What is Computation? (How) Does Nature Compute?

1. Quantum vs. Classical Computation

Adrian German: About five years ago at MIT Stephen Wolfram was asked a question[1] at the end of his presentation:

> Question (on tape): "How does your Principle of Computational Equivalence (PCE) explain the separation in complexity between a classical cellular automaton (CA) and a quantum CA?"
>
> Stephen Wolfram (answers the question on tape): "If you take the current formalism of quantum mechanics and you say: let's use it, let's just assume that we can make measurements infinitely quickly and that the standardized formalism of quantum mechanics is an exact description of the actual world, not some kind of an idealization, as we know that it is – because we know that in fact when we make measurements, we have to take some small quantum degree of freedom and amplify it to almost an infinite number of degrees of freedom, and things like that – but let's assume we don't worry about that, let's take the idealization that, as the formalism suggests, we just do a measurement and it happens. So then the question is: how does what I am talking about relate to the computation that you can get done in a quantum case vs. the classical case. And the first thing I should say is that, ultimately, the claim of my PCE is that in our universe it is in fact not possible to do things that are more sophisticated than what a classical CA, for example, can do. So, if it turned out that quantum CAs (or quantum computers of some kind) can do more sophisticated things than any of these classical CAs then this PCE of mine would just be wrong."
>
> Question (on tape): "By more sophisticated do you mean with a better complexity then, or do you mean in a universal sense?"

[1]Plays recording from `http://mitworld.mit.edu/video/149/` in which the selected paragraph appears at the 1:28:28 mark. Tape starts.

> Wolfram (on tape): "So – to be more specific, one question is: can it do more than a universal Turing machine, can it break Church's thesis? Another one is: can it do an NP-complete computation in polynomial time? And on the second issue I am not so sure. But on the first issue I certainly will claim very strongly that in our actual universe one can't do computations that are more sophisticated than one can get done by a standard classical Turing machine. Now, having said that there's already then a technical question which is: if you allow standard idealizations of quantum mechanics, with complex amplitudes that have an arbitrary number of bits in them, and things like that, even within the formalism can you do computations that are more sophisticated than you can do in a standard classical universal Turing machine? I think that people generally feel that you probably can't. But it would be interesting to be able to show that. For example, the following is true: if you say that you do computations just with polynomials then there is a result from the 1930s that says that every question you ask is decidable. This was Tarski's result. So, if the kinds of primitives that you have in your theory would be only polynomial primitives then it is the case that even using these arbitrary precision amplitudes in quantum mechanics you couldn't compute something more than what we can classically compute. But as soon as you allow transcendental functions that is no longer the case – because for example even with trigonometric functions you can easily encode an arbitrary diophantine equation in your continuous system. So if you allow such functions then you can immediately do computations that cannot be done by a classical Turing machine, and if there was a way to get quantum mechanics to make use of those transcendental functions – which certainly doesn't seem impossible – then it would imply that quantum mechanics in its usual idealization would be capable of doing computations beyond the Church's thesis limit. Then you can ask questions about the NP completeness level of things and, as I said, I'm less certain on that issue. My own guess, speculation if you want, is that if you try and eventually unravel the idealizations that are made in quantum mechanics you will find that in fact you don't succeed in doing things that are more sophisticated than you could do with a classical Turing machine type system – but I don't know that for sure!"

Adrian German: Where this answer ends, our conference starts. As quantum computation continues to generate significant interest while the interest in NKS grows steadily we decided to ask ourselves: Is there really a tension between the two? If there is – how is that going to inform us? What if the conflict is only superficial? It was then that we remembered a quote from Richard Feynman from the 1964 Messenger Lectures at Cornell, later published as the book "The Character of Physical Law":

> "It always bothers me that, according to the laws as we understand

> them today, it takes a computing machine an infinite number of logical operations to figure out what goes on in no matter how tiny a region of space, and no matter how tiny a region of time. So I have often made the hypothesis that ultimately physics will not require a mathematical statement, that in the end the machinery will be revealed, and the laws will turn out to be simple, like the chequer board with all its apparent complexities[2]."

The seed of that hypothesis may have been planted by one of our guests today: Ed Fredkin[3] who says that in spite of his extensive collaboration with Feynman was never sure that his theories on digital physics had actually made an impact with the legendary physicist – until he heard it in the lectures as stated above. (Ed Fredkin, seated next to Greg Chaitin at the table, confirms: "That's right.")

2. The Idea of Computation

George Johnson: Well, good morning, and I want to thank you all for coming out and also for inviting me to what's turned out to be a really, really fascinating conference. Friday night just after I got here I had dinner with my old friend Douglas Hofstadter who of course lives in Bloomington and I was thus reminded why (and how) I later became interested in computation in the abstract. I was already very interested in computers when Doug's book "Godel, Escher, Bach" came out – and I remember in graduate school picking up a copy in a bookstore on Wisconsin Ave., in Washington, DC, and just being absolutely sucked into a vortex – and then eventually buying the book and reading it. And as I was reading it I thought "Well, this is great, because I am a science journalist," and

[2]However, by 1981 this hypothesis seems to have lost its appeal for Feynman; in his talk at the *First Conference on the Physics of Computation*, held at MIT, he observed that it appeared to be impossible in general to simulate an evolution of a quantum system on a classical computer in an efficient way. He proposed a basic model for a quantum computer that would be capable of such simulations. That talk, published in 1982, has proved to be much more influential and more often quoted than the theme paragraph of our conference. At the same conference Tommaso Toffoli introduced the reversible Toffoli gate, which, together with the NOT and XOR gates provides a universal set for quantum computation. In 1985 in his notable paper, Deutsch was the first to establish a solid ground for the theory of quantum computation by introducing a fully quantum model for computation and giving the description of a universal quantum computer. After the pioneering work of David Deutsch, quantum computation still remained a marginal curiosity in the theory of computation until 1994, when Peter W. Shor introduced his celebrated quantum algorithms for factoring integers and extracting discrete logarithms in polynomial time. The importance of these algorithms is well-known, however the theory still remains far more developed than the practice.

[3]Richard Feynman's interaction with Ed Fredkin started in 1962 when Fredkin and Minsky were in Pasadena and one evening not knowing what to do with their time "sort of invited [them]selves to Feynman's house," and is very well documented by Tony Hey. Twelve years later, in 1974, Fredkin visited Caltech again, this time as a Fairchild scholar and spent one year with Feynman discussing quantum mechanics and computer science. "They had a wonderful year of creative arguments," writes Hey, "and Fredkin invented Conservative Logic and the Fredkin Gate, which led to Fredkin's billiard ball computer." (See `http://www.cs.indiana.edu/~dgerman/hey.pdf`)

at the time I was working for a daily newspaper in Minneapolis, "I think we really need to do a profile of this Hofstadter person." It was a great way to get introduced to this topic and I had my first of many other trips to Bloomington to meet Doug and over the years we got to know each other pretty well – in fact he copy-edited my very first book "The Machinery of the Mind."

And it was while writing that book that I came across a quote from one of tonight's guests Tommaso Toffoli that I just remembered while I was sitting here on these sessions and I looked it up the other night and I just wanted to use it to get us started. This was an article in 1984 in Scientific American by Brian Hayes[4] that mentions Stephen Wolfram's early work on CAs and mentions Norman Packard who was at the Institute for Advanced Studies (IAS) at the time (he later went on to co-found a prediction company in my hometown of Santa Fe and is now researching Artificial Life) and quotes[5] Dr. Toffoli as saying:

> "... in a sense Nature has been continually computing the 'next state' of the universe for billions of years; all we have to do – and actually, all we can do – is 'hitch a ride' on this huge ongoing computation."

It was reading that and things like that really got me hooked and excited about this idea of computation as a possible explanation for the laws of physics. We were talking about a way to get started here to have you briefly introduce yourselves, although most of us all know who you are and why you're important, but just to talk in that context of how you first got hooked to this idea and the excitement of computation in the abstract.

Gregory Chaitin: Being a kid what I was looking for was the most exciting new idea that I could find. And there were things like Quantum Mechanics and General Relativity that I looked at along the way and Gödel's Incompleteness Theorem but at some basic level what was clearly the big revolution was the idea of computation, embodied in computer technology. Computer technology was exciting but I was even more interested in the idea of computation as a deep mathematical and philosophical idea. To me it was already clear then that this was a really major new mathematical idea – the notion of computation is like a magic wand that transforms everything you touch it with, and gives you a different way of thinking about everything. It's a major paradigm shift at a technological level, at the level of applied mathematics, pure mathematics, as well as the level of fundamental philosophy, really fundamental questions in philosophy. And the part about the physical universe – that part was not obvious to me at all. But if you want to discover the world by pure thought, who cares how this world is actually built? So I said: let us design a world with pure thought that is computational, with computation as its foundational building block. It's like playing God, but it would be a world that we can understand, no? If we invent it we can understand it – whereas if we try to figure out how

[4] http://bit-player.org/bph-publications/AmSci-2006-03-Hayes-reverse.pdf

[5] http://www.americanscientist.org/issues/id.3479,y.0,no.,content.true,page.1,css.print/issue.aspx

this world works it turns into metaphysics again[6]. So, obviously, I've hitched a ride on the most exciting wave, the biggest wave I could see coming!

Ed Fredkin: When I was in the Air Force and was stationed at Lincoln Labs I had the good fortune to meet Marvin Minsky almost right away and not long afterwards I met John McCarthy and I'd get lots of advice from them. I don't know exactly when I first thought of the idea of the Universe being a simulation on a computer but it was at about that time. And I remember when I told John McCarthy this idea (that had to be around 1960) he said to me something that in our long time of talking to each other has probably said to me maybe a hundred times: "Yes, I've had that same idea," about it being a computer. And I said "Well, what do you think of it?" and he said "Yeah, well we can look for roundoff or truncation errors in physics..." and when he said it I immediately thought "Oh! He thinks I'm thinking of an IBM 709 or 7090 (I guess wasn't out yet) in the sky..." and I was thinking of some kind of computational process... I wasn't thinking of roundoff error... or truncation error. When I told this idea to Marvin he suggested that I look at cellular automata and I hadn't heard of cellular automata at that point so what I had to do was to find a paper by... or, find out what I could, I remember I couldn't find the paper that von Neumann had done and from that point on – which was around 1960 (say, 1959 or 1960) – I remained interested in it[7]. And on one of my first experiments on the computer I decided to find the simplest rule that is symmetrical in every possible way, so I came up with the von Neumann neighborhood and binary cellular automata and I thought "what function that's symmetrical is possible?" And, as it turned out it's XOR. And so I programmed that one up and I was kind of amazed by the fact that that simple rule was at least a little bit interesting.

Rob de Ruyter: I also have to go back to my youth, or maybe high-school, when (in the time that I was in high-school, at least) there was a lot of ebullience still around about biology and there were a number of wonderful discoveries

[6]Beginning in the late 1960s, Chaitin made contributions to algorithmic information theory and metamathematics, in particular a new incompleteness theorem in reaction to Gödel's incompleteness theorem. He attended the Bronx High School of Science and City College of New York, where he (still in his teens) developed the theories that led to his independent discovery of Kolmogorov complexity. Chaitin has defined Chaitin's constant Ω a real number whose digits are equidistributed and which is sometimes informally described as an expression of the probability that a random program will halt. Ω has the mathematical property that it is definable but not computable. Chaitin's early work on algorithmic information theory paralleled the earlier work of Kolmogorov.

Chaitin also writes about philosophy, especially metaphysics and philosophy of mathematics (particularly about epistemological matters in mathematics). In metaphysics, Chaitin claims that algorithmic information theory is the key to solving problems in the field of biology (obtaining a formal definition of 'life', its origin and evolution) and neuroscience (the problem of consciousness and the study of the mind). Indeed, in recent writings, he defends a position known as digital philosophy. In the epistemology of mathematics, he claims that his findings in mathematical logic and algorithmic information theory show there are 'mathematical facts that are true for no reason, they're true by accident. They are random mathematical facts'. Chaitin proposes that mathematicians must abandon any hope of proving those mathematical facts and adopt a quasi-empirical methodology.

[7]`http://www.digitalphilosophy.org/`

made and are still being made about how life on the cellular scale works. So that was a kind of an incredibly fascinating world for me that opened as I got interested in biology and learned ever more intricate mechanisms. At the same time of course we learned physics – actually in Holland in high-school you learn them both at the same time, as opposed to in this country. And in physics you get all these beautiful laws, mathematical descriptions of nature and they give you a real sense that you can capture nature in pure thought. Now it was obvious then already that going from that physical picture where you can describe everything very precisely to the workings of a biological cell that there's an enormous range of things that you have to cover in order to understand how this biological cell works and we still don't know that in terms of underlying physical principles. At the same time at that age when you're in high school you also go through all kind of hormonal, etc. transformations and one of the things you start to realize is that (a) as you look at the world you take information in from the world outside but (b) you also have an own mind with which you can also think and do introspection. And you know that this introspection, your own thoughts, somehow have to reside, they have to be built out of matter – that probably, out there somewhere, somehow, that has to happen. Then that's a question that has always fascinated me tremendously and still really fascinates me which is (if you want to put it in shorthand) the question of how matter leads to mind, the mind-matter question and everything that's related to that.

