

Transcript for Terence Tao: Hardest Problems in Mathematics, Physics & the Future of AI | Lex Fridman Podcast #472

This is a transcript of Lex Fridman Podcast #472 with Terence Tao. The timestamps in the transcript are clickable links that take you directly to that point in the main video. Please note that the transcript is human generated, and may have errors. Here are some useful links:

- Go back to [this episode's main page](#)
- Watch the [full YouTube version of the podcast](#)

Table of Contents

Here are the loose “chapters” in the conversation. Click link to jump approximately to that part in the transcript:

- [0:00 – Introduction](#)
- [0:49 – First hard problem](#)
- [6:16 – Navier–Stokes singularity](#)
- [26:26 – Game of life](#)
- [33:01 – Infinity](#)
- [38:07 – Math vs Physics](#)
- [44:26 – Nature of reality](#)
- [1:07:09 – Theory of everything](#)
- [1:13:10 – General relativity](#)
- [1:16:37 – Solving difficult problems](#)
- [1:20:01 – AI-assisted theorem proving](#)
- [1:32:51 – Lean programming language](#)
- [1:42:51 – DeepMind's AlphaProof](#)
- [1:47:45 – Human mathematicians vs AI](#)
- [1:57:37 – AI winning the Fields Medal](#)
- [2:04:47 – Grigori Perelman](#)

- 2:17:30 – Twin Prime Conjecture
- 2:34:04 – Collatz conjecture
- 2:40:50 – $P = NP$
- 2:43:43 – Fields Medal
- 2:51:18 – Andrew Wiles and Fermat's Last Theorem
- 2:55:16 – Productivity
- 2:57:55 – Advice for young people
- 3:06:17 – The greatest mathematician of all time

Terence Tao: Hardest Problems in Mathematics, Physics & the Future of AI | ...



Introduction

Lex Fridman

(00:00:00) The following is a conversation with Terence Tao, widely considered to be one of the greatest mathematicians in history, often referred to as The Mozart of Math. He won the Fields Medal and the Breakthrough Prize in Mathematics, and has contributed groundbreaking work to a truly astonishing range of fields in mathematics and physics. This was a huge honor for me for many reasons, including the humility and kindness that Terry showed to me

throughout all our interactions. It means the world. This is the Lex Fridman Podcast. To support it, please check out our sponsors in the description or at LexFridman.com/sponsors. And now, dear friends, here's Terence Tao.

First hard problem

Lex Fridman

(00:00:49) What was the first really difficult research-level math problem that you encountered, one that gave you pause maybe?

Terence Tao

(00:00:57) Well, in your undergraduate education you learn about the really hard impossible problems like the Riemann Hypothesis, the Twin-Primes Conjecture. You can make problems arbitrarily difficult. That's not really a problem. In fact, there's even problems that we know to be unsolvable. What's really interesting are the problems just on the boundary between what we can do rather easily and what are hopeless, but what are problems where existing techniques can do 90% of the job and then you just need that remaining 10%. I think as a PhD student, the Kakeya Problem certainly caught my eye. And it just got solved actually. It's a problem I've worked on a lot in my early research. Historically, it came from a little puzzle by the Japanese mathematician Soichi Kakeya in 1918 or so. So, the puzzle is that you have a needle on the plane or think like driving on a road something, and you want it to execute a U-turn, you want to turn the needle around, but you want to do it in as little space as possible. So, you want to use this little area in order to turn it around, but the needle is infinitely maneuverable. So, you can imagine just spinning it around. As the unit needle, you can spin it around its center, and I think that gives you a disc of area, I think π over four. Or you can do a three-point U-turn, which is what we teach people in their driving schools to do. And that actually takes area of π over eight, so it's a little bit more efficient than a rotation. And so for a while people thought that was the most efficient way to turn things around, but Besicovitch showed that in fact you could actually turn the needle around using as little area as you wanted. So, 0.01, there was some really fancy multi back and forth U-turn thing that you could do that you could turn a needle around and in so doing it would pass through every intermediate direction. Is

Lex Fridman

(00:02:51) This in the two-dimensional plane?

Terence Tao

(00:02:52) This is in the two-dimensional plane. So, we understand everything in two dimensions. So, the next question is: what happens in three dimensions? So, suppose the Hubble space Telescope is tube in space, and you want to observe every single star in the universe, so you want to rotate the telescope to reach every single direction. And here's unrealistic part, suppose that space is at a premium, which totally is not, you want to occupy as little volume as possible in order to rotate your needle around, in order to see every single star in the sky. How small a volume do you need to do that? And so you can modify Besicovitch's construction. And so if your telescope has zero thickness, then you can use as little volume as you need. That's a simple modification of the two-dimensional construction. But the question is that if your telescope is not zero thickness, but just very, very thin, some thickness δ , what is the minimum volume needed to be able to see every single direction as a function of δ ?

(00:03:45) So, as δ gets smaller, as the needle gets thinner, the volume should go down. But how fast does it go down? And the conjecture was that it goes down very, very slowly like logarithmically roughly speaking, and that was proved after a lot of work. So, this seems like a puzzle. Why is it interesting? So, it turns out to be surprisingly connected to a lot of problems in partial differential equations, in number theory, in geometry, combinatorics. For example, in wave propagation, you splash some water around, you create water waves and they travel in various directions, but waves exhibit both particle and wave-type behavior. So, you can have what's called a wave packet, which is a very localized wave that is localized in space and moving a certain direction in time. And so if you plot it in both space and time, it occupies a region which looks like a tube. What can happen is that you can have a wave which initially is very dispersed, but it all focuses at a single point later in time. You can imagine dropping a pebble into a pond and the ripples spread out, but then if you time-reverse that scenario, and the equations of wave motion are time-reversible, you can imagine ripples that are converging to a single point and then a big splash occurs, maybe even a singularity. And so it's possible to do that. And geometrically what's going on is that there's also light rays, so if this wave represents light, for example, you can imagine this wave as a superposition of photons all traveling at the speed of light.

(00:05:15) They all travel on these light rays and they're all focusing at this one point. So, you can have a very dispersed wave focus into a very concentrated wave at one point in space and time, but then it de-focuses again, it separates. But potentially if the conjecture had a negative solution, so what that meant is that there's a very efficient way to pack tubes pointing different directions to a very, very narrow region of a very narrow volume. Then you

would also be able to create waves that start out some... There'll be some arrangement of waves that start out very, very dispersed, but they would concentrate, not just at a single point, but there'll be a lot of concentrations in space and time. And you could create what's called a blowup, where these waves amplitude becomes so great that the laws of physics that they're governed by are no longer wave equations, but something more complicated and nonlinear.

Navier-Stokes singularity

(00:06:08) And so in mathematical physics, we care a lot about whether certain equations and wave equations are stable or not, whether they can create these singularities. There's a famous unsolved problem called the Navier-Stokes regularity problem. So, the Navier-Stokes equations, equations that govern the fluid flow for incompressible fluids like water. The question asks: if you start with a smooth velocity field of water, can it ever concentrate so much that the velocity becomes infinite at some point? That's called a singularity. We don't see that in real life. If you splash around water in the bathtub, it won't explode on you or have water leaving at the speed of light or anything, but potentially it is possible.

(00:06:49) And in fact, in recent years, the consensus has drifted towards the belief that, in fact, for certain very special initial configurations of, say, water, that singularities can form, but people have not yet been able to actually establish this. The Clay Foundation has these seven Millennium Prize Problems as a \$1 million prize for solving one of these problems, and this is one of them. Of these of these seven, only one of them has been solved, at the Poincare Conjecture [inaudible 00:07:18]. So, the Keakeya Conjecture is not directly directly related to the Navier-Stokes Problem, but understanding it would help us understand some aspects of things like wave concentration, which would indirectly probably help us understand the Navier-Stokes Problem better.

Lex Fridman

(00:07:32) Can you speak to the Navier-Stokes? So, the existence of smoothness, like you said, Millennium Prize Problem, You've made a lot of progress on this one. In 2016, you published a paper, Finite Time Blowup For An Average Three-Dimensional Navier-Stokes Equation. So, we're trying to figure out if this thing... Usually it doesn't blow up, but can we say for sure it never blows up?

Terence Tao

(00:07:56) Right, yeah. So yeah, that is literally the \$1 million question. So, this is what distinguishes mathematicians from pretty much everybody else. If something holds 99.99% of the time, that's good enough for most things. But mathematicians are one of the few people who really care about whether really 100% of all situations are covered by it. So, most fluid, most of the time water does not blow up, but could you design a very special initial state that does this?

Lex Fridman

(00:08:29) And maybe we should say that this is a set of equations that govern in the field of fluid dynamics, trying to understand how fluid behaves. And it's actually turns out to be a really... Fluid is extremely complicated thing to try to model.

Terence Tao

(00:08:43) Yeah, so it has practical importance. So this Clay Prize problem concerns what's called the Incompressible Navier-Stokes, which governs things like water. There's something called the Compressible Navier-Stokes, which governs things like air, and that's particularly important for weather prediction. Weather prediction, it does a lot of computational fluid dynamics. A lot of it's actually just trying to solve the Navier-Stokes equations as best they can. Also gathering a lot of data, so that they can initialize the equation. There's a lot of moving parts, so it's very important from practically.

Lex Fridman

(00:09:09) Why is it difficult to prove general things about the set of equations like it not not blowing up?

Terence Tao

(00:09:17) Short answer is Maxwell's Demon. So, Maxwell's Demon is a concept in thermodynamics. If you have a box of two gases in oxygen and nitrogen, and maybe you start with all the oxygen on one side and nitrogen on the other side, but there's no barrier between them. Then they will mix and they should stay mixed. There's no reason why they should un-mix. But in principle, because of all the collisions between them, there could be some sort of weird conspiracy that maybe there's a microscopic demon called Maxwell's Demon that will... every time an oxygen and nitrogen atom collide, they'll bounce off in such a way that the oxygen sort of drifts onto one side and then nitrogen goes to the other. And you could have an extremely improbable configuration emerge, which we never see, and which statistically it's extremely unlikely, but mathematically it's possible that this can happen and we can't rule that out.

(00:10:06) And this is a situation that shows up a lot in mathematics. A basic example is the digits of pi 3.14159 and so forth. The digits look like they have no pattern, and we believe they have no pattern. On the long-term, you should see as many ones and twos and threes as fours and fives and sixes, there should be no preference in the digits of pi to favor, let's say seven over eight. But maybe there's some demon in the digits of pi that every time you compute more and more digits, it biases one digit to another. And this is a conspiracy that should not happen. There's no reason it should happen, but there's no way to prove it with our current technology. So, getting back to Navier-Stokes, a fluid has a certain amount of energy, and because the fluid is in motion, the energy gets transported around.

(00:10:53) And water is also viscous, so if the energy is spread out over many different locations, the natural viscosity of the fluid will just damp out the energy and will go to zero. And this is what happens when we actually experiment with water. You splash around, there's some turbulence and waves and so forth, but eventually it settles down and the lower the amplitude, the smaller velocity, the more calm it gets. But potentially there is some sort of demon that keeps pushing the energy of the fluid into a smaller and smaller scale, and it'll move faster and faster. And at faster speeds, the effect of viscosity is relatively less. And so it could happen that it creates some sort of what's called a self-similar blob scenario where the energy of the fluid starts off at some large scale and then it all sort of transfers energy into a smaller region of the fluid, which then at a much faster rate moves into an even smaller region and so forth.

(00:11:55) And each time it does this, it takes maybe half as long as the previous one, and then you could actually converge to all the energy concentrating in one point in a finite amount of time. And that's scenario is called finite time blowup. So, in practice, this doesn't happen. So, water is what's called turbulent. So, it is true that if you have a big eddy of water, it will tend to break up into smaller eddies, but it won't transfer all energy from one big eddy into one smaller eddy. It will transfer into maybe three or four, and then those ones split up into maybe three or four small eddies of their own. So the energy gets dispersed to the point where the viscosity can then keep everything under control. But if it can somehow concentrate all the energy, keep it all together, and do it fast enough that the viscous effects don't have enough time to calm everything down, then this blowup can occur.

(00:12:51) So, there were papers who had claimed that, "Oh, you just need to take into account conservation of energy and just carefully use the viscosity and you can keep everything under control for not just the Navier-Stokes, but for many, many types of equations like this."

And so in the past there have been many attempts to try to obtain what's called global regularity for Navier-Stokes, which is the opposite of finite time blowup, that velocity stays smooth. And it all failed. There was always some sign error or some subtle mistake and it couldn't be salvaged.

(00:13:17) So, what I was interested in doing was trying to explain why we were not able to disprove finite time blowup. I couldn't do it for the actual equations of fluids, which are too complicated, but if I could average the equations of motion of Navier-Stokes, basically if I could turn off certain types of ways in which water interacts and only keep the ones that I want. So, in particular, if there's a fluid and it could transfer as energy from a large eddy into this small eddy or this other small eddy, I would turn off the energy channel that would transfer energy to this one and direct it only into this smaller eddy while still preserving the lower conservation energy.

Lex Fridman

(00:13:58) So, you're trying to make a blowup?

Terence Tao

(00:14:00) Yeah, yeah. So, I basically engineer a blowup by changing rules of physics, which is one thing that mathematicians are allowed to do. We can change the equation.

Lex Fridman

(00:14:08) How does that help you get closer to the proof of something?

Terence Tao

(00:14:11) Right. So, it provides what's called an obstruction in mathematics. So, what I did was that basically if I turned off the certain parts of the equation, which usually when you turn off certain interactions, make it less nonlinear, it makes it more regular and less likely to blow up. But I find that by turning off a very well-designed set of interactions, I could force all the energy to blow up in finite time. So, what that means is that if you wanted to prove the regularity for Navier-Stokes for the actual equation, you must use some feature of the true equation, which my artificial equation does not satisfy. So, it rules out certain approaches.

(00:14:55) So, the thing about math, it's not just about taking a technique that is going to work and applying it, but you need to not take the techniques that don't work. And for the problems that are really hard, often though are dozens of ways that you might think might apply to solve the problem, but it's only after a lot of experience that you realize there's no way that these methods are going to work. So, having these counterexamples for nearby

problems rules out... it saves you a lot of time because you're not wasting energy on things that you now know cannot possibly ever work.

Lex Fridman

(00:15:30) How deeply connected is it to that specific problem of fluid dynamics or is this some more general intuition you build up about mathematics?

Terence Tao

(00:15:38) Right. Yeah. So, the key phenomenon that my technique exploits is what's called super-criticality. So, in partial [inaudible 00:15:46] equations, often these equations are like a tug of war between different forces. So, in Navier-Stokes, there's the dissipation force coming from viscosity, and it's very well understood. It's linear, it calms things down. If viscosity was all there was, then nothing bad would ever happen, but there's also transport that energy from... in one location of space can get transported because the fluid is in motion to other locations. And that's a nonlinear effect, and that causes all the problems. So, there are these two competing terms in the Navier-Stokes Equation, the dissipation term and the transport term. If the dissipation term dominates, if it's large, then basically you get regularity. And if the transport term dominates, then we don't know what's going on. It's a very nonlinear situation, it's unpredictable, it's turbulent.

(00:16:32) So, sometimes these forces are in balance at small scales but not in balance at large scales or vice versa. Navier-Stokes is what's called supercritical. So at smaller and smaller scales, the transport terms are much stronger than the viscosity terms. So, the viscosity terms are things that calm things down. And so this is why the problem is hard. In two dimensions, so the Soviet mathematician Ladyzhenskaya, she in the '60s shows in two dimensions there was no blowup. And in two dimensions, the Navier-Stokes Equation is what's called critical, the effect of transport and the effect of viscosity about the same strength even at very, very small scales. And we have a lot of technology to handle critical and also subcritical equations and prove regularity. But for supercritical equations, it was not clear what was going on, and I did a lot of work, and then there's been a lot of follow up showing that for many other types of supercritical equations, you can create all kinds of blowup examples.

(00:17:27) Once the nonlinear effects dominate the linear effects at small scales, you can have all kinds of bad things happen. So, this is sort of one of the main insights of this line of work is that super-criticality versus criticality and subcriticality, this makes a big difference. That's a key qualitative feature that distinguishes some equations for being sort of nice and predictable and... Like planetary motion, there's certain equations that you can predict for

millions of years or thousands at least. Again, it's not really a problem, but there's a reason why we can't predict the weather past two weeks into the future because it's a supercritical equation. Lots of really strange things are going on at very fine scales.

Lex Fridman

(00:18:04) So, whenever there is some huge source of nonlinearity, that can create a huge problem for predicting what's going to happen?

Terence Tao

(00:18:13) Yeah. And if non-linearity is somehow more and more featured and interesting at small scales. There's many equations that are nonlinear, but in many equations you can approximate things by the bulk. So, for example, planetary motion, if you want to understand the orbit of the Moon or Mars or something, you don't really need the microstructure of the seismology of the Moon or exactly how the mass is distributed. Basically, you can almost approximate these planets by point masses, and it's just the aggregate behavior is important. But if you want to model a fluid, like the weather, you can't just say, "In Los Angeles the temperature is this, the wind speed is this." For supercritical equations, the fine scale information is really important.

Lex Fridman

(00:18:54) If we can just linger on the Navier-Stokes Equations a little bit. So, you've suggested, maybe you can describe it, that one of the ways to solve it or to negatively resolve it would be to construct a kind of liquid computer, and then show that the halting problem from computation theory has consequences for fluid dynamics, so show it in that way. Can you describe this idea?

Terence Tao

(00:19:22) Right, yeah. So, this came out of this work of constructing this average equation that blew up. So, as part of how I had to do this, so there's this naive way to do it, you just keep pushing. Every time you get one scale, you push it immediately to the next scale as fast as possible. This is sort of the naive way to force blowup. It turns out in five and higher dimensions, this works, but in three dimensions there was this funny phenomenon that I discovered, that if you change laws of physics, you just always keep trying to push the energy into smaller and smaller scales, what happens is that the energy starts getting spread out into many scales at once, so that you have energy at one scale. You're pushing it into the next scale, and then as soon as it enters that scale, you also push it to the next scale, but there's still some energy left over from the previous scale.

(00:20:16) You're trying to do everything at once, and this spreads out the energy too much. And then it turns out that it makes it vulnerable for viscosity to come in and actually just damp out everything. So, it turns out this direct abortion doesn't actually work. There was a separate paper by some other authors that actually showed this in three dimensions. So, what I needed was to program a delay, so kind of like airlocks. So, I needed an equation which would start with a fluid doing something at one scale, it would push this energy into the next scale, but it would stay there until all the energy from the larger scale got transferred. And only after you pushed all the energy in, then you open the next gate and then you push that in as well.

