notion of proof, one of the most fundamental notions of civilization." A possible instance of the 'extroversion' (to use C. Papadimitriou's term) of complexity theory,[2] their claim is apparently that

[2]In calling complexity theory 'extroverted,' Papadimitriou refers to the dissemination

interactive proofs, zero knowledge proofs, and probabilistic checkable proofs[3] have something deep to teach philosophers, especially epistemologists, about 'proof.' In particular, if the 'proof' in 'interactive zero knowledge proof' is relevant to the mathematical conception of proof, then this computer science research presumably has something deep to teach philosophers about the epistemology of mathematics.

I am not so sure. In the second part of this essay, I will critically examine the claim that interactive zero knowledge proofs can significantly contribute to our philosophical understanding of proofs and evidence in mathematics. Doing so will raise a host of interesting questions: In what sense exactly are these protocols 'proofs'? Are interactive proofs mathematical proofs and are instances of such common decision problems as Vertex Cover or the Traveling Salesman Problem even pieces of genuine mathematics? If interactive proofs are not mathematical proofs, do any of their features inform the epistemology of mathematics? For example, is there a place for probabilistic methods within our theories of mathematical evidence? And is a proof that convinces us of a particular fact without providing any explanation or leading to any understanding of why something is the case any less valuable than a more explanatory proof? I fear that my answers to these questions might disappoint. For though I think the concepts of interactive proofs and zero knowledge are fascinating and ingenious, I will argue that they do not cut much ice in the philosophy of mathematics.[4] That said, I do think this research in complexity theory suggests the potential for developing an epistemology of theoretical computer science that is distinct from the mathematical case. In the final section of this essay, I will explore how the comparison of interactive proofs with mathematical proofs points to some of the salient ways in which the concepts, perspectives, and epistemic principles adopted by some theoretical computer scientists may differ from those found in the mathematical community.

More generally, this project is situated at the junction of two youthful philosophical currents. Firstly, a pioneering group of philosophers of science have recently turned their attention to computer science in earnest, recognizing the Philosophy of Computer Science (PCS) as a new branch of philosophical inquiry. Current research topics in this field include the relationship between mathematics and computer science, abstraction in computer sci-

---

of its thirty-years worth of ideas and inventions across other disciplines. For example: the widespread use of NP-completeness, work on biological algorithms and the price of anarchy, and the testing of theoretical physics furnished by scientists' attempts to build a quantum computer. But though interactive zero knowledge proofs fall under the scope of complexity theory, 'complexity theory' and 'theoretical computer science' should not be conflated. In addition to computational complexity, theoretical computer science includes such branches as automata theory, type theory, formal semantics for programming languages, etc.

[3]Probabilistic checkable proofs, where the verifier must only read a few random bits in the proofs, are an interesting topic in their own right but I will not discuss them here. For a nice survey, see Goldreich [2008b], §3.

[4]In the interest of full disclosure, I should mention that my own views have changed significantly since the early stages of this project. Initially, I thought there were important lessons to be gleaned from interactive zero knowledge proofs for the epistemology of mathematics. But as evinced in the current version of this essay, I have since adopted a more critical stance. I am especially grateful to one of the anonymous referees for motivating this change in view.

ence, the use of mechanized 'proof assistants' in justificatory efforts, and the position of computer science among the empirical sciences. Colburn [2000], Floridi [2003], and special issues of *The Monist*, *Minds and Machines*, and this journal on PCS are good examples.[5] Secondly, a growing body of philosophers of mathematics are forgoing the standard inquiries into mathematical ontology and the foundational debates in favor of closer investigations of different aspects of contemporary mathematical practice. A new volume *The Philosophy of Mathematical Practice* edited by Paolo Mancosu includes an assortment of this work, with essays on mathematical explanation and understanding, visualization, diagrammatic reasoning, purity of methods, mathematical concepts, and the use of computers in mathematical inquiry. As part of both these currents, my case study exemplifies how this new wave of research in the philosophy of mathematics can help clarify the philosophical impact of a particular development in the modern theory of computing, one that challenges our epistemology of mathematics. Though I ultimately conclude that interactive zero knowledge proofs are *not* mathematical proofs and do *not* revolutionalize our understanding of 'proof' in any mathematically relevant sense, my investigation will nevertheless indicate how mathematicians and complexity theorists can differ in their interpretation of 'proof,' suggesting potential differences between their respective practices.

2 A Cat and Mouse Game

Let us begin by analyzing Fievel's protocol in detail. Before the challenge, Fievel has no knowledge of Dimitri's new maze. If Dimitri is honest, the maze will look like the one on the left in the figure below with a path between the North and South openings (you can verify this). If Dimitri is cunning, no thoroughfare will exist, as in the maze on the right. Fievel suggests that in each of K trials, Dimitri drops him at some random spot in the maze. Fievel then randomly chooses either 'North' or 'South' and Dimitri must lead him by some random walk to the chosen opening.[6] If Dimitri succeeds in doing so, Fievel accepts. If not, Fievel rejects. If the maze is a fair one, Dimitri will be able to lead Fievel out of the maze no matter which opening he chooses, so Fievel accepts in all K trials.[7] By contrast, if the maze is a trick, then no matter where Fievel is placed in a particular trial, Dimitri will have probability only $1/2$ of leading Fievel out. In the right maze for example, if Fievel picks 'North,' then Dimitri is defeated; if Fievel picks 'South,' then Dimitri can lead him out. Iterating this over K trials for a trick maze, Fievel has probability only $(1/2)^K$ of accepting in all K trials, or equivalently, probability $1 - (1/2)^K$ of rejecting in at least one of the K trials. So by following this protocol and choosing K large enough, Fievel can

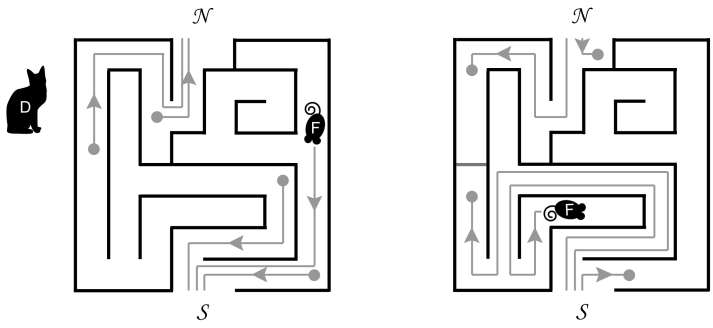
[5] *The Monist* 82:1, 1999; *Minds and Machines* 17:2, 2007; see the PCS website hosted at the University of Essex for more references.

[6] To make this more precise, assume Dimitri considers the set of all acyclic paths from Fievel's current location in the maze to the chosen entrance and picks one member at random.