I was lucky enough to find a place in the physics department where there they had a biophysics group and I could study questions of mind – probably a simple mind, the mind of an insect, or at least little bits of the mind of an insect – and try to study them in a very, highly quantitative way that harkens back to the nice, beautiful mathematical description that you get in physics. So I think this tension between matter and how you describe it in mathematical formalisms and thought – how you introspectively know what thought is, and presumably to some extent all animals have thoughts – and thought to some extent takes the form of computation, of course, then [that] is the kind of thing that drives me in my work[8]. And as an experimentalist I find it very pleasing that you can do experiments where you can at least take little little slivers off these questions and perhaps make things a little bit more clear about the mind.

Tony Leggett: I guess I am going to be "skunk" of the group, because I actually am not convinced that a computational approach is going to solve at least those fundamental problems of physics which I find most interesting[9]. I

[8]Rob de Ruyter van Steveninck is Gill Professor in the Dept. of Physics and Program in Neural Science at Indiana University Bloomington. Fellow of the American Physical Society, Member on the Advisory Board for the Linda and Jack Gill Center at IU he has co-authored *Spikes: Exploring the Neural Code* (MIT Press, 1999) with William Bialek and others

[9]Sir Anthony James Leggett, KBE, FRS (born 26 March 1938, Camberwell, London, UK), aka Tony Leggett, is the John D. and Catherine T. MacArthur Chair and Center for Advanced Study Professor of Physics at the University of Illinois at Urbana-Champaign since 1983. Dr. Leggett is widely recognized as a world leader in the theory of low-temperature physics, and his pioneering work on superfluidity was recognized by the 2003 Nobel Prize in Physics. He has shaped the theoretical understanding of normal and superfluid helium liquids and strongly coupled superfluids. He set directions for research in the quantum physics

certainly think that the sort of ideas, like some of those that we've discussed at this conference, are extremely intriguing and they may well be right – I don't know. And one area in which I think that one could quite clearly demonstrate that this confluence of computational science and physics has been fruitful is of course the area of quantum information and in particular quantum computing. Certainly what's happened in that area is that, as I said earlier, an approach coming from computer science gives you a completely new way of looking at the problems, posing them as questions that you would not have thought about otherwise. But quite interesting actually that at least in the decade from 1995 to 2005, when quantum information clearly gained way, a lot of the papers which appeared in Physical Review Letters at that time in some sense could have easily appeared in 1964, but they hadn't. Why not? Because people then didn't have the knowledge (or intuition) to ask such particular type(s) of questions. So certainly I think that was a very very useful and fruitful interaction.

But I think that if someone told me, for example, that such and such a problem which can be easily posed in physics cannot be answered by a computer with a number of bits which is greater or equal to the total number of particles in the universe: I don't think I would be too impressed. And my reaction would probably be: "All right, so what?" I don't think that's useful, I don't think that nature goes around computing what it has to do – I think it just doesn't! But I think perhaps a little more fundamentally, one reason I'm a little skeptical about the enterprise is that I actually think that the fundamental questions in physics have much more to do with the interface between the description of a physical world and our own consciousness. One of the most obvious cases is the infamous quantum measurement problem, which I certainly do regard as a very serious problem; and secondly the question of the arrow of time: that we can remember the past, that the past influences the future and viceversa. It's something that underlines some of the most fundamental aspects of our existence as human beings and I don't think we understand them at all, at least I don't think we understand them in physical terms, and I personally find it very difficult to imagine how a computational approach could enhance our approach to any of these problems.

Now one of the reasons I come to conferences like this is that I hope that I may in the future somehow be convinced. So we'll have to wait and see.

Cristian Calude: I guess I am probably the most conventional and ordinary person[10] in this context but my interests are in computability, complexity and randomness, and I try to understand why we can do mathematics and what makes us capable of understanding mathematical ideas. I am also a little bit skeptical regarding the power of quantum computing. I've been involved in the last few years in a small project in dequantizing various quantum algorithms,

---

of macroscopic dissipative systems and use of condensed systems to test the foundations of quantum mechanics.

[10]Cristian Calude is Chair Professor at the University of Auckland, New Zealand. Founding director of the Centre for Discrete Mathematics and Theoretical Computer Science. Member of the Academia Europaea. Research in algorithmic information theory and quantum computing.

i.e., constructing classical versions of quantum algorithms which are as quick as their quantum counterparts. And I have come to I believe that quantum computing and quantum information say much more about physics than they could become useful tools of computation. Finally, I am very interested to understand the nature and the power of quantum randomness, whether this type of hybrid computation where you have your favorite PC as the source of quantum randomness ... can get you more. Can we surpass the Turing barrier, at least in principle, with this kind of hybrid computation or not?

Tom Toffoli: Well, I was born by a historical accident in a post office and that may have something to do with all of this (laughter) because it was exactly organized like Ethernet: press a key and you bring a whole lot of lines (40 miles) from a high value to ground, so you have to listen to what you are typing and if what you get is different from what you are typing, or what you're expecting to hear, then it means that someone else is on the line too. So you both stop. And then both of you start again, at random – exactly in the spirit of Ethernet. But anyway, when I was five, it was right after the end of the war, my mother already had four kids and she was very busy trying to find things for us to eat from the black market and such – so I was left off for the most part of the day in a place on the fifth or sixth floor. Where we lived we had long balconies with very long rails, for laundry etc. and the rail was not raised in the middle, it was just hooked, stuck in the wall at the endpoints on the two sides. So I was like a monkey in a cage in that balcony and I had to think and move around and find a way to spend time in some way. So one of the things I noticed was that as I was pulling and shaking this thing, the rope, somehow I discovered that by going to more or less one third of its length and shaking it in a certain way, eventually the thing would start oscillating more and more and more. And I discovered what you could call resonance, phase locking and other things like that – and I felt really proud about this thing, about this power unleashed into your [tiny] hands. But I also got very scared because – as a result of my early scientific activity – the plaster had come out at the place where the rail was hooked in the wall and there was even plaster outside on the sidewalk if you looked down from the sixth floor balcony where I was, so I was sure I was going to get into trouble when my mother would come home later that day.

In any event, the point that I'd like to make is that "I cannot understand why people cannot understand" (as Darwin wrote to a friend) "that there's no such thing as simply experiments – there are only experiments 'for' or 'against' a subject." In other words you have to ask a question first, otherwise the experiment is not a very useful thing. Now you will say: "Fine, but how are you going to answer the question that you're making the experiment for?" And what we do with computation, cellular automata and so on, is that we just make our own universe. But the one that we make, we know it, we make the rules so the answers that we get there are precise answers. We know exactly whether we can make life within this CA (like von Neumann) and we know whether we can compute certain things, so it's a world that we make so that we have precise answers – for our limited, invented world – and that hopefully can give us some light on answers for the questions in the real world. And so we have to go continually back and forth between the worlds that we know because we have complete control [over them] and we can really answer questions and the other worlds about which we would really like to answer questions. And my feeling is that computation, CA and all these things that Stephen has called our attention to are just attempts to get the best of the two worlds, essentially. The only world we can answer questions for is the one we make, and yet we want to answer the questions about the world in which we are – which has all of its imperfections, and where resonance does not give you an infinite peak but still gives you crumbled plaster on the floor near the wall and so on.

But I think that we should be able to live with one foot in one world and the other in the other world.

Stephen Wolfram: Let's see, the question I believe was how did one get involved with computation, and how did one get interested in computation and then interested in these kind of things. Well, the story for me is a fairly long one by now: when I was a kid I was interested in physics. I think I got interested in physics mainly because I wanted to understand how stuff works and at the time, it was in the 1960s, roughly, physics was the most convincing kind of place to look for an understanding of how stuff works. By the time I was 10-11 I started reading college physics books and so on and I remember a particular moment, when I was 12, a particular physics book that had a big influence on me: it was a book about statistical mechanics which had on the cover this kind of a simulated movie strip that purported to show how one would go from a gas of hard spheres or something like that, that was very well organized, to a gas of hard spheres that looked completely random. This was an attempt to illustrate the second law of thermodynamics. And I thought it was a really interesting picture, so I read the whole mathematical explanation of the book as to why it was this way.

And the mathematical explanation to me was pretty unconvincing, not at all in the least because this was one of those explanations which – it's about actually the analog of time basically – ended with a statement that said "well, the whole explanation could be just as well run in reverse, but somehow, that isn't how it works." (General laughter) Not very convincing, so I decided: all right, I will figure out on my own what was really going on there, I would make my own version of this movie strip. At the time I had access to a simple computer, it was an English computer: Eliott 903, computers we used mostly to operate tanks and such. It had eight kilowords of eighteen bit memory. It was otherwise a fine computer and I started trying to program this thing to work out the second law of thermodynamics. In some respects the computer was fairly primitive especially when doing floating point operations so I tried to simplify the model that I had for these hard sphere gas bouncing around. What I realized many years later was that the model that I had actually come up with was a two dimensional cellular automata system. The thing that went wrong was: (a) I simulated this 2DCA system and (b) I found absolutely nothing interesting and (c) I could not reproduce the second law of thermodynamics. So that was when

I was about 12 years old. And at that point I said: maybe there was something else going on, maybe I don't understand this so I'm going to work in an area where I could learn about stuff, where I think I can understand better what's going on. And the area that I got involved with was particle physics. And so I got to know all sorts of stuff about particle physics and quantum field theory and figured out lots of things that made me kind of a respectable operative in the particle physics business at the time.

And doing particle physics something that happened was that we had to do these complicated algebraic calculations, something that I was never particularly good at. I also have the point of view that one should always try to figure out the best possible tools that one can build to help one out. And even though I wasn't very good myself at doing things like algebra I learned to build computer tools for doing algebra. And eventually I decided that the best way to do the kinds of computations I needed to do was to build my own tool for doing those computations. And so I built a system called SMP which was the precursor of Mathematica – this was around 1981. In order to build this computer system I had to understand at a very fundamental level (a) how to set up a wide range of computations and (b) how to build up a language that should describe a wide range of computations. And that required inventing these primitives that could be used to do all sorts of practical computations. The whole enterprise worked out rather well, and one of my conclusions from that was that I was able to work out the primitives for setting things up into a system like SMP. And then for a variety of reasons when I got to started thinking about basic science again my question was: "Couldn't I take these phenomena that one sees in nature and kind of invent primitives that would describe how they work – in the same kind of way that I succeeded in inventing primitives for this computer language that I built?" I actually thought that this would be a way of getting to a certain point in making fairly traditional physics models of things, and it was only somewhat coincidentally that later I actually did the natural science to explore abstractly what the computer programs that I was setting up did. And the result of that was that I found all sorts of interesting phenomena in CAs and so on.

That got me launched on taking more seriously the idea of just using computation, computer programs, as a way to model lots of things. I think that in terms of the conviction for example that the universe can be represented by computation I certainly haven't proved that yet. I've only become increasingly more convinced that it's plausible – because I did use to say just like everybody else: "Well, all these simple programs and things: they can do this amount of stuff but there'll be this other thing (or things) that they can't do." So, for instance, say, before I worked on the NKS book one of the things that I believed was that while I could make simple programs representing the standard phenomena in physics, that standard phenomena – not at the level of fundamental physics, but at the level of things like how snowflakes are formed, and so on – I somehow believed that, for example, biological systems with adaptation and natural selection would eventually lead to a higher level of complexity that couldn't be captured by these very simple programs that I was looking at. And I kind of realized as a result of working on the NKS book and so on that that's just not true. That's a whole separate discussion, but I kind of believed that that there would be different levels of things that could and couldn't be achieved by simple programs. But as I worked on more and more areas, I got more and more convinced that the richness of what's available easily in the computational universe is sufficiently great that it's much less crazy to think that our actual universe might be made in that kind of way. But we still don't know if that's right until we actually have the final theory so to speak.

3. Is the Universe a Computer?

George Johnson: Very good. And now we have a fairly formal linear exercise for you. Let me preface it by saying that early in the conference there was, I thought, this very interesting exchange between Dr. Toffoli and Seth Lloyd, who's another old Santa Fe friend. And I thought it was funny that he said he had to leave to help his kids carve jack-o-lanterns, because shortly after I met Seth he invited me to his house in Santa Fe that he was renting and he was carving jack-o-lanterns then as well – this was before he was married and had children. Our other jack-o-lantern carver then was Murray Gell-Mann. I was writing a biography of Murray at the time and he had not agreed to cooperate, in fact he was being somehow obstructive, but somehow sitting there with a Nobel prize winner carving jack-o-lanterns helped break the ice.

