A Bit of Conversation: A Scientific Fiction

Srinivasan Keshav

Srinivasan Keshav is the Robert Sansom Professor of Computer Science in the Department of Computer Science and Technology at the University of Cambridge. He received a PhD in Computer Science from the University of California, Berkeley, in 1991. He was subsequently a Member of Technical Staff at AT&T Bell Labs and an Associate Professor at Cornell. In 1999 he left academia to co-found Ensim Corporation and GreenBorder Technologies Inc. Returning to academia, he was at the University of Waterloo in Canada from 2003 to 2019, holding a Canada Research Chair and later the Cisco Chair in Smart Grid. He is a Fellow of the Royal Society of Canada, ACM, and IEEE, and a Distinguished Alumnus of IIT Delhi.

SCENE. Manhattan. Street with jazz club and apartment building on a muggy late summer evening.

The tired leaves on the trees are damp with rain. Yellow cabs whoosh by, nearly drowned out by a thrum of cicadas. A door swings open from the jazz club, releasing a bubble of music into the street and ALAN, CLAUDE, and JOHNNY, three young men engaged in animated conversation.

ALAN: [*To CLAUDE.*] Brilliant. Thanks for bringing us. We couldn't have heard this in London.

CLAUDE: You're welcome. Should we walk over to my place for a drink? It's only a block away. [*To JOHNNY.*] OK, Johnny?

JOHNNY: Sure!

They saunter down the street and enter CLAUDE's apartment building. It is a mess, with papers, books, magazines, empty beer bottles, sheet music, and electronic gear strewn everywhere. CLAUDE sweeps the papers off two chairs and onto the floor and gestures to ALAN and JOHNNY to sit.

CLAUDE: Don't mind the mess. I'll get to it one of these days.

JOHNNY: So, what did you mean when you said in the jazz club that 'It's all the same thing'? What's all the same thing?

CLAUDE: Well, let's see. Did you like the jazz performance? [*JOHNNY nods.*] OK, so what you heard was a series of notes, right?

JOHNNY: Yes.

CLAUDE: So if I had a way to write every note down, then the concert could be replayed just as you heard it?

JOHNNY: Yes, of course. That's like cutting an LP record.

CLAUDE: Exactly. So, think about the groove on the LP, which is a spiral you could unroll into a long line. What defines it?

JOHNNY: Its depth at each point.

CLAUDE: What if I made a table of depths and represented each possible depth by a symbol? Like so. [*He grabs a sheet of paper off the floor and writes '0.001 inch: 1; 0.002 inch: 2; ...', saying the measurements as he writes. He shows it to JOHNNY.*]

JOHNNY: Yes, I see that.

CLAUDE: So, you agree that the jazz performance can be represented by a groove and now you see that the groove depth is represented by a symbol.

JOHNNY: Yes, so I see that the performance can be represented by a series of symbols. Got it.

CLAUDE: Now, let me show you something else. See this magazine here? Let's find a black-and-white picture. [*JOHNNY picks out an ad for a car.*] Give that to me and hand me a pen. [*JOHNNY does so. CLAUDE draws a fine grid on the ad and labels the columns 'A', 'B', 'C', and so on, and the rows '1', '2', '3', and so on, narrating this as he does so. He holds it out to JOHNNY.*] Now, if you look at each grid cell, say B17, you'll see that there is only one shade of grey in that cell.

JOHNNY: Yes.

CLAUDE: Well, not really, but I could make the grid fine enough and that would be true.

JOHNNY: OK.

CLAUDE: Now I can play the same trick as before. I'll create a symbol to represent each shade of grey and then I can represent the image as a series of symbols.

JOHNNY: I get it. So, what you're saying is that you can represent music and images as strings of symbols.

CLAUDE: Yes! And more. A movie is a series of images, so that's also a string of symbols. I claim that any information you hear, see, or read can be reduced to such a string.

JOHNNY: Oh, very interesting.

CLAUDE: [*Spinning around, suddenly intense.*] It's all the same, all strings of symbols!

JOHNNY: Claude, this is very good. But I have a question.