[7] Assume the maze cannot contain an isolated point which is not reachable from either entrance.

become convinced with as high a probability as he would like that the maze is a fair one.

Now here is the crucial point: if the maze is fair, then Dimitri can convince Fievel of this fact while providing *no additional knowledge that Fievel could not have easily gained on his own.* To see this, consider the case where Fievel and Dimitri follow the protocol for the full K trials. By the end of the protocol, Fievel has been led from K random spots in the maze along random walks to either the North or South opening. He knows of K random walks $j_1, ..., j_K$ and is also convinced that a path through the maze exists. But consider this alternative: for K trials, Fievel picks one of the openings at random and starting from this opening, he begins to walk randomly through the maze, stopping after some random interval of time has passed and returning back the way he came. In this case, Fievel is no longer convinced the maze is fair (assume he does not traverse the entire maze or exhaust all possible paths) as Fievel can go on these walks in a trick maze as well. But Fievel again has knowledge of K random walks $j'_1, ..., j'_K$ so this cannot be knowledge gained exclusively in his interaction with Dimitri. The two alternative protocols are pictured below.



Dimitri leads Fievel out of the maze

Fievel goes for some strolls

My claim is not that Fievel gains *no* additional knowledge in his interaction with Dimitri beyond knowledge that the maze is fair, but only the milder one that Fievel gains no additional knowledge that he could not have easily acquired by himself. To be sure, Fievel does gain substantive knowledge of the walks $j_1, ..., j_K$ and there is even a chance that the union of some subset of these walks is a path through the maze. The point is that Fievel could have arrived at the same epistemic state by going for his own walks instead of being led out of the maze by Dimitri. Through slight revisions of the original protocol, this additional knowledge can also be reduced. Perhaps after Dimitri places Fievel in the maze, he blindfolds Fievel and only takes off the blindfold as they approach the chosen opening. Unless Fievel has an excellent non-visual sense of direction, knowledge of $j_1, ..., j_K$ is now useless.

3 IP and ZK

The Cat and Mouse example captures the spirit of zero knowledge interactive proofs well. There is a repeated dynamic exchange between a prover (Dimitri) and verifier (Fievel) where the prover attempts to convince the verifier of the truth of a particular claim (that the maze is fair). Both the prover and verifier can randomize their actions and the prover generally knows something (the maze layout) that the verifier does not. If the claim holds, the verifier always accepts at the end of the interaction. If the claim does not hold, the verifier *almost always* rejects. When the claim does hold, the verifier gains no additional knowledge beyond the truth of the asserted claim from the interaction that he could not have gained independently.

These intuitions can be formalized in a computational complexity framework. Goldwasser, Micali and Rackoff [1985] introduced the concept of an *interactive proof system*[8] as a pair of Turing machines, each with a read-only random tape and a read-write work-tape, with communication tapes for sending messages back and forth.[9] The prover machine P is computationally unbounded while the verifier machine V is probabilistic polynomial-time (it has limited computing power). In an interactive proof, the machines send messages to each other as the prover attempts to convince the verifier of the membership of an input x in a set S. In the Cat and Mouse game for example, Dimitri tries to convince Fievel that his new maze is contained in the set of mazes with thoroughfares between the openings. Following Trevisan [2007], I denote the interaction between the prover P and verifier V in a particular protocol by $V \leftrightarrow P$ and read '$V \leftrightarrow P$ accepts x' as: verifier V accepts after interacting with the prover P on common input x.

A formal definition of an interactive proof can now be given (adapted from Goldreich [2008b], p. 8):

Def 1. An interactive proof system (P, V) for a set S is a two-party game between a probabilistic polynomial-time verifier V and probabilistic prover P satisfying the following conditions:

Completeness:[10] if $x \in S$ then $\Pr(V \leftrightarrow P \text{ accepts } x) = 1$

Soundness:[11] if $x \notin S$ then for every prover P^*, $\Pr(V \leftrightarrow P^* \text{ accepts } x) \leq 1/2$

[8] As is commonplace in the literature, I will use the terms 'proof' and 'proof system' interchangeably in the interactive zero knowledge context. 'Proof systems' are discussed in more detail in §5.

[9] Earlier versions of this work date back to 1982. Instead of conceiving of the prover and verifier as interactive machines, one can alternatively think of these 'players' in terms of the strategies they employ: functions from the common input x, a player's internal random bits and the messages it has received so far, to the player's next move (Goldreich [2008b], p. 7-8).

[10] Relaxing this *perfect completeness* condition to allow for two-sided error (*i.e.*, $V \leftrightarrow P$ can reject when $x \in S$) does not increase the power of interactive proof systems (Goldreich [2008b], p. 16).

[11] The soundness condition is sometimes relaxed so that it only refers to provers P^* that can be implemented by a family of polynomial-size circuits. In this case, the condition is called *computational soundness* (Goldreich [2008b], p. 20).

The class of sets with interactive proofs is denoted by **IP**.[12]

The completeness condition is straightforward: if the input x is in S, the prover P can always convince the verifier V of this fact. The soundness condition is more complex: if the input x is not in S (the maze is a trick) then $V \leftrightarrow P^*$ rejects x with probability at least $1/2$, though by repeating the interaction over multiple trials the soundness error (*i.e.*, the probability that an input x is mistakenly accepted) can be made arbitrarily small. The soundness condition must hold for *any* possible prover P^* that interacts with V, indicating that the prover need not be trusted and hinting at the cryptographic applications of interactive proofs. Assume for example that Dimitri does not follow the protocol. Dimitri might always drop Fievel at the same spot in the maze or lead him through the maze on non-random routes. The soundness condition ensures that none of these deviations from the protocol will trick Fievel. Irrespective of Dimitri's actions, Fievel will still reject a bad maze at least half of the time.

Whereas traditional static proofs can be written down in a textbook or journal, Goldwasser, Micali and Rackoff [1985] compare interactive proofs to those that can be 'explained in class':

> Informally, in a classroom, the lecturer can take full advantage of the possibility of interacting with the 'recipients' of the proof. They may ask questions at crucial points of the argument and receive answers. This makes life much easier. Writing down a proof that can be checked by everybody without interaction is a much harder task. In some sense, because one has to answer in advance all possible questions. (p. 292)

Intuitively, one would expect interactive proofs to be more powerful than static ones and, in fact, it has been proven that they likely are. For complexity theorists, the canonical proof systems are **NP**-proofs which are essentially interactive proofs without the interaction and randomness. In an **NP**-proof system, the prover is only implicit and their one message must be deterministically verifiable in polynomial time. Shamir's [1990] celebrated **IP** Theorem states that $\mathbf{IP} = \mathbf{PSPACE}$.[13] We know that $\mathbf{NP} \subseteq \mathbf{PSPACE}$ and it is generally believed that $\mathbf{NP} \subset \mathbf{PSPACE}$. The former inclusion implies that interactive proofs are at least as powerful as **NP**-proofs and if the latter strict inclusion holds, then $\mathbf{NP} \subset \mathbf{IP}$ so interactive proofs are more powerful than **NP**-proofs.