Seth of course has described his grand vision of the universe as a quantum computer – the universe *is* a quantum computer, he said, and then Dr. Toffoli [had] an interesting subtle objection and basically I think you said that it was this idea of the computational relation between the observer and the system, that somebody has to be looking at it. And later, actually – to skip out to another context during the same session – when Seth made some joke about the election and you said, quoted Stalin saying that it doesn't matter who votes, it's who counts the votes, that kind of seemed to connect back to it and got me thinking about this whole idea of endo-physics of looking at the universe from within or from without. So what I am wondering is this: does Seth's view of looking at the universe not *as* a computer but saying that *it is* a computer, implies a God's eye view of somebody who's watching the computation? What is the universe computing, if it is computing? Is the universe *a* computer, or is the universe *like* a computer– or do the two state the exact same thing?

Tom Toffoli: Are you asking anyone in particular?

George Johnson: Anyone. Please jump in.

Stephen Wolfram: I can think of a few fairly obvious things to say. You know, there's the question of: "What are models?" In other words, if you say, in the traditional physics view of things: "Is the motion of the earth a differential equation?" No, it's not – it's *described* by a differential equation, it isn't a differential equation. And I don't think, maybe other people here think differently about it, but my view of computation as the underlying thing in physics is that computation is a *description* of what happens in physics. There's no particular sense in which it is useful to say the universe is a computer. It's merely something that can be described as a computation[al process] or something that operates according to the rules of some program.

Cristian Calude: I agree and I would add one more idea. When you have a real phenomenon, and a model – a model typically is a simplified version of the reality. In order to judge whether this model is useful or not you have to get some results (using the model). So I would judge the merit of this idea about the universe being modeled faithfully by a gigantic computer primarily by saying: "Please tell me three important facts that this model reveals about the universe that the classical models can't."

Ed Fredkin: I will tell you one in a minute. The thing about a computational process is that we normally think of it as a bunch of bits that are evolving over time plus an engine – the computer. The interesting thing about informational processes (digital ones) is that they're independent of the engine in the sense of what the process is: any engine that is universal (and it's hard to make one that isn't) can – as long as it has enough memory – exactly produce the same informational process as any other. So, if you think that physics is an informational process then you don't have to worry about the design of the engine – because the engine isn't here. In other words, if the universe is an informational process then the engine, if there is one, is somewhere else.

Gregory Chaitin: George you're asking a question which is a basic philosophical question. It's epistemology versus ontology. In other words when you say the Universe looks like a computer, that this is a model that is helpful – that's an epistemological point of view, it helps us to understand, it gives us some knowledge. But a deeper question is what is the universe really. Not just we have a little model, that sort of helps us to understand it, to know things. So, that is a more ambitious question! And the ancient Greeks, the pre-Socratics had wonderful ontological ideas: the world is number, the world is this, the world is that. And fundamental physics also wants to answer ontological questions: "What is the world really built of at the fundamental level?" So, it's true, we very often modestly work with models, but when you start looking at fundamental physics and you make models for that and if a model is very successful – you start to think [that] the model *is* the reality. That this is really an ontological step forward. And I think modern philosophy doesn't believe in metaphysics and it certainly doesn't believe in ontology. It's become unfashionable. They just look at epistomological questions or language, but mathematicians and physicists we still care about the hard ontological question: If you're doing fundamental physics you *are* looking for the ultimate reality, you *are* working on ontology. And a lot of the work of lots of physicists nowadays really resembles metaphysics– when you start talking about all possible worlds, like Max Tegmark does or many other people do, or David Deutsch for that matter. So philosophers have become [gotten] very timid, but some of us here, we are continuing in the tradition of the pre-Socratics.

Ed Fredkin: The problem – one problem – we have is the cosmogeny problem which is the origin of the universe. We have these two sort of contradictory facts: one is [that] we have a lot of conservation laws, in particular mass energy is conserved and we have the observation that the universe began, it seems, with a big bang not so long ago, you know, just thirteen or fourteen billion years ago. Well, basically there's a contradiction in those two statements: because if something began and you have a conservation law where did everything come from, or how did it come about? And also the idea that it began at some time is problematic – the laws of physics can't help us right there because, if you think a lot about of those details it's not so much why matter suddenly appeared but why is there physics and why this physics and stuff like that.

And there is a way[11] to wave your hands, and come up with a kind of answer that isn't very satisfying. Which is that you have to imagine that there is some other place – I just call it 'other' – and in this other place for whatever reason there is an engine and that engine runs an informational process. And one can actually come to some kind of feeble conclusions about this other place, because there are some numbers that we can state about how big that computer must be. In other words if we ask the question what would it take to run a computer that emulated our universe, well – we can guess some quantitative numbers if we can figure out a few things, and so you can make a few statements [educated guesses] about that other kind of place. The point about 'other' is that it is a place that doesn't need to have conservation laws, it is a place that doesn't need to have concepts such as 'beginnings' and 'ends' – so there aren't that many constraints. And one of the wonderful things about computation is that it is one of the least demanding concepts, if you say: well, what do you need in order to have a computational engine? Well, you need a space of some kind. What kind of space? How many dimensions does it have to have. Well, it could have three, it could have two, it could have one, it could have seven – it doesn't matter. They can all do the same computations. Of course, if you have a one-dimensional [space] you can spend a lot of time overcoming that handicap. But does it need the laws of physics as we know them? No, you don't need to have the laws of physics as we know them. In fact, the requirements are so minimal for having a computation compared to the wonderful rich physics we have that it's very, very simple. I am, in fact, reminded of a science-fiction story, by a Polish author where there's a robot that could make everything that started with the letter 'n' and in Polish, like in Russian, or in English, the word 'nothing' starts with an 'n' so someone bored said: "OK, make nothing." And the robot started working and where the sky had previously been white with so many stars and galaxies just minutes earlier, it slowly started to fade away, little by little, galaxy by galaxy. Admittedly, it's just a science fiction story, but the point is that one could even inquire as to what would be the motivations to create an emulation like this? You can imagine that there is some question, and [then] one needs to think about: what could the question be?

We can speculate about that. But the point is that this is an explanation

[11] `http://en.wikipedia.org/wiki/Digital_philosophy#Fredkin.27s_ideas_on_physics`

that says: well there's this thing called 'other' that we don't know anything about – as opposed to all other explanations that imply that some kind of magic happened. Well: I don't like magic, myself.

Rob deRuyter: A couple of sentences from a biological perspective: Let's take a naive standpoint that there's a world out there and that there's a brain and this brain needs to understand what's happening in the world, what's going around and unfortunately, maybe – or fortunately for us – this brain is an engine that is really well adapted to information processing in the savannah, or in the trees, or wherever. I don't think that the brain itself is a universal computational engine – at least I don't see it that way – but it's a device that is extremely well adapted to processing information that comes to our sensory organs, in from the world that we happen to inhabit. So if we want to start thinking about more complex things or deeper things we need to develop tools that allow (or that help) us translate our thoughts about the phenomena that we observe in the world. Or the other way around: just like we developed hammers and pliers, in dealing with phenomena in the world we need to develop tools to think [about them]. I have no idea of what the limitations that the structure of our brain are, and that impose – I mean, there must be limitations (in the way we think) that impose structure on those tools that we develop to help think about things. But computation, I think, in a sense, is one of the tools [that] we tried to develop in dealing with the world around us. And what computation allows us to do, like mathematics, is to be able to develop long chains of reasoning that we normally (don't use, I think) but that we can extend to reason about very long series, sequences of complicated observations about phenomena [events] in the world around us. So what interests me is this relationship between the way we think – and the way we have evolved to think about the world around us – and the things that we think about now in terms of scientific observations and [thoughts about] the origin of the universe. To what extent does the hardware that we carry around inform and determine the kinds of tools and computations and the strategies that we're using?

Tom Toffoli: I would like to give some examples to illustrate why the question of whether the universe is a computer, is a really hard question. In some sense it resembles the question of what is life. Let's take for example the concept of randomness: say you buy a random number generator and you start producing numbers with it. First number you get out is 13, then you get 10, 17, and so on – and then you start asking yourself about the numbers that you obtained: how random are they? What is random? Is 10 a random number? Is 13 a random number? How about 199999 – is it a random number? Then you realize that, of course, randomness is not a property of the number, it is a property of the process. You pay for a random number generator because you want to be surprised. You want not to know what will come out. If you knew it – it would not be random to you because it would be perfectly predictable. So we use that term 'a random number' as an abbreviation for whatever [sequence] is produced by a random number generator. I'll give you another example: somebody tried to trademark icons. When they were invented icons were small: 16 by 16 pixels black and white, and you can draw, you know, some simple things with those 16 by 16 bits. So you may want to trademark them. And some greedy businessman said: look I will go to the judge and I will try to I trademark *the entire set* of 16 by 16 pixel set (icons,) all of them, and everybody has to pay me. So now if you are the judge and you have to decide whether you can or cannot allow to someone the right to trademark not just one icon, but all of the icons that can be made that way. And if you are the judge you really have two ways: you say (a) either you have to give me a reason why this icon is really something interesting or (b) you pay 5 cents for each one of the icons that you register and being that there are 256 items you know, you don't have to exercise any judgment, it would just turn into a big contribution to the comunity.

In other words, the question is: what makes an icon what it is? Is it the fact that it is 16 by 16 bits or that you have reason to believe that there is something useful [in it]? Brian Hayes, whom you mentioned a moment ago, once said: "What surprises me is that most people don't use the computer for what makes it unique and powerful – which is that it is a programmable machine." My partial definition of a computer is: something that can compute (evaluate) a lot of different functions. If it can evaluate just one function, then I wouldn't call it a computer I would call it a special purpose machine or whatever it is. So we may get the surprise that if we find the formula that gives us the universe as a computer, then at that very point the universe itself becomes a special purpose machine. I mean, we know the formula[, we know the machine,] we know the initial conditions, and we just go: tick, tick, tick, tick. And it's the largest computer but nobody can program it – if this is the universe, we cannot program it because we are inside[12] of it.

So, the definition of a computer is a bit like the definition of life and the definition of evolution or being adaptive: if there isn't a component of adaptiveness, I wouldn't call the thing a computer. Now the thing can be formalized better, I just said it in an intuitive way, but I'm asking some of these questions to try to clarify a bit what exactly it was that we wanted.

Gregory Chaitin: Look, I'd like to be aggresive about this – the best way for me to think about something is to make claims that are much too strong (at least it brings out the idea). So the universe *has to* be a computer, as Stephen said, because the only way to understand something is to program it. I myself use the same paradigm. Every time I try to understand something the way I do it, is: I write a computer program. So the only possible working model of the universe has to be a computer – a computational model. I say a working model because that's the only way we can understand something: by writing a program, and getting it to work and debugging it. And then trying to run it on examples and such. So you say that you understand something only if you can program it. Now what if the universe decides however that it's not a

[12] First off the universe is calculating all possible states, the easiest way to see this is to consider Everett's interpretation of quantum mechanics. Furthermore a nuclear explosion is a very crude way of programming the universe from the inside, any explosion is. Programs can modify themselves too, erase themselves from memory. Wolfram's concept of free will.

– that you *can't* do a computational model about it. Well, then: no problem. It just means we used the wrong computers. You know, if this universe is more powerful than a computer model of it can be, that means that our notion of what a computer is is wrong and we just need a notion of computer that is more powerful, and then things are in sync. And by the way, there is a way to define the randomness of individual numbers based on a computer (we hear: infinite ones, from Toffoli) well, anyway, but that's another issue[13].

Stephen Wolfram[14]: One point to make, in relation to using computation as a model of a universe: we're used to a particular thing happening when we do modeling, we're used to models being idealizations of things. We say we're going to have a model of a snowflake or a brain or something like that, we don't imagine that we're going to make a *perfect* model! It's a very unusual case that we're dealing with in modeling fundamental physics (perhaps modeling isn't the right term, because what we imagine is that we're going to actually have a precise model that reproduces our actual universe in every detail.) It's not the same kind of thing as has been the tradition of modeling in natural science, it's much more. So when you say, when you talk about what runs it and so on, it's much more like talking about mathematics: you wouldn't ask when you think about a mathematical result, [if you] work out some results in number theory, for example, one wouldn't be asking all the time "Well, what's *running* all these numbers?" It's just not a question that comes up when one is dealing with something where [what you have] is a precise model of things.

One other point to make regarding the question of to what extent our efforts of modeling relate to what our brains are good at doing and so on. One of the things I am curious about is: if it turns out to be the case that we can find a precise representation, a new representation (better word than model) for our universe in terms of a simple program and we find that it's, you know, program number such and such – what do we conclude in that moment? It's a funny scientific situation, it kinds of reminds one of a couple of previous scientific situations, like for instance Newton was talking about working out the orbits of planets and so on and made the statement that once the planets are put in their orbits then we can use the laws of gravity and so on to work out what would happen – but he couldn't imagine what would have set the planets originally in motion in their orbit in the first place. So he said, "Well, the Hand of God must have originally put the planets in motion. And we can only with our science figure out what happens after that." And it's the same with Darwin's theories: once we have life happening then natural selection will lead us inexorably to all the things that we see in biology. But how to cause life in the first place – he couldn't imagine. So some of the things I'd be curious about would be: if in fact we do come up with a precise representation of the universe as a simple program – what do we do then and can we imagine what kind of a conclusion we

[13]Not to confuse the number with its representation: 3^{100} looks random in base 2 but not very random in base 3. Same number, different representations.