CLAUDE: OK.

JOHNNY: What's the smallest unit of information? What would be an 'atom' of information?

CLAUDE: [*He pauses for a moment.*] Hmmm… perhaps the choice between yes and no? 0 and 1?

JOHNNY: Exactly. And if I were to encode each of your symbols as a string of 0s and 1s, then all information would be reduced to strings of 0s and 1s!

ALAN: Well, with a string of, say, 5 0s and 1s, we can represent up to 2x2x2x2x2 = 32 different symbols. And adding just one to the length of the string lets you represent twice as many strings, 64.

CLAUDE: Yes. So, even if we have many types of symbols, we can represent them with fairly short strings of 0s and 1s. This means you can represent all information as 0s and 1s, which are binary digits, or 'bits'. [*ALAN and JOHNNY nod.*] It's all bits!

ALAN: There's a problem here. I agree that bits can represent symbols, but how best should we assign bits to symbols? It doesn't make sense to use many bits when only a few are needed.

CLAUDE: True, but now that you told me about bits, I think I know how to solve that. Say you had a picture with 975 shades of grey, but the most common colour was black. If you assigned all shades equal length strings, you would need to encode at least 975 values, which would need all shades to be represented by 10 bits (since 2x2x...x2 10 times is 1024). But I could be clever about it, and assign 0 to black, so that most of the time I used only one bit for that. The less common the shade, the longer the bit string assigned to it, and we'd still come out ahead!

ALAN: Yes, I see that. What's going on is that a picture that is mostly black doesn't have much information content.

CLAUDE: Yes, and what I've been thinking of is a way to exploit this intuition to find the smallest string of symbols needed to represent anything. Indeed, what I can prove is that given a set of images, or, indeed, for any body of information, it is possible to encode each message in the set using the shortest possible string of symbols.

ALAN: That sounds very interesting! Let me tell you about what I've been working on. You've talked about symbols representing information, but what if we use symbols to represent actions?

CLAUDE: Hmm... give me an example.

ALAN: OK, let's say '00' means walk one step east, '01' means walk one step north, '10' means walk one step west, and '11' means walk one step south. Then, an army's marching orders would be a series of bits.

CLAUDE: Right.

ALAN: So, we could build a machine that reads a magnetic tape containing marching orders using a playback or read head, and then tells the army what to do.

CLAUDE: OK.

ALAN: But now I'll make it more interesting. Suppose I had some tape and a head that can both read and write symbols. I'll also need an operator who can read what's on the tape and has an instruction manual and a blackboard to keep track of the current 'state'. By this I mean some private information about the status of the machine, which can be represented by a number like 1, 2, 3, and so on. The idea is that the action taken by the operator, which is printed in the instruction manual, depends not only on what is read from the tape but also on the current state. The state is initially set to 0 and the state changes as a result of the prior state and what is read from the tape, because of the carrying out of an instruction. For example, here is an instruction. 'If you are in state 1 and you see the symbol 0, then write a 1, move left one step, and change to state 2; if you are in state 1 and you see the symbol 1, write a 0 and move right one step, and change to state 2; if you are in state 2, no matter what you read, move left one step, and stay in state 2.' You can see that at the end of each instruction, we learn the new state and where the head has moved to, and if the tape has been changed.

CLAUDE: Ah, since you know the state at the start and you know the new state at each step, it's possible to keep track of the state quite easily. But this does seem more complicated than what you had before. What does that buy you?

ALAN: [*Grinning.*] This machine can calculate anything you'd like! More precisely, it can perform any calculation that can be specified as the outcome of a deterministic set of instructions (also called an algorithm). My thesis advisor, Alonzo Church, and I have proved this mathematically. That's why some people call this machine a 'Turing Machine'.

CLAUDE: Really! But how could such a simple machine do anything interesting. Can you use it to add two numbers, say two and three?