A more surprising feature of **IP** is that given certain intractability assumptions,[14] every set which has an interactive proof has a 'zero knowledge' interactive proof as well. Introduced along with interactive proof systems in Goldwasser, Micali and Rackoff [1985], zero knowledge proofs are the limiting cases of interactive proofs that convince V of the truth of the claim

[12] A finer hierarchy of classes $\mathbf{IP(k(n))}$ can be defined where for $k : \mathbb{N} \to \mathbb{N}$, $\mathbf{IP(k(n))}$ is the class of sets with interactive proofs in which the interaction $V \leftrightarrow P$ involves at most $k(n)$ messages on inputs x of length n (Trevisan [2007], p. 1).

[13] **PSPACE** is the class of sets whose membership can be decided by a deterministic Turing machine that needs only a polynomial amount of space on the tape.

[14] *I.e.*, that one-way functions exist; see Goldreich [2004], p. 6-7.

$x \in S$ but provide no additional knowledge. This raises immediate conceptual difficulties: What account of knowledge is applicable here? And how can we measure the amount of 'additional knowledge' gained in an interaction? In the zero knowledge context, the clever stroke of computer scientists is to largely sidestep these questions altogether and focus on what it means to 'gain nothing' from an interaction. Looked at from this angle, characterizing zero knowledge becomes tractable: "*the adversary* ***gains nothing*** *if whatever it can obtain by unrestricted adversarial behavior can be obtained within essentially the same computational effort by a benign (or prescribed) behavior.*" (Goldreich [2008b], p. 23) In the so-called 'simulation paradigm,' 'benign behavior' refers to a probabilistic polynomial-time simulation based only on the common input x, so $V \leftrightarrow P$ yields no additional knowledge if the verifier can simulate the entire interaction herself from x. I have already provided an example of such a simulation with Fievel's random independent strolls, an alternative way to generate K random walks through the maze without being led by Dimitri.[15]

This characterization of zero knowledge can be made precise (adapted from Goldreich [2008b], p. 24-5):

Def 2. A prover strategy P over a set S is zero knowledge if for every probabilistic polynomial-time verifier V^*, there exists a probabilistic polynomial-time simulator A^* such that $(P, V^*)(x)$ and $A^*(x)$ are computationally indistinguishable[16] for every $x \in S$, where $(P, V^*)(x)$ is a random variable representing the output of $V^* \leftrightarrow P$ and $A^*(x)$ is a random variable representing the output of A^* on x. An interactive proof system (P, V) is zero knowledge if P is zero knowledge.

The class of sets with zero knowledge proofs is denoted by **ZK**.[17]

Like the verifier V^* the simulator A^* has a random tape. Consequently, the simulator's output on input x, denoted $A^*(x)$, will be a random variable. The zero knowledge condition says that for all $x \in S$, the probability distribution of $A^*(x)$ is 'effectively similar' (see n. 16) to the distribution of the output of $V^* \leftrightarrow P$, *i.e.*, that for good inputs the simulator does its job well. Note that a simulator A^* must exist for *any* V^*, indicating that whereas in the soundness condition it was the prover that could not be trusted, the verifier's honesty is now in question. By asking predetermined questions when a protocol

[15] Though the Cat and Mouse example captures the intuition behind the simulation paradigm, it is not quite right. Depending on the nature of Dimitri's maze, the distribution of outputs of the two alternative protocols may diverge significantly. If there exists a point in the maze that is reachable by two distinct routes from the South opening, say, then the probability that Dimitri leads Fievel along one of these routes may be lower than the probability that Fievel strolls along it. I am grateful to Kenny Easwaran for making this point.

[16] Roughly, two distributions X and Y are *computationally indistinguishable* if no efficient algorithm can distinguish between them (for a precise definition, see Goldreich [2004], p. 7). This standard zero knowledge condition is sometimes referred to as *computational zero knowledge*. When the condition is strengthened so that $A^*(x)$ and $(P, V^*)(x)$ must be identical, there are *perfect zero knowledge* proofs and the class **PZK**; when the condition is relaxed to allow for minor deviations between the distributions, there are *statistical zero knowledge* proofs and the class **SZK** (*ibid*, p. 9-11).

[17] This definition is somewhat simplified. For details, see Goldreich [2008b], p. 24-5.

asks for random ones (*e.g.*, always choosing 'South'), V might attempt to pry additional knowledge from P. In a zero knowledge proof system, such attempts are in vain.

To clarify the concepts discussed in this section, I recommend that the curious reader work through some examples of interactive zero knowledge proofs in the literature. Goldreich [2008b], p. 29-31, for instance, presents an accessible zero knowledge proof for the Graph 3-colorability problem. Since Graph 3-colorability is **NP**-complete, the proof also establishes that given the existence of one-way functions (see n. 14), every **NP** set has a zero knowledge interactive proof, that $\mathbf{NP} \subseteq \mathbf{ZK}$.[18]

4 The Epistemology of Mathematics

Acquainted with interactive proofs and zero knowledge, we can now ask whether these concepts have any significance for the philosophy of mathematics. That they do is *prima facie* far from clear. On one hand, recall Goldreich and Wigderson's opening claim that zero knowledge interactive proofs 'introduce a deep and fruitful revolution in the understanding of the notion of proof.' Given the mathematical flavor of such decision problems as Graph 3-colorability, their assertion *suggests* that the proof systems challenge our traditional conception of 'mathematical proof' and have something deep to teach philosophers about the epistemology of mathematics. However, Goldwasser, Micali and Rackoff certainly did not introduce the concepts of interactive proofs and zero knowledge with the intention of revising our epistemological ideas, but rather had an eye on cryptographic applications.[19] In this section, I consider what philosophical lessons, if any, can be gleaned from this complexity theory research that are relevant to philosophers of mathematics. Whatever one's initial impressions, I think the strong mathematical and epistemological nature of interactive zero knowledge proofs merits such a philosophical analysis.

As a first step, consider the following proposal:

Possible Lesson 1: Interactive zero knowledge proofs are bona fide mathematical proofs and claims like 'This graph possesses a Hamiltonian circuit' are pieces of genuine mathematics. Consequently, the standard model of mathematical proof as a sequence of statements that are either axioms or derived from previous statements by rules of inference is too narrow. Mathematical proofs are not abstract objects but rather interactive processes and a complete picture of the proper warrants for mathematical truth must account for this interaction.