[14]Tony Leggett has been signaling and waiting for the microphone for some time now, Gerardo points out. Stephen starts, Tony Leggett to follow.

can come to, about why this program and not another program and so on? So one of the possibilities would be that we find out that it's, you know, program number 1074 or whatever it is. The fact that it is such a small number might be a consequence of the fact that our brains are set up because they are made from this universe in such a way that it is inevitable, and in [all] the enumerations that we [might] use our universe will turn out to be a small number universe. I don't think that's the case – but that's one of those self fulfilling prophecies: because we exist in our universe our universe will have to have laws that will somehow seem simple [and intuitive] to us. I think it's more clear than that, but that's one of the potential resolutions of this question: so now we have our representation of the universe, what do we conclude metaphysically from the fact that it is this particular universe [representation] and not the other one.

George Johnson: Dr. Leggett?

Anthony J. Leggett[15]: I think with regard to the strong and forceful statement that could be related to Seth Lloyd's argument namely that the universe is a computer, I just have a little very naive and simple question: it seems to me that if a statement is called for then the converse is not called for. So my question to Seth Lloyd is: "What would it be like for the universe *not* to be a computer?" And so far I fear I don't find that particular statement truly helpful. I do find quite helpful the thesis that it may be useful to look at the universe in the particular framework of computational science and to ask different questions about [it, although] I have to say that I'm not yet convinced that looking at it in this way is going to help us to answer some very obvious questions, some of which [are usually met with a] certain amount of friction, namely is the anthropic principle physically meaningful? That is, why do the constants[16] of nature as we know them have the particular values[/relevance] that they have? Is it perhaps for some arbitrary reason, or maybe for some other deeper reason. Now, of course, there have been plenty of arguments and speculations [here] about [all sorts of things but] as far as I can see [they don't seem to give any] direct relationship (of the universe with a specific computational model, or the universe as a computer.) But I would really like to hear a plausible argument as to why this point of view takes us further on these types of questions [that I just mentioned.]

Ed Fredkin: I want to react a little bit to what Stephen was saying. There exist areas where we use computers to write programs that are *exact* models, exact and perfect in every possible way – and that is when you design a program to emulate another computer. This is done all the time both for writing a trace program [map and] for debugging, where you write an emulator for the computer that the software is running on. Or you want to run software that is for another computer like the Mac did when it switched CPUs from the 68000 to the PowerPC: they made an emulator, which is an exact implementor of the software on another computer.

[15] Has been waiting patiently since mid-Chaitin or so.

[16] `http://en.wikipedia.org/wiki/Anthropic_principle`

This relates to something that I used to call 'The Tyranny of Universality.' Which is: "Gee, we can never understand the design of the computer that runs physics since any universal computer can do it." In other words if there's a digital computer running all physics of course then any computer can do it, but then, after convincing myself that that point of view made sense a long time later I came up with a different perspective: that if the process that runs physics is digital and it is some kind of CA there will exist a model that's one to one onto in terms of how it operates. And it would probably be possible to find in essence the simplest such model so that if some kind of experimental evidence showed us that physics is some kind of discrete, digital [physical] process like a CA I believe we should be able to find the exact process (or, you know, one of a small set of processes) that implement it exactly.

4. Is the Universe Discrete or Continuous?

George Johnson: Maybe a good way to get to Tony Leggett's question that he raised at the end is just to ask the same question that our host Adrian asked after we saw that brief film [that he showed us] first thing this morning which is: is there a fundamental difference between a computational physics, or a computational model, or emulation of the universe and quantum mechanics? Or is there a fundamental distinction between a discrete and a continuous physics? Does anyone have a reaction to that?

Stephen Wolfram: (speaking to George Johnson): So if you're asking is there some definitive test for whether the universe is somehow discrete or somehow fundamentally continuous...

George Johnson: Yeah – if there's a conflict that [it] could possibly be both at a deeper level.

Stephen Wolfram: If you're asking for that – for example in the kinds of models that I made some effort to study, there are so many different ways to formulate these models that this question "Is it discrete [or] is it continuous?" becomes kind of bizarre. I mean, you could say: we'll represent it in some algebraic form [in which] it looks like it's talking about these very continuous objects – and what matters about it may yet turn out to be discrete, it may turn out to be a discrete representation (which is much easier to deal with). So I think that at the level of models that I consider plausible the distinction between continuity and discreteness is much less clear than we expect. I mean, if you ask this question I think you end up asking 'very non-physics questions' like, for example, how much information can in principle be in this volume of space. I'm not sure that without operationalizing that question that it's a terribly interesting or meaningful question.

Tom Toffoli: I would like to say something that will be very brief. Look at CAs, they seem to be a paradigm for discreteness. But as it turns out one of the characterizations of CAs is that they are a dynamical system [that perform certain kinds of translations] and they are continuous with respect to a certain topology which is identical to the Cantor set topology, continuous in exactly that very sense of that definition of continuity that is studied in freshman calculus. But the interesting thing is that this is the Cantor set topology, invented by Cantor (the one with the interval where you remove the middle third, etc.) And as it turns out, this topology for CAs is – not in the sense of geometrical topology, but in the sense of set topology of circuits with gates that have a finite number of inputs and a finite number of outcomes [outputs] – that is exactly the topology of the Cantor set, so it's sort of a universal topology for computation. And so we come full circle that (a) something that was not invented by Cantor to describe computers in fact represents the natural topology to describe discrete computers and (b) the moment you take on an in[de]finite lattice then you have continuity exactly the kind of continuity, continuity of state, of the dynamics that you get when you study continuous functions. So these are some of the surprises that one gets by working on things.

Stephen Wolfram: I just want to say with respect to the question of how do we tell, you know, the sort of thing that Tony is saying: "How do we tell that this is [not] all just complete nonsense?" Right?

Tony Leggett: Refutation of the negative statement.

Stephen Wolfram: Yes, right. We're really only going to know for sure if and when we finally get a theory, a representation that is the universe and that can be represented conveniently in computational form. Then people will say: "Great! This computational idea is right, it was obvious all along, everybody's thought about it for millions of years..."

Tom Toffoli: At that point they will probably say: "In fact it's trivial!"

Stephen Wolfram (agrees laughing, everybody laughs): And I think that until that time one could argue back and forth forever about what's more plausible than what and it's always going to be difficult to decide it based on just that. Yet these things tend to be decided in science in a surprisingly sociological way. For example the fact that people would seriously imagine that aspects of string theory should be taken seriously as ways to [model the reality of a physical universe] it's – it's interesting and it's great mathematics – but it's a[n interesting] sociological phenomenon that causes [or forces] that to be taken seriously at the expense of other kinds of approaches. And it's a matter of history that the approach we're using (computational ideas) isn't the dominant theme in thinking about physics right now. I think it's purely a matter of history. It could be that in place of string theory people could be studying all kinds of bizarre CAs or network systems or whatever else and weaving the same kind of elaborate mathematical type web that's been done in string theory and be as convinced as the string theorists are that they're on to the right thing. I think at this stage until one has the definitive answer, one simply doesn't know enough to be able to say anything with certainty and it's really a purely sociological thing whether we can say that this is the right direction or this isn't the right direction. It's very similar actually to the AI type thing, people will argue forever and ever about whether it's possible to have an AI and so on – and some of us are actually putting a lot of effort into trying to do practical things that might be identified as relevant to that. I think that actually the AI question is harder to decide than the physics question. Because in the physics case once we'll have it it's likely (it seems to me) that we'll be able to show that the representation is of the universe that is obviously the actual universe and the question will be closed. Whereas the question of AI will be harder to close.

Cristian Calude: Apparently there is an antagonism between the discrete and continuous view. But if we look at mathematics there are mathematical universes in which discrete and continuous are co-existing. Of course, what Tom said, the Cantor space is a very interesting example, but it might be too simple for the problem that we are discussing. For instance, non-standard analysis is another universe where you find [this same phenomenon] and you have discreteness and you have continuity and maybe, to the extents that mathematics can say something about the physical universe, it could be just a blend of continuous and discreteness and some phenomena may be revealed through discreteness and some others will be revealed through continuity and continuous functions.

Gregory Chaitin: Again I am going to exaggerate – on purpose. I think the question is like this: discreteness vs. continuity. And I'm going to say why I am on the side of discreteness.

The reason is this: I want the world to be comprehensible! Now there are various ways of saying this. One way would be: God would not create a world that we couldn't understand. Or everything happens for a reason (the principle of sufficient reason). And other ways. So I guess I qualify as a neo-Pythagorean because I think the world is more beautiful, [if it] is more comprehensible. We are thinkers, we are rationalists – we're not mystics. A mystic is a person that gets in a communion with an incomprehensible world and feels some kind of [comm]unity and is able to relate to it. But we want to understand rationally so the best universe is one that can be completely understood and if the universe is discrete we can understand it – it seems to me. This is something that you said, at one point, Ed – it is absolutely totally understandable because you run the model and the model is exactly what is happening.

Now a universe which uses continuity is a universe where no equation is exact, right? Because we only have approximations up to a certain order. So I would also say: a universe would be more beautiful if it were discrete! And although we now end up in aesthetics, which is even more complicated, I would still say that a discrete universe is more beautiful, a greater work of art for God to create – and I'm not religious, by the way. But I think it's a very good metaphor to use – or maybe I am religious in some sense, who knows?

Another way to put it is let's say this universe does have continuity and messy infinite precision and everything – well, too bad for it. Why didn't God create as beautiful a universe as he could have?

Tom Toffoli: He should have asked you, Greg!

Gregory Chaitin (laughs, everybody laugs): What? ... No... No ... maybe at this point I think Stephen is the leading candidate for coming up with a ... [everybody is still laughing, including Chaitin who continues] ... so that would be more beautiful it seems to me. You see, it would be more understandable it would be more rational it would show the power of reason. Now maybe reason is a mistake as may be to postulate that the universe is comprehensible – either as a fundamental postulate or because [you know] God is perfect and good and would not create such a universe, if you want to take an ancient theological view. Maybe it's all a mistake, but this one of the reasons that I'm a neo-Pythagorean, because I think that would be a more beautiful, or comprehensible universe.

Stephen Wolfram: I have a more pragmatic point of view, which is that if the universe is something that can be represented by [something like] a simple discrete program, then it's realistic to believe that we can just find it by searching for it. And it would be embarrassing if the universe would indeed be out there in the first, you know, billion universes that we can find by enumeration and we never bothered to even look for it. [There's a very sustained reaction from the rest of the round table members, especially Greg Chaitin, whom we hear laughing.] It may turn out that, you know, the universe isn't findable that way – but we haven't excluded that yet! And that's the stage we're at, right now. Maybe in, you know, 10-20-50 years we will be able to say: yes we looked at all the first I don't know how many – it will be like looking for counterexamples of the Riemann hypothesis, or something like that – and we'll say that we've looked at the first quadrillion possible universes and none of them is our actual universe, so we're beginning to lose confidence that this approach is going to work. But [right now] we're not even at the basic stage of that yet.

George Johnson wants to give the microphone to Tony Leggett. Ed Fredkin says: "Just one comment!" indicating that his comment is short.

Ed Fredkin : If this were a one dimensional universe, Steve, you would have found the rule by now, right? Because you've explored all of them...

George Johnson: Tony!

Tony Leggett: Well, George, I think, raised the question whether quantum mechanics has any relevance to this question, so let me just comment briefly on that. I think if one thinks about the general structure of quantum mechanics, and the ways in which we verify its predictions, you come to the conclusion that almost all the experiments (and I'll stick my neck out and say *all* the experiments) which have really shown us interesting things about quantum mechanics do measure discrete variables in fact. Experiments on the so-called macroscopic quantum coherence, experiments on Bell's theorem, and so forth – they all basically use discrete variables in practice. Now of course the formalism of quantum mechanics is a continuous formalism. You allow amplitudes to have arbitrary values, but you never really measure those things. And I think that all one can say when one does sometimes measure things like position and momentum which are [apparently] continuous variables – if you look at it hard you'll see that the actual operational setup is such that you are really measuring discrete things. So measurements within the framework of quantum mechanics which claim to be of continuous variables usually are of discrete variables. So I think from the, as it were, the ontological point of view, one can say that quantum mechanics does favor a discrete point of view.

George Johnson: Well, I think we're supposed to take a break now then we'll jump right in. So we'll have a fifteen minute break.

5. Is the Universe Random?

George Johnson: First, just a technical matter, I've been asked by the people recording this to remind you to speak closer into the microphone. And, also, a procedural announcement: in the last half hour which will be starting at about 11:45 we'll take questions from the audience, so if you can – start thinking about the questions you wanted to ask about the conference so far.