(00:21:01) So, by doing that, the energy inches forward, scale by scale in such a way that it's always localized at one scale at a time, and then it can resist the effects of viscosity because it's not dispersed. So, in order to make that happen, I had to construct a rather complicated nonlinearity. And it was basically... It was constructed like an electronic circuit. So, I actually thank my wife for this because she was trained as an electrical engineer, and she talked about she had to design circuits and so forth. And if you want a circuit that does a certain thing, maybe have a light that flashes on and then turns off and then on and off. You can build it from more primitive components, capacitors and resistors and so forth, and you have to build a diagram.

(00:21:47) And these diagrams, you can sort of follow up your eyeballs and say, "Oh yeah, the current will build up here and it will stop, and then it will do that." So, I knew how to build analog of basic electronic components, like resistors and capacitors and so forth. And I would stack them together in such a way that I would create something that would open one gate. And then there'd be a clock, and then once the clock hits a certain threshold, it would close it. It would become a Rube Goldberg type machine, but described mathematically. And this ended up working. So, what I realized is that if you could pull the same thing off for the actual equations, so if the equations of water support a computation... So, you can imagine a steampunk, but it's really water-punk type of thing where... So, modern computers are electronic, they're powered by electrons passing through very tiny wires and interacting with other electrons and so forth.

(00:22:39) But instead of electrons, you can imagine these pulses of water moving a certain velocity. And maybe there are two different configurations corresponding to a bit being up or down. Probably that if you had two of these moving bodies of water collide, they would come out with some new configuration, which would be something like an AND gate or OR gate,

that the output would depend in a very predictable way on the inputs. And you could chain these together and maybe create a Turing machine. And then you have computers which are made completely out of water. And if you have computers, then maybe you can do robotics, so hydraulics and so forth. And so you could create some machine which is basically a fluid analog, what's called a von Neumann machine.

(00:23:26) So, von Neumann proposed if you want to colonize Mars, the sheer cost of transporting people in machines to Mars is just ridiculous, but if you could transport one machine to Mars, and this machine had the ability to mine the planet, create some more materials, smelt them and build more copies of the same machine, then you could colonize a whole planet over time. So, if you could build a fluid machine, which yeah, so it's a fluid robot. And what it would do, its purpose in life, it's programmed so that it would create a smaller version of itself in some sort of cold state. It wouldn't start just yet. Once it's ready, the big robot configuration of water would transfer all its energy into the smaller configuration and then power down. And then they clean itself up, and then what's left is this newest state which would then turn on and do the same thing, but smaller and faster.

(00:24:19) And then the equation has a certain scaling symmetry. Once you do that, it can just keep iterating. So, this, in principle, would create a blowup for the actual Navier-Stokes. And this is what I managed to accomplish for this average Navier-Stokes. So, it provided this sort of roadmap to solve the problem. Now, this is a pipe dream because there are so many things that are missing for this to actually be a reality. So, I can't create these basic logic gates. I don't have these special configurations of water. There's candidates, these include vortex rings that might possibly work. But also analog computing is really nasty compared to digital computing because there's always errors. You have to do a lot of error correction along the way.

(00:25:05) I don't know how to completely power down the big machine, so it doesn't interfere the writing of the smaller machine, but everything in principle can happen. It doesn't contradict any of the laws of physics, so it's sort of evidence that this thing is possible. There are other groups who are now pursuing ways to make Navier-Stokes blow up, which are nowhere near as ridiculously complicated as this. They actually are pursuing much closer to the direct self-similar model, which can... It doesn't quite work as is, but there could be some simpler scheme they want to just describe to make this work.

Lex Fridman

(00:25:40) There is a real leap of genius here to go from Navier-Stokes to this Turing machine. So, it goes from what the self-similar blob scenario that you're trying to get the smaller and smaller blob to now having a liquid Turing machine gets smaller and smaller and smaller, and somehow seeing how that could be used to say something about a blowup. That's a big leap.

Game of life

Terence Tao

(00:26:08) So, there's precedent. So, the thing about mathematics is that it's really good at spotting connections between what you might think of as completely different problems, but if the mathematical form is the same, you can draw a connection. So, there's a lot of previously on what called cellular automata, the most famous of which is Conway's Game of Life. There's this infinite discrete grid, and at any given time, the grid is either occupied by a cell or it's empty. And there's a very simple rule that tells you how these cells evolve. So, sometimes cells live and sometimes they die. And when I was a student, it was a very popular screen saver to actually just have these animations go on, and they look very chaotic. In fact, they look a little bit like turbulent flow sometimes, but at some point people discovered more and more interesting structures within this Game of Life. So, for example, they discovered this thing called glider.

(00:27:00) So, a glider is a very tiny configuration of four or five selves which evolves and it just moves at a certain direction. And that's like this vortex rings [inaudible 00:27:09]. Yeah, so this is an analogy, the Game of Life is a discrete equation, and the fluid Navier-Stokes is a continuous equation, but mathematically they have some similar features. And so over time people discovered more and more interesting things that you could build within the Game of Life. The Game of Life is a very simple system. It only has like three or four rules to do it, but you can design all kinds of interesting configurations inside it. There's some called a glider gun that does nothing that spit out gliders one at a time. And then after a lot of effort, people managed to create AND gates and OR gates for gliders.

(00:27:48) There's this massive ridiculous structure, which if you have a stream of gliders coming in here and a stream of gliders coming in here, then you may produce extreme gliders coming out. Maybe if both of the streams have gliders, then there'll be an output stream, but if only one of them does, then nothing comes out. So, they could build something like that. And once you could build these basic gates, then just from software engineering, you

can build almost anything. You can build a Turing machine. It's enormous steampunk type things. They look ridiculous. But then people also generated self-replicating objects in the Game of Life, a massive machine, a [inaudible 00:28:31] machine, which over a huge period of time and always look like glider guns inside doing these very steampunk calculations. It would create another version of itself which could replicate.

Lex Fridman

(00:28:42) That's so incredible.

Terence Tao

(00:28:42) A lot of this was like community crowdsourced by amateur mathematicians actually. So, I knew about that work. And so that is part of what inspired me to propose the same thing with Navier-Stokes. Seriously, analog is much worse than digital. It's going to be... You can't just directly take deconstructions in the Game of Life and plunk them in. But again, it shows it's possible.

Lex Fridman

(00:29:06) There's a kind of emergence that happens with these cellular automata local rules... maybe it's similar to fluids, I don't know, but local rules operating at scale can create these incredibly complex dynamic structures. Do you think any of that is amenable to mathematical analysis? Do we have the tools to say something profound about that?

Terence Tao

(00:29:34) The thing is, you can get these emergent very complicated structures, but only with very carefully prepared initial conditions. So, these glider guns and gates and self-propelled machines, if you just plunk on randomly some cells and you unlink them, you will not see any of these. And that's the analogous situation with Navier-Stokes again, that with typical initial conditions, you will not have any of this weird computation going on. But basically through engineering, by specially designing things in a very special way, you can make clever constructions.

Lex Fridman

(00:30:07) I wonder if it's possible to prove the negative of... basically prove that only through engineering can you ever create something interesting.

Terence Tao

(00:30:16) Yeah. This is a recurring challenge in mathematics that I call the dichotomy between structure and randomness, that most objects that you can generate in mathematics

are random. They look like random, like the digital supply, well, we believe is a good example. But there's a very small number of things that have patterns. But now, you can prove something has a pattern by just constructing... If something has a simple pattern and you have a proof that it does something like repeat itself every so often, you can do that and you can prove that... For example, you can prove that most sequences of digits have no pattern. So, if you just pick digits randomly, there's something called law of large numbers. It tells you you're going to get as many ones as twos in the long run. But we have a lot fewer tools to...

(00:31:01) If I give you a specific pattern like the digits of pi, how can I show that this doesn't have some weird pattern to it? Some other work that I spent a lot of time on is to prove what are called structure theorems or inverse theorems that give tests for when something is very structured. So, some functions are what's called additive. If you have a function of natural numbers of the natural numbers, so maybe two maps to four, three maps to six and so forth, some functions are what's called additive, which means that if you add two inputs together, the output gets added as well. For example, a multiply by constant. If you multiply a number by 10... If you multiply A plus B by 10, that's the same as multiplying A by 10 and B by 10, and then adding them together. So, some functions are additive, some functions are kind of additive but not completely additive.

(00:31:47) So, for example, if I take a number, and I multiply by the square of two and I take the integer part of that, so 10 by square root of two is like 14 point something, so 10 up to 14, 20 up to 28. So, in that case, additivity is true then, so 10 plus 10 is 20 and 14 plus 14 is 28. But because of this rounding, sometimes there's round-up errors, and sometimes when you add A plus A, this function doesn't quite give you the sum of the two individual outputs, but the sum plus/minus one. So, it's almost additive, but not quite additive.

(00:32:21) So, there's a lot of useful results in mathematics, and I've worked a lot on developing things like this, to the effect that if a function exhibits some structure like this, then it's basically there's a reason for why it's true. And the reason is because there's some other nearby function, which is actually completely structured, which is explaining this sort of partial pattern that you have. And so if you have these inverse theorems, it creates this dichotomy that either the objects that you study are either have no structure at all or they are somehow related to something kind of structured. And in either way, in either case, you can make progress. A good example of this is that there's this old theorem in mathematics-

Terence Tao

(00:33:01) A good example of this is that there's this old theorem in mathematics called Szemerédi's Theorem, proven in the 1970s. It concerns trying to find a certain type of pattern in a set of numbers, the patterns of arithmetic progression. Things like three, five, and seven or 10, 15 and 20, and Szemerédi, Endre Szemerédi proved that any set of numbers that are sufficiently big, what's called positive density, has arithmetic progressions in it of any length you wish.

(00:33:28) For example, the odd numbers have a density of one half, and they contain arithmetic progressions of any length. So in that case, it's obvious, because the odd numbers are really, really structured. I can just take 11, 13, 15, 17, I can easily find arithmetic progressions in that set, but Szemerédi's theorem also applies to random sets. If I take a set of odd numbers and I flip a coin for each number, and I only keep the numbers for which I got a heads... So I just flip coins, I just randomly take out half the numbers, I keep one half. That's a set that has no patterns at all, but just from random fluctuations, you will still get a lot of arithmetic progressions in that set.

Lex Fridman

(00:34:10) Can you prove that there's arithmetic progressions of arbitrary length within a random-

Terence Tao

(00:34:17) Yes. Have you heard of the infinite monkey theorem? Usually, mathematicians give boring names to theorems, but occasionally they give colorful names.

Lex Fridman

(00:34:24) Yes.

Terence Tao

(00:34:24) The popular version of the infinite monkey theorem is that if you have an infinite number of monkeys in a room, each with typewriter, they type out text randomly, almost surely, one of them is going to generate the entire script of Hamlet, or any other finite string of text. It'll just take some time, quite a lot of time, actually, but if you have an infinite number, then it happens.

(00:34:44) So basically, the theorem is that if you take an infinite string of digits or whatever, eventually any finite pattern you wish will emerge. It may take a long time, but it will eventu-

ally happen. In particular, arithmetic progressions of any length will eventually happen, but you need an extremely long random sequence for this to happen.

Lex Fridman

(00:35:04) I suppose that's intuitive. It's just infinity.

Terence Tao

(00:35:08) Yeah, infinity absorbs a lot of sins.

Lex Fridman

(00:35:11) Yeah. How we humans supposed to deal with infinity?

Terence Tao

(00:35:15) Well, you can think of infinity as an abstraction of a finite number of which you do not have a bound. So nothing in real life is truly infinite, but you can ask yourself questions like, "What if I had as much money as I wanted?", or, "What if I could go as fast as I wanted?", and a way in which mathematicians formalize that is mathematics has found a formalism to idealize, instead of something being extremely large or extremely small, to actually be exactly infinite or zero, and often the mathematics becomes a lot cleaner when you do that. I mean, in physics, we joke about assuming spherical cows, real world problems have got all kinds of real world effects, but you can idealize, send some things to infinity, send some things to zero, and the mathematics becomes a lot simpler to work within.

Lex Fridman

(00:36:06) I wonder how often using infinity forces us to deviate from the physics of reality.

Terence Tao

(00:36:17) So there's a lot of pitfalls. So we spend a lot of time in undergraduate math classes teaching analysis, and analysis is often about how to take limits and whether...

(00:36:28) So for example, $A + B$ is always $B + A$. So when you have a finite number of terms and you add them, you can swap them and there's no problem, but when you have an infinite number of terms, they're these sort of show games you can play where you can have a series which converges to one value, but you rearrange it, and it suddenly converges to another value, and so you can make mistakes. You have to know what you're doing when you allow infinity. You have to introduce these epsilons and deltas, and there's a certain type of wave of reasoning that helps you avoid mistakes.

(00:36:58) In more recent years, people have started taking results that are true in infinite limits and what's called finitizing them. So you know that something's true eventually, but you

don't know when. Now give me a rate. So such... If I don't have an infinite number of monkeys, but a large finite number of monkeys, how long do I have to wait for Hamlet to come out? That's a more quantitative question, and this is something that you can attack by purely finite methods, and you can use your finite intuition, and in this case, it turns out to be exponential in the length of the text that you're trying to generate.

(00:37:36) So this is why you never see the monkeys create Hamlet. You can maybe see them create a four letter word, but nothing that big, and so I personally find once you finitize an infinite statement, it does come much more intuitive, and it's no longer so weird.

Lex Fridman

(00:37:51) So even if you're working with infinity, it's good to finitize so that you can have some intuition?

Terence Tao

(00:37:57) Yeah, the downside is that the finitized groups are just much, much messier. So the infinite ones are found first usually, decades earlier, and then later on, people finitize them.

Math vs Physics

Lex Fridman

(00:38:07) So since we mentioned a lot of math and a lot of physics, what is the difference between mathematics and physics as disciplines, as ways of understanding, of seeing the world? Maybe we can throw engineering in there, you mentioned your wife is an engineer, give it new perspective on circuits. So this different way of looking at the world, given that you've done mathematical physics, so you've worn all the hats.

Terence Tao

(00:38:30) Right. So I think science in general is interaction between three things. There's the real world, there's what we observe of the real world, observations, and then our mental models as to how we think the world works.

(00:38:46) We can't directly access reality. All we have are the observations, which are incomplete and they have errors, and there are many, many cases where we want to know, for example, what is the weather like tomorrow, and we don't yet have the observation, but we'd like to. A prediction.

(00:39:04) Then we have these simplified models, sometimes making unrealistic assumptions, spherical cow type things. Those are the mathematical models.

(00:39:11) Mathematics is concerned with the models. Science collects the observations, and it proposes the models that might explain these observations. What mathematics does, we stay within the model, and we ask what are the consequences of that model? What observations, what predictions would the model make of future observations, or past observations? Does it fit? Observe data?

(00:39:35) So there's definitely a symbiosis. I guess mathematics is unusual among other disciplines is that we start from hypotheses, like the axioms of a model, and ask what conclusions come up from that model. In almost any other discipline, you start with the conclusions. "I want to do this. I want to build a bridge, I want to make money, I want to do this," and then you find the paths to get there. There's a lot less sort of speculation about, "Suppose I did this, what would happen?". Planning and modeling. Speculative fiction maybe is one other place, but that's about it, actually. Most of the things we do in life is conclusions driven, including physics and science. I mean, they want to know, "Where is this asteroid going to go? What is the weather going to be tomorrow?", but mathematics also has this other direction of going from the axioms.

Lex Fridman

(00:40:32) What do you think... There is this tension in physics between theory and experiment. What do you think is the more powerful way of discovering truly novel ideas about reality?

Terence Tao

(00:40:42) Well, you need both, top down and bottom up. It's really an interaction between all these... So over time, the observations and the theory and the modeling should both get closer to reality, but initially, and this is always the case out there, they're always far apart to begin with, but you need one to figure out where to push the other.

(00:41:04) So if your model is predicting anomalies that are not predicted by experiment, that tells experimenters where to look to find more data to refine the models. So it goes back and forth.

(00:41:21) Within mathematics itself, there's also a theory and experimental component. It's just that until very recently, theory has dominated almost completely. 99% of mathematics is theoretical mathematics, and there's a very tiny amount of experimental mathematics.

People do do it. If they want to study prime numbers or whatever, they can just generate large data sets.

(00:41:41) So once we had the computers, we had to do it a little bit. Although even before... Well, like Gauss for example, he discovered a conjecture, the most basic theorem in number theory, called the prime number theorem, which predicts how many primes up to a million, up to a trillion. It's not an obvious question, and basically what he did was that he computed, mostly by himself, but also hired human computers, people whose professional job it was to do arithmetic, to compute the first hundred thousand primes or something, and made tables and made a prediction. That was an early example of experimental mathematics, but until very recently, it was not...

(00:42:22) I mean, theoretical mathematics was just much more successful. Of course, doing complicated mathematical computations was just not feasible until very recently, and even nowadays, even though we have powerful computers, only some mathematical things can be explored numerically.

(00:42:37) There's something called the combinatorial explosion. If you want us to study, for example, Szemerédi's theorem, you want to study all possible subsets of numbers one to a thousand. There's only 1000 numbers. How bad could it be? It turns out the number of different subsets of one to a thousand is two to the power of 1000, which is way bigger than any computer can currently enumerate.

(00:42:59) So there are certain math problems that very quickly become just intractable to attack by direct brute force computation. Chess is another famous example. The number of chess positions, we can't get a computer to fully explore, but now we have AI, we have tools to explore this space, not with 100% guarantees of success, but with experiment. So we can empirically solve chess now. For example, we have very, very good AIs that don't explore every single position in the game tree, but they have found some very good approximation, and people are using actually these chess engines to do experimental chess. They're revisiting old chess theories about, "Oh, when you do this type of opening... This is a good type of move, this is not," and they can use these chess engines to actually refine, and in some cases, overturn conventional wisdom about chess, and I do hope that that mathematics will have a larger experimental component in the future, perhaps powered by AI.

Lex Fridman

(00:44:05) We'll, of course, talk about that, but in the case of chess, and there's a similar thing in mathematics, I don't believe it's providing a kind of formal explanation of the different po-

sitions. It's just saying which position is better or not that you can intuit as a human being, and then from that, we humans can construct a theory of the matter.

Nature of reality

(00:44:27) You've mentioned the Plato's cave allegory. In case people don't know, it's where people are observing shadows of reality, not reality itself, and they believe what they're observing to be reality. Is that, in some sense, what mathematicians and maybe all humans are doing, is looking at shadows of reality? Is it possible for us to truly access reality?