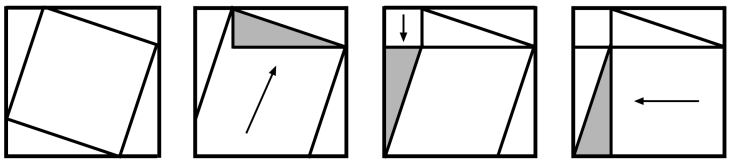
I think most philosophers would reject this characterization of mathematical proofs as *interactive processes*. An anomaly on the battlefield of endless controversies, there is actually general agreement among philosophers of mathematics that the standard model of proof provided by mathematical

[18] Since $\mathbf{ZK} \subseteq \mathbf{IP}$ by definition, it follows that $\mathbf{NP} \subseteq \mathbf{IP}$. And given the existence of one-way functions, it is also known that $\mathbf{ZK} = \mathbf{IP}$ (Goldreich [2008b], p. 31).

[19] I am grateful to one of the anonymous referees for making this point.

logic which explicates proofs as formal deductive arguments successfully articulates the proper warrants for mathematical knowledge.[20] Of course, such tight gapless deductions are idealized notions that one rarely comes across in actual mathematical practice and, moreover, we are perfectly comfortable calling the imprecise and incomplete arguments we do come across in mathematics textbooks and journals 'proofs.' But ordinary proofs, as the story goes, are only informal guides to the low-level proofs, indicating to the punctilious reader how they can work out the details should they have the paper and patience.[21] In any case, Goldreich [2008b], p. 4, even concedes in a recent survey that interactive proofs are not even informal mathematical proofs: "the motivation for the definition of interactive proof systems is not replacing the notion of a mathematical proof, but rather capturing other forms of proofs that are of natural interest." What Goldreich has in mind here are the 'daily proofs' found in more dynamic social contexts, such as the withstanding of a cross-examination in a law court (which 'proves' the defendant's innocence) or a debate in the political or scientific domain (*ibid*, p. 5). Though we may call these interactive exchanges 'proofs,' we are not using the term in any mathematical sense.

Regarding the ordinary mathematics proofs found in textbooks and journals, I do not even think that the claim that such proofs are dynamic and interactive processes is all that revolutionary. Consider my favorite visual proof of the Pythagorean theorem:



It begins with the leftmost square made up of four identical copies of a right triangle and a tilted middle square whose area is the square of the hypotenuse of the triangle. After some shifting, it ends with the rightmost square made up of the same four copies of the right triangle and two smaller squares whose areas are the squares of the sides of the triangle. As the areas of the two large squares are equal, the Pythagorean theorem holds. Other examples from mathematics, in particular mathematical logic, abound: in priority arguments in recursion theory, we enumerate sets in dynamic layered constructions to ensure certain requirements are met (Soare [1987], Chapters VII and VIII); in Henkin's Compactness proof for first-order logic, the sentences of a formal language are dynamically considered in turn (Hodges [1997], p. 124-6); and for a more interactive example, in an Ehrenfeucht-Fraïsse back-and-forth game in model theory, the fictitious players ∀belard and ∃loise take turns choosing elements from two abstract mathematical structures, ∀belard try-

[20] There is the separate issue of what justifies the axioms, or first principles, of mathematics but I will not go into this here.

[21] Whether they agree with this particular story or not, most philosophers of mathematics still consider ordinary mathematical proofs to be *objects* (static texts and/or diagrams) rather than *processes*.

ing to show they are different while ∃loise tries to show they are structurally identical (*ibid*, p. 74-81).

A possible reply here is that the dynamism and interaction found in interactive proofs is of an entirely different sort. In an Ehrenfeucht-Fraïsse game, the interaction enters heuristically in the form of a game, a nonessential mode of presentation used by the prover to convince the verifier that an isomorphism exists, say, between two countable atomless Boolean algebras.[22] In my other mathematical examples, the dynamism also stays on the (implicit) prover's side. But in interactive proofs, by contrast, there is nontrivial interaction *between* the prover P and verifier V: the players usually exchange multiple messages back and forth as the prover attempts to convince the verifier of the truth of a particular claim. Mathematical proofs can still be regarded as **NP**-proof systems (as I have already been suggesting), special cases of interactive proof systems where a 'transcendental' prover sends a single message to an efficient deterministic verifier, but this is clearly an impoverished notion of multiplayer interaction. However, can we not potentially go further? For looking at mathematical proofs in a wider communal context involving classrooms, conferences, and refereed journals, shifting the focus from a printed proof to the robust mathematical activity leading to its discovery, development, and publication, proofs lose their static solitary character. Adopting a sociological and historical stance towards mathematical proofs can imbue them with a form of interaction similar to that found in interactive proofs. Now this broad open-ended approach is a bit of a stretch, I know. Conceiving of mathematical proofs as socio-historical processes is a radical move and, furthermore, I still do think there is a sharp qualitative difference between the interaction in interactive proofs and that found in Ehrenfeucht-Fraïsse games and mathematical activity at the community level. Hence, I am content to make the weaker claim that characterizing mathematical proofs as dynamic and interactive is not as revolutionary as it initially seems.

So much then for Lesson 1. Interactive proofs are *not* mathematical proofs and viewed in a particular light, informal mathematical proofs are already interactive. But our philosophical investigation is not over. An important issue still remains: though interactive proofs are orthogonal to our mathematical conception of proof, we can still ask whether the subject matter of these proof systems, claims like 'The number 3571 is prime' or 'This graph has a maximum independent set of size 3,' are pieces of genuine mathematics and if so, whether such assertions are of interest to philosophers of mathematics. If they are mathematically significant, interactive zero knowledge proofs might still inform our theories of *mathematical evidence* (but not our models of *mathematical proof*) and impact our general views on the epistemology of mathematics.[23] If they are not, there is no need to go on.

Fortunately for the reader enjoying my present analysis, I think that the

[22] The dynamic concept of a *back-and-forth game* can be replaced by the static concept of a *back-and-forth system*, a set of isomorphisms between substructures of the original mathematical structures. See Hodges [1997], p. 76-8, for details.