When Hector, Gerardo and I were talking about good questions that would stimulate debate we thought that perhaps we should ask something that would be really really basic about randomness – and the great thing about being a journalist particularly a science journalist is that you get this license of asking really really smart people questions about things that have been puzzling you. And this is something that has always kind of bugged me – the SETI (Search for ExtraTerrestrial Intelligence) where we get these signals from space which are then analyzed by these computers, both by super computers and by SETI at home, where you donate some CPU time on your PC, computer cycles etc. They're looking for some structure in what appears to be random noise. And I was wondering we're getting a signal that seems to be pure noise but to some extent – as I think Tomasso Toffoli has suggested – perhaps the randomness is only in the eye of the beholder.

If, for example we're getting this noisy signal that just seems to be static – how do we know we're not getting the ten billionth and fifty seventh digit of the expansion of π forward? How do we know that we're not getting line ten trillion quatrillion stage forward of the computation of the rule 30 automata? So I am wondering if you can help me with that.

Tom Toffoli: This is not an answer. It's just something to capture the imagination. Suppose that people are serious about computing and they say: "Look: you're not using your energy efficiently because you're letting some energy – that has not completely degraded – out." So they start to make better recirculators, filters and so on and now whatever thermal energy comes out is as thermalized as possible. Because if it's not, they would have commited a thermodynamical sin. But this is exactly what happens when you look at the stars. They are just sending close to thermodynamical equilibrium a certain temperature – so you can say well probably then this is prima facie evidence that that there are people there computing and they are computing so efficiently that they are just throwing away garbage, they're not throwing away something that is still recyclable! And this could be an explanation as to why we see all these stars with all these temperatures.

Stephen Wolfram: You know I think the question about SETI and how it relates to the type of things we're talking about – I think it gets us into lots of interesting things. I used to be a big SETI enthusiast and because I'm a practical guy I was thinking years ago about how you could make use of unused communication satellites and use them to actually detect signals and so on. And now I have worked on the NKS book for a long time, and thought about the PCE and I have became a deep SETI non-enthusiast. Because what I realized is that it goes along with statements like "the weather has a mind of its own". There's this question of what would constitute – you know, when we say that we're looking for extra terrestrial intelligence – what actually is the abstract version of intelligence? It's similar to the old question about what is life, and can we have an abstract definition of life that's divorced from our particular experience with life on the earth. I mean, on the earth it is pretty easy to tell whether something – reasonably easy to tell whether something – is alive or not. Because if it's alive it probably has RNA it has some membranes it has all kinds of historical detail that connects it to all the other life that we know about. But if you say, abstractly: what is life? It is not clear what the answer is. At times, in antiquity it was that things that can move themselves are alive. Later on it was that things that can do thermodynamics in a different way than other things do thermodynamics are alive. But we still – we don't have – it's not clear what the abstract definition of life is divorced from the particular history. I think the same is true with intelligence. The one thing that most people would (I think) agree with – is that to be intelligent you must do some computation. And with this principle of computational equivalence idea what one is saying is that there are lots of things out there that are equivalent in the kind of computation that they can do ...

Tom Toffoli (who is seated next to Wolfram, and has a microphone): But you can also do computation without being intelligent! So ... it's a similar ... (everybody laughs)

Stephen Wolfram (replying to Toffoli): Well[, so] that's precisely the question: can you – what is the difference, what is the distinctive feature of intelligence? If we look at history, it's a very confusing picture: a famous example that I like is when Marconi had developed radio and (he had a yacht that he used to ply the Atlantic with, and) at one point he was in the middle of the Atlantic and he had this radio mast – because that was the business that he was in – and he could hear these funny sounds, you know: ... wooo ... oooeeo ... eeooo ... woooo ... this kind of sounds out in the middle of the Atlantic. So what do you think he concluded that these sounds were? He concluded that they must be radio signals from the martians! Tesla, was in fact more convinced that they were radio signals from the martians. But, what were they in fact? They were in fact some modes of the ionosphere on the earth, they were physical processes that – you know, something that happens in the plasma. So the question was, how do you distinguish the genuinely intelligent thing, if there is some notion of that, from the thing that is the ... the computational thing that is.

The same thing happened with pulsars, when the first pulsars were discovered. In the first days of discovery it seemed like this periodic millisecond thing must be some extraterrestrial beacon. And then it seemed like it was too simple. We [now] think it's too simple to be of intelligent origin. It also relates to this question about the anthropic principle and the question of whether our universe is somehow uniquely set up to be capable of supporting intelligence like us. When we realize that there isn't an abstract definition of [actual] intelligence, it is (as I think) just a matter of doing computation. Then the space of possible universes that support something like intelligence becomes vastly broader and we kind of realize that this notion of an anthropic principle with all these detailed constraints – just doesn't make much sense.

There's so much more we can say about this, and I'll let others do so.

George Johnson: Randomness: is it in the eye of the beholder?

Greg Chaitin: George, I suppose it would be cowardly of me not to defend the definition of randomness that I have worked on all my life, but I think it is more fun to say (I was defending rationalism, you know) that a world is more understandable because it is discrete, and for that reason it is more beautiful. But in fact I've spent my life, my professional life, working on a definition of randomness and trying to find, and I think I have found, randomness in pure mathematics which is a funny place to find something that is random. When you say that something is random you're saying that you can't understand it, right? So defining randomness is the rational mind trying to find its own limits, because to give a rational definition [to randomness is odd] there's something paradoxical in being able to know *that*, you know, being able to define randomness, or being able to know that something is random – because something is random [when] it escapes ... I'm not formulating this well, [actually] improvising it, but there are some paradoxes involved in that. The way it works out in these paradoxes is that you can define randomness but you can't know that something is random, because if you could know that something is random then it wouldn't be random. Randomness would just be a property like any others, and it could be used [would enable you] to classify things. But I do think that there is a definition of randomness for individual numbers and you don't take into account the process by which the numbers are coming to you: you can look at individual strings of bits – base 10 number – and you can at least mathematically say what it means for this to be random. Now, although most numbers or most sequences of bits are random according to this definition, the paradoxical thing about it is that you can never be sure that one individual number is random – so I think it is possible to define a notion of randomness which is intrinsic and structural[17] and doesn't depend on the process from which something comes but there is a big problem with this definition which is: it's useless. Except to create a paradox

[17]Maybe Chaitin means 'representation of a number' instead. 2^{10000} does not look random in base 2. It does look random in base 3. The number is the same. Its representation isn't. The representation is a process (just as Toffoli said).

or except to show limits to knowledge, or limits to mathematical reason. But I think that's fun, so that's what I've been doing my whole life.

So I don't know if this is relevant to SETI? I guess it is, because if something looks random it then follows that it probably doesn't come from an intelligent source. But what if these superior beings remove redundancy from their messages? They just run it through a compression algorithm because they are sending us enormous messages, they're sending us all their knowledge and wisdom and philosophy everything they know in philosophy, because their star is about to go nova, so this is an enormous text encompassing all their accomplishments of their thinking and civilization – so obviously they think any intelligent mind would take this information and compress it, right? And the problem is, we're getting this LZD compressed message and we think that it's random noise and in fact it's this wonderfully compact message encapsulating the wisdom and the legacy of this great civilization?

George Johnson: Oh, but do they include the compression algorithm?

Gregory Chaitin: Well, they might think that a priori this is the only conceivable compression algorithm, that it is so simple that any intelligent being would use this compression algorithm – I don't know ... (people laughing)

Stephen Wolfram: I think it's an interesting question – about SETI. For example, if you imagine that there was a sufficiently advanced civilization that it could move stars around, there's an interesting kind of question: in what configuration would the stars be moved around and how would you know that there is evidence of intelligence moving the stars around? And there's a nice philosophical quote from Kant who said "if you see a nice hexagon drawn in the sand you know that it must come from some sort of intelligent entity [that has created it]" And I think that it's particularly charming that now in the last few years it's become clear that there are these places in the world where there are hexagonal arrangements of stones that have formed and it is now known that there is a physical process that has causes a hexagonal arrangement of stones to be formed. That's sort of a charming version of this ... (Ed Fredkin starts).

Ed Fredkin: One of the poles of Saturn has this beautiful hexagon – at the pole and we have pictures of them.

Stephen Wolfram (continues): ... right, so the question is what do you have to see to believe that we have evidence that there was an intention, that there was a purpose. It's just like Gauss for example, [he] had the scheme of carving out in the Syberian forest the picture of the Pythagorean theorem, because that would be the thing that would reveal the intelligence. And if you look at the Earth now a good question to ask an astronaut is: "What do you see on the Earth that makes you know that there is some sort of a civilization?" And I know the answer: the thing that is most obvious to the astronauts is – two things, OK? One is: in the great salt lake in Utah [there is] a causeway that divides a region which has one kind of algae which tend to be of orangeish color, from a region that has another kind of algae that tend to be bluish, and there's a straight line that goes between these two colored bodies of water. It's perfectly straight and that is thing number one. Thing number two is in New Zealand. There's a perfect circle that is visible in New Zealand [from the space]. I was working on the NKS book and I was going to write a note about this particularly thing. We contacted them, this was before the web was as developed as it is today, so we contacted the New Zealand Geological Survey to get some information about this perfect circle and they said: "If you are writing a geology book (the circle is around a volcano,) please *do not* write that this volcano produces this perfect circle, because it isn't true." What's actually true is that there is a national park that was circumscribed around the volcano and it happens to be perfectly circular and there are sheep that have grazed inside the national park but not outside, so it's a human produced circle. But it's interesting to see what is there on the Earth that sort of reveals the intelligence of its source.

And actually, just to make one further comment about randomness, and integers – just to address this whole idea of whether there are random integers or not, and does it matter, and how can you tell, and so on – we have a little project called 'integer base' which basically is a directory of integers. And the question is to find is the simplest program that makes each of these integers. And it's interesting, it's a very pragmatical thing [project] actually trying to fill in actual programs that make integers. We have to have some kind of metric as to what counts as simple[18]. When you use different kinds of mathematical functions, you use different kinds of programming constructs, you actually have to concretely decide, [measure, quantify] simplicity. And there are lots of ways, for example: how many times does this function appear [is referenced] on the web, that could be a criterion as to how much weight should be given; or how long is this function's name in Mathematica; or other kinds of criteria like that. So it's kind of a very concrete version of this question about random integers.

Tom Toffoli: I'd like to say something that throws out another corollary. There's this self-appointed guru of electronics, Don Lancaster, he's very well known in circles and he said something [that is] very true. He said: the worst thing that could happen to humanity is to find an energy source that is inexhaustible and free. You know, we are all hoping that we will find something like that, but if we found it it would be a disaster, because then the Earth would be turned into a cinder in no time.

If you don't have any of the current constraints it can turn very dangerous. For example, you have a house there, and you have a mountain, and in the afternoon you would like to have sunshine. And in the morning you would like to have shading from the cold or whatever, so if energy is free you just take the mountain from where it is in the morning you take it away and you plant it back in the evening. And this is what we're doing in essence when we're commuting from the suburbs to the center of Boston. You see this river of cars that is rushing in every day, and rushing out every day, with energy that is costing quite a bit. Imagine if it was completely free. So, again, let's try to think up what the answer to our question would be if we really got it and then see the

[18]Like in genetic algorithms, or genetic programming.

consequences of that [first].

And, again: this comment is in relation to what I said earlier about the randomness of the stars i[f it']s an indication of a super-intelligence, or super-stupidity. Who knows, it could be [that they're one and] the same thing?

Rob deRuyter: Just to take your question completely literally: there's a lot of randomness in our eyes, as we look around the world. And especially outside in moonlight conditions there are photons flying around, but they are not that many and we are very aware of [the fact that] information that we're getting into our visual system, information that we have to process in order to navigate successfully, is of low quality and the interesting thing is that we as organisms are used to walking around in a world that is random and we're very conscious of it. Yesterday I spoke about how flies cope with this – we cope with this too and we cope with it at all levels, from adaptation of [in] photoreceptors in our eyes to the adaptation in the computational algorithms that our brain is using [or executes] to where you are in an environment where you are subject to large levels of noise, because there are not that many photons around. [In that case] you tend to move very cautiously – you don't start running, unless maybe the tiger is just following you, but that is a very rare situation – so, I think, in a lot of senses we're used to the measurements that our sensors make being more or less random depending on how the situation is at the moment. And so as computational engines we are very well aware of that.

Cristian Calude: I am interested in the quality of quantum randomness. We were able to prove that quantum randomness, under some mild assumptions on the quantum model of physics we agree on, is not computable. So this means no Turing machine can reproduce the outcome of a quantum generated sequence of bits (finitely many) and this gives you a weak form of relation between Greg's theory, Greg's definition of algorithmic randomness and what one would consider to be the best possible source of randomness in this this universe, i.e., quantum randomness. And one of the things that is delicate as we are thinking and experimenting is a way to distinguish quantum randomness from Mathematica generated randomness. Is it possible, by using finitely many well-chosen tests, to find a mark of this distinction you know between something that is computer [computably] generated from something that is not generated [in that way]?

So whereas here we have some information about the source, you know, like Tom said – we know very well that there is an asymptotic definition of the way that these bits can be generated – it is still very difficult to account in a finite amount of tests for that difference.