Terence Tao

(00:44:55) Well, there are these three ontological things. There's actual reality, there's observations and our models, and technically they are distinct, and I think they will always be distinct, but they can get closer over time, and the process of getting closer often means that you have to discard your initial intuitions. So astronomy provides great examples, like an initial model of the world is flat because it looks flat and it's big, and the rest of the universe, the skies, is not. The sun, for example, looks really tiny.

(00:45:38) So you start off with a model, which is actually really far from reality, but it fits the observations that you have. So things look good, but over time, as you make more and more observations, bring it closer to reality, the model gets dragged along with it, and so over time, we had to realize that the earth was round, that it spins, it goes around the solar system, solar system goes around the galaxy, and so on and so forth, and the universe was expanding. Expansions is self-expanding, accelerating, and in fact, very recently this year... So even the acceleration of the universe itself, this evidence now is non-constant.

Lex Fridman

(00:46:13) The explanation behind why that is...

Terence Tao

(00:46:16) It's catching up.

Lex Fridman

(00:46:18) It's catching up. I mean, it's still the dark matter, dark energy, this kind of thing.

Terence Tao

(00:46:23) We have a model that explains, that fits the data really well. It just has a few parameters that you have to specify. So people say, "Oh, that's fudge factors. With enough fudge

factors, you can explain anything,” but the mathematical point over the model is that you want to have fewer parameters in your model and data points in your observational set.

(00:46:43) So if you have a model with 10 parameters that explains 10 observations, that is a completely useless model, it's what's called overfitted, but if you have a model with two parameters and it explains a trillion observations, which is basically the dark matter model, I think it has 14 parameters, and it explains petabytes of data that the astronomers have.

(00:47:06) You can think of a theory. One way to think about a physical mathematical theory is it's a compression of the universe, and a data compression. So you have these petabytes of observations, you like to compress it to a model which you can describe in five pages and specify a certain number of parameters, and if it can fit, to reasonable accuracy, almost all of your observations, the more compression that you make, the better your theory.

Lex Fridman

(00:47:32) In fact, one of the great surprises of our universe and of everything in it is that it's compressible at all. That's the unreasonable effectiveness of mathematics

Terence Tao

(00:47:40) Yeah, Einstein had a quote like that. “The most incomprehensible thing about the universe is that it is comprehensible.”

Lex Fridman

(00:47:45) Right, and not just comprehensible. You can do an equation like $E=MC^2$.

Terence Tao

(00:47:49) There is actually some possible explanation for that. So there's this phenomenon in mathematics called universality. So, many complex systems at the macro scale are coming out of lots of tiny interactions at the macro scale, and normally, because of the commutative explosion, you would think that the macro scale equations must be infinitely, exponentially more complicated than the macro scale ones, and they are, if you want to solve them completely exactly. If you want to model all the atoms in a box of air...

(00:48:21) Like Avogadro's number is humongous. There's a huge number of particles. If you actually tried to track each one, it'll be ridiculous, but certain laws emerge at the microscopic scale that almost don't depend on what's going on at the macro scale, or only depend on a very small number of parameters.

(00:48:35) So if you want to model a gas of a quintillion particles in a box, you just need to know is temperature and pressure and volume, and a few parameters, like five or six, and it

models almost everything you need to know about these 10 to 23 or whatever particles. So we don't understand universality anywhere near as we would like mathematically, but there are much simpler toy models where we do have a good understanding of why universality occurs. The most basic one is the central limit theorem that explains why the bell curve shows up everywhere in nature, that so many things are distributed by what's called a Gaussian distribution, famous bell curve. There's now even a meme with this curve.

Lex Fridman

(00:49:18) And even the meme applies broadly. The universality to the meme.

Terence Tao

(00:49:22) Yes, you can go meta if you like, but there are many, many processes. For example, you can take lots of independent random variables and average them together in various ways. You can take a simple average or more complicated average, and we can prove in various cases that these bell curves, these Gaussians, emerge, and it is a satisfying explanation.

(00:49:44) Sometimes they don't. So if you have many different inputs and they're all correlated in some systemic way, then you can get something very far from a bell curve to show up, and this is also important to know when [inaudible 00:49:55] fails. So universality is not a 100% reliable thing to rely on. The global financial crisis was a famous example of this. People thought that mortgage defaults had this sort of Gaussian type behavior, that if a population of a hundred thousand Americans with mortgages ask what proportion of them would default on their mortgages, if everything was de-correlated, it would be an asset bell curve, and you can manage risk of options and derivatives and so forth, and there's a very beautiful theory, but if there are systemic shocks in the economy that can push everybody to default at the same time, that's very non-Gaussian behavior, and this wasn't fully accounted for in 2008.

(00:50:45) Now I think there's some more awareness that this systemic risk is actually a much bigger issue, and just because the model is pretty and nice, it may not match reality. So the mathematics of working out what models do is really important, but also the science of validating when the models fit reality and when they don't... You need both, but mathematics can help, because for example, these central limit theorems, it tells you that if you have certain axioms like non-correlation, that if all the inputs were not correlated to each other, then you have this Gaussian behavior and things are fine. It tells you where to look for weaknesses in the model.

(00:51:25) So if you have a mathematical understanding of Szemerédi's theorem, and someone proposes to use these Gaussian [inaudible 00:51:32] or whatever to model default risk, if you're mathematically trained, you would say, "Okay, but what are the systemic correlation between all your inputs?", and so then you can ask the economist, "How much of a risk is that?", and then you can go look for that. So there's always this synergy between science and mathematics.

Lex Fridman

(00:51:52) A little bit on the topic of universality, you're known and celebrated for working across an incredible breadth of mathematics, reminiscent of Hilbert a century ago. In fact, the great Fields Medal winning mathematician Tim Gowers has said that you are the closest thing we get to Hilbert. He's a colleague of yours.

Terence Tao

(00:52:16) Oh yeah, good friend.

Lex Fridman

(00:52:16) But anyway, so you are known for this ability to go both deep and broad in mathematics. So you're the perfect person to ask. Do you think there are threads that connect all the disparate areas of mathematics? Is there a kind of a deep, underlying structure to all of mathematics?

Terence Tao

(00:52:36) There's certainly a lot of connecting threads, and a lot of the progress of mathematics can be represented by taking... By stories of two fields of mathematics that were previously not connected, and finding connections.

(00:52:50) An ancient example is geometry and number theory. So in the times of the ancient Greeks, these were considered different subjects. I mean, mathematicians worked on both. Euclid worked both on geometry, most famously, but also on numbers, but they were not really considered related. I mean, a little bit, like you could say that this length was five times this length because you could take five copies of this length and so forth, but it wasn't until Descartes, who developed analytical geometry, that you can parameterize the plane, a geometric object, by two real numbers. So geometric problems can be turned into problems about numbers.

(00:53:35) Today this feels almost trivial. There's no content to this. Of course, a plane is X and Y , because that's what we teach and it's internalized, but it was an important develop-

ment that these two fields were unified, and this process has just gone on throughout mathematics over and over again. Algebra and geometry were separated, and now we have this fluid, algebraic geometry that connects them, and over and over again, and that's certainly the type of mathematics that I enjoy the most.

(00:54:06) I think there's sort of different styles to being a mathematician. I think hedgehogs and fox... A fox knows many things a little bit, but a hedgehog knows one thing very, very well, and in mathematics, there's definitely both hedgehogs and foxes, and then there's people who can play both roles, and I think ideal collaboration, British mathematicians involves very... You need some diversity, like a fox working with many hedgehogs or vice versa, but I identify mostly as a fox, certainly. I like arbitrage, somehow. Learning how one field works, learning the tricks of that wheel, and then going to another field which people don't think is related, but I can adapt the tricks.

Lex Fridman

(00:54:49) So see the connections between the fields.

Terence Tao

(00:54:52) Yeah. So there are other mathematicians who are far deeper than I am. They're really hedgehogs. They know everything about one field, and they're much faster and more effective in that field, but I can give them these extra tools.

Lex Fridman

(00:55:05) I mean, you've said that you can be both a hedgehog and the fox, depending on the context, depending on the collaboration. So can you, if it's at all possible, speak to the difference between those two ways of thinking about a problem? Say you're encountering a new problem, searching for the connections versus very singular focus.

Terence Tao

(00:55:26) I'm much more comfortable with the fox paradigm. Yeah. So yeah, I like looking for analogies, narratives. I spend a lot of time... If there's a result, I see it in one field, and I like the result, it's a cool result, but I don't like the proof, it uses types of mathematics that I'm not super familiar with, I often try to re-prove it myself using the tools that I favor.

(00:55:53) Often, my proof is worse, but by the exercise they're doing, so I can say, "Oh, now I can see what the other proof was trying to do," and from that, I can get some understanding of the tools that are used in that field. So it's very exploratory, very... Doing crazy things in crazy fields and reinventing the wheel a lot, whereas the hedgehog style is, I think, much

more scholarly. You're very knowledge-based. You stay up to speed on all the developments in this field, you know all the history, you have a very good understanding of exactly the strengths and weaknesses of each particular technique. I think you rely a lot more on calculation than sort of trying to find narratives. So yeah, I can do that too, but other people are extremely good at that.

Lex Fridman

(00:56:44) Let's step back and maybe look at a bit of a romanticized version of mathematics. So I think you've said that early on in your life, math was more like a puzzle-solving activity when you were young. When did you first encounter a problem or proof where you realized math can have a kind of elegance and beauty to it?

Terence Tao

(00:57:11) That's a good question. When I came to graduate school in Princeton, so John Conway was there at the time, he passed away a few years ago, but I remember one of the very first research talks I went to was a talk by Conway on what he called extreme proof.

(00:57:28) So Conway just had this amazing way of thinking about all kinds of things in a way that you wouldn't normally think of. So he thought proofs themselves as occupying some sort of space. So if you want to prove something, let's say that there's infinitely many primes, you have all different proofs, but you could rank them in different axes. Some proofs are elegant, some proofs are long, some proofs are elementary and so forth, and so there's this cloud, so the space of all proofs itself has some sort of shape, and so he was interested in extreme points of this shape. Out of all these proofs, what is one of those, the shortest, at the expense of everything else, or the most elementary or whatever?

(00:58:09) So he gave some examples of well-known theorems, and then he would give what he thought was the extreme proof in these different aspects. I just found that really eye-opening, that it's not just getting a proof for a result that was interesting, but once you have that proof, trying to optimize it in various ways, that proofing itself had some craftsmanship to it.

(00:58:40) It's certainly informed my writing style, like when you do your math assignments and as you're an undergraduate, your homework and so forth, you're sort of encouraged to just write down any proof that works and hand it in, and as long as it gets a tick mark, you move on, but if you want your results to actually be influential and be read by people, it can't just be correct. It should also be a pleasure to read, motivated, be adaptable to generalize to other things. It's the same in many other disciplines, like coding. There's a lot of analogies be-

tween math and coding. I like analogies, if you haven't noticed. You can code something, spaghetti code, that works for a certain task, and it's quick and dirty and it works, but there's lots of good principles for writing code well so that other people can use it, build upon it so it has fewer bugs and whatever, and there's similar things with mathematics.

Lex Fridman

(00:59:37) Yeah, first of all, there's so many beautiful things there, and [inaudible 00:59:42] is one of the great minds in mathematics ever, and computer science, just even considering the space of proofs and saying, "Okay, what does this space look like, and what are the extremes?"

(00:59:56) Like you mentioned, coding as an analogy is interesting, because there's also this activity called the code golf, which I also find beautiful and fun, where people use different programming languages to try to write the shortest possible program that accomplishes a particular task, and I believe there's even competitions on this, and it's also a nice way to stress test not just the programs, or in this case, the proofs, but also the different languages. Maybe that's a different notation or whatever to use to accomplish a different task.

Terence Tao

(01:00:31) Yeah, you learn a lot. I mean, it may seem like a frivolous exercise, but it can generate all these insights, which, if you didn't have this artificial objective to pursue, you might not see...

Lex Fridman

(01:00:43) What, to you, is the most beautiful or elegant equation in mathematics? I mean, one of the things that people often look to in beauty is the simplicity. So if you look at $E=MC^2$... So when a few concepts come together, that's why the Euler identity is often considered the most beautiful equation in mathematics. Do you find beauty in that one, in the Euler identity?

Terence Tao

(01:01:08) Yeah. Well, as I said, what I find most appealing is connections between different things that... So if you... Pi equals minus one. So yeah, people use all the fundamental constants. Okay. I mean, that's cute, but to me...

(01:01:24) So the exponential function, which is by Euler, was to measure exponential growth. So compound interest or decay, anything which is continuously growing, continuously decreasing, growth and decay, or dilation or contraction, is modeled by the exponential

function, whereas π comes around from circles and rotation, right? If you want to rotate a needle, for example, a hundred degrees, you need rotate by π radians, and i , complex numbers, represents the swapping imaginary axes of a 90 degree rotation. So a change in direction.

(01:01:53) So the exponential function represents growth and decay in the direction that you already are. When you stick an i in the exponential, now instead of motion in the same direction as your current position, the motion is at a right angle to your current position. So rotation, and then, so $e^{i\pi}$ equals minus one tells you that if you rotate for a time π , you end up at the other direction. So it unifies geometry through dilation and exponential growth or dynamics through this act of complexification, rotation by i . So it connects together all these two as mathematics, dynamics, geometry and complex numbers. They're all considered almost... They were all next-door neighbors in mathematics because of this identity.

Lex Fridman

(01:02:37) Do you think the thing you mentioned as Q , the collision of notations from these disparate fields, is just a frivolous side effect, or do you think there is legitimate value in when notation... Although our old friends come together in the night?

Terence Tao

(01:02:54) Well, it's confirmation that you have the right concepts. So when you first study anything, you have to measure things, and give them names, and initially sometimes, because your model is, again, too far off from reality, you give the wrong things the best names, and you only find out later what's really important.

Lex Fridman

(01:03:14) Physicists can do this sometimes, but it turns out okay.

Terence Tao

(01:03:18) So actually, physics [inaudible 01:03:19] $E=MC^2$. So one of the big things was the E , right? So when Aristotle first came up with his laws of motion, and then Galileo and Newton and so forth, they saw the things they could measure, they could measure mass and acceleration and force and so forth, and so Newtonian mechanics, for example, $F=ma$, was the famous Newton's second law of motion. So those were the primary objects. So they gave them the central billing in the theory.

(01:03:44) It was only later after people started analyzing these equations that there always seemed to be these quantities that were conserved. So in particular, momentum and energy,

and it's not obvious that things have an energy. It's not something you can directly measure the same way you can measure mass and velocity, so both, but over time, people realized that this was actually a really fundamental concept.

(01:04:05) Hamilton, eventually in the 19th century, reformulated Newton's laws of physics into what's called Hamiltonian mechanics, where the energy, which is now called the Hamiltonian, was the dominant object. Once you know how to measure the Hamiltonian of any system, you can describe completely the dynamics like what happens to all the states. It really was a central actor, which was not obvious initially, and this change of perspective really helped when quantum mechanics came along, because the early physicists who studied quantum mechanics, they had a lot of trouble trying to adapt their Newtonian thinking, because everything was a particle and so forth, to quantum mechanics, because everything was a wave, but it just looked really, really weird.

(01:04:51) You ask, "What is the quantum version of $F=ma$?", and it's really, really hard to give an answer to that, but it turns out that the Hamiltonian, which was so secretly behind the scenes in classical mechanics, also is the key object in quantum mechanics, that there's also an object called a Hamiltonian. It's a different type of object. It's what's called an operator rather than a function, but again, once you specify it, you specify the entire dynamics.

(01:05:17) So there's something called Schrodinger's equation that tells you exactly how quantum systems evolve once you have a Hamiltonian. So side by side, they look completely different objects. One involves particles, one involves waves and so forth, but with this centrality, you could start actually transferring a lot of intuition and facts from classical mechanics to quantum mechanics. So for example, in classical mechanics, there's this thing called Noether's theorem. Every time there's a symmetry in a physical system, there was a conservation law. So the laws of physics are translation invariant. Like if I move 10 steps to the left, I experience the same laws of physics as if I was here, and that corresponds to conservation momentum. If I turn around by some angle, again, I experience the same laws of physics. This corresponds to the conservation of angular momentum. If I wait for 10 minutes, I still have the same laws of physics.

Terence Tao

(01:06:00) If I wait for 10 minutes, I still have the same laws of physics. So there's time transition invariance. This corresponds to the law of the conservation of energy. So there's this fundamental connection between symmetry and conservation. And that's also true in quantum mechanics, even though the equations are completely different, but because they're

both coming from the Hamiltonian, the Hamiltonian controls everything, every time the Hamiltonian has a symmetry, the equations will have a conservation law. Once you have the right language, it actually makes things a lot cleaner.

(01:06:32) One of the problems why we can't unify quantum mechanics and general relativity, yet we haven't figured out what the fundamental objects are. For example, we have to give up the notion of space and time being these almost Euclidean-type spaces, and it has to be, we know that at very tiny scales there's going to be quantum fluctuations. There's space-time foam and trying to use Cartesian coordinates X , Y , Z . It's a non-starter, but we don't know what to replace it with. We don't actually have the concepts, the analog Hamiltonian that sort of organized everything.

Theory of everything

Lex Fridman

(01:07:09) Does your gut say that there is a theory of everything, so this is even possible to unify, to find this language that unifies general relativity and quantum mechanics?

Terence Tao

(01:07:19) I believe so. The history of physics has been out of unification much like mathematics over the years. [inaudible 01:07:26] magnetism was separate theories and then Maxwell unified them. Newton unified the motions of heavens for the motions of objects on the Earth and so forth. So it should happen. It's just that, again, to go back to this model of the observations and theory, part of our problem is that physics is a victim of its own success. That our two big theories of physics, general relativity and quantum mechanics are so good now is that together they cover 99.9% of all the observations we can make. And you have to either go to extremely insane particle accelerations or the early universe or things that are really hard to measure in order to get any deviation from either of these two theories to the point where you can actually figure out how to combine together. But I have faith that we've been doing this for centuries and we've made progress before. There's no reason why we should stop.

Lex Fridman

(01:08:18) Do you think you'll be a mathematician that develops a theory of everything?

Terence Tao

(01:08:24) What often happens is that when the physicists need some theory of mathematics, there's often some precursor that the mathematicians worked out earlier. So when Einstein started realizing that space was curved, he went to some mathematician and asked, "Is there some theory of curved space that mathematicians already came up with that could be useful?" And he said, "Oh yeah, I think Riemann came up with something." And so yeah, Riemann had developed Riemannian geometry, which is precisely a theory of spaces that are curved in various general ways, which turned out to be almost exactly what was needed by Einstein's theory. This is going back to weakness and unreasonable effectiveness of mathematics. I think the theories that work well, that explain the universe, tend to also involve the same mathematical objects that work well to solve mathematical problems. Ultimately, they're just both ways of organizing data in useful ways.