[23] Whereas our models of mathematical proof are concerned with the proper warrants for mathematical knowledge, our theories of mathematical evidence are concerned with when it is rational to believe that such warrants exist.

type of decision problems with interactive zero knowledge proof systems are of significant mathematical interest. To be sure, a mathematical theorem of the form '$x \in S$' is somewhat unusual. Mathematicians are for the most part in the business of proving general theorems and studying abstract structures and the mappings between them, while interactive proofs concern individual objects such as specific finite graphs, natural numbers, or Boolean formulae. That said, important mathematical theorems of the form '$x \in S$' and specific combinatorial problems *do* exist: Pythagoras famously proved, for example, that $\sqrt{2}$ is in the set of irrational numbers; Euler's 1732/3 proof that $2^{2^5} + 1$ is *not* prime (is in the set of composite numbers) refuted a long-standing conjecture of Fermat that for every natural number n, $2^{2^n} + 1$ is prime[24] (Avigad [2006], p. 110); and familiar examples from discrete geometry which resemble Graph 3-colorability and other **NP** problems include the Four Color theorem, Kepler conjecture, and Kelvin problem. Collectively, I think these examples suffice to show that the claims referred to by interactive proof systems are, to use G. H. Hardy's term, 'genuine mathematics.'

Decision problems like Boolean Satisfiability or the Traveling Salesman Problem, however, may still be uninteresting to philosophers of mathematics. Goldreich [2008b], p. 6, calls their instances 'mundane theorems' and Canto 11 of Hardy's classic 'A Mathematician's Apology' opens:[25]

> A chess problem is genuine mathematics, but it is in some way 'trivial' mathematics. However ingenious and intricate, however original and surprising the moves, there is something essential lacking. Chess problems are *unimportant*. The best mathematics is *serious* as well as beautiful—'important' if you like, but the word is very ambiguous, and 'serious' expresses what I mean much better.

Are the various 'genuine' mathematical theorems that concern complexity theorists working with interactive zero knowledge proof systems similar to chess problems in being pieces of 'trivial' mathematics, 'unimportant' and 'unserious'? Compared to enigmas like the Twin Prime conjecture or the Continuum hypothesis, isn't the existence of a Hamiltonian circuit in a particular graph only a minor footnote in the onward march of mathematics, disconnected from deeper mathematical developments in, say, algebraic graph theory and unlikely to lead to any important mathematical or scientific advances?[26] I think that both questions should be answered 'No.' Firstly,

[24] Indeed, the numbers $2^{2^0} + 1$, $2^{2^1} + 1$, $2^{2^2} + 1$, $2^{2^3} + 1$, and $2^{2^4} + 1$ are all prime.

[25] I came across this section of Hardy's 'Apology' via Fallis [1997], p. 170. An anonymous referee also cheekily writes: "A penny may be 'genuine' money, but it would hardly get anyone's attention."

[26] In the next paragraph of Canto 11, Hardy continues: "The 'seriousness' of a mathematical theorem lies, not in its practical consequences, which are usually negligible, but in the *significance* of the mathematical ideas which it connects. We may say, roughly, that a mathematical idea is 'significant' if it can be connected, in a natural and illuminating way, with a large complex of other mathematical ideas. Thus a serious mathematical theorem, a theorem which connects significant ideas, is likely to lead to important advances in mathematics itself and even in other sciences. No chess problem has ever affected the general development of scientific thought: Pythagoras, Newton, Einstein have in their times changed its whole direction."

computers have been increasingly used in so-called 'experimental mathematics,' especially in testing and falsifying open mathematical problems (Avigad [2008a], p. 303-4). As of February 2008, for example, the Goldbach conjecture has been verified by Oliveira e Silva[27] for $n \leq 11 \cdot 10^{17}$. Checking particular instances of such conjectures is not unlike solving the types of arithmetic problems that have interactive proofs. And though verifying a general conjecture's instances cannot *prove* the conjecture true, computerized tests can nevertheless provide us with *evidence* that the conjecture holds or in some cases even lead to a counterexample.[28] Secondly, computer-assisted brute force combinatorial enumeration has become a common technique in discrete geometry (Avigad [2008a], p. 304). Appel and Haken's 1977 proof of the Four Color Theorem required them to show that almost 2,000 specific graph configurations could not appear in a minimal counterexample to the theorem. Hale's 1998 proof of the Kepler conjecture similarly required him to consider thousands of 'tame' graphs. Each of these examples indicates how the proof of an important mathematical theorem can reduce to showing that a particular property holds of each member of an exhaustive finite set of mathematical objects, again not unlike the subject matter of interactive proof systems. And whatever one's views are on lengthy computerized proofs, I think all would agree that research by Appel, Haken and Hales has garnered substantial *evidence* for the truth of the Four Color Theorem and Kepler conjecture.[29]

The kind of garden-variety results obtained by interactive proof systems, then, may have an important role to play in our theories of mathematical evidence. Despite their seeming triviality, such claims can be mathematically significant: in isolation, they can serve as counterexamples to important open conjectures; in the thousands, they can complete a 'proof by exhaustion' in discrete geometry; in the billions, they can provide considerable support for a general hypothesis. This suggests another epistemic lesson:

Possible Lesson 2: Though interactive proofs are not mathematical proofs, the claims verified in interactive zero knowledge proofs can constitute evidence for the truth or falsity of serious mathematical conjectures. Accordingly, we must broaden our theories of mathematical evidence to accommodate certain features of interactive proof systems, in particular, their randomness and soundness error. Our theories of evidence must provide us with a framework in which to assess probabilistic claims like 'It is nearly certain that 3571 is prime' or 'It is highly likely that the Goldbach conjecture is true.'

Recall that interactive proofs can be fallible. If an interactive proof sys-

[27] See Tomás Oliveira e Silva's webpage 'Goldbach conjecture verification' hosted at the Departamento de Electrónica, Telecomunicações e Informática, Universidade de Aveiro, Portugal.

[28] How long would Fermat's conjecture that for all n, $2^{2^n} + 1$ is prime have remained open in our modern age of experimental mathematics?

[29] Though there was some controversy surrounding the original Appel and Haken proof (mostly among philosophers), the proof of the Four Color Theorem was recently formalized and checked by G. Gonthier in 2005 using the mechanized proof assistant Coq (see Arkoudas and Bringsjord [2007]). Meanwhile, Hales has launched the 'Flyspeck' project to produce a formal proof of the Kepler conjecture and estimates that the project may take up to 20 years to complete.

tem has non-zero soundness error, a successful proof (*i.e.*, where $V \leftrightarrow P$ accepts) will only convince the verifier of the truth of the considered claim with high probability, not complete certainty. Moreover, this fallibility is *built-in*. Unlike a mathematical proof which, when correct, provides an *a priori* warrant for the implication between premises and conclusion,[30] many interactive proofs are explicitly structured to only convince V 'beyond a reasonable doubt' that $x \in S$. In fact, this randomness and soundness error gives interactive proof systems their power (see n. 14). What Lesson 2 tells us is that since interactive zero knowledge proofs can play an important role in justificatory efforts in mathematics, the built-in fallibility of some proof systems requires philosophers of mathematics to seriously consider probabilistic evidence.