Ed Fredkin: Just a funny story about random numbers: in the early days of computers people wanted to have random numbers for Monte Carlo simulations and stuff like that and so a great big wonderful computer was being designed at MIT's Lincoln laboratory. It was the largest fastest computer in the world called TX2 and was to have every bell and whistle possible: a display screen that was very fancy and stuff like that. And they decided they were going to solve the random number problem, so they included a register that always yielded a random number; this was really done carefully with radioactive material and Geiger counters, and so on. And so whenever you read this register you got a truly random number, and they thought: “This is a great advance in random numbers for computers!” But the experience was contrary to their expectations! Which was [that] it turned into a great disaster and everyone ended up hating it: no one writing a program could debug it, because it never ran the same way twice, so ... This was a bit of an exaggeration, but [as a result] everybody decided that the random number generators of the traditional kind, i.e., shift register sequence generated type and so on, were much better. So that idea got abandoned, and I don’t think it has ever reappeared.

Stephen Wolfram: Actually it has reappeared, in the current generation of Pentium chips there’s a hardware random generator that’s based on double Johnson noise in the resistor. But in those days programs could be run on their own. In these days there are problems in that programs can no longer be run on their own: they are accessing the web, they’re doing all sorts of things, essentially producing random noise from the outside, not from quantum mechanics but they’re producing random noise from the outside world. So the same problem has come up again.

George Johnson: This reminds me that in the dark ages before Mathematica the RAND corporation published this huge tome – I found a reprint of it called “One hundred thousand random numbers” (someone corrects, Calude I think, George Johnson repeats the correction out loud: “One *million* random numbers”) in case you needed some random numbers – and Murray Gell-Mann used to tell a story that he was working at RAND and at one time, I think when they were working on the book, they had to print an errata sheet! (There is laughter, but Wolfram quickly wants to make a point.)

Stephen Wolfram: Well, the story behind the erratum sheet I think is interesting because those numbers were generated from (I think a triode, or something) some vacuum tube device and the problem was that when they first generated the numbers, they tried to do it too quickly, and basically didn’t wait for the junk in the triode to clear out between one bit being found and the next bit being found. This is exactly the same cause of difficulty in randomness that you get in trying to get perfect randomness from radioactive decay! I think the null experiment for quantum computing, one that’s perhaps interesting to talk about here, is this question of how do you get – I mean, can you get – a perfect sequence of random bits from a quantum device? What’s involved in doing that? And, you know, my suspicion is the following: my suspicion would be that every time you get a bit out you have to go from the quantum level up to the measured classical level, you have to kind of spread the information about this bit out in this bowl of thermodynamic soup of stuff so that you get a definite measurement; and the contention, or my guess, would be that there is a rate at which that spreading can happen and [...] that in the end you won’t get out bits that are any more random than the randomness that you could have got out just through the spreading process alone without kind of the little quantum seed. So that's an extreme point of view, that the extra little piece (bit) of quantumness doesn't really add anything to your ability to get out random bits. I don't know if that is correct but you know I, at least a long time ago, I did try looking at some experiments and the typical thing that's found is that you try to get random bits out of a quantum system quickly and you discover that you have $\frac{1}{f}$ noise fluctuations because of correlations in the detector and so on. So I think that the minimal question from quantum mechanics is: can you genuinely get sort of random bits and what's involved in doing that? What actually happens in the devices to make that happen? I'd be curious to know the answer to this [question].

Tom Toffoli: I know the answer, Intel already says: yes, you can get good quantum numbers if you have a quantum generator of random numbers. Just generate one random number, then throw away your generator and buy a new one, because the one that you have used is alreay entangled with the one it generated. So you buy a new one and you solved the problem. (Wolfram says: this is also a good commercial strategy/solution ... people laugh)

Ed Fredkin: There's a great story in the history of – back in the '50s people doing various electronic things needed noise. They wanted noise, random noise so [they thought:] what would be a good source of it. And so it was discovered that a particular model of photomultiplier, if you covered it up and let no light into it gave beautiful random noise. And [as a result] various people conducted experiments, they characterized this tube and it was essentially like a perfect source of random noise, and the volume [of sales] started [to pick up]. Various people started building these circuits and using them all over. Meanwhile, back at the tube factory which was RCA, someone noticed: "Hey, that old [noisy] photomultiplier that we had trouble selling lately, sales are picking up, we better fix that design so it isn't so noisy!" So they fixed it and that was the end of that source of random[ness] noise.

6. Information vs. Matter

George Johnson: I think I'll ask another question about one other thing that has been bothering me before we start letting the audience jump in. I first ran across this when I was writing a book called "Fire of the mind" and the subtitle was 'Science, faith and the search for order.' And I was re-reading a book that I read in college that really impressed me at the time, and seeing that it still [stood up,] which was Robert Pirsig's book "Zen and the art of motorcycle maintenance" – and it did stand up in my opinion.

There's a scene early on in the book the protagonist who called himself Phædrus after Plato's dialogue is taking a motorcycle trip around the country with his son Chris – who in real life later was tragically murdered in San Francisco where he was attending a Zen monastery, which is neither here or there – but, in the book this person's running around with Chris and they're sitting around the campfire at night and drinking whisky and talking and telling ghost stories and at one point Chris asks his father: “Do you believe in ghosts?”

And he says “Well, no, of course I don’t believe in ghosts because ghosts contain no matter and no energy and so according to the laws of physics they cannot exist.” And then he thinks for a moment and says: “Well, of course the laws of physics are also not made of matter or energy and therefore they can’t exist [either].” And this really seemed like an interesting idea to me, and when I was learning about computational physics, this made me wonder: *where* are the laws of physics? Do you have to be a Platonist and think that the laws of physics are written in some theory realm? Or does this computational physics gives us a way to think of them as being embedded within the very systems that they explain? [waits a little, sees Chaitin wanting to answer, says:] Greg!

Gregory Chaitin: George, we *have* ghosts: information! Information is non-material.

George Johnson: Information is physical, right?

Gregory Chaitin: No, no – it’s ...

George Johnson: Well, wouldn’t Landaurer say that?

Gregory Chaitin: Well, maybe Rolf would say that, but ontologically we’ve come up with this new concept of information and those of us that do digital physics somehow take information as more primary than matter. And this is a very old philosophical debate: is the world built of spirit or mind or is it built of matter? Which is primary which is secondary? And the traditional view of what we see as the reality, is that everything is made of matter. But another point of view is that the universe is an idea and therefore (and information is much closer to that) made of spirit, and matter is a secondary phenomenon. So, as a matter of fact, perhaps everything is ghost. If you believe in a computational model informational model of the universe, then there is no matter! It’s just information – patterns of information from which matter is built.

George Johnson: Does that sound right to you, Tony?

Tony Leggett: Who, me?

George Johnson: Yes. Does it sound right to you that information is more fundamental than matter or energy? I think most of us – you know, the average person in the street – asked about information would think about information as a human construct that is imposed on matter or energy [that we make.] But I really like the idea that Gregory [has] suggested – and I actually wrote about it quite a bit in this book – that information is actually in the basement there, and that matter and energy are somehow [built/conjured] out of that. So I was just wondering, from your perspective: is *that* something [that] ...

Tom Toffoli (starts answering the question): Well, ideally, I mean...

George Johnson: Actually I was asking Tony...

Tom Toffoli: I will be very brief – will be done right away.

George Johnson: Sure, go ahead.

Tom Toffoli: You can ask the same thing about correlation rather than information because it conveys the same meaning. Furthermore one can bring up and discuss the notion of entanglement, and in the same spirit: it's not here nor there. Where is it? Very related issues. That's the point I wanted to make.

Tony Leggett: Well I think I would take the slightly short-sighted point of view [chuckle] that information seems to me to be meaningless – unless it is information *about something*. Then one has to ask the question: "what is the something?" I would like to think that *that something* has something to do with the matter of interest, and the matter and energy that's [involved].

Stephen Wolfram: This question about whether abstract formal systems are *about something* or not is a question that obviously has come up from mathematics. And my guess about the answer to this question: is information the primary thing or is matter the primary thing? I think that the answer to that question would probably end up being that they are really the same kind of thing. That there's no difference between [them]. That matter is merely our way of representing to ourselves things that are in fact some pattern of information, but we can also say that matter is the primary thing and information is just our representation of that. It makes little difference, I don't think there's a big distinction – if one's right that there's an ultimate model for [the] representation of universe in terms of computation.

But I think that one can ask this question about whether formal systems are about something – this comes up in mathematics a lot, we can invent some axioms system and then we can say: is this axiom system describing something really, or is it merely an axiom system that allows us to make various deductions but it's not really about anything. And, for example, one of the important consequences of Gödel's theorem is that you might have thought that the Peano axioms are really just about integers and about arithmetic but what Gödel's theorem shows is that these axioms also admit different [various] non-standard arithmetics, which are things that are not really like the ordinary integers, but still consistent with its axioms. I think it's actually a confusion of the way in which mathematics has been built in its axiomatic form that there is this issue about 'aboutness' so to speak – and maybe this is getting kind of abstract. But when we think about computations we set things up, we have particular rules, [and] we just say "OK, we run the rules, you know, [and] what happens – happens." Mathematics doesn't think about [these] things in those terms, typically. Instead, it says: let's come up with axiom systems which constrain how things could possibly work. That's a different thing from saying let's just throw down some rules and then the rules run and then things just happen. In mathematics we say: let's come up with axioms which sort of describe how things have to work, same thing was done in physics with equations – the idea of, you know, let's make up an equation that describe what's possible to happen in the world – and not (as we do in computation) let's do something where we set up a rule and then the rule just runs. So, for example, that's why in gravitation theory there's a whole discussion about "What do the solutions to the Einstein's equations look like?" and "What is the set of possible solutions?" and not (instead) "How will the thing run?" but the question traditionally asked there is: "What are the possible things consistent with these constraints?"

So in mathematics we end up with these axiom systems, and they are trying to sculpt things, to ensure that they're really talking about the thing that you originally imagine[d you were] talking about, like integers. And what we know from Gödel's theorem and so on, is that that kind of sculpting can never really work. We can never really use this constraint-based model of how to understand things to actually make our understanding be about a certain thing. So, I think that's kind of the ultimate problem with this idea of whether the laws that one's using to describe things and the things themselves are one and the same or they are different, and if so what the distinction really is.

Gregory Chaitin: There's this old idea that maybe the world is made of mathematics and that the ultimate reality is mathematical. And for example people have thought so about continuous mathematics, differential equations and partial differential equations, and that view was monumentally successful, so that already is a non-materialistic view of the world. Also let me say that quantum mechanics is not a materialistic theory of the world; whatever the Schrödinger wave equation is it's not matter, so materialism is definitely dead as far as I am concerned. The way Bertrand Russell put it is: if you take the view that reality is just what the normal day appearances are, and modern science shows that everyday reality is not the real reality, therefore – I don't know how he called this ... 'naive realism' I think – and if naive realism is true then it's false, therefore it's false. That's another way to put it. So the only thing we're changing in this view that the actual structure of the world is mathematical, the only thing new that we're adding to this now is we're saying mathematics is ultimately computational, or ultimately it is about information, and zeroes and ones. And that's a slight refinement on a view that is quite classical.

So I am saying in a way we're not as revolutionary as it might seem, this is just a natural evolution in an idea. In other words, this question of idealism vs. materialism, or "Is the world built of ideas or is the world built of matter?" it might sound crazy, but it's the question of "Is the ultimate structure of the world mathematical?" versus matter. That sounds less theological and more down to earth. And we have a new version of this, as ideas keep being updated, the current version of this idea is just: "Is it matter or is it information?" So these are old ideas that morph with time, you know, they evolve, they recycle, but they're still distinguishably not that far from their origin.

Cristian Calude: Information vs. matter, discrete vs. continuous: interesting philosophical contrasts and I found Stephen's description to be extremely interesting, because I too think that these views should in fact coexist in a productive duality. And it depends on your own abilities, it depends on your own problems if one of them would be more visible or useful. And at the end of the day what really counts is – if you have a view that in a specific problem information prevails – what can you get (from that)? Can you prove a theorem, can you get some result, can you build a model which you know answers an important question or not? In some cases one view may be the right one, in other cases the other one is, so from my point of view, which is, I would guess, more pragmatic, I would say: look, you choose whatever view you wish in order to get a result and if you get the result that [means that] for this specific problem that choice was the correct one.

Ed Fredkin: But in fact the world either is continuous or discrete, and we can call it anything we want, to get results and so on, to add convenience – but there really is an answer to it and one answer is right and the other is wrong. I go with Kronecker who said "God invented the integers and all else is the work of man." So you can do anything with discrete models and/or continuous models but that doesn't mean that the world is both, or can be, or could be either. No, the world is either one – or the other.