Lex Fridman

(01:09:17) It just feels like you might need to go some weird land that's very hard to intuit. You have string T=theory.

Terence Tao

(01:09:25) Yeah, that was a leading candidate for many decades. I think it's slowly pulling out of fashion. It's not matching experiment.

Lex Fridman

(01:09:33) So one of the big challenges of course, like you said, is experiment is very tough because of how effective both theories are. But the other is just you're talking about you're not just deviating from space-time. You're going into some crazy number of dimensions. You're doing all kinds of weird stuff that to us, we've gone so far from this flat earth that we started at, like you mentioned, and now it's very hard to use our limited ape descendants of a cognition to intuit what that reality really is.

Terence Tao

(01:10:10) This is why analogies are so important. So yeah, the round earth is not intuitive because we're stuck on it. But round objects in general, we have pretty good intuition over and we have interest about light works and so forth. And it's actually a good exercise to actually work out how eclipses and phases of the sun and the moon and so forth can be really easily explained by round earth and round moon and models. And you can just take a basketball and a golf ball and a light source and actually do these things yourself. So the intuition is there, but you have to transfer it.

Lex Fridman

(01:10:47) That is a big leap intellectually for us to go from flat to round earth because our life is mostly lived in flat land. To load that information and we're all like, take it for granted. We take so many things for granted because science has established a lot of evidence for this kind of thing, but we're on a round rock flying through space. Yeah, that's a big leap. And you have to take a chain of those leaps. The more and more and more we progress,

Terence Tao

(01:11:15) Right, yeah. So modern science is maybe, again, a victim of own success is that in order to be more accurate, it has to move further and further away from your initial intuition. And so for someone who hasn't gone through the whole process of science education, it looks more suspicious because of that. So we need more grounding. There are scientists who do excellent outreach, but there's lots of science things that you can do at home. Lots of YouTube videos I did at YouTube video recently, Grant Sanderson, we talked about this earlier, that how the ancient Greeks were able to measure things like the distance of the moon, distance the earth, and using techniques that you could also replicate yourself. It doesn't all have to be fancy space telescopes and very intimidating mathematics.

Lex Fridman

(01:12:01) Yeah, I highly recommend that. I believe you give a lecture and you also did an incredible video with Grant. It's a beautiful experience to try to put yourself in the mind of a person from that time shrouded in mystery. You're on this planet, you don't know the shape of it, the size of it. You see some stars, you see some things and you try to localize yourself in this world and try to make some kind of general statements about distanced places.

Terence Tao

(01:12:29) Change of perspective is really important. You say travel broadens the mind, this is intellectual travel. Put yourself in the mind of the ancient Greeks or person some other time period, make hypotheses, spherical [inaudible 01:12:41], whatever, speculate. And this is what mathematicians do and some other, what artists do actually.

Lex Fridman

(01:12:48) It's just incredible that given the extreme constraints, you could still say very powerful things. That's why it's inspiring. Looking back in history, how much can be figured out when you don't have much to figure out stuff with.

Terence Tao

(01:13:01) If you propose axioms, then the mathematics does. You follow those axioms to their conclusions and sometimes you can get quite a long way from initial hypotheses.

General relativity

Lex Fridman

(01:13:10) If we can stay in the land of the weird. You mentioned general relativity. You've contributed to the mathematical understanding, Einstein's field equations. Can you explain this work and from a mathematical standpoint, what aspects of general relativity are intriguing to you? Challenging to you?

Terence Tao

(01:13:31) I have worked on some equations. There's something called the wave maps equation or the Sigma field model, which is not quite the equation of space-time gravity itself, but of certain fields that might exist on top of space-time. So Einstein's equations of relativity just describe space and time itself. But then there's other fields that live on top of that. There's the electromagnetic field, there's things called Yang-Mills fields, and there's this whole hierarchy of different equations of which Einstein's considered one of the most nonlinear and difficult, but relatively low on the hierarchy was this thing called the wave maps equation. So it's a wave which at any given point is fixed to be on a sphere. So I can think of a bunch of arrows in space and time. Yeah, so it's pointing in different directions, but they propagate like waves. If you wiggle an arrow, it would propagate and make all the arrows move kind of like sheaves of wheat in a wheat field.

(01:14:27) And I was interested in the global regularity problem. Again for this question, is it possible for the energy here to collect at a point? So the equation I considered was actually what's called a critical equation where it's actually the behavior at all scales is roughly the same. And I was able barely to show that you couldn't actually force a scenario where all the energy concentrated at one point, that the energy had to disperse a little bit at the moment, just a little bit. It would stay regular. Yeah, this was back in 2000. That was part of why I got interested in [inaudible 01:14:58] afterwards actually. So I developed some techniques to solve that problem. So part of it, this problem is really nonlinear because of the curvature of the sphere. There was a certain nonlinear effect, which was a non-perturbative effect. It was when you sort looked at it normally it looked larger than the linear effects of the wave equation. And so it was hard to keep things under control even when your energy was small.

(01:15:23) But I developed what's called a gauge transformation. So the equation is kind of like an evolution of sheaves of wheat, and they're all bending back and forth, so there's a lot of motion. But if you imagine stabilizing the flow by attaching little cameras at different

points in space, which are trying to move in a way that captures most of the motion, and under this stabilized flow, the flow becomes a lot more linear. I discovered a way to transform the equation to reduce the amount of nonlinear effects, and then I was able to solve the equation. I found the transformation while visiting my aunt in Australia, and I was trying to understand the dynamics of all these fields, and I couldn't do a pen and paper, and I had not enough facility of computers to do any computer simulations.

(01:16:08) So I ended up closing my eyes being on the floor and just imagining myself to actually be this vector field and rolling around to try to see how to change coordinates in such a way that somehow things in all directions would behave in a reasonably linear fashion. And yeah, my aunt walked in on me while I was doing that and she was asking, "Why am I doing this?"

Lex Fridman

(01:16:28) It's complicated as the answer.

Terence Tao

(01:16:30) "Yeah, yeah. And okay, fine. You are a young man. I don't ask questions."

Solving difficult problems

Lex Fridman

(01:16:34) I have to ask about how do you approach solving difficult problems if it's possible to go inside your mind when you're thinking, are you visualizing in your mind the mathematical objects, symbols, maybe what are you visualizing in your mind? Usually when you're thinking?

Terence Tao

(01:16:57) A lot of pen and paper. One thing you pick up as a mathematician is I call it cheating strategically. So the beauty of mathematics is that you get to change the problem and change the rules as you wish. You don't get to do this by any other field. If you're an engineer and someone says, "Build a bridge over this river," you can't say, "I want to build this bridge over here instead," or, "I want to put it out of paper instead of steel," but a mathematician, you can do whatever you want on. It's like trying to solve a computer game where there's unlimited cheat codes available. And so you can set this, there's a dimension that's large. I've set it to one. I'll solve the one-dimensional problem first. So there's a main term and an error term. I'm going to make a spherical call assumption [inaudible 01:17:45] term is zero.

(01:17:45) And so the way you should solve these problems is not in this Iron Man mode where you make things maximally difficult, but actually the way you should approach any reasonable math problem is that if there are 10 things that are making your life difficult, find a version of the problem that turns off nine of the difficulties, but only keeps one of them and solve that. And then so you solve nine cheats. Okay, you solve 10 cheats, then the game is trivial, but you solve nine cheats. You solve one problem that teaches you how to deal with that particular difficulty. And then you turn that one-off and you turn someone else something else on, and then you solve that one. And after you know how to solve the 10 problems, 10 difficulties separately, then you have to start merging them a few at a time.

(01:18:26) As a kid, I watched a lot of these Hong Kong action movies from our culture, and one thing is that every time it's a fight scene, so maybe the hero gets swarmed by a hundred bad-guy goons or whatever, but it'll always be choreographed so that he'd always be only fighting one person at a time and it would defeat that person and move on. And because of that, he could defeat all of them. But whereas if they had fought a bit more intelligently and just swarmed the guy at once, it would make for much worse cinema, but they would win.

Lex Fridman

(01:19:02) Are you usually pen and paper? Are you working with computer and LaTeX?

Terence Tao

(01:19:08) Mostly pen and paper actually. So in my office I have four giant blackboards and sometimes I just have to write everything I know about the problem on the four blackboards and then sit my couch and just see the whole thing.

Lex Fridman

(01:19:20) Is it all symbols like notation or is there some drawings?

Terence Tao

(01:19:23) Oh, there's a lot of drawing and a lot of bespoke doodles that only makes sense to me. And the beauty of a blackboard is you erase and it's a very organic thing. I'm beginning to use more and more computers, partly because AI makes it much easier to do simple coding things that if I wanted to plot a function before, which is moderately complicated, has some iteration or something, I'd had to remember how to set up a Python program and how does a full loop work and debug it and it would take two hours and so forth. And now I can do it in 10, 15 minutes as much. I'm using more and more computers to do simple explorations.

AI-assisted theorem proving

Lex Fridman

(01:20:01) Let's talk about AI a little bit if we could. So maybe a good entry point is just talking about computer-assisted proofs in general. Can you describe the Lean formal proof programming language and how it can help as a proof assistant and maybe how you started using it and how it has helped you?

Terence Tao

(01:20:25) So Lean is a computer language, much like standard languages like Python and C and so forth, except that in most languages the focus is on using executable code. Lines of code do things, they flip bits or they make a robot move or they deliver your text on the internet or something. So lean is a language that can also do that. It can also be run as a standard traditional language, but it can also produce certificates. So a software language like Python might do a computation and give you that the answer is seven. Okay, does the sum of three plus four equal to seven?

(01:20:59) But Lean can produce not just the answer, but a proof that how it got the answer of seven as three plus four and all the steps involved. So it creates these more complicated objects, not just statements, but statements with proofs attached to them. And every line of code is just a way of piecing together previous statements to create new ones. So the idea is not new. These things are called proof assistance, and so they provide languages for which you can create quite complicated mathematical proofs. They produce these certificates that give a 100% guarantee that your arguments are correct if you trust the compiler of Lean, but they made the compiler really small and there are several different compilers available for the Lean.

Lex Fridman

(01:21:45) Can you give people some intuition about the difference between writing on pen and paper versus using Lean programming language? How hard is it to formalize statement?

Terence Tao

(01:21:56) So Lean, a lot of mathematicians were involved in the design of Lean. So it's designed so that individual lines of code resemble individual lines of mathematical argument. You might want to introduce a variable, you want to prove our contradiction. There are various standard things that you can do and it's written. So ideally should like a one-to-one correspondence. In practice, it isn't because Lean is explaining a proof to extremely pedantic colleague who will point out, "Okay, did you really mean this? What happens if this is zero? Okay, how do you justify this?" So Lean has a lot of automation in it to try to be less annoying. So

for example, every mathematical object has to come with a type. If I talk about X , is X a rule number or a natural number or a function or something? If you write things informally, it's often if you have context. You say, "Clearly X is equal to let X be the sum of Y and Z and Y and Z were already rule number, so X should also be a rule number." So Lean can do a lot of that, but every so often it says, wait a minute, can you tell me more about what this object is? What type of object it is? You have to think more at a philosophical level, not just computations that you're doing, but what each object actually is in some sense.

Lex Fridman

(01:23:17) Is he using something like LLMs to do the type inference or you match with the real number?

Terence Tao

(01:23:23) It's using much more traditional what's called good old-fashioned AI. You can represent all these things as trees, and there's always algorithm to match one tree to another tree.

Lex Fridman

(01:23:30) So it's actually doable to figure out if something is a real number or a natural number.

Terence Tao

(01:23:36) Every object comes with a history of where it came from, and you can kind of trace it.

Lex Fridman

(01:23:40) Oh, I see.

Terence Tao

(01:23:41) Yeah. So it's designed for reliability. So modern AIs are not used in, it's a disjoint technology. People are begin to use AIs on top of lean. So when a mathematician tries to program proven in lean, often there's a step. Okay, now I want to use the fundamental thing on calculus, say to do the next step. So the lean developers have built this massive project called Mathlib, a collection of tens of thousands of useful facts about methodical objects.

(01:24:09) And somewhere in there is the fundamental calculus, but you need to find it. So a lot of the bottleneck now is actually lemma search. There's a tool that you know is in there somewhere and you need to find it. And so there are various search engine engines specialized for Mathlib that you can do, but there's now these large language models that you can

say, "I need the fundamental calculus at this point." And it was like, okay, for example, when I code, I have GitHub Copilot installed as a plugin to my IDE, and it scans my text and it sees what I need. Says I might even type, now I need to use the fundamental calculus. And then it might suggest, "Okay, try this," and maybe 25% of the time it works exactly. And then another ten-fifty percent of the time it doesn't quite work, but it's close enough that I can say, oh yeah, if I just change it here and here, it'll work. And then half the time it gives me complete rubbish. But people are beginning to use AIs a little bit on top, mostly on the level of basically fancy auto-complete that you can type half of one line of a proof and it will find, it'll tell you.

Lex Fridman

(01:25:11) Yeah, but a fancy, especially fancy with the sort of capital letter F, remove some of the friction mathematician might feel when they move from pen and paper to formalizing.

Terence Tao

(01:25:23) Yes. Yeah. So right now I estimate that the time and effort taken to formalize it, proof is about 10 times the amount taken to write it out. So it's doable, but it's annoying.

Lex Fridman

(01:25:36) But doesn't it kill the whole vibe of being a mathematician? Having a pedantic coworker?

Terence Tao

(01:25:42) Right? Yeah, if that was the only aspect of it, but there's some cases was actually more pleasant to do things formally. So there's a theorem I formalized, and there was a certain constant 12 that came out in the final statement. And so this 12 had been carried all through the proof and everything had to be checked all these other numbers that had to be consistent with this final number 12. And so we wrote a paper through this theorem with this number 12. And then a few weeks later someone said, "Oh, we can actually improve this 12 to an 11 by reworking some of these steps." And when this happens with pen and paper, every time you change your parameter, you have to check line by line that every single line of your proof still works. And there can be subtle things that you didn't quite realize, some properties, not number 12, that you didn't even realize that you were taking advantage of. So a proof can break down at a subtle place.

(01:26:29) So we had formalized the proof with this constant 12, and then when this new paper came out, we said, "Oh," so that took three weeks to formalize and 20 people to formalize this original proof. I said, "Now let's update the proof to 11." And what you can do with Lean is in your headline theorem, you change your 12 to 11, you run the compiler and off the

thousands of lines of code, you have 90% of them still work and there's a couple that are lined in red. Now, I can't justify these steps, but immediately isolates which steps you need to change, but you can skip over everything, which works just fine.

(01:27:04) And if you program things correctly with good programming practices, most of your lines will not be red. And there'll just be a few places where you, if you don't hard code your constants, but you use smart tactics and so forth, you can localize the things you need to change to a very small period of time. So within a day or two, we had updated our proof because it's this very quick process, you make a change. There are 10 things now that don't work. For each one, you make a change and now there's five more things that don't work, but the process converges much more smoothly than with pen and paper.

Lex Fridman

(01:27:40) So that's for writing? Are you able to read it? If somebody else has a proof, are you able to, what's the versus paper and?

Terence Tao

(01:27:48) Yeah, so the proofs are longer, but each individual piece is easier to read. So if you take a math paper and you jump to page 27 and you look at paragraph six and you have a line of text of math, I often can't read it immediately because it assumes various definitions, which I have to go back and maybe on 10 pages earlier this was defined and the proof is scattered all over the place, and you basically are forced to read fairly sequentially. It's not like say a novel where in a theory you could open up a novel halfway through and start reading. There's a lot of context. But when [inaudible 01:28:23] Lean, if you put your cursor on a line code, every single object there, you can hover over it and it would say what it is, where it came from, where stuff is justified. You can trace things back much easier than flipping through a math paper.

(01:28:34) So one thing that Lean really enables is actually collaborating on proofs at a really atomic scale that you really couldn't do in the past. So traditionally with pen and paper, when you want to collaborate with another mathematician, either you do it at a blackboard where you can really interact, but if you're doing it sort of by email or something, basically, yeah, you have to segment it. Say, "I'm going to finish section three, you do section four," but you can't really work on the same thing, collaborate at the same time.

(01:29:03) But with Lean, you can be trying to formalize some portion of proof and say, "I got stuck at line 67 here. I need to prove this thing, but it doesn't quite work. Here's the three lines of code I'm having trouble with." But because all the context is there, someone else can

say, “Oh, okay, I recognize what you need to do. You need to apply this trick or this tool,” and you can do extremely atomic-level conversations. So because of Lean, I can collaborate with dozens of people across the world, most of who I have never met in person, and I may not know actually even whether they’re how reliable they are in the proof-taking field, but Lean gives me a certificate of trust so I can do trustless mathematics.

Lex Fridman

(01:29:43) So there’s so many interesting questions there. So one, you’re known for being a great collaborator. So what is the right way to approach solving a difficult problem in mathematics when you’re collaborating? Are you doing a divide and conquer type of thing? Or are you focused in on a particular part and you’re brainstorming?

Terence Tao

(01:30:05) There’s always a brainstorming process first. Yeah, so math research projects, by their nature, when you start, you don’t really know how to do the problem. It’s not like an engineering project where somehow the theory has been established for decades and it’s implementation is the main difficulty. You have to figure out even what is the right path. So this is what I said about cheating first. It’s like to go back to the bridge building analogy. So first assume you have infinite budget and unlimited amounts of workforce and so forth. Now can you build this bridge? Okay, now have infinite budget, but only finite workforce, right? Now can you do that? And so forth. So, of course no engineer can actually do this. Like I say, they have fixed requirements. Yes, there’s this sort of jam sessions always at the beginning where you try all kinds of crazy things and you make all these assumptions that are unrealistic, but you plan to fix later.

(01:30:57) And you try to see if there’s even some skeleton of an approach that might work. And then hopefully that breaks up the problem into smaller sub problems, which you don’t know how to do. But then you focus on the sub ones. And sometimes different collaborators are better at working on certain things. So one of my themes I’m known for is a theme of Ben Green, which now called the Green-Tao theorem. It’s a statement that the primes contain mathematic progressions of an event. So it was a modification of his [inaudible 01:31:26] already. And the way we collaborated was that Ben had already proven a similar result for progressions of length three. He showed that such like the primes contain loss and loss of progressions of length three, even subsets of the primes, certain subsets do, but his techniques only worked for the three progressions. They didn’t work for longer.