ALAN: Yes, let me tell you how. First, we need to represent the numbers as strings of, say, 1's. So, '11' would be 2 and '111' would be 3. We'll need a way to separate them on a tape, say with a 0. When the machine starts, it reads the first 1 from the first number, replaces it with a blank, and changes to a state that causes it to move all to the right end and write a 1. Then it changes to a state that will move it all the way back, until it hits a blank space, when it moves right and repeats the process. I'll spare you the details, but when the machine stops, there will be 5 1's on the tape, and the two inputs will be erased.

CLAUDE: Fascinating. Could you do something more complex?

ALAN: Yes, here's another example. Suppose you had a picture in shades of grey and you wanted to make it black and white. You could write instructions to move the head in such a way that all values of grey closer to black would be overwritten by the string for black, and all other values would be overwritten by the string for white.

CLAUDE: So your machine can deal not only with numbers but also with images!

ALAN: Yes, and as you just showed us, since all information can be represented as bits, my machine can manipulate all types of information!

CLAUDE: Amazing. But this solution is not elegant, in that you have this complicated instruction manual to be consulted by an operator at each step.

ALAN: Yes, but there's a way around that. Suppose I could represent the instructions themselves as bits on another tape, call it the instruction tape. Then, a more complicated machine could read the instructions it was supposed to carry out from this tape and carry them out. Admittedly, this machine would be more complex than the one before, since it would need to keep track of not just the state, but also the instruction manual. But this machine would be a *universal* machine, a machine that could emulate any other Turing Machine, just by putting the right instructions on the instruction tape.

JOHNNY: [*Excited.*] Did you know the US military has been building special-purpose machines that they use to calculate missile trajectories, taking into account wind direction and weather conditions? I have been consulting with them. So I know exactly how to build a machine that does one computation well. This is like the first machine Alan described. But this conversation has solved a problem I've been grappling with for a while. The trajectory-computing machines (we call them 'computers') can only compute one thing at a time. If we need to change what they compute, we have to go in and rearrange the wiring. But if we use Alan's idea of a universal machine, we could store the wiring diagram, so to speak, on the instruction tape, and the computer would automatically rewire itself.

ALAN: Fascinating. So perhaps my thought experiments were not all a waste of time. I agree that you could store the instructions for the trajectory computer as a 'program' to tell the computer what to do. Feed this as the instruction tape, and your trajectory computer would never need rewiring!

CLAUDE: [*Suddenly.*] Johnny, you said the trajectory computer prints out trajectories, right? Where does it print them to?

JOHNNY: Well, to punched cards, actually, which we then feed to a printer.

CLAUDE: Instead of punching cards, could the trajectory computer make sounds instead? A different sound for each symbol it wanted to print?

ALAN: In theory, yes, why not?

CLAUDE: But then you could have the computer send these sounds on a telephone line. And if another computer were listening, it would allow the two to 'talk' to each other.

ALAN: Yes, that could be arranged.

CLAUDE: And if one computer that received messages were to act like a telephone operator, it could relay messages to yet other computers.

ALAN: Yes, again, in theory, this could be done. But why bother? There aren't many computers in the world today—maybe 6 or 7. And telephone lines can be really noisy, so the sounds could get corrupted, changing the symbols being exchanged.

CLAUDE: [*He shifts and looks down, before speaking softly.*] I've been thinking about that, too. I've figured out how fast you can send information symbols on a noisy line. It turns out that it is possible to send symbols with perfect recognition even on a noisy channel, such as a telephone line, if they're properly encoded. For example, you could send a sound corresponding to a symbol for a long time, longer than any noise would last, and this way the recipient could decode the symbol despite the noise, albeit at a slower rate. I'd been thinking of sending military codes over telegraph wires, but I think you could do the same for computer sounds, too.

ALAN: So, you'd get a network of computers that could talk to each other. But does that matter?

CLAUDE: Don't you see? If we can represent any information as bits, and computers can manipulate these bits with arbitrary precision and can talk to each other, then *all* information can be processed and communicated. [*Sweeping his hand across the mess in the room.*] Look at all this! All these papers, books, magazines. You could send them all from the creator's computer to a central store computer, and from the store to anyone who wanted to read it anywhere, on demand. The world's encyclopaedias, every film ever made, every song ever sung, they could all be in memory stores and someone with a computer could access them!