Stephen Wolfram: This whole question about mechanism [versus] mathematics and so on – it is kind of amusing. For example with CAs, models and things like that, people in traditional physics (not so in other areas but people in traditional physics) have often viewed this kinds of models with a great deal of skepticism. And it's kind of an amusing turn of historical fate, because, in the pre-Newtonian period people always had mechanistic models for things – whether there were angels pushing the Earth around the orbit, or other kinds of mechanistic type of things. And then along came this kind of purely abstract mathematical description of law of gravity and so on and everybody said – but only after a while! – everybody said: "Well, it's all just mathematics and there isn't a material reality, there isn't a mechanism behind these things!" And so, when one comes with these computational models which seem to have much more of a tangible mechanism, that is viewed as suspicious and kind of non-scientific by people who spend their lives working in the mathematical paradigm, that it can't be simple enough if there's an understandable mechanism to [behind] things. So it's kind of an interesting turning around of the historical process. As you know I work on various different kinds of things and, as I said, I've not been working much on finding a fundamental theory of physics lately. I actually find this discussion [as] an uptick in my motivation and enthusiasm to actually go and find a fundamental theory of physics. Because I think, in a sense, what with all of these metaphysical kinds of questions about what might be there, what might not be there and so on: damn it, we can actually answer these things!

Tom Toffoli: I know a way out of this! It is similar to the one that Ed proposed a long time ago. He said (about computability, referring to exponential, polynomial problems) he said that one can turn all problems into linear problems. You know, *all* the exponential problems! And people of course said: "Ed you are an undisciplined amateur you say these things without knowing what you're talking about." But he said: "Look, we have Moore's law! And with Moore's law everything doubles its speed every so many years. So we just wait long enough and we get the solution – as long as it takes."

So the key then, is to wait, as long as we need to. With this, I am now giving a simpler solution to this problem, [of finding a fundamental theory of physics,] starting from the observation that domesticated animals become less intelligent than wild animals. This has been proven recently with research on wolves. And maybe domesticated animals are somewhat less intelligent in certain ways but they can see what humans want and they obey. But then some more experiments were run and the findings were that wild wolves once they put them in an environment with humans they learn humans faster than domesticated animals to anticipate the wills of their trainers. It's not that they follow their will, but they anticipate it faster.

Now we are doing a big experiment on ourselves, on humanity – humanity is domesticating itself. And there was a time when people said: we discovered differential equations and very few people can understand them so we have the monopoly, we are the scientists. And eventually somebody said: "Well, wait a second, but why can't we find a model like, you know, Ed or Stephen – that is, sort of computational, discrete so that everyone can own it and possess it and so on?" But [one forgets that] we are domesticating ourselves! To the point that the computer that – according to Brian Hayes the first thing about a computer is that you can program it, make it do whatever we want – now most people don't even know that that is possible, they just think of the computer as an appliance. And soon even the computer will be a mystery to most people! The programmable digital computer [like] the differential equations were a generation ago – so we solved the problem, in that sense, just wait long enough and nobody will even [be able to] care about these things, it will just be a mystery for us[19].

Cristian Calude: Well, I would like to just add a small remark, suggested by your idea about physical computation. So, this is essentially a personal remark about the P vs. NP problem: and I believe this is a very challenging and deep and interesting mathematical question, but I think one that has no computer science meaning whatsoever. For the simple fact that P is not an adequate model of physical computation, and there are lots of results – both theoretical and experimental – which point out that P does not model [properly] what we understand as physical computation. Probably the simplest example is to think about simplex which is exponentially difficult, but works much better in practice than all the known polynomial solutions.

7. Unmoderated Audience Questions

George Johnson: Thank you. I guess this is a good time to move on to questions from the audience and I think the best way to do this is that if you ask a question of a particular speaker [if] the speaker can repeat the question we will then have a record of what was asked – like Charles Bennett's raindrop craters that we saw the other day – along with what was answered.

[19] And we'll get used to it like we get used with the idea that we are not immortal!

Jason Cawley (a bit incomprehensible, has no microphone): So I have a question [for Greg Chaitin but for everyone else as well.] You said that the world would be more intelligible and prettier, and rationalism [would be] morphing into ... [... inner system is ...] discrete – which was very [attractive] to me. But how intelligible would it be, really, even if I grant you finiteness and discreteness, even if we find the rule? Won't it have all these pockets of complexity in it, wouldn't it have [...] computation, won't it be huge compared to us, [which are] finite, wouldn't it last much longer than us – and then we still have all kinds of ways that would be mysterious to us in all the little detail?

George Johnson asks Gregory Chaitin to repeat the question.

Gregory Chaitin: Well, I didn't catch all of that... (collects his thoughts then proceeds to summarize the question as best as he heard it) ... Oh, I guess the remark is disagreeing with what I said that a discrete universe would be more beautiful ... no, no, more comprehensible ... right? ... and you gave a lot of reasons why you think it would be ugly, incomprehensible, disgusting – and I can't argue, if you feel that way! [But in that case] I don't think that's a question, I view that as a comment that doesn't need to be answered ...

[**Gerardo Ortiz** takes the microphone from the moderator's table to Jason Cawley, whom we can now hear well, reformulating his question.]

Jason Cawley: Sorry. The question is "How intelligible is a perfectly discrete universe?" The reason I am concerned about this is: I happen to like rationalism too, but I don't want people concluding, when they see non intelligibility in the universe, that it is evidence against [the] rational.

Gregory Chaitin refines his answer: Oh, okay, great! Well, in that case, as Stephen has pointed out in his book, it could be that all the randomness in the world is just pseudo randomness, you know, and things only *look* unintelligible, but they are actually rational. The other thing is he's also pointed out – and this is sort of his version of Gödel's incompleteness theorem – is that something can be simple and discrete and yet we would not be able to prove things about it. And Stephen's version of this (which I think is very interesting) is that in the way that the universe is created, because you have to run a computation, in general you have to run a physical system to see what it will do, you can't have a shortcut to the answer[20]. So that can be viewed as bad, but it also means that you could have a simple theory of the world that wouldn't help us much to predict things. And you can also look at it as good, because it means that the time evolution of the universe is creative and surprising, it's actually doing something that we couldn't [know] in advance – by just sitting at our desks[, and thinking] – so I view this as fundamental and creative! And in regards to creativity Bergson was talking 100 years ago about "L'Evolution Créatrice[21]" – at this point, this would be a new version of that. But over aestetics one can't

[20]Via the Principle of Computational Irreducibility (which is a corollary of the PCE).

[21]1907 book by French philosopher Henri Bergson. Its English translation appeared in 1911. The book provides an alternate explanation for Darwin's mechanism of evolution.

ultimately argue too much. But still, I think it was a good question.

Gerardo Ortiz: So, further questions?

New question from the audience[22]: First a question about SETI: I may be naive, but it seems strange to me why we would like to look at perfect circles and dividing lakes when one can simply look at Chicago and New York emanating light from Earth at night. If I were a Martian that's what I would do. But what interests me more and that's the question for Sir Leggett is this: suppose that there were phase transitions from quantum to classical. Would building a quantum computer – would the approach to build a quantum computer be different than trying to reduce decoherence as it's being done presently?

Tony Leggett: I'm not entirely clear what you mean by postulating that there was a phase transition from quantum to classical. Could you elaborate a bit?

Question is refined: Ah, well, I am a bit ignorant about your theory but, if there is a theory that it's not just decoherence but in fact there are phase transitions from quantum to classical at some level – that's why we don't see Schroedinger's cat after some [... W]ould this imply perhaps a different approach of building a quantum computer?

Tony Leggett: If you mean – if you're referring to theories, for example, of the GRWB type which postulate that there are physical mechanisms [systems] which will meet the linear formalism of quantum mechanics[23] which will have to be modified – and it will have to be modified more severly as it goes from the microscopic to the macroscopic – then I think the answer to your question is that to the extent that we want to use macroscopic or semi-macroscopic systems as qubits in our quantum computer, it wouldn't work. On the other hand I don't think that theories of the GRWB – scenarios of the GRWB type – necessarily [...] against an attempt to build a quantum computer always keeping the individual bits at the microscopic level. You have to look at it in, of course, in detail in a specific context of a particular computer built around a particular algorithm, such as, say, Shor's algorithm. But I don't think it is a priori essential that a GRWB type scenario would destroy the possibility of it [...].

Question ends, and **Gerardo Ortiz** invites new questions.

New question from the audience[24]: Well, we still haven't answered the question of "How does nature compute?" We are just discussing [differences] between discrete and continuous, classical vs. quantum computation but if we see mathematics as a historical accident and we try to push it aside, shouldn't we try to look at nature, and try to understand how nature in fact computes? And not just try to traduce[25] it into a mathematical context. For example one can watch plants and see that there is some kind of parallel type of computation

[22] I don't know the name

[23] Composed of kets

[24] Still don't know the name, but Hector might be able to identify him.

[25] I don't know if the speaker means, literally, 'traduce' – or, perhaps 'translate' and the meaning is [somewhat] lost in the ad-hoc translation...

that is going on, that is, not based on our mathematics, but like *they* do it, the plants ... you know... Do you have you any thoughts on that?

Stephen Wolfram answers it: I think that one of the things that makes that somewhat concrete is the question of how we should build useful computational devices. Whether our useful computational devices should have ALUs (Arithmetic Logic Unit) inside them [or not]. The very fact that every CPU that's built has a piece that's called "the arithmetic logic unit," tells you that in our current conception of computation we have mathematics somewhere in the middle of it. So an interesting thing is: can we achieve useful computational tasks without ever having an ALU in the loop, so to speak. I think the answer is: definitely yes, but as the whole engineering development of computers has been so far we've just been optimizing this one particular model that's based on mathematics. And as we try to build computers that are more at the molecular scale, we could, actually, use the same model: we could take the design of the Pentium chip and we can shrink it down really really small and have it implemented in atoms. But an alternative would be that we can have atoms do things that they are more naturally good at doing. And I think the first place where this would come up are things like algorithmic drugs, where you want to have something that is in essence a molecule operating in some biological, biomedical context and it wants to actually do a computation as it figures out whether to bind to some site or not, as opposed to saying "I am the right shape so I'm going to bind there!" So that's a place where computation might be done. But it's not going to be computation that will be done through arithmetic but we'll be forced to think about computation at a molecular scale in its own terms, simply because that's the scale at which the thing has to operate. And I'm going to guess that there will be a whole series of devices and things, that – mostly driven by the molecular case – where we want to do computation but where the computation doesn't want to go through the intermediate layer of the arithmetic.

Cristian Calude: Yes! There is a lot of research that we have started about ten years ago in Aukland, and a series of conferences called "Unconventional Computation." And this is one of the interesting questions. You know, there are basically two streams of thought: one is quantum computation, and the other one is molecular computing. And in quantum computing, you have this tendency of using the embedded mathematics inside. But in molecular computing, you know, you go completely wild [because] mathematics is not there [so] you use all sorts of specific biological operations for computation. And if you look at the results, some of them are quite spectacular.

Small refinement: Yeah, but they are still based on logic gates and ... you know ... molecular computing is still based on trying to build logic gates and ...

Cristian Calude listens and continues: No! There is no logic gate! That's the difference, because the philosophy of approach in quantum computation for instance [is:] you do these logical gates at the level of atoms or other particles – but in molecular computation there is no arithmetical instruction, there are no numbers, you know, everything is [a] string, and the way they are manipulated is based exactly on biological type of processing. No arithmetic.

Gregory Chaitin: No boolean algebra, not [even] *and*'s and *or*'s?.

Cristian Calude: No, nothing! And then this is the beauty, and in a sense this was the question that I posed to Seth Lloyd in '98. [I said] you know, why don't you do in quantum computing something similar? Why don't you try to think of some kind of rules of processing – not imposed from the classical computation, from Turing machines, but rules which come naturally from the quantum processes – just like the typical approach in molecular computation.

Tom Toffoli: I would like to give a complementary answer to this. I've been teaching microprocessors and microcontrollers – you have them in your watches and cell phones. They're extremely complicated objects. And you would say, I mean given the ease with which we can fabricate these things, whenever we want to run a program or algorithm, we could make a very special purpose computer rather than using a microprocessor to program it. Apparently it turns out that it's much more convenient, if someone had designed a microprocessor with an ALU and a cache and the other things in between, to just take it as a given and then the [process is complete]. If we look at biology, biology has done the same thing. We have, at a certain moment, hijacked mitochondria that do the conversion of oxygen and sugar into recharging the ATP batteries. That was a great invention. And now, after probably three billion years, or something like that we still keep using that instead of inventing a method that is, maybe a little more efficient, but would have to be a very special choice for every different [circumstance]. Essentially we could optimize more, but we would do that at the cost of losing flexibility, modularity and so on [but, apparently] it's much more convenient. For three billion years we've kept using this kind of energy microprocessor, that worked fairly well ... So there is, essentially, the flexibility or modular evolution that really suggests that choices like the ALU ... is not an optimal choice, but – empirically – is a very good choice. This is my viewpoint.