Some philosophers already have. In his discussion of the relevance of computer usage in mathematics on theories of mathematical evidence, Avigad ([2008a], §11.4) mentions Pólya, Hacking, Good, Gaifman, and Corfield as proponents of a 'qualitative theory of mathematical plausibility.'[31] But despite their efforts, a worked-out generally accepted theory of mathematical evidence that includes probability doesn't exist. In one respect then, Lesson 2 seems correct. Interactive proofs do point to the need for a theory of probabilistic mathematical evidence, arguably an important extension of our philosophy of mathematics. That said, a lot of other things do as well: Pólya called for a theory of mathematical plausibility as early as 1941 when theoretical computer science was still in its infancy (*ibid,* p. 309); Avigad's discussion of inductive evidence in mathematics also makes no explicit mention of interactive proofs, though it is motivated by the prevalence of computational methods in modern mathematical practice, such as the probabilistic primality test of Solovay and Strassen (*ibid,* p. 305); and Fallis' investigation of probabilistic proofs (n. 31) centers on Adelman's DNA proofs for determining whether a graph has a Hamiltonian path (that have imperfect completeness). Together, I think these examples show that there is still something misleading about Lesson 2. For neither Adleman's remarkable 'molecular proofs' nor the Solovay-Strassen primality test are interactive proofs but like interactive proofs, both motivate a theory of mathematical evidence that

[30] At least on the standard rationalist view of mathematical proof (Arkoudas and Bringsjord [2007], p. 188).

[31] Fallis [1997] also argues that there is no epistemic reason for preferring the proofs widely accepted in the mathematical community to probabilistic proofs, though he *does* think that "mathematicians probably do have good reasons (for example, sociological and/or pedagogical ones) for not using probabilistic methods" (p. 166). However, as Avigad ([2008a], p. 307) rightly points out, the incorporation of probability in our theories of *mathematical evidence* (as Lesson 2 teaches) should not be confused with the acceptance of inductive evidence in *mathematical proofs* (as Fallis argues):

> one need not conflate the attempt to provide an idealized account of the proper warrants for mathematical knowledge with the attempt to provide an account of the activities we may rationally pursue in service of this ideal, given our physical and computational limitations. It is such a conflation that has led [Fallis] to wonder why mathematicians refuse to admit inductive evidence in mathematical proofs. The easy answer to Fallis's bemusement is simply that inductive evidence is not the right sort of thing to provide mathematical knowledge, as it is commonly understood.

incorporates probability. And there are plenty of other such probabilistic examples around. In light of these, I think the general claim that research on interactive zero knowledge proof systems impels a broadening of our theories of mathematical evidence to accommodate likelihood claims in mathematics is exaggerated. Regardless of the seminal work of Goldwasser, Micali and Rackoff, such a 'broadening of our epistemological scope,' to use Avigad's phrase, has been under consideration for a long time.

So much for Lesson 2 as well. To summarize, I have been trying to mine work on zero knowledge interactive proofs for epistemic lessons in the philosophy of mathematics. In doing so, I focused on two ways in which interactive proofs differ markedly, at least at first glance, from the traditional mathematical conception of proof: they involve nontrivial interaction between the prover and verifier and allow for randomness and built-in error. In the first vein, I considered the possible lesson that interactive proofs require us to revise our models of mathematical proof to incorporate their novel interactive element. In the second vein, I considered the possible lesson that interactive proofs require us to extend our theories of mathematical evidence to incorporate their fallibility. In both cases, the lessons foundered: interactive proofs are not mathematical proofs and though they do suggest a need for a theory of probabilistic mathematical evidence, this is nothing new.

Changing tack, I now consider whether philosophers of mathematics can instead learn something from the 'zero knowledge' rather than the 'interactive proof' component of interactive zero knowledge proofs. Here is a final proposal:

Possible Lesson 3: Zero knowledge proofs underline how the principal function of a proof is to convince.[32] Despite providing the verifier with no explanation, no understanding, and no knowledge of the asserted claim beyond its truth, zero knowledge interactive proofs are still proofs, though not in any mathematical sense. Nevertheless, their import still carries over to the mathematical case: the real value of a mathematical proof lies in its persuasive power; mathematical explanation and understanding are negligible considerations in the effort to confirm mathematical facts.

This lesson is more subtle than the earlier ones. While Lessons 1 and 2 call for substantial changes in the epistemology of mathematics, Lesson 3 only calls for a shift in perspective, teaching us that the cardinal virtue of a proof, mathematical or otherwise, lies in its ability to convince the verifier, nothing more: "The glory attached to the creativity involved in finding proofs makes us forget that it is the less glorified process of verification that gives proofs their value." (Goldreich [2008b], p. 1) Zero knowledge interactive proofs are the extreme cases where finding solutions to problems is entirely subjugated to the verification process. After K trips through Dimitri's new maze, Fievel still does not know a path between the openings but may be convinced that a thoroughfare exists. Using the zero knowledge interactive protocol for Graph 3-colorability mentioned at the end of Section 3, the prover can convince the verifier that a graph is 3-colorable without disclosing a particular 3-coloring.

[32] Goldreich [2008b] opens with this quote by Shimon Even: "A proof is whatever convinces me."

Now neither of these proofs explain *why* the proven claim holds or provide the verifier with any understanding but according to Lesson 3, they still, as proofs, do the necessary work. As beautiful and interesting as the many alternative explanations of the Pythagorean theorem may be, for example, Lesson 3 tells us these are all bonus. Only persuasion really matters.

I think this lesson is actually a step backward in the philosophy of mathematics. Part of the new movement in the philosophy of mathematics towards a philosophy of mathematical practice has been a push for a more encompassing view of 'mathematical proof': Can we develop a general criterion that distinguishes proofs that explain from those that do not? How can we make sense of the multiplicity of proofs of the same mathematical fact and the general feeling among mathematicians that each of these proofs is valuable for providing a different sort of understanding? Avigad [2006], p. 106, aptly writes, "the challenge is to explain what can be gained from a proof beyond knowledge that the resulting theorem is true" and he later elaborates:

> we have discerned a grab bag of virtues that mathematical proofs can enjoy. Some of these virtues may be classified as explanatory: a proof can explain how it might have been discovered, how an associated problem was solved, or why certain features of the statement of the theorem are relevant. Proofs may also establish stronger statements than the theorem they purport to prove; they may introduce definitions and methods that are useful in other contexts; they may introduce definitions and methods that can fruitfully be generalized; or they may suggest solutions to a more (sic) general problem. They can also suggest related theorems and questions. We can add a few more fairly obvious virtues to the list: a good proof should be easy to read, easy to remember, and easy to reconstruct. Sometimes our criteria are at odds with one another: for example, we may value a proof for providing explicit algorithmic information, whereas we may value another proof for downplaying or suppressing calculational detail. (*ibid,* p. 128-9)

On this catholic view of proofs, persuasive power is only one virtue among many. For the 'new' philosopher of mathematics interested in mathematical practice, the challenge is to make sense of the constellation of virtues that mathematical proofs may exhibit, rather than focusing solely on the appropriate axioms and inference rules for proofs.[33] So much then for Lesson 3.