(01:31:46) But I had these techniques coming from a [inaudible 01:31:48] theory, which is something that I had been playing with and I knew better than Ben at the time. And so if I could justify certain randomness properties of some set relating for primes, there's a certain technical condition, which if I could have it, if Ben could supply me to this fact, I could conclude the theorem. But what I asked was a really difficult question in number theory, which he said, "There's no way we can prove this." So he said, "Can you prove your part of the theorem using a weak hypothesis that I have a chance to prove it?" And he proposed something which he could prove, but it was too weak for me. I can't use this. So there was this conversation going back and forth, a hacker-

Lex Fridman

(01:32:29) Different cheats to-

Terence Tao

(01:32:31) Yeah, yeah, I want to cheat more. He wants to cheat less, but eventually we found a property which A, he could prove, and B, I could use, and then we could prove our theorem. So there are all kinds of dynamics. Every collaboration has some story. No two are the same.

Lean programming language

Lex Fridman

(01:32:51) And then on the flip side of that, like you mentioned with Lean programming, now that's almost like a different story because you can create, I think you've mentioned a blueprint for a problem, and then you can really do a divide and conquer with Lean where you're working on separate parts and they're using the computer system proof checker essentially to make sure that everything is correct along the way.

Terence Tao

(01:33:17) So it makes everything compatible and trustable. Yeah, so currently only a few mathematical projects can be cut up in this way. At the current state of the art, most of the Lean activity is on formalizing proofs that have already been proven by humans. A math paper basically is a blueprint in a sense. It is taking a difficult statement like big theorem and breaking up into me a hundred little lemmas, but often not all written with enough detail that each one can be sort of directly formalized.

(01:33:46) A blueprint is a really pedantically written version of a paper where every step is explained as much detail as possible and just trying to make each step kind of self-contained or depending on only a very specific number of previous statements that have been proven so that each node of this blueprint graph that gets generated can be tackled independently of the others. And you don't even need to know how the whole thing works. So it's like a modern supply chain. If you want to create an iPhone or some other complicated object, no one person can build up a single object, but you can have specialists who just, if they're given some widgets from some other company, they can combine them together to form a slightly bigger widget.

Lex Fridman

(01:34:27) I think that's a really exciting possibility because you can have, if you can find problems that could be broken down in this way, then you could have thousands of contributors, right? To be completely distributed.

Terence Tao

(01:34:39) So I told you before about this split between theoretical and experimental mathematics. And right now most mathematics is theoretical, only a tiny bit it's experimental. I think the platform that Lean and other software tools, so GitHub and things like that will allow experimental mathematics to scale up to a much greater degree than we can do now. So right now, if you want to do any mathematical exploration of some mathematical pattern or something, you need some code to write out the pattern. And I mean, sometimes there are some computer algebra packages that could help, but often it's just one mathematician coding lots and lots of Python or whatever. And because coding is such an error-prone activity, it's not practical to allow other people to collaborate with you on writing modules for your code because if one of the modules has a bug in it, the whole thing is unreliable. So you get these bespoke spaghetti code written by non-professional programmers, mathematicians, and they're clunky and slow. And so because of that, it's hard to really mass-produce experimental results.

(01:35:45) But I think with Lean, I'm already starting some projects where we are not just experimenting with data, but experimenting with proofs. So I have this project called the Equational Theories Project. Basically we generated about 22 million little problems in abstract algebra. Maybe I should back up and tell you what the project is. Okay, so abstract algebra studies operations like multiplication, addition and the abstract properties. So multiplication for example, is commutative. X times Y is always Y times X , at least for numbers. And

it's also associative. X times Y times Z is the same as X times Y times Z . So these operations obey some laws that don't obey others. For example, X times X is not always equal to X . So that law is not always true. So given any operation, it obeys some laws and not others. And so we generated about 4,000 of these possible laws of algebra that certain operations can satisfy.

(01:36:38) And our question is which laws imply which other ones, so for example, does commutativity imply associativity? And the answer is no, because it turns out you can describe an operation which obeys the commutative law but doesn't obey the associative law. So by producing an example, you can show that commutativity does not imply associativity. But some other laws do imply other laws by substitution and so forth, and you can write down some algebraic proof. So we look at all the pairs between these 4,000 laws and this up to 22 million of these pairs. And for each pair we ask, does this law imply this law? If so, give a proof. If not, give a counterexample. So 22 million problems, each one of which you could give to an undergraduate algebra student, and they had a decent chance of solving the problem, although there are a few, at least 22 million, like a hundred or so that are really quite hard, but a lot are easy. And the project was just to work out to determine the entire graph which ones imply which other ones.

Lex Fridman

(01:37:31) That's an incredible project, by the way. Such a good idea, such a good test that the very thing we've been talking about at a scale that's remarkable.

Terence Tao

(01:37:38) So it would not have been feasible. The state of the art in the literature was like 15 equations and sort of how they applied, that's at the limit of what a human with pen and paper can do. So you need to scale that up. So you need to crowdsource, but you also need to trust all the, no one person can check 22 million of these proofs. You need it to be computerized. And so it only became possible with Lean. We were hoping to use a lot of AI as well. So the project is almost complete. So at these 22 million, all but two had been settled.

Lex Fridman

(01:38:11) Wow.

Terence Tao

(01:38:12) Well, actually, and of those two, we have a pen and paper proof of the two, and we're formalizing it. In fact, this morning I was working on finishing it, so we're almost done on this.

Lex Fridman

(01:38:22) It's incredible.

Terence Tao

(01:38:22) Yeah. Fantastic.

Lex Fridman

(01:38:25) How many people were you able to get?

Terence Tao

(01:38:26) About 50, which in mathematics is considered a huge number.

Lex Fridman

(01:38:30) It's a huge number. That's crazy.

Terence Tao

(01:38:32) Yeah. So we're going to have a paper of 50 authors and a big appendix of who contributed what.

Lex Fridman

(01:38:38) Here's an question, not to maybe speak even more generally about it. When you have this pool of people, is there a way to organize the contributions by level of expertise of the people, of the contributors? Now, okay, I'm asking a lot of pothead questions here, but I'm imagining a bunch of humans, and maybe in the future, some AIs, can there be an ELO rating type of situation?

Lex Fridman

(01:39:00) Can there be an Elo-rating type of situation where a gamification of this.

Terence Tao

(01:39:07) The beauty of these lean projects is that automatically you get all this data, so everything's being uploaded for this GitHub. GitHub tracks who contributed what. So you could generate statistics at any later point in time. You can say, "Oh, this person contributed this many lines of code" or whatever. These are very crude metrics. I would definitely not want this to become part of your tenure review or something. But I mean, I think already in enterprise computing, people do use some of these metrics as part of the assessment of performance of an employee. Again, this is the direction which is a bit scary for academics to go down. We don't like metrics so much.

Lex Fridman

(01:39:49) And yet academics use metrics. They just use old ones, number of papers.

Terence Tao

(01:39:56) Yeah, it's true that...

Lex Fridman

(01:39:59) It feels like this is a metric, while flawed, is going in more in the right direction. Right.

Terence Tao

(01:40:05) Yeah.

Lex Fridman

(01:40:06) It's interesting. At least it's a very interesting metric.

Terence Tao

(01:40:08) Yeah, I think it's interesting to study. I think you can do studies of whether these are better predictors. There's this problem called Goodhart's Law. If a statistic is actually used to incentivize performance, it becomes gamed, and then it's no longer a useful measure.

Lex Fridman

(01:40:22) Oh, humans. Always gaming the...

Terence Tao

(01:40:25) It's rational. So what we've done for this project is self-report. So there are actually standard categories from the sciences of what types of contributions people give. So there's the concept and validation and resources and coding and so forth. So there's a standard list of twelve or so categories, and we just ask each contributor to... There's a big matrix of all the authors and all the categories just to tick the boxes where they think that they contributed, and just give a rough idea. Also, you did some coding and you provided some compute, but you didn't do any of the pen- and-paper verification or whatever.

(01:41:02) And I think that that works out. Traditionally, mathematicians just order alphabetically by surname. So we don't have this tradition as in the sciences of "lead author" and "second author" and so forth, which we're proud of. We make all the authors equal status, but it doesn't quite scale to this size. So a decade ago I was involved in these things called polymath projects. It was the crowdsourcing mathematics but without the lean component. So it was limited by, you needed a human moderator to actually check that all the contributions coming in were actually valid. And this was a huge bottleneck, actually, but still we had projects that were 10 authors or so. But we had decided, at the time, not to try to decide who

did what, but to have a single pseudonym. So we created this fictional character called DHJ Polymath in the spirit of [inaudible 01:41:51]. This is the pseudonym for a famous group of mathematicians in the 20th century.

(01:41:56) And so the paper was authored on the pseudonym, so none of us got the author credit. This actually turned out to be not so great for a couple of reasons. So one is that if you actually wanted to be considered for tenure or whatever, you could not use this paper in your... As you submitted as one your publications, because it didn't have the formal author credit. But the other thing that we've recognized much later is that when people referred to these projects, they naturally referred to the most famous person who was involved in the project. So "This was Tim Gower's playoff project." "This was Terence Tao's playoff project," and not mention the other 19 or whatever people that were involved.

Lex Fridman

(01:42:36) Oh, yeah.

Terence Tao

(01:42:37) So we're trying something different this time around where we have, everyone's an author, but we will have an appendix with this matrix, and we'll see how that works.

DeepMind's AlphaProof

Lex Fridman

(01:42:45) So both projects are incredible, just the fact that you're involved in such huge collaborations. But I think I saw a talk from Kevin Buzzard about the Lean Programming Language just a few years ago, and you're saying that this might be the future of mathematics. And so it's also exciting that you're embracing one of the greatest mathematicians in the world embracing this, what seems like the paving of the future of mathematics.

(01:43:12) So I have to ask you here about the integration of AI into this whole process. So DeepMind's alpha proof was trained using reinforcement learning on both failed and successful formal lean proofs of IMO problems. So this is sort of high-level high school?

Terence Tao

(01:43:32) Oh, very high-level, yes.

Lex Fridman

(01:43:33) Very high-level, high-school level mathematics problems. What do you think about the system and maybe what is the gap between this system that is able to prove the high-school level problems versus gradual-level problems?

Terence Tao

(01:43:47) Yeah, the difficulty increases exponentially with the number of steps involved in the proof. It's a commentarial explosion. So the thing of large language models is that they make mistakes and so if a proof has got 20 steps and your [inaudible 01:44:01] has a 10% failure rate at each step of going the wrong direction, it's extremely unlikely to actually reach the end.

Lex Fridman

(01:44:09) Actually, just to take a small tangent here, how hard is the problem of mapping from natural language to the formal program?

Terence Tao

(01:44:19) Oh yeah. It's extremely hard, actually. Natural language, it's very fault-tolerant. You can make a few minor grammatical errors and speak in the second language, can get some idea of what you're saying. But formal language, if you get one little thing wrong, then the whole thing is nonsense. Even formal to formal is very hard. There are different incompatible prefaces and languages. There's Lean, but also Coq and Isabelle and so forth. And even converting from a formal action to formal language is an unsolved problem.

Lex Fridman

(01:44:52) That is fascinating. Okay. But once you have an informal language, they're using their RL trained model, something akin to AlphaZero that they used to go to then try to come up with tools, also have a model. I believe it's a separate model for geometric problems.

Terence Tao

(01:44:52) Yes.

Lex Fridman

(01:45:12) So what impresses you about the system, and what do you think is the gap?

Terence Tao

(01:45:18) Yeah, we talked earlier about things that are amazing over time become kind of normalized. So now somehow, it's of course geometry is a silver bullet problem.

Lex Fridman

(01:45:27) Right. That's true, that's true. I mean, it's still beautiful to...

Terence Tao

(01:45:31) Yeah, these are great work that shows what's possible. The approach doesn't scale currently. Three days of Google's server time can solve one high school math format there. This is not a scalable prospect, especially with the exponential increase as the complexity increases.

Lex Fridman

(01:45:49) We should mention that they got a silver medal performance. The equivalent of the silver medal performance.

Terence Tao

(01:45:55) So first of all, they took way more time than was allotted, and they had this assistance where the humans started helped by formalizing, but also they're giving themselves full marks for the solution, which I guess is formally verified. So I guess that's fair. There are efforts, there will be a proposal at some point to actually have an AI math Olympiad where at the same time as the human contestants get the actual Olympiad problems, AI's will also be given the same problems, the same time period and the outputs will have to be graded by the same judges, which means that it'll have to be written in natural language rather than formal language.

Lex Fridman

(01:46:37) Oh, I hope that happens. I hope that this IMO happens. I hope next one.

Terence Tao

(01:46:41) It won't happen this IMO. The performance is not good enough in the time period. But there are smaller competitions, there are competitions where the answer is a number rather than a long form proof. And AI is actually a lot better at problems where there's a specific numerical answer, because it's easy to do reinforcement learning on it." Yeah, you've got the right answer, you've got the wrong answer." It's a very clear signal, but a long form proof either has to be formal, and then the Lean can give it thumbs up or down, or it's informal, but then you need a human to create it to tell. And if you're trying to do billions of reinforcement learning runs, you can't hire enough humans to grade those. It's already hard enough for the last language to do reinforcement learning on just the regular text that people get. But now we actually hire people, not just give thumbs up, thumbs down, but actually check the output mathematically, yeah, that's too expensive.

Human mathematicians vs AI

Lex Fridman

(01:47:45) So if we just explore this possible future, what is the thing that humans do that's most special in mathematics, that you could see AI not cracking for a while? So inventing new theories? Coming up with new conjectures versus proving the conjectures? Building new abstractions? New representations? Maybe an AI turnstile with seeing new connections between disparate fields?

Terence Tao

(01:48:17) That's a good question. I think the nature of what mathematicians do over time has changed a lot. So a thousand years ago, mathematicians had to compute the date of Easter, and they really complicated calculations, but it is all automated, the order of centuries, we don't need that anymore. They used to navigate to do spherical navigation, circle trigonometry to navigate how to get from the Old World to the New or something, like very complicated calculation. Again, have been automated. Even a lot of undergraduate mathematics even before AI, like Wolfram Alpha for example. It's not a language model, but it can solve a lot of undergraduate-level math tasks. So on the computational side, verifying routine things, like having a problem and saying, "Here's a problem in partial differential equations, could you solve it using any of the 20 standard techniques?" And say, "Yes, I've tried all 20 and here are the 100 different permutations and my results."

(01:49:12) And that type of thing, I think it will work very well, type of scaling to once you solve one problem to make the AI attack a hundred adjacent problems. The things that humans do still... So where the AI really struggles right now is knowing when it's made a wrong turn. It can say, "Oh, I'm going to solve this problem. I'm going to split up this one into these two cases. I'm going to try this technique." And sometimes, if you're lucky and it's a simple problem, it's the right technique and you solve the problem and sometimes it will have a problem, it would propose an approach which is just complete nonsense, but it looks like a proof.

(01:49:53) So this is one annoying thing about LLM-generated mathematics. So yeah, we've had human generated mathematics as a very low quality, like submissions who don't have the formal training and so forth, but if a human proof is bad, you can tell it's bad pretty quickly. It makes really basic mistakes. But the AI-generated proofs, they can look superficially flawless. And it's partly because what the reinforcement learning has actually trained them to do, to make things to produce tech that looks like what is correct, which for many ap-

plications is good enough. So the air is often really subtle and then when you spot them, they're really stupid. Like no human would've actually made that mistake.

Lex Fridman

(01:50:36) Yeah, it's actually really frustrating in the programming context, because I program a lot, and yeah, when a human makes low quality code, there's something called "code smell", right? You can tell immediately there's signs, but with the AI generated code...

Terence Tao

(01:50:53) [inaudible 01:50:53].

Lex Fridman

(01:50:52) And you're right, eventually you find an obvious dumb thing that just looks like good code.

Terence Tao

(01:50:59) Yeah.

Lex Fridman

(01:51:00) It's very tricky, too, and frustrating, for some reason, to have to work.

Terence Tao

(01:51:05) So the sense of smell, this is one thing that humans have, and there's a metaphorical mathematical smell that it's not clear how to get the AI to duplicate that eventually. So the way AlphaZero and so forth make progress on Go and chess and so forth, is in some sense they have developed a sense of smell for Go and chess positions, that this position is good for white, it's good for black. They can't initiate why, but just having that sense of smell lets them strategize. So if AIs gain that ability to a sense of viability of certain proof strategies, because I'm going to try to break up this problem into two small subtasks and they can say, "Oh, this looks good. The two tasks look like they're simpler tasks than your main task and they've still got a good chance of being true. So this is good to try." Or "No, you've made the problem worse, because each of the two subproblems is actually harder than your original problem," which is actually what normally happens if you try a random thing to try normally it's very easy to transform a problem into an even harder problem. Very rarely do you transform into a simpler problem. So if they can pick up a sense of smell, then they could maybe start competing with a human level of mathematicians.

Lex Fridman

(01:52:24) So this is a hard question, but not competing but collaborating. Okay, hypothetical. If I gave you an Oracle that was able to do some aspect of what you do and you could just collaborate with it, what would you like that Oracle to be able to do? Would you like it to maybe be a verifier, like check, do the codes? Like “Yes, Professor Tao, correct, this is a promising fruitful direction”? Or would you like it to generate possible proofs and then you see which one is the right one? Or would you like it to maybe generate different representation, different totally different ways of seeing this problem?

Terence Tao

(01:53:10) Yeah, I think all of the above. A lot of it is we don’t know how to use these tools, because it’s a paradigm that... We have not had in the past. Systems that are competent enough to understand complex instructions that can work at massive scale, but are also unreliable. It’s an interesting... A bit unreliable in subtle ways, whereas was providing sufficiently good output. It’s an interesting combination. I mean, you have graduate students that you work with who kind of like this, but not at scale. And we had previous software tools that can work at scale, but very narrow, so we have to figure out how to use, so Tim Gowers is actually, you mentioned he actually foresaw in 2000. He was envisioning what mathematics would look like in actually two and a half decades.

Lex Fridman

(01:54:08) That’s funny.

Terence Tao

(01:54:09) Yeah, he wrote his article, a hypothetical conversation between a mathematical assistant of the future and himself. He’s trying to solve a problem and they would have a conversation. Sometimes the human would propose an idea and the AI would evaluate it, and sometimes the AI would propose an idea and sometimes a competition was required and AI would just go and say, “Okay, I’ve checked the 100 cases needed here,” or “The first you set this through for all N , I’ve checked N up to 100 and it looks good so far,” or “Hang on, there’s a problem at N equals 46.” So just a freeform conversation where you don’t know in advance where things are going to go, but just based on, “I think ideas are going to propose on both sides.” Calculations could propose on both sides.