Stephen Wolfram: Well – as is always the case in the history of technological evolution it is inconceivable to go back and sort of restart [the whole design process]. Because there is just far too much investment in that particular thing. The point that I want to make is that – what will happen is that there will be certain particular technological issues, which will drive different types of computing. My guess is that the first ones that will actually be important are these biomedical ones, because they have to operate on this molecular scale, because that's the scale on which biomedicine operates. And, you know, one can get away with having much bigger devices for other purposes – but biomedicine is a place where potentially a decision has to be made by a molecule. And whether – maybe there will be ways of hacking around that [but] in time, it won't be very long before [the first applications of this kind will be finalized].

And what's interesting about [this] is that if you look at the time from Gödel and Turing to the time when computers became generic and everybody had one, and the time from Crick and Watson and DNA to the time when genomics becomes generic – it's about the same interval of time. It hasn't yet happened for genomics, it has more or less happened for computers. Computers were invented, the idea of computers is twenty-three years earlier or something like that, than the DNA idea. Anyway, [it will soon happen for genomics as well, and] in time we will routinely be able to sequence things, in real time, from ourselves, and we'll do all kinds of predictions that yes, you know we detect that today you have a higher population of antibodies that have a particular form so we'll be able to run some simulation that this means that you should go and by a supply of T-cells, that has this particular characteristic and so on. And there's a question whether the decisions about that will be made externally by ALU based computers, or whether they will be made internally by some kind of molecular device – more like the way biology actually does it. And if it ends up that they are made externally then there won't be any drive from the technology side to make a different substructure for computing.

Roshan James: I study computer science. And one of the ideas I find truly fascinating and I think it [really] is – is this thing called the Curry-Howard isomorphism, which basically relates propositions to types and rules to terms and proof normalization to program evaluation. And since you're [discussing] models of computation, I was wondering if you have encountered something similar for cellular automata and such.

Stephen Wolfram: This is an area I don't know much about, so ... [He turns and asks Calude: "Do you know about this question?" Calude gestures, then apparently makes a suggestion. Wolfram turns back, speaks to the student:] OK, so you have to explain it to us a bit more. Tell us, explain to us what you're talking about, so we can learn something and try to address the question.

Roshan James: OK. I think that the classification, this particular relation is very explicit in the simply typed lambda calculus where program terms, which are lambda expressions, can be given types – and the types are pretty much propositional logic expressions. And if you can prove a certain proposition the structure of the proof in natural deduction style will actually look like a program type and if the proof is not a normal proof then the process of proof normalization is basically the process of the evaluation of the term into a normal form of the term, which basically means that if you have this computational model which is the lambda calculus, reductions of the calculus will correspond to normalizations of the proofs, and the types serve as a way of classifying programs, types are a way of saying: these particular programs that behave in such and such a way don't have such and such properties. And I feel that this might be something that carries over to other notions of computations as well, because there's nothing intrinsic about the lambda calculus that makes this [uniquely applicable to it].

Stephen Wolfram: Let me start by saying that I'm a very anti-type person. And it turns out that, you know, in the history types were invented as a hack by Russell basically to avoid certain paradoxes – and types then became this kind of "great thing" that were used as an example and then as a practical matter of engineering in the early computer languages there was this notion of integer types versus real types and so on, and the very idea of types became very inflated – at least that's how I see it.

So, for example in Mathematica there are no types. It's a symbolic system where there is only one type: a symbolic expression. And in practical computing the most convincing use of types is the various kinds of checking but in a sense when something is checkable using types that involves a certain kind of rigidity in programs that you can write, that kind of restricts the expressivity of the language that you have. And what we found over and over again in Mathematica, as we thought about putting things in that are like types, that to do that would effectively remove the possibility of all sorts of *between paradigm* kinds of programming, so to speak, that exist when you don't really have types.

So, having said that, this question of the analogy between proof processes and computation processes is an interesting one. I've thought about that a lot, and there's more than just a few things to say about it. But one thing to think about is: "What is a proof?" and "What's the point of doing a proof?" I mean, the real role of a proof is as a way to convince (humans, basically) that something is true. Because when we do a computation, in the computation we just follow through certain steps of the computation and assuming that our computer is working correctly the result will come out according to the particular rules that were given for the computation. The point of a proof is somehow to be able to say to a human – look at this: you can see what all the steps were and you can verify that it's correct. I think the role of proofs in modern times has become, at best, a little bizarre. Because, for example, so here's a typical case of this: when Mathematica first existed twenty years ago one would run into mathematicians who would say "How can I possibly use this, I can't prove that any of the results that are coming out are correct!" OK? That's what they were concerned about. So, I would point out sometimes that actually when you think you have a proof, in some journal for example, it's been maybe checked by one person – maybe – if you're lucky. In Mathematica we can automate the checking of many things and we can do automatic quality assurance, and it's a general rule that – in terms of how much you should trust things – the more people use the thing that you're using the more likely it is that any bugs in it will have been found [by the time you use it]. So, you know, if you say: "Well, maybe it's a problem in the software that I myself am writing, maybe it's a problem in the system (like Mathematica) that I am using, maybe it's a problem in the underlying hardware of the computer" – it gets less and less plausible that there's a problem, the broader the use of the thing is.

So I think as a practical matter when people say: "I want a proof!" that the demand for proof, at least in the kind of things that Mathematica does, decayed dramatically in the first few years that Mathematica existed because it became clear that most likely point of failure is where you as a human were trying to explain to the computer what to do. Now, having said that it's interesting [that] in Mathematica we have more and more types of things that are essentially proof systems – various kinds of things, for example proof systems for real algebra, we just added in the great new Mathematica 7.0 all sorts of stuff of doing computation with hundreds of thousands of variables and so on. Those are effectively places where what we've done was to add a proof system for those kinds of things. We also added a general equational logic proof system, but again I think that this question whether people find a proof interesting or whether they just want the results – it seems that the demand for presentation of proofs is very low.

Tom Toffoli (wants to add something, Wolfram uses the break to drink some water): If you would give me the microphone for one second: coming back to Russell when he published Principia Mathematica – most of the theorems were right, but a good fraction of the proofs were found wrong. I mean this was Russell, OK? He was wrong, but there was no problem, because he was still convinced, by his own ways he was convinced of the theorems. So he put together some proofs (arguments) to try to convince the readers that the theorems were right, and he convinced them. But the proofs, as actual mechanical devices, were not working. So his proofs were coming out of just a heuristic device and he derived, and you can always derive the right theorem with the wrong proof. What are you going to do in that case? [Wolfram makes a sign that he can take the the microphone back, Toffoli returns it]

Stephen Wofram: No, I think it's actually interesting this whole analogy between proof and computation and so on. One of the things that I have often noticed is that if you look at people's earlier attempts to formalize mathematics – the thing that they focused on formalizing was the process of doing proofs, and that was what Whitehead, Russell, Peano before him and so on [worked on]. It turned out that that direction of formalization was fairly arid. Not much came from it. The direction that turned out to be the most interesting direction of formalization was, in fact, the formalization of the process of computation. So, you know, in the construction of Mathematica, what we were trying to do was to formalize the process of computing things. Lots of people used that and did various interesting things with it. The ratio of people who do computation with formalized mathematics to the number who do proofs with formalized mathematics is a huge ratio. The proof side turned out not to be that interesting. A similar kind of thing, and an interesting question, is how one would go about formalizing every day discourse: one can take (the) everyday language and one can come up with a formalized version of it that expresses things in a sort of formal, symbolic way. But the thing that I've not figured out – actually I think that I've now figured it out, but I *hadn't* figured it out – was "So, what's the point of doing that?" In other words the Russell-Whitehead effort of formalizing proofs turned out not not lead to much. The right idea about formalizing mathematics was the one about formalizing the process of computation. Similarly formalizing everyday discourse as a way to make the semantic web or some such other thing, probably has the same kind of issue as the kind of formalization of proof as was done in mathematics and I think that maybe there is another path for what happens when you formalize everyday discourse, and it's an interesting analogy to what would happen in the mathematics case. You know: what's the point of formalization and what can you do with a formalized system like that.

Gregory Chaitin: Let me restate this very deep remark that Stephen has just made about proofs versus computation: if you look at a[ny] formal system for a mathematical formal theory, Gödel shows in 1931 that it will always be incomplete. So any artificial language for doing mathematics will be incomplete, will never be universal, will never have every possible mathematical argument. There is no formal language for mathematics where every possible mathematical argument or proof can be written in. Zermelo-Fraenkel set theory, you know, as a corollary of Gödel – may be wonderful for everything we have but it is incomplete. Now the exact opposite – the terminology is different – when you talk about a programming language you don't talk about completeness and incompleteness, you talk about universality. And the amazing thing is [that] the drive for formalism, and Russell-Whitehead is one data point on that, another data point is Hilbert's program, the quest for formalization started off in mathematics, and the idea was to formalize reasoning; and the amazing thing is that this failed. Gödel in 1931 and Turing in 1936 showed that there are fundamental obstacles– it can't work! But the amazing thing is that this is a wonderful failure! I mean what can be formalized beautifully, is not proof, or reasoning but: *computation*. And there almost any language you come up with is universal, which is to say: complete – because every algorithm can be expressed in it. So this is the way I put what Stephen was saying.

So Hilbert's dream, and Russell and Whitehead failed gloriously – is not good for reasoning but it's good as a technology, is another way to put it, if you're trying to shock people. The quest for a firm foundation for mathematics failed, but gave rise to a trillion dollar industry. (Toffoli asks for the microphone)

Tom Toffoli: Let me add something to this. You've heard of Parkinson's law[26], the one that says: "a system will use as many resources as are available." It was formulated because it was noticed that the whole British Empire was run by essentially a basement of a few dozen people for two hundred years. And then, the moment the British started losing their empire, they started de-colonizing and so on, then they had a ministry of the colonies and this ministry grew bigger and bigger and bigger as the colonies became fewer and fewer and fewer. And it was not working as well as before. And I think that formalization is often something like that. You can think about it, but you don't want to actually do it, even von Neumann, you know, the moment he decided that CAs were, sort of, plausible to give life – he didn't go through the process of [developing] the whole thing. The point was already made.

Gerardo Ortiz: So, there is room for a last question ...

New question[27] from the audience: If I may, can I add something about the Curry-Howard isomorphism before I ask my question? Yes? Maybe short ... I think that the revolution that we are living in physics is only part of a wider revolution and there are many questions in physics to which the answer uses the notion of algorithm. For instance: what are the laws of nature? They are

[26] `http://en.wikipedia.org/wiki/Parkinson's_Law`

[27] Hector may be able to identify him.

algorithms. So you may give this answer, only because you have the notion of algorithm. And for centuries we didn't have it. So, to these questions we had either no answer or ad-hoc answers. For instance: what are the laws of nature? Compositions. When we had compositions we didn't have algorithms. It was a good way to answer it. And there are many many areas in knowledge where there are many questions to which now we answer: it is an algorithm. And one of the very first questions on which we changed our mind, was the question: what is a proof? And: what is a proof? The original answer was a sequence of formulas verifying deduction rules and so on. And starting with Kolmogorov – because behind the Curry-Howard isomorphism there is [the] Kolmogorov interpretation – is this idea that proofs, like the laws of nature are (in fact) algorithms. So they are two facets, or two elements of a wider revolution, and I think that they are connected in this way.

Now consider this question[28]: Does the Higgs boson[29] exist? Today, I guess, there are people who believe that the Higgs boson exists, there are people who believe that it doesn't exist, but anyone – you can take the electron, if you want (instead of the Higgs boson) [which] most people I guess, believe it exists – but I guess that everyone agrees that we have to find some kind of procedural [means of verifying such a prediction]. It can be an experiment, it may be anything you want – in this case – that will allow eventually, if we are lucky enough, to solve this question. And I would be very uncomfortable if someone told me that the Higgs boson exists in the eye of the beholder, or that Higgs boson has to exist because then the theory would be more beautiful, or if the Higgs boson does not exist then it's actually a problem of the Universe and not ours and we can continue to postulate that it exists because it would be nicer. So I wonder if the two questions we have (not?) been discussing today are not of the same kind: we should try to look for a kind of procedure to answer the question at some point. One: is the universe computable? The other: is the universe discrete? In some sense are these questions only metaphysical questions or are they questions related to experiments that we could [and should, in fact] carry out?

George Johnson: Well thank you, and I'm sorry we won't going to have time to get into that, which could easily be another hour because I'm told by Adrian that we're on a very very draconian time schedule here because of our lunch and also because we have to check out of our rooms. So I want to thank the speakers very much very much for a wonderfully ...

Stephen Wolfram: Can we respond to that?

George Johnson: Perhaps we could resume at the next conference and this would be another reason to look forward to it.

Stephen Wolfram: So many things can happen between now and then – by then we may even have a definitive answer. Right now there's so much to say about it but it sounds like we don't have the time for that.

[28]George Johnson says: "Well, thank you!" for the first time.

[29]`http://en.wikipedia.org/wiki/Higgs_boson`

Everybody is in good spirits and the round table debate, along with the conference, ends (applause etc.)

End of transcript.

Acknowledgments

Transcript by Adrian German (Indiana University) with some assistance from Anthony Michael (UC Berkeley).

References

No time to make a list of references, just some footnotes in the text.