In the end, it seems that interactive zero knowledge proofs, despite their mathematical and epistemological tinge, have nothing much to teach philosophers of mathematics. Perhaps this theoretical computer science research does have important lessons for the epistemology of mathematics that I have overlooked. But in this section I have already explored several features that distinguish interactive zero knowledge proofs from mathematical ones (non-trivial interaction between the prover and verifier, built-in soundness error,

[33]There is a growing body of literature on mathematical explanation and understanding. See P. Mancosu's 'Mathematical Explanation: Why it Matters,' J. Hafner and P. Mancosu's 'Beyond Unification,' and J. Avigad's 'Understanding Proofs' in Mancosu (ed.) *The Philosophy of Mathematical Practice* for some recent examples of this work.

and absence of knowledge generation and explanatory force) and none of these have led to any novel insights in the philosophy of mathematics. I conclude that interactive zero knowledge proof systems *do not* revolutionize our understanding of mathematical proof and evidence.

5 The Epistemology of Theoretical Computer Science

Still, Goldreich and Wigderson are not necessarily wrong. The analysis in the preceding section does not refute their revolutionary claim that opened this essay, that 'interactive proofs and zero knowledge introduce a deep and fruitful revolution in the understanding of the notion of proof, one of the most fundamental notions of civilization.' For though work on interactive zero knowledge proofs does not inform our epistemology of mathematics, this research may still have something significant to teach philosophers about how 'proof' is interpreted *within* the theoretical computer science community. Indeed, my discussion in the previous sections underlines some of the ways in which mathematicians and complexity theorists can sometimes differ in their conception of proof, differences catalyzed by the introduction and subsequent development of interactive zero knowledge proofs by Goldwasser, Micali, Rackoff, Goldreich, and others. To the extent that these differences reflect broader perspectives and epistemic principles adopted by these theoretical computer scientists, work on interactive zero knowledge proof systems may also motivate the development of a separate epistemology of theoretical computer science (or at least complexity theory) that departs from the theory of mathematical knowledge advanced by logicians and philosophers of mathematics.

Let us take this as our starting point: research on interactive proofs extends the concept of 'proof' to the concept of 'proof system.' Despite the conflation of these terms in the theoretical computer science literature, I find it helpful to distinguish between them. By the term 'proof,' I refer to an object, a synthesis of text and diagrams that establishes the truth of a mathematical claim. Both formal and informal mathematical proofs fall under this description which seems to loosely capture the sense of 'proof' as the term is used within mathematics. By the term 'proof system,' I refer to a particular kind of environment, one containing both a prover and verifier with certain properties (such as being polynomial-time, zero knowledge, etc.) in which a particular kind of process takes place: the prover and verifier exchange messages back and forth and at the end of the interaction the verifier either accepts or rejects. I have already contrasted these concepts in §4. But note that when 'proof' is interpreted in the above loose sense, the concept of 'proof system' need not be seen as a replacement of the concept of 'proof' but rather as its generalization. In cases where an implicit prover sends only a single message to the verifier, as in **NP**-proof systems, the difference between 'proof' and 'proof system' becomes so subtle that it almost disappears.

Since some complexity theorists have embraced the concept of 'proof system,' it is tempting to conclude, as Goldreich and Widerson do, that the

epistemological views of these computer scientists are revolutionary. And in some sense this does seem right. The extension to 'proof system' does lead to a richer conception of proof, making explicit certain features that are either trivial or absent in mathematical proofs. Firstly, in proof systems, the verification procedure has a dominant role: "Conceptually speaking, proofs are secondary to the verification process; whereas technically speaking, proof systems are defined in terms of their verification procedures." (Goldreich [2008b], p. 1) For mathematicians, by contrast, verification is presumably just 'following the logic' of a proof and this is not given much thought. Secondly, in proof systems, the computing resources of both the prover and verifier are explicitly acknowledged, whereas the computational complexity[34] of mathematical proofs is rarely given serious attention. Thirdly, in proof systems, proof and verification processes can be probabilistic, breaking from the strict determinism that characterizes the traditional conception of mathematical proof. Fourthly, proof systems can have built-in fallibility as completeness and soundness are explicitly recognized as key properties of proof systems and these can be imperfect. Standard mathematical proofs, however, are purported to have perfect completeness and perfect soundness so these properties are not mentioned. The four features come together in the definitions of particular proof systems: an **NP**-proof system is one with perfect completeness and perfect soundness where the verifier implements a deterministic polynomial-time strategy, while an interactive proof system is one with perfect completeness and (potentially) imperfect soundness where the verifier implements a probabilistic polynomial-time strategy.

Initially, it also seems plausible that these four features of proof systems reflect various attitudes and epistemic principles that are entrenched in some parts of the theoretical computer science community and are, for the most part, absent in the mathematical community. One might infer from the probabilistic nature and built-in fallibility of interactive proofs that some complexity theorists are tolerant of uncertainty, that knowing that 'the number 3571 is prime' or some other result holds beyond a reasonable doubt can be good enough for justificatory purposes (*i.e.*, complete certainty, or the semblance of it, is not always required for knowledge claims in theoretical computer science). One might also infer from the emphasis on verification in proof systems and, in particular, work on zero knowledge proofs that there exist significant differences in what mathematicians and some theoretical computer scientists value about proofs. I think it is fair to say that, *ceteris paribus*, mathematicians value explanatory proofs more than proofs that do not explain, proofs that introduce useful new techniques more than proofs in which standard methods are applied, and proofs that increase our understanding more than those that do not. Zero knowledge researchers, on the other hand, appear to have far different values, valuing the persuasive power of a proof above everything else. Lastly, one might also infer from the algorithmic approach to 'zero knowledge' in the simulation paradigm and the focus on the computational resources of the prover and verifier in interactive proof systems that the epistemological views of some theoretical

[34]Not to be confused with the *computability* of formal mathematical proofs which is of fundamental importance in logic.

computer scientists are intertwined with the 'algorithmic lens' (to use another of C. Papadimitriou's terms) through which these scientists view the world in their quasi-imperialistic efforts to work everything out in a computational setting.[35] By viewing epistemic notions like 'proof' and 'knowledge' through the algorithmic lens, some complexity theorists reinterpret these notions in novel algorithmic ways.