(01:54:53) I’ve had conversations with AI where I say, “Okay, we’re going to collaborate to solve this math problem,” and it’s a problem that I already know the solution to, so I try to prompt it. “Okay, so here’s the problem.” I suggest using this tool, and it’ll find this.” Okay, it might start using this, and then it’ll go back to the tool that it wanted to do before. You have

to keep railroading it onto the path you want, and I could eventually force it to give the proof I wanted, but it was like herding cats. And the amount of personal effort I had to take to not just prompt it but also check its output because a lot of what it looked like is going to work, I know there's a problem on line 17, and basically arguing with it. It was more exhausting than doing it unassisted, but that's the current state of the art.

Lex Fridman

(01:55:44) I wonder if there's a phase shift that happens to where it's no longer feels like herding cats. And maybe you'll surprise us how quickly that comes.

Terence Tao

(01:55:54) I believe so. In formalization, I mentioned before that it takes 10 times longer to formalize a proof than to write it by hand. With these modern AI tools and also just better tooling, the Lean developers are doing a great job adding more and more features and making it user-friendly going from nine to eight to seven... Okay, no big deal, but one day it'll drop a little one. And that's a phase shift, because suddenly it makes sense when you write a paper to write it in Lean first, or through a conversation with AI, which is generally on the fly with you, and it becomes natural for journals to accept. Maybe they'll offer expedite refereeing. If a paper has already been formalized in Lean, they'll just ask the referee to comment on the significance of the results and how it connects to literature and not worry so much about the correctness, because that's been certified. Papers are getting longer and longer in mathematics, and it's harder and harder to get good refereeing for the really long ones unless they're really important. It is actually an issue, and the formalization is coming in at just the right time for this to be.

Lex Fridman

(01:57:04) And the easier and easier to guess because of the tooling and all the other factors, then you're going to see much more like math lib will grow potentially exponentially, as it's a virtuous cycle.

Terence Tao

(01:57:16) I mean, one phase shift of this type that happened in the past was the adoption of LaTeX. So LaTeX is this typesetting language that all mathematicians use now. So in the past people used all kinds of word processors and typewriters and whatever, but at some point LaTeX became easier to use than all other competitors, and people would switch within a few years. It was just a dramatic base shift.

AI winning the Fields Medal

Lex Fridman

(01:57:37) It's a wild, out-there question, but what year, how far away are we from a AI system being a collaborator on a proof that wins the Fields Medal? So that level.

Terence Tao

(01:57:55) Okay, well it depends on the level of collaboration, right?

Lex Fridman

(01:57:58) No, it deserves to be get the Fields Medal. So half-and-half

Terence Tao

(01:58:03) Already. I can imagine if it was a medal-winning paper, having some AI assistance in writing it just like the order complete alone is already, I use it speeds up my own writing. You can have a theorem and you have a proof, and the proof has three cases, and I write down the proof of first case and the autocomplete just suggests that. Now here's how the proof of second case could work. And it was exactly correct. That was great. Saved me like five, ten minutes of typing.

Lex Fridman

(01:58:30) But in that case, the AI system doesn't get the Fields Medal. Are we talking 20 years, 50 years, a hundred years? What do you think?

Terence Tao

(01:58:42) Okay, so I gave a prediction in print by 2026, which is now next year, there will be math collaborations with the AI, so not Fields-Medal winning, but actual research-level papers.

Lex Fridman

(01:58:54) Published ideas that are in part generated by AI.

Terence Tao

(01:58:58) Maybe not the ideas, but at least some of the computations, the verifications.

Lex Fridman

(01:59:03) Has that already happened?

Terence Tao

(01:59:04) That already happened. There are problems that were solved by a complicated process conversing with AI to propose things and the human goes and tries it and the contract doesn't work, but it might pose a different idea. It's hard to disentangle exactly. There are certainly math results which could only have been accomplished because there was a human authentication and an AI involved, but it's hard to disentangle credit. I mean, these tools, they do not replicate all the skills needed to do mathematics, but they can replicate some non-trivial percentage of them, 30, 40%, so they can fill in gaps. So coding is a good example. So it's annoying for me to code in Python. I'm not a native, I'm a professional programmer, but with AI, the friction cost of doing it is much reduced. So it fills in that gap for me. AI is getting quite good at literature review.

(02:00:15) I mean, it's still a problem with hallucinating references that don't exist, but this, I think, is a solvable problem. If you train in the right way and so forth and verify using the internet, you should, in a few years, get to the point where you have a lemma that you need and say, "Has anyone proven this lemma before?" And it will do basically a fancy web search and say, yeah, there are these six papers where something similar has happened. I mean, you can ask it right now and it'll give you six papers of which maybe one is legitimate and relevant, one exists but is not relevant, and four are hallucinated. It has a non-zero success rate right now, but there's so much garbage, so much the signal-to-noise ratio is so poor, that it's most helpful when you already somewhat know the relationship, and you just need to be prompted to be reminded of a paper that was already subconsciously in your memory.

Lex Fridman

(02:01:14) Versus helping you discover new you were not even aware of, but is the correct citation.

Terence Tao

(02:01:20) Yeah, that it can sometimes do, but when it does, it's buried in a list of options for which the other-

Lex Fridman

(02:01:26) That are bad. I mean, being able to automatically generate a related work section that is correct. That's actually a beautiful thing. That might be another phase shift because it assigns credit correctly. It breaks you out of the silos of thought.

Terence Tao

(02:01:42) Yeah, no, there's a big hump to overcome right now. I mean it's like self-driving cars. The safety margin has to be really high for it to be feasible. So yeah, there's a [inaudible

02:01:54]-Morrow problem with a lot of AI applications that they can develop tools that work 20%, 80% of the time, but it's still not good enough. And in fact, even worse than good, in some ways.

Lex Fridman

(02:02:08) I mean, another way of asking the Fields Medal question is what year do you think you'll wake up and be like real surprised? You read the headline, the news or something happened that AI did, real breakthrough. Something. Like Fields Medal, even a hypothesis. It could be really just this AlphaZero Go moment would go that kind of thing.

Terence Tao

(02:02:33) Yeah, this decade I can see it making a conjecture between two things that people would thought was unrelated.

Lex Fridman

(02:02:42) Oh, interesting. Generating a conjecture. That's a beautiful conjecture.

Terence Tao

(02:02:45) Yeah. And actually has a real chance of being correct and meaningful.

Lex Fridman

(02:02:50) Because that's actually kind of doable, I suppose, but the word of the data is...

Terence Tao

(02:02:56) Yeah.

Lex Fridman

(02:02:56) No, that would be truly amazing.

Terence Tao

(02:02:59) The current models struggle a lot. I mean, so a version of this... The physicists have a dream of getting the AI to discover new laws of physics. The dream is you just feed it all this data, and this is here is a new patent that we didn't see before, but it actually, even the current state of the art even struggles to discover old laws of physics from the data. Or if it does, there's a big concern of contamination, that it did it only because it's somewhere in this training, somehow new, Boyle's Law or whatever you're trying to reconstruct.

(02:03:35) Part of it is we don't have the right type of training data for this. So for laws of physics, we don't have a million different universes with a million different laws of nature. And a lot of what we are missing in math is actually the negative space of... So we have pub-

lished things of things that people have been able to prove, and conjectures that end up being verified or we counter examples produced, but we don't have data on things that were proposed and they're kind of a good thing to try, but then people quickly realized that it was the wrong conjecture and then they said, "Oh, but we should actually change our claim to modify it in this way to actually make it more plausible."

(02:04:16) There's a trial and error process, which is a real integral part of human mathematical discovery, which we don't record because it's embarrassing. We make mistakes, and we only like to publish our wins. And the AI has no access to this data to train on. I sometimes joke that basically AI has to go through grad school and actually go to grad courses, do the assignments, go to office hours, make mistakes, get advice on how to correct the mistakes and learn from that.

Grigori Perelman

Lex Fridman

(02:04:47) Let me ask you if I may, about Grigori Perelman, you mentioned that you try to be careful in your work and not let a problem completely consume you just you've really fall in love with the problem and it really cannot rest until you solve it. But you also hastened to add that sometimes this approach actually can be very successful, and an example you gave is Grigori Perelman who proved the Poincare Conjecture and did so by working alone for seven years, with basically little contact with the outside world. Can you explain this one Millennial Prize problem that's been solved, Poincare Conjecture, and maybe speak to the journey that Grigori Perelman has been on?

Terence Tao

(02:05:31) All right, so it's a question about curved spaces. Earth is a good example. So think of Earth as a 2-D surface. Injecting around you could maybe be a torus with a hole in it or can have many holes and there are many different topologies, a priori, that a surface could have, even if you assume that it's bounded and smooth and so forth. So we have figured out how to classify surfaces as a first approximation. Everything is determined by some called the genus, how many holes it has. So a sphere has genus zero, or a donut has genus one, and so forth. And one way you can tell the surfaces apart, probably the sphere has, which is called simply connected. If you take any closed loop on the sphere, like a big closed loop of rope, you can contract it to a point while staying on the surface. And the sphere has this property, but a torus doesn't. If you're on a torus and you take a rope that goes around say the outer

diameter of torus, there's no way... It can't get through the hole. There's no way to contract it to a point.

(02:06:25) So it turns out that the sphere is the only surface with this property of contractibility, up to continuous deformations of the sphere. So things that are what called topologically equivalent of the sphere. So Poincare asks the same question, higher dimensions, so it becomes hard to visualize because surface you can think of as embedded in three dimensions, but a curved three-space, we don't have good intuition of four-dimensional space to live it. And there are also three-dimensional spaces that can't even fit into four dimensions. You need five or six or higher. But anyway, mathematically you can still pose this question, that if you have a bounded three-dimensional space now, which also has this simply connected property that every loop can be contracted, can you turn it into a three-dimensional version of the sphere? And so this is the Poincare conjecture.

(02:07:09) Weirdly, in higher dimensions, four and five was actually easier. So it was solved first in higher dimensions, there's somehow more room to do the deformation. It is easier to move things around to your sphere. But three was really hard. So people tried many approaches. There's sort of commentary approaches where you chop up the surface into little triangles or tetrahedra and you just try to argue based on how the faces interact each other. There were algebraic approaches, there's various algebraic objects like things called the fundamental group that you can attach to these homologies and co-homology and all these very fancy tools. They also didn't quite work, but Richard Hamilton's proposed a partial differential equations approach.

(02:07:52) So the problem is that... So you have this object, which is secret is a sphere, but it's given to you in a weird way. So I think of a ball that's being crumpled up and twisted, and it's not obvious that it's the ball, but if you have some sort of surface, which is a deformed sphere, you could for example, think that as a surface of a balloon, you could try to inflate it, you blow it up and naturally as you fill it with air, the wrinkles will sort of smooth out and it will turn into a nice round sphere, unless of course it was a torus or something, which case it would get stuck at some point.

(02:08:32) If you inflate a torus, there be a point in the middle when the inner ring shrinks to zero, you get a singularity and you can't blow up any further and you can't flow further. So he created this flow, which is now called Ricci Flow, which is a way of taking an arbitrary surface or space and smoothing it out to make it rounder and rounder to make it look like a sphere. And he wanted to show that either this process would give you a sphere, or it would create a

singularity, actually very much like how PDEs either they have global regularity or finite and blow up. Basically, it's almost exactly the same thing. It's all connected. And he showed that for two dimensions, two-dimensional surfaces, if you start to simply connect it, no singularities ever formed, you never ran into trouble and you could flow and it will give you a sphere. So he got a new proof of the two-dimensional result.

Lex Fridman

(02:09:20) But by the way, that's a beautiful explanation of Ricci flow in its application in this context. How difficult is the mathematics here for the 2D case? Is it?

Terence Tao

(02:09:27) Yeah, these are quite sophisticated equations on par with the Einstein equations. Slightly simpler, but they were considered hard nonlinear equations to solve, and there's lots of special tricks in 2D that helped. But in 3D, the problem was that this equation was actually super critical. The same problem as [inaudible 02:09:48]. As you blow up, maybe the curvature could get concentrated in smaller and smaller regions, and it looked more and more nonlinear and things just looked worse and worse. And there could be all kinds of singularities that showed up. Some singularities, these things called neck pinches where the surface behaves like a barbell and it pinches at a point. Some singularities are simple enough that you can sort of see what to do next. You just make a snip and then you can turn one surface into two and e-bolt them separately. But there was the prospect that there's some really nasty knotted singularities showed up that you couldn't see how to resolve in any way, that you couldn't do any surgery to. So you need to classify all the singularities, like what are all the possible ways that things can go wrong? So what Perelman did was, first of all, he made the problem, he turned the problem from a super critical problem to a critical problem. I said before about how the invention of energy, the Hamiltonian, really clarified Newtonian mechanics. So he introduced something which is now called Perelman's reduced volume and Perelman's entropy. He introduced new quantities, kind of like energy, that looked the same at every single scale, and turned the problem into a critical one where the non-linearities actually suddenly looked a lot less scary than they did before. And then he had to solve... He still had to analyze the singularities of this critical problem. And that itself was a problem similar to this wave map thing I worked on actually. So on the level of difficulty of that.

(02:11:18) So he managed to classify all the singularities of this problem, and show how to apply surgery to each of these. And through that was able to resolve the Poincare Conjecture. So quite a lot of really ambitious steps, and nothing that a large language model

today, for example, could... At best, I could imagine a model proposing this idea as one of hundreds of different things to try, but the other 99 would be complete dead ends. But you'd only find out after months of work, he must have had some sense that this was the right track to pursue. It takes years to get from A to B.

Lex Fridman

(02:11:54) So you've done, like you said, actually, you see even strictly mathematically, but more broadly in terms of the process, you've done similar-

Lex Fridman

(02:12:01) In terms of the process, you've done similarly difficult things. What can you infer from the process he was going through because he was doing it alone? What are some low points in a process like that when you start to, you've mentioned hardship, AI doesn't know when it's failing. What happens to you, you're sitting in your office when you realize the thing you did for the last few days, maybe weeks is a failure?

Terence Tao

(02:12:27) Well, for me, I switch to a different problem. So I'm a fox, I'm not a hedgehog.

Lex Fridman

(02:12:33) But you're generally, that is a break that you can take, is to step away and look at a different problem?

Terence Tao

(02:12:37) Yeah, yeah. You can modify the problem too. I mean, you can ask some cheater if there's a specific thing that's blocking you that some bad case keeps showing up, that for which your tool doesn't work. You can just assume by fiat this bad case doesn't occur. So you do some magical thinking, but strategically okay for the point to see if the rest of the argument goes through. If there's multiple problems with your approach, then maybe you just give up. But if this is the only problem but everything else checks out, then it's still worth fighting. So yeah, you have to do some forward reconnaissance sometimes too.

Lex Fridman

(02:13:18) And that is sometimes productive to assume like, "Okay, we'll figure it out eventually"?

Terence Tao

(02:13:21) Oh, yeah, yeah. Sometimes actually it's even productive to make mistakes. So one of, there was a project which actually we won some prizes for with four other people. We

worked on this PDE problem. Again, actually this blow-off regularity type problem, and it was considered very hard. Jean Bourguignon was another Fields mathematicist who worked on a special case of this, but he could not solve the general case. And we worked on this problem for two months and we thought we solved it. We had this cute argument that if everything fit, and we were excited, we were planning celebration, to all get together and have champagne or something, and we started writing it up. And one of us, not me actually, but another co-author said, "Oh, in this lemma here, we have to estimate these 13 terms that show up in this expansion.

(02:14:13) And we estimate 12 of them, but in our notes, I can't find the estimation of the 13th. Can someone supply that?" And I said, "Sure, I'll look at this." Yeah, we didn't cover it, we completely omitted this term and this term turned out to be worse than the other 12 terms put together. In fact, we could not estimate this term. And we tried for a few more months and all different permutations, and there was always this one term that we could not control. And so this was very frustrating. But because we had already invested months and months of effort in this already, we stuck at this, which we tried increasingly desperate things and crazy things. And after two years we found an approach that was somewhat different, but quite a bit from our initial strategy, which actually didn't generate these problematic terms and actually solve the problem.

(02:14:58) So we solve the problem after two years, but if we hadn't had that initial false dawn of nearly solving a problem, we would've given up by month two or something and worked on an easier problem. If we had known it would take two years, not sure we would've started the project. Sometimes actually having the incorrect, it's like Columbus struggling in the new world, they had an incorrect measurement of the size of the Earth. He thought he was going to find a new trade route to India, or at least that was how he sold it in his prospectus. I mean, it could be that he actually secretly knew, but.

Lex Fridman

(02:15:31) Just from a psychological element, do you have emotional or self-doubt that just overwhelms you in moments like that? Because this stuff, it feels like math is so engrossing that it can break you when you invest so much of yourself in the problem and then it turns out wrong. You could start to... A similar way chess has broken some people.

Terence Tao

(02:15:59) Yeah, I think different mathematicians have different levels of emotional investment in what they do. I mean, I think for some people it's as a job, you have a problem and if

it doesn't work out, you go on the next one. So the fact that you can always move on to another problem, it reduces the emotional connection. I mean, there are cases, so there are certain problems that are what are called mathematical diseases where just latch onto that one problem and they spend years and years thinking about nothing but that one problem. And maybe their career suffers and so forth, but they say, "Okay, I've got this big win. Once I finish this problem, it will make up for all the years of lost opportunity." I mean, occasionally it works, but I really don't recommend it for people without the right fortitude.

(02:16:54) So I've never been super invested in any one problem. One thing that helps is that we don't need to call our problems in advance. Well, when we do grant proposals, we say we will study this set of problems, but even though we don't promise, definitely by five years I will supply a proof of all these things. You promise to make some progress or discover some interesting phenomena. And maybe you don't solve the problem, but you find some related problem that you can say something new about and that's a much more feasible task.

Twin Prime Conjecture

Lex Fridman

(02:17:27) But I'm sure for you, there's problems like this. You have made so much progress towards the hardest problems in the history of mathematics. So is there a problem that just haunts you? It sits there in the dark corners, twin prime conjecture, Riemann hypothesis, Goldbach's conjecture?

Terence Tao

(02:17:48) Twin prime, that sounds... Look, again, I mean, the problems like the Riemann hypothesis, those are so far out of reach.

Lex Fridman

(02:17:54) You think so?

Terence Tao

(02:17:55) Yeah, there's no even viable stretch. Even if I activate all the cheats that I know of in this book, there's just still no way to get from A to B. I think it needs a breakthrough in another area of mathematics to happen first and for someone to recognize that it would be a useful thing to transport into this problem.

Lex Fridman

(02:18:18) So we should maybe step back for a little bit and just talk about prime numbers.

Terence Tao

(02:18:22) Okay.

Lex Fridman

(02:18:23) So they're often referred to as the atoms of mathematics. Can you just speak to the structure that these atoms provide?