JOHNNY: Pipe dreams! A computer takes up half a building! And it costs millions of dollars.

CLAUDE: Have you heard of John Bardeen at Bell Labs? He's one of the solid-state physicists in Building 1, who works with Walter Brattain and William Shockley. He's working on a 'transfer resistor' or 'transistor' that can replace vacuum tubes. He says that a computer made from transistors would not be much larger than a refrigerator! Can you imagine that? And would only cost about $200,000.

JOHNNY: Still too expensive.

CLAUDE: OK, but transistors are made from silica, which is just sand. There's no reason you couldn't make them smaller and cheaper. Someday, computers may be small enough to carry in your pocket.

JOHNNY: And pigs will fly!

CLAUDE: No, really, I imagine that by 2020 nearly everyone will carry a pocket computer and communicator that can talk to any other computer in the world and access all the world's information and entertainment.

ALAN: What if it were true? That wouldn't change human nature. Will the power of technology change that?

JOHNNY: What do you think?

ALAN: Well, we have electricity, aircraft, and telephones today. None of these existed a century ago. But human nature hasn't evolved in that time. We are still driven by ego, greed, and passions, as were our ancestors. Just as high explosives have made military generals more powerful, these new technologies will empower politicians and charlatans, leaders and rogues. Someone with access to the world's information may use it to build more powerful bombs or breed a virus more dangerous than smallpox. Who is to say what might happen? And who can police that?

CLAUDE: Another problem is that I think that humans suffer from a paradox of choice. When given too many choices, they pick one and stick to it, no matter what, because the sheer mental effort in making a choice is unpleasant. For example, when I go to the grocery store and have a choice of 30 types of bread, I pick one that looks reasonable, then get the same one every time after. In the same way, when given access to the world's information, I think people will filter out all but a few things that they like, and live inside an information bubble. A Republican bubble. A Democrat bubble. A Christian bubble. An atheist bubble. Then, they will ignore all else. This might result in a loss of societal consensus on issues that matter.

JOHNNY: Yes, there is a danger that each person will create their own reality and be reachable only by gifted influencers who can break through the people's bubbles. Politicians of the future will need to be communicators, not makers of policy, yet could exercise vast powers. A dangerous mix!

ALAN: I admit this does look rather bad, but surely there are some benefits arising from our work! I can think of at least one: the spread of education. No one need be deprived of access to education by accident of geography.

JOHNNY: There's much more. Doctors around the world can learn from each other. And by allowing anyone to share their ideas without intervention by editors and censors, creativity might flourish.

CLAUDE: Like-minded people around the world could find each other and perhaps offer mutual support. This might counteract the heavy hand of prejudice over time.

ALAN: As always, we seem to be on the brink of a world that could either tip into cataclysm or grow into glory. It has ever been thus. Perhaps the power we enable will cure the very ills our work creates!

The three look at each other, with half-bemused smiles.

NARRATOR: One of these young men has figured out how to represent all information as bits and how to communicate those bits. The second has dreamt up machines that could manipulate bits in arbitrary ways. And the third knows how to actually build these machines. Each has an inkling that their work would give rise to a new world that they might not much like, but they could not help but create.

CLAUDE: [*He shrugs, walks over to the phonograph, and puts on a jazz LP.*] True. We can only warn, work, and wait. Let's drink to that!

NARRATOR: We are the inheritors of Claude, Alan, and Johnny's work. They are Shannon, Turing, and von Neumann, who gave rise to the digital world. Where next? I wonder!

The scientific ideas in this work are factual and the characters are real, but the meeting and the conversation are, of course, fiction.

Further reading

Shannon CE, 'A mathematical theory of communication' (1948) 27(3) The Bell System Technical Journal 379.

Turing AM, 'On computable numbers, with an application to the Entscheidungsproblem' (1937) 2(1) Proceedings of the London Mathematical Society 230.

Burks AW, Goldstine HH, and von Neumann J, *Preliminary discussion of the logical design of an electronic computer instrument* (1946).