Now if these considerations are accurate, then an epistemology of theoretical computer science, or at least its fragment in which the 'proof system' concept is applicable, will look rather different from our standard epistemology of mathematics. Such a distinct epistemology would account for the ways in which complexity theorists and mathematicians differ in their thinking about proofs, knowledge, evidence, justification, etc. But as enamored as I am by the prospect of a rich, distinctive epistemology of theoretical computer science, I am skeptical that the above picture which calls for its development is all that accurate. For as far as I can tell, the proofs found in theoretical computer science journals and textbooks are just formal mathematical proofs and even the warrants for meta claims regarding interactive proofs are standard 'written proofs.'[36] Despite their generalization of the concept 'proof' to 'proof system,' complexity theorists only seem to treat proof systems as objects of analysis, to be analyzed and incorporated into their ever-growing universe of complexity classes, rather than as a means to acquire knowledge in their own field. What's more, complexity theorists do not seem all that tolerant of uncertainty or single-mindedly bent on persuasion either. Go and tell some theoretical computer scientists that '$\mathbf{P} \neq \mathbf{NP}$ with 99% probability' and they will not be very impressed. And complexity theorists, like mathematicians, value new proofs of previously established theorems, such as Dinur's [2005] proof of the PCP Theorem or Lautemann's [1983] proof of the Sipser-Gacs Theorem, a phenomenon left unexplained by a picture in which the sole function of a proof is to convince. Though in certain cryptographic settings, the very features of proofs that mathematicians praise are hindrances and probabilistic fallible processes are appreciated, it is important to keep in mind that it is the cryptographers (or parties using the cryptographic protocol) rather than the complexity theorists who appreciate the zero knowledge component.

Perhaps a more thorough examination of the practice of complexity theory would establish the existence of significant differences in how theoretical computer scientists and mathematicians go about deciding what to believe and justifying knowledge claims. But unless it can be demonstrated that, say, inductive methods can be the sort of thing to provide knowledge and not simply evidence in theoretical computer science, it remains unclear why the development of a separate epistemology of theoretical computer science

[35] I have already mentioned several applications of the 'algorithmic lens' in biology, network economics, and physics in n. 2. Some other examples are: algorithmic game theory, computational population genetics, and graph theoretic investigations of the world-wide web.

[36] In branches of theoretical computer science such as program verification, the use of mathematical proofs is so essential that it is hardly worth mentioning. What is relevant here is that theorems in computational complexity are also proven in the standard mathematical sense.

would be a worthwhile pursuit, given that our epistemology of mathematics already does the necessary work. In two respects then, this essay has been a cautionary tale. In §4, I argued that research on interactive zero knowledge proof systems, while initially appearing to radically challenge our traditional conception of mathematical proof and other epistemological ideas, has little to teach philosophers of mathematics. In this final section, I have also challenged one who would take research on interactive zero knowledge proofs as a call to develop a partially independent epistemology of theoretical computer science. Note that I am not denying that the development of such a distinct epistemology may ultimately prove to be a fruitful enterprise but only expressing my initial skepticism for this project.

Acknowledgements

The first version of this essay grew out of a series of lectures by Luca Trevisan and an independent study with Christos Papadimitriou in 2007. I am grateful to these computer scientists for providing the initial inspiration for my project. Parts of the essay were subsequently presented at both the Richard Wollheim Society and Student Logic Colloquium at the University of California, Berkeley. I am grateful to the participants for comments and to the anonymous referees of this journal for sharp criticisms.

References

Jeremy Avigad. Mathematical method and proof. *Synthese*, 153:105-159, 2006.

Jeremy Avigad. Computers in Mathematical Inquiry. In P. Mancosu (ed). *The Philosophy of Mathematical Practice*, pages 302-316. Oxford University Press, Oxford, 2008a.

Jeremy Avigad. Understanding Proofs. In P. Mancosu (ed). *The Philosophy of Mathematical Practice*, pages 317-353. Oxford University Press, Oxford, 2008b.

Timothy Colburn. *Philosophy and Computer Science*. M. E. Sharpe, Armonk, NY, 2000.

Irit Dinur. The PCP theorem by gap amplification. Technical Report TR05-046, ECCC, 2005.

Don Fallis. The Epistemic Status of Probabilistic Proof. *Journal of Philosophy*, 94:165-186, 1997.

Luciano Floridi (ed). *The Blackwell Guide to the Philosophy of Computing and Information*. Blackwell, Oxford, 2003.

Oded Goldreich. Zero-Knowledge twenty years after its invention, March 2004. Full version available from
`http://www.wisdom.weizmann.ac.il/~oded/zk-tut02.html`.

Oded Goldreich. *Computational Complexity: A Conceptual Perspective*. Cam-

bridge University Press, Cambridge, 2008a.

Oded Goldreich. Probabilistic Proof Systems: A Primer, June 2008b. Full version available from
`http://www.wisdom.weizmann.ac.il/~oded/pps.html`.

Oded Goldreich and Avi Wigderson. The Theory of Computing: A Scientific Perspective, June 2001. Full version available from
`http://www.wisdom.weizmann.ac.il/~oded/toc-sp2.html`.

Shafi Goldwasser, Silvio Micali and Charles Rackoff. The knowledge complexity of interactive proof-systems. *Proceedings of the 17th ACM Symposium on the Theory of Computing*: 291-304, 1993.

Johannes Hafner and Paolo Mancosu. Beyond Unification. In P. Mancosu (ed). *The Philosophy of Mathematical Practice*, pages 151-178. Oxford University Press, Oxford, 2008.

G. H. Hardy. A Mathematician's Apology, November 1940. Full version available from
`http://www.math.ualberta.ca/~mss/misc/A%20Mathematician's%20Apology.pdf`.

Wilfrid Hodges. *A Shorter Model Theory*. Cambridge University Press, Cambridge, 1997.

Clemens Lautemann. BPP and the Polynomial Hierarchy. *Information Processing Letters*, 17:215-217, 1983.

Paolo Mancosu (ed). *The Philosophy of Mathematical Practice*. Oxford University Press, Oxford, 2008.

Paolo Mancosu. Mathematical Explanation: Why it Matters. In P. Mancosu (ed). *The Philosophy of Mathematical Practice*, pages 134-150. Oxford University Press, Oxford, 2008.

Adi Shamir. IP = PSPACE. *Proceedings of the 31st Annual Symposium on Foundations of Computer Science*: 11-15, 1990.

Robert I. Soare. *Recursively Enumerable Sets and Degrees*. Springer-Verlag, Berlin, 1987.

Luca Trevisan. Notes on Zero Knowledge, May 2007. Full version available from `http://www.cs.berkeley.edu/~luca/cs172/notezk.pdf`.