Terence Tao

(02:18:31) So the natural numbers have two basic operations, addition, and multiplication. So if you want to generate the natural numbers, you can do one of two things. You can just start with one and add one to itself over and over again. And that generates you the natural numbers. So additively, they're very easy to generate one, two, three, four, five. Or you can take the prime number if you want to generate multiplicatively, you can take all the prime numbers, two, three, five, seven and multiply them all together. Together that gives you all the natural numbers except maybe for one. So there are these two separate ways of thinking about the natural numbers from an additive point of view and a multiplicative point of view. And separately, they're not so bad. So any question about that natural was it only was addition, it's relatively easy to solve.

(02:19:11) And any question that only was multiplication is relatively easy to solve. But what has been frustrating is that you combine the two together and suddenly you get the extremely rich... I mean, we know that there are statements in number theory that are actually as undecidable. There are certain polynomials in some number of variables. Is there a solution in the natural numbers? And the answer depends on an undecidable statement whether the axioms of mathematics are consistent or not. But even the simplest problems that combine something more applicative such as the primes with something additives such as shifting by two, separately we understand both of them well, but if you ask when you shift the prime by two, can you get up? How often can you get another prime? It's been amazingly hard to relate the two.

Lex Fridman

(02:19:59) And we should say that the twin prime conjecture is just that, it pauses that there are infinitely many pairs of prime numbers that differ by two. Now the interesting thing is that you have been very successful at pushing forward the field in answering these complicated questions of this variety. Like you mentioned the Green-Tao Theorem. It proves that prime numbers contain arithmetic progressions of any length.

Terence Tao

(02:20:25) Right.

Lex Fridman

(02:20:25) It's just mind-boggling that you could prove something like that.

Terence Tao

(02:20:27) Right. Yeah. So what we've realized because of this type of research is that different patterns have different levels of indestructibility. What makes the twin prime problem hard is that if you take all the primes in the world, three, five, seven, 11, and so forth, there are some twins in there, 11 and 13 is a twin prime, pair of twin primes and so forth. But you could easily, if you wanted to redact the primes to get rid of these twins. The twins, they'd show up and they're infinitely many of them, but they're actually reasonably sparse. There's not, I mean, initially there's quite a few, but once you got to the millions, the trillions, they become rarer and rarer. And you could actually just, if someone was given access to the database of primes, you just edit out a few primes here and there.

(02:21:15) They could make the twin prime conjecture false by just removing 0.01% of the primes or something, just well-chosen to do this. And so you could present a censored database of the primes, which passes all of these statistical tests of the primes. It obeys things like the polynomial theorem and other effects of the primes, but doesn't contain any twin primes anymore. And this is a real obstacle to the twin-prime conjecture. It means that any proof strategy to actually find twin primes in the actual primes must fail when applied to these slightly edited primes. And so it must be some very subtle, delicate feature of the primes that you can't just get from aggregate statistical analysis.

Lex Fridman

(02:22:01) Okay, so that's out.

Terence Tao

(02:22:02) Yeah. On the other hand, progressions has turned out to be much more robust. You can take the primes and you can eliminate 99% of the primes actually, and you can take any 90% you want. And it turns out, and another thing we proved is that you still get arithmetic progressions. Arithmetic progressions are much, they're like cockroaches.

Lex Fridman

(02:22:21) Of arbitrary length though.

Terence Tao

(02:22:23) Yes. Yes.

Lex Fridman

(02:22:24) That's crazy.

Terence Tao

(02:22:25) Yeah.

Lex Fridman

(02:22:25) So for people who don't know, arithmetic progressions is a sequence of numbers that differ by some fixed amount.

Terence Tao

(02:22:32) Yeah. But it's again like, it's an infinite monkey type phenomenon. For any fixed length of your set, you don't get arbitrary length progressions. You only get quite short progressions.

Lex Fridman

(02:22:40) But you're saying twin-prime is not an infinite monkey phenomena. I mean, it's a very subtle monkey. It's still an infinite monkey phenomena.

Terence Tao

(02:22:48) Right. Yeah. If the primes were really genuinely random, if the primes were generated by monkeys, then yes, in fact the infinite monkey theorem would-

Lex Fridman

(02:22:56) Oh, but you're saying that twin prime, you can't use the same tools. It doesn't appear random almost.

Terence Tao

(02:23:05) Well, we don't know. We believe the primes behave like a random set. And so the reason why we care about the twin prime conjecture is a test case for whether we can genuinely confidently say with 0% chance of error that the primes behave like a random set. Random versions of the primes we know contain twins at least with 100% probably, or probably tending to 100% as you go out further and further. So the primes, we believe that they're random. The reason why arithmetic progressions are indestructible is that regardless of whether it looks random or looks structured like periodic, in both cases the arithmetic progressions appear, but for different reasons. And this is basically all the ways in which the thing was... There are many proofs of this sort of arithmetic progression-type theorems.

(02:23:54) And they're all proven by some sort of dichotomy where your set is either structured or random and in both cases you can say something and then you put the two together. But in twin primes, if the primes are random, then you are happy, you win. If the primes are structured, they could be structured in a specific way that eliminates the twins. And we can't rule out that one conspiracy.

Lex Fridman

(02:24:16) And yet you were able to make, as I understand, progress on the K-tuple version

Terence Tao

(02:24:21) Right. Yeah. So the one funny thing about conspiracies is that any one conspiracy theory is really hard to disprove. That if you believe the world is one by lizards is that here's some evidence that it's not [inaudible 02:24:32] work, that it was just talked about lizards. You might have encountered this kind of phenomena.

Lex Fridman

(02:24:38) Yes.

Terence Tao

(02:24:41) There's almost no way to definitively rule out a conspiracy. And the same is true in mathematics. A conspiracy that is solely devoted to eliminating twin primes, you would have to also infiltrate other areas of mathematics, but it could be made consistent at least as far as we know. But there's a weird phenomenon that you can make one conspiracy rule out other conspiracies. So if the world is run by lizards, it can't also be one by aliens, right?

Lex Fridman

(02:25:09) Right.

Terence Tao

(02:25:09) So one unreasonable thing is hard to disprove, but more than one, there are tools. So yeah, so for example, we know there's infinitely many primes that no two, which... So there are infinite pairs of primes which differ by at most, 246 actually is the code.

Lex Fridman

(02:25:26) Oh, so there's like a bound on the-

Terence Tao

(02:25:28) Right. So there's twin primes, there's a thing called cousin primes that differ by four. There's a thing called sexy primes that differ by six.

Lex Fridman

(02:25:36) What are sexy primes?

Terence Tao

(02:25:38) Primes that differ by six. The name is much less... It causes much less exciting than the name suggests.

Lex Fridman

(02:25:43) Got it.

Terence Tao

(02:25:45) So you can make a conspiracy rule out one of these, but once you have 50 of them, it turns out that you can't rule out all of them at once. It requires too much energy somehow in this conspiracy space.

Lex Fridman

(02:25:55) How do you do the bound part? How do you develop a bound for the differentiated team deposit-

Terence Tao

(02:26:01) Okay.

Lex Fridman

(02:26:01) ... that there's an infinite number of?

Terence Tao

(02:26:03) So it's ultimately based on what's called the pigeonhole principle. So the pigeonhole principle is a statement that if you have a number of pigeons, and they all have to go into pigeonholes and you have more pigeons than pigeonholes, then one of the pigeonholes has to have at least two pigeons in it. So there has to be two pigeons that are close together. So for instance, if you have 100 numbers and they all range from one to 1,000, two of them have to be at most 10 apart because you can divide up the numbers from one to 100 into 100 pigeonholes. Let's say if you have 101 numbers. 101 numbers, then two of them have to be a distance less than 10 apart because two of them have to belong to the same pigeonhole. So it's a basic feature of a basic principle in mathematics.

(02:26:45) So it doesn't quite work with the primes already because if the primes get sparser and sparser as you go out, that there are fewer and fewer numbers are prime. But it turns out that there's a way to assign weights to numbers. So there are numbers that are kind of

almost prime, but they don't have no factors at all other than themselves and one. But they have very few factors. And it turns out that we understand almost primes a lot better than primes. And so for example, it was known for a long time that there were twin almost primes. This has been worked out. So almost primes are something we can understand. So you can actually restrict the attention to a suitable set of almost primes. And whereas the primes are very sparse overall relative to the almost primes actually are much less sparse.

(02:27:33) You can set up a set of almost primes where the primes have density like say 1%. And that gives you a shot at proving by applying some sort of pigeonhole principle that there's pairs of primes that are just only 100 apart. But in order to prove the twin prime conjecture, you need to get the density of primes, this having also up to a threshold of 50%. Once you get up to 50%, you will get twin primes. But unfortunately, there are barriers. We know that no matter what kind of good set of almost primes you pick, the density of primes can never get above 50%. It's what the parity barrier and I would love to fight. So one of my long-term dreams is to find a way to breach that barrier because it would open up not only the twin prime conjecture but the Goldbach conjecture.

(02:28:12) And many other problems in number theory are currently blocked because our current techniques would require going beyond this theoretical parity barrier. It's like going fast as the speed of light.

Lex Fridman

(02:28:24) Yeah. So we should say a twin prime conjecture, one of the biggest problems in the history of mathematics. Goldbach conjecture also. They feel like next-door neighbors. Has there been days when you felt you saw the path?

Terence Tao

(02:28:37) Oh, yeah. Yeah. Sometimes you try something and it works super well. You again, get the sense of mathematical smell we talked about earlier. You learn from experience when things are going too well because there are certain difficulties that you sort of have to encounter. I think the way of colleague might put it is that if you are on the streets of New York and you're put in a blindfold and you're put in a car and after some hours the blindfold is off and then you're in Beijing. I mean that was too easy somehow. There was no ocean being crossed. Even if you don't know exactly what was done, you're suspecting that something wasn't right.

Lex Fridman

(02:29:21) But is that still in the back of your head? Do you return to the prime numbers every once in a while to see?

Terence Tao

(02:29:29) Yeah, when I have nothing better to do, which is less and less. I get busy with so many things these days. But when I have free time and I'm not, I'm too frustrated to work on my real research projects, and I also don't want to do my administrative stuff or I don't want to do some errands for my family. I can play with these things for fun. And usually you get nowhere. You have to just say, "Okay, fine. Once again, nothing happened. I will move on." Very occasionally one of these problems or actually solved. Well, sometimes as you say, you think you solved it and then you forward for maybe 15 minutes and then you think, "I should check this. This is too easy, too good to be true." And it usually is.

Lex Fridman

(02:30:11) What's your gut say about when these problems would be solved, the twin prime and Goldbach?

Terence Tao

(02:30:16) The twin prime, I'll think we'll-

Lex Fridman

(02:30:18) 10 years?

Terence Tao

(02:30:19) ... keep getting more partial results. It does need at least one... This parity barrier is the biggest remaining obstacle. There are simpler versions of the conjecture where we are getting really close. So I think in 10 years we will have many more much closer results, we may not have the whole thing. So twin primes is somewhat close. The Riemann hypothesis I have no clue. It has happened by accident I think.

Lex Fridman

(02:30:47) So the Riemann hypothesis is a kind of more general conjecture about the distribution of prime numbers, right?

Terence Tao

(02:30:53) Right. Yeah. It's states that sort of viewed multiplicatively, for questions only involving multiplication, no addition. The primes really do behave as randomly as you could hope. So there's a phenomenon in probability called square root cancellation that if you want to poll, say America on some issue, and you ask one or two voters and you may have sampled a

bad sample, and then you get a really imprecise measurement of the full average. But if you sample more and more people, the accuracy gets better and better. And it accuracy improves the square root of the number of people you sample. So if you sample 1, 000 people, you can get a 2 or 3% margin of error. So in the same sense, if you measure the primes in a certain multiplicative sense, there's a certain type of statistic you can measure and it's called the Riemann's data function, and it fluctuates up and down.

(02:31:42) But in some sense, as you keep averaging more and more, if you sample more and more, the fluctuation should go down as if they were random. And there's a very precise way to quantify that. And the Riemann hypothesis is a very elegant way that captures this. But as with many other ways in mathematics, we have very few tools to show that something really genuinely behaves really random. And this is actually not just a little bit random, but it's asking that it behaves as random as it actually random set, this square root cancellation. And we know because of things related to the parity problem actually, that most of us' usual techniques cannot hope to settle this question. The proof has to come out of left field. But what that is, no one has any serious proposal. And there's various ways to solve. As I said, you can modify the primes a little bit and you can destroy the Riemann hypothesis.

(02:32:37) So it has to be very delicate. You can't apply something that has huge margins of error. It has to just barely work. And there's all these pitfalls that you dodge very adeptly.

Lex Fridman

(02:32:50) The prime numbers is just fascinating.

Terence Tao

(02:32:52) Yeah. Yeah, yeah.

Lex Fridman

(02:32:53) What to you is most mysterious about the prime numbers?

Terence Tao

(02:33:00) That's a good question. Conjecturally, we have a good model of them. I mean, as I said, I mean they have certain patterns, like the primes are usually odd, for instance. But apart from there's some obvious patterns, they behave very randomly and just assuming that they behave. So there's something called the Kramer random model of the primes that after a certain point, primes just behave like a random set. And there's various slight modifications to this model. But this has been a very good model. It matches the numerics. It tells us what to predict. I can tell you with complete certainty the twin prime conjecture is true.

The random model gives overwhelming odds it is true, I just can't prove it. Most of our mathematics is optimized for solving things with patterns in them.

(02:33:39) And the primes have this anti-pattern, as do almost everything really, but we can't prove that. I guess it's not mysterious that the primes be random because there's no reason for them to have any kind of secret pattern. But what is mysterious is what is the mechanism that really forces the randomness to happen? This is just absent.

Collatz conjecture

Lex Fridman

(02:34:04) Another incredibly surprisingly difficult problem is the Collatz conjecture.

Terence Tao

(02:34:09) Oh yes.

Lex Fridman

(02:34:10) Simple to state, beautiful to visualize in its simplicity and yet extremely difficult to solve. And yet you have been able to make progress. Paul Erdos said about the Collatz conjecture that mathematics may not be ready for such problems. Others have stated that it is an extraordinarily difficult problem, completely out of reach, this is in 2010, out of reach of present-day mathematics, and yet you have made some progress. Why is it so difficult to make? Can you actually even explain what it is, is the key to-

Terence Tao

(02:34:41) Oh, yeah. So it's a problem that you can explain. It helps with some visual aids. But yeah, so you take any natural number, like say 13, and you apply the following procedure to it. So if it's even, you divide it by two, and if it's odd, you multiply it by three and add one. So even numbers get smaller, odd numbers get bigger. So 13 would become 40 because 13 times 3 is 39, add one you get 40. So it's a simple process. For odd numbers and even numbers, they're both very easy operations. And then you put it together, it's still reasonably simple. But then you ask what happens when you iterate it? You take the output that you just got and feed it back in. So 13 becomes 40, 40 is now even divide by two is 20. 20 is still even divide by 2, 10, five, and then five times three plus one is 16, and then eight, four, two, one. And then from one it goes one, four, two, one, four, two, one. It cycles forever. So this sequence I just described, 13, 40, 20, 10, so both, these are what is known as hailstorm sequences, because there's an oversimplified model of hailstorm formation which is not actual-

ly quite correct but it's still somehow taught to high school students as a first approximation, is that a little nugget of ice gets an ice crystal forms and clouded. It goes up and down because of the wind. And sometimes when it's cold it acquires a bit more mass and maybe it melts a little bit. And this process of going up and down creates this partially melted ice which eventually causes hailstorm, and eventually it falls down to the earth. So the conjecture is that no matter how high you start up, you take a number which is in the millions or billions, this process that goes up, if you are odd and down, it eventually comes down to Earth all the time.

Lex Fridman

(02:36:23) No matter where you start with very simple algorithm, you end up at one. And you might climb for a while-

Terence Tao

(02:36:29) Right.

Lex Fridman

(02:36:29) ... you come down.

Terence Tao

(02:36:29) Yeah. So yeah, if you plotted these sequences, they look like Brownian motion. They look like the stock market. They just go up and down in a seemingly random pattern. And in fact, usually that's what happens, that if you plug in a random number, you can actually prove, at least initially, that it would look like a random walk. And that's actually a random walk with a downward drift. It's like if you are always gambling on a roulette at the casino with odds slightly weighted against you. So sometimes you win, sometimes you lose. But over in the long run, you lose a bit more than you win. And so normally your wallet will go to zero if you just keep playing over and over again.

Lex Fridman

(02:37:07) So statistically it makes sense that we go here?

Terence Tao

(02:37:11) Yes. So the result that I proved roughly speaking such that statistically like 99% of all inputs would drift down to maybe not all the way to one, but to be much, much smaller than what you started. So it's like if I told you that if you go to a casino, most of the time you end up, if you keep playing it for long enough, you end up with a smaller amount in your wallet than when you started. That's kind of like the result that I proved.

Lex Fridman

(02:37:35) So why is that result... Can you continue down that thread to prove the full conjecture?

Terence Tao

(02:37:42) Well, the problem is that I used arguments from probability theory, and there's always this exceptional event. So in probability, we have this law of large numbers, which tells you things like if you play a casino with a game at a casino with a losing expectation over time you are guaranteed, almost surely with probability as close to 100% as you wish, you're guaranteed to lose money. But there's always this exceptional outlier. It is mathematically possible that even in the game is the odds are not in favor, you could just keep winning slightly more often than you lose. Very much like how in Navier-Stokes it could be, most of the time your waves can disperse, there could be just one outlier choice of initial conditions that would lead you to blow up. And there could be one outlier choice of a special number they stick in that shoots off infinity while all other numbers crash to Earth, crash to one.

(02:38:40) In fact, there's some mathematicians who, Alex Kontorovich for instance, who've proposed that actually these collapse iterations are like these similar Automator. Actually, if you look at what they happen in binary, they do actually look a little bit like these game of life type patterns. And in analogy to how the game of life can create these massive self-replicating objects and so forth, possibly you could create some sort of heavier-than-air flying machine. A number which is actually encoding this machine, which is just whose job it's to encode, is to create a version of something which is larger.

Lex Fridman

(02:39:17) Heavier-than-air machine encoded in a number-

Terence Tao

(02:39:20) Yeah.

Lex Fridman

(02:39:20) ... that flies forever.

Terence Tao

(02:39:22) So Conway in fact, worked on this problem as well.

Lex Fridman

(02:39:25) Oh wow.

Terence Tao

(02:39:26) Conway, so similar, in fact, that was more on inspirations for the Navier-Stokes project. Conway studied generalizations of the collapse problem where instead of multiplying by three and adding one or dividing by two, you have more complicated branching list. But instead of having two cases, maybe you have 17 cases and then you go up and down. And he showed that once your iteration gets complicated enough, you can actually encode Turing machines and you can actually make these problems undecidable and do things like this. In fact, he invented a programming language for these kind of fractional linear transformations. He called it frac-trat as a play on full-trat. And he showed that you can program, it was Turing-complete, you could make a program that if your number you insert in was encoded as a prime, it would sink to zero.

(02:40:13) It would go down, otherwise it would go up and things like that. So the general class of problems is really as complicated as all the mathematics.

Lex Fridman

(02:40:23) Some of the mystery of the cellular automata that we talked about, having a mathematical framework to say anything about cellular automata, maybe this same kind of framework is required. Yeah, Goldbach's conjecture.

Terence Tao

(02:40:35) Yeah. If you want to do it, not statistically, but you really want 100% of all inputs to for the earth. Yeah. So what might be feasible is, yeah, statistically 99% go to one, but everything, that looks hard.

P = NP

Lex Fridman

(02:40:50) What would you say is out of these within reach famous problems is the hardest problem we have today? Is it the Riemann hypothesis?

Terence Tao

(02:40:59) Well, it's up there. P equals NP is a good one because that's a meta problem. If you solve that in the positive sense that you can find a P equals NP algorithm, potentially, this solves a lot of other problems as well.

Lex Fridman

(02:41:14) And we should mention some of the conjectures we've been talking about. A lot of stuff is built on top of them now. There's ripple effects. P equals NP has more ripple effects than basically any other-

Terence Tao

(02:41:24) Right. If the Riemann hypothesis is disproven, that'd be a big mental shock to the number theorists. But it would have follow-on effects for cryptography, because a lot of cryptography uses number theory, uses number-theory constructions involving primes and so forth. And it relies very much on the intuition that number-theories are built over many, many years of what operations involving primes behave randomly and what ones don't? And in particular, encryption methods are designed to turn text-written information on it into text, which is indistinguishable from random noise. And hence, we believe to be almost impossible to crack, at least mathematically. But if something has caught our beliefs as the Riemann hypothesis is wrong, it means that there are actual patterns of the primes that we're not aware of.

(02:42:21) And if there's one, there's probably going to be more. And suddenly a lot of our crypto systems are in doubt.

Lex Fridman

(02:42:27) Yeah. But then how do you then say stuff about the primes-

Terence Tao

(02:42:33) Yeah.

Lex Fridman

(02:42:34) ... that you're going towards because of the Collatz conjecture again? Because, do you want it to be random, right?

Terence Tao

(02:42:41) Yes.

Lex Fridman

(02:42:41) You want it to be random?

Terence Tao

(02:42:43) Yeah. So more broadly, I'm just looking for more tools, more ways to show that things are random. How do you prove a conspiracy doesn't happen?

Lex Fridman

(02:42:49) Right. Is there any chance to you that P equals NP ? Can you imagine a possible universe?

Terence Tao

(02:42:57) It is possible. I mean, there's various scenarios. I mean, there's one where it is technically possible, but in fact it's never actually implementable. The evidence is sort of slightly pushing in favor of no, that probably P is not a good NP .

Lex Fridman

(02:43:11) I mean, it seems like it's one of those cases similar to Riemann hypothesis. I think the evidence is leaning pretty heavily on the no.

Terence Tao

(02:43:20) Certainly more on the no than on the yes. The funny thing about P equals NP is that we have also a lot more obstructions than we do for almost any other problem. So while there's evidence, we also have a lot of results ruling out many, many types of approaches to the problem. This is the one thing that the computer science has actually been very good at. It's actually saying that certain approaches cannot work. No-go theorems. It could be undecidable, yeah, we don't know.

Fields Medal

Lex Fridman

(02:43:43) There's a funny story I read that when you won the Fields Medal, somebody from the internet wrote you and asked, what are you going to do now that you've won this prestigious award? And then you just quickly, very humbly said that a shiny metal is not going to solve any of the problem I'm currently working on, so I'm going to keep working on them. First of all, it's funny to me that you would answer an email in that context, and second of all, it just shows your humility. But anyway, maybe you could speak to the Fields Medal, but it's another way for me to ask about Gregorio Perelman. What do you think about him famously declining the Fields Medal and the Millennium Prize, which came with a \$1 million of prize money. He stated that, "I'm not interested in money or fame. The prize is completely irrelevant for me. If the proof is correct, then no other recognition is needed."

Terence Tao

(02:44:40) Yeah, no, he's somewhat of an outlier, even among mathematicians who tend to have somewhat idealistic views. I've never met him. I think I'd be interested to meet him one

day, but I've never had the chance. I know people who met him. He's always had strong views about certain things. I mean, it's not like he was completely isolated from the math community. I mean, he would give talks and write papers and so forth, but at some point he just decided not.

Terence Tao

(02:45:00) ... He talks and write papers and so forth, but at some point he just decided not to engage with the rest of the community. He was disillusioned or something, I don't know. And he decided to peace out and collect mushrooms in St. Petersburg or something. And that's fine, you can do that. That's another sort of flip side. A lot of our problems that we solve, some of them do have practical application and that's great. But if you stop thinking about a problem, so he hasn't published since in this field, but that's fine. There's many, many other people who've done so as well.

(02:45:39) Yeah. So I guess one thing I didn't realize initially with the Fields Medal is that it sort of makes you part of the establishment. So most mathematicians, just career mathematicians, you just focus on publishing the next paper, maybe promote it one rank, and starting a few projects, may have taken some students or something. But then suddenly people want your opinion on things and you have to think a little bit about things that you might just foolishly say, because you know no one's going to listen to you, it's more important now.

Lex Fridman

(02:46:11) Is it constraining to you? Are you able to still have fun and be a rebel and try crazy stuff and play with ideas?

Terence Tao

(02:46:19) I have a lot less free time than I had previously, mostly by choice. I always say I have the option to sort of decline, so I decline a lot of things. I could decline even more or I could acquire a reputation of being so unreliable that people don't even ask anymore.

Lex Fridman

(02:46:38) I love the different algorithms here. This is great.

Terence Tao

(02:46:41) It's always an option, but there are things that I don't spend as much time as I do as a postdoc, just working on one problem at a time or fooling around. I still do that a little bit. But yeah, as you advance in your career, the more soft skills, so math somehow front-loads all the technical skills to the early stages of your career. So as a postdoc, you publish or

perish. You're incentivized to basically focus on proving very technical theorems, so prove yourself as well as prove the algorithms. But then as you get more senior, you have to start mentoring and giving interviews and trying to shape direction of field both research-wise and sometimes you have to do various administrative things. And it's kind the right social contract because you need to work in the trenches to see what can help mathematicians.

Lex Fridman

(02:47:40) The other side of the establishment, the really positive thing is that you get to be a light that's an inspiration to a lot of young mathematicians or young people that are just interested in mathematics. It's like-

Terence Tao

(02:47:52) Yeah, yeah.

Lex Fridman

(02:47:52) ... just how the human mind works. This is where I would probably say that I like the Fields Medal, that it does inspire a lot of young people somehow. This is just how human brains work. At the same time, I also want to give sort of respect to somebody like Grigori Perelman, who is critical of awards. In his mind, those are his principles and any human that's able for their principles to do the thing that most humans would not be able to do, it's beautiful to see.

Terence Tao

(02:48:25) Some recognition is necessary and important, but yeah, it's also important to not let these things take over your life and only be concerned about getting the next big award or whatever. So again, you see these people try to only solve really big math problems and not work on things that are less sexy, if you wish, but actually still interesting and instructive. As you say, the way the human mind works, we understand things better when they're attached to humans, and also if they're attached to a small number of humans. The way our human mind is wired, we can comprehend the relationships between 10 or 20 people. But once you get beyond like 100 people, there's a limit, I think there's a name for it, beyond which it just becomes the other.

(02:49:18) And so you have to simplify the [inaudible 02:49:21] 99.9% of humanity becomes the other. Often these models are incorrect, and this causes all kinds of problems. So yeah, to humanize a subject, if you identify a small number of people and say these are representative people of a subject, role models, for example, that has some role, but it can also be too much of it can be harmful because I'll be the first to say that my own career path is not that

of a typical mathematician. The very accelerated education, I skipped a lot of classes. I think I always had very fortunate mentoring opportunities, and I think I was at the right place at the right time. Just because someone doesn't have my trajectory, it doesn't mean that they can't be good mathematicians. They would be, but in a very different style, and we need people of a different style.

(02:50:16) And sometimes too much focus is given on the person who does the last step to complete a project in mathematics or elsewhere that's really taken centuries or decades with lots and lots of, building on lots of previous work. But that's a story that's difficult to tell if you're not an expert. It's easier to just say one person did this one thing. It makes for a much simpler history.

Lex Fridman

(02:50:40) I think on the whole, it is a hugely positive thing. To talk about Steve Jobs as a representative of Apple, when I personally know and of course everybody knows the incredible design, the incredible engineering teams, just the individual humans on those teams. They're not a team. They're individual humans on a team, and there's a lot of brilliance there, but it's just a nice shorthand, like π , Steve Jobs, π .

Terence Tao

(02:51:08) Yeah, as a starting point, as a first approximation that's how you-

Lex Fridman

(02:51:13) And then read some biographies and then look into much deeper first approximation.

Andrew Wiles and Fermat's Last Theorem

Terence Tao

(02:51:17) Yeah.

Lex Fridman

(02:51:17) That's right. So you mentioned you were at Princeton too. Andrew Wiles at that time-

Terence Tao

(02:51:22) Oh yeah.

Lex Fridman

(02:51:22) ... he was a professor there. It's a funny moment how history is just all interconnected, and at that time, he announced that he proved Fermat's Last Theorem. What did you think, maybe looking back now with more context about that moment in math history?

Terence Tao

(02:51:37) Yeah, so I was a graduate student at the time. I vaguely remember there was press attention and we all had the same, we had pigeonholes in the same mail room, so we all got mail and suddenly Andrew Wiles' mailbox exploded to be overflowing.

Lex Fridman

(02:51:53) That's a good metric.

Terence Tao

(02:51:54) Yeah. We all talked about it at tea and so forth. We didn't understand. Most of us sort of didn't understand the proof. We understand high level details. In fact, there's an ongoing project to formalize it in Lean. Kevin Buzzard is actually-

Lex Fridman

(02:52:09) Yeah. Can we take that small tangent? How difficult is that 'cause as I understand the proof for Fermat's Last Theorem has super complicated objects?

Terence Tao

(02:52:20) Yeah.

Lex Fridman

(02:52:21) It's really difficult to formalize now.

Terence Tao

(02:52:22) Yeah, I guess. Yeah, you're right. The objects that they use, you can define them. So they've been defined in Lean, so just defining what they are can be done. That's really not trivial, but it's been done. But there's a lot of really basic facts about these objects that have taken decades to prove in all these different math papers. And so lots of these have to be formalized as well. Kevin Buzzard's goal, actually he has a five-year grant to formalize Fermat's Last Theorem, and his aim is that he doesn't think he will be able to get all the way down to the basic axioms, but he wants to formalize it to the point where the only things that he needs to rely on is black boxes, are things that were known by 1980 to a number of theorists at the time, and then some other person or some other work would have to be done to get from there.

(02:53:13) So it's a different area of mathematics than the type of mathematics I'm used to. In analysis, which is my area, the objects we study are kind of much closer to the ground. I study things like prime numbers and functions and things that are within scope of a high school math education to at least define. But then, there's this very advanced algebraic side of number theory where people have been building structures upon structures for quite a while, and it's a very sturdy structure. It's been very... At the base, at least it's extremely well-developed with textbooks and so forth. But it does get to the point where if you haven't taken these years of study and you want to ask about what is going on at level six of this tower, you have to spend quite a bit of time before they can even get to the point where you can see something that you recognize.

Lex Fridman

(02:54:07) What inspires you about his journey that was similar, as we talked about, seven years mostly working in secret?

Terence Tao

(02:54:15) Yeah, so it kind of fits with the romantic image I think people have of mathematicians to the extent that they think of them all as these kind of eccentric wizards or something. So that's certainly kind of accentuated that perspective. It is a great achievement. His style of solving problems is so different from my own, which is great. We need people like that.

Lex Fridman

(02:54:46) Can you speak to it, like in terms of you like the collaborative?

Terence Tao

(02:54:49) I like moving on from a problem if it's giving too much difficulty.

Lex Fridman

(02:54:54) Got it.

Terence Tao

(02:54:55) But you need the people who have the tenacity and the fearlessness. I've collaborated with people like that where I want to give up 'cause the first approach that we tried didn't work and the second one didn't work. But they're convinced and they have third, fourth, and the fifth, which works. And I'd have to eat my words, "Okay. I didn't think this was going to work, but yes, you were right all along."

Productivity

Lex Fridman

(02:55:16) And we should say for people who don't know, not only are you known for the brilliance of your work, but the incredible productivity, just the number of papers, which are all very high quality. So there's something to be said about being able to jump from topic to topic.

Terence Tao

(02:55:31) Yeah, it works for me. But there are also people who are very productive and they focus very deeply. I think everyone has to find their own workflow. One thing which is a shame in mathematics is that mathematics has a sort a one-size-fits-all approach to teaching mathematics, and so we have a certain curriculum and so forth. Maybe if you do math competitions or something, you get a slightly different experience. But I think many people, they don't find their native math language until very late or usually too late. So they stop doing mathematics and they have a bad experience with a teacher who's trying to teach them one way to do mathematics that they don't like it.

(02:56:12) My theory is that humans don't come, evolution has not given us a math center of a brain directly. We have a vision center and a language center and some other centers, which evolution has honed, but we don't have an innate sense of mathematics. But our other centers are sophisticated enough that we can repurpose other areas of our brain to do mathematics. So some people have figured out how to use the visual center to do mathematics, and so they think things very visually when they do mathematics. Some people have repurposed their language center and they think very symbolically. Some people, if they are very competitive and they're gaming, there's a part of your brain that's very good at solving puzzles and games, and that can be repurposed.

(02:57:02) But when I talk about the mathematicians, they don't quite think that, I can tell that they're using some other different styles of thinking, not disjoint, but they may prefer visual. I don't actually prefer visual so much. I need lots of visual aids myself. Mathematics provides a common language, so we can still talk to each other even if we are thinking in different ways.

Lex Fridman

(02:57:26) But you could tell there's a different set of subsystems being used in the thinking process?

Terence Tao

(02:57:32) Yeah, they take different paths. They're very quick at things that I struggle with and vice versa, and yet they still get to the same goal.

Lex Fridman

(02:57:39) That's beautiful.

Terence Tao

(02:57:41) But the way we educate, unless you have a personalized tutor or something, education, sort of just financial skill has to be mass-produced, you have to teach the 30 kids. If they have 30 different styles, you can't teach 30 different ways.

Advice for young people

Lex Fridman

(02:57:55) On that topic, what advice would you give to students, young students who are struggling with math, but are interested in it and would like to get better? Is there something in this complicated educational context? What would you advise?

Terence Tao

(02:58:10) Yeah, it's a tricky problem. One nice thing is that there are now lots of sources for mathematical enrichment outside the classroom. So in my days, there were math competitions and there are also popular math books in the library. But now you have YouTube. There are forums just devoted to solving math puzzles. And math shows up in other places. For example, there are hobbyists who play poker for fun and they, for very specific reasons, are interested in very specific probability questions. And actually, there's a community of amateur probabilists in poker, in chess, in baseball. There's math all over the place, and I'm hoping actually with these new tools for Lean and so forth, that actually we can incorporate the broader public into math research projects. This almost doesn't happen at all currently.

(02:59:13) So in the sciences, there's some scope for citizen science, like astronomers. There are amateurs who would discover comets, and there's biologists that people who could identify butterflies and so forth. And in math, there are a small number of activities where amateur mathematicians can discover new primes and so forth. But previously, because we had to verify every single contribution, most mathematical research projects, it would not help to have input from the general public. In fact, it'll just be time-consuming because just error checking and everything. But one thing about these formalisation projects is that they are bringing in more people. So I'm sure there are high school students who've already con-

tributed to some of these formalizing projects, who've contributed to mathlib. You don't need to be a PhD holder to just work on one atomic thing.

Lex Fridman

(03:00:03) There's something about the formalisation here that also, as a very first step, opens it up to the programming community too. The people who are already comfortable with programming. It seems like programming is somehow maybe just the feeling, but it feels more accessible to folks than math. Math is seen as this extreme, especially modern mathematics is seen as this extremely difficult-to-enter area, and programming is not. So that could be just an entry point.

Terence Tao

(03:00:31) You can execute code and you can get results. You can print out the world pretty quickly. If programming was taught as an almost entirely theoretical subject where you're just taught the computer science, the theory of functions and routines and so forth, and outside of some very specialized homework assignments, you're not actually programming, like on the weekend for fun, they would be as considered as hard as math. So as I said, there are communities of non-mathematicians where they're deploying math for some very specific purpose, like optimizing their poker game, and for them, then math becomes fun for them.

Lex Fridman

(03:01:13) What advice would you give in general to young people how to pick a career, how to find themselves, what they could be good at?

Terence Tao

(03:01:25) That's a tough, tough, tough question. Yeah, so there's a lot of uncertainty now in the world. There was this period after the war where, at least in the West, if you came from a good demographic, there was a very stable path to it, to a good career. You go to college, you get an education, you pick one profession and you stick to it. It's becoming much more a thing of the past. So I think you just have to be adaptable and flexible. I think people will have to get skills that are transferable, like learning one specific programming language or one specific subject of mathematics or something. That itself is not a super transferable skill, but sort of knowing how to reason with abstract concepts or how to problem solve when things go wrong. Anyway, these are things which I think we will still need even as our tools get better, and you'll be working with AI supports and so forth.

Lex Fridman

(03:02:13) But actually you're an interesting case study. You're one of the great living mathematicians, and then you had a way of doing things, and then all of a sudden you start learning. First of all, you kept learning new fields, but you learned Lean. That's not a non-trivial thing to learn. For a lot of people, that's an extremely uncomfortable leap to take, right?

Terence Tao

(03:02:40) Yeah.

Lex Fridman

(03:02:41) A lot of mathematicians.

Terence Tao

(03:02:42) First of all, I've always been interested in new ways to do mathematics. I feel like a lot of the ways we do things right now are inefficient. Many of my colleagues, who spend a lot of time doing very routine computations or doing things that other mathematicians would instantly know how to do and we don't know how to do them, like how we search and get a quick response and so forth. So that's why I've always been interested in exploring new workflows.

(03:03:09) About four or five years ago, I was on a committee where we had to ask for ideas for interesting workshops to run at a math institute. And at the time, Peter Scholze had just formalized one of his new theorems, and there were some other developments in computer-assisted proof that look quite interesting. And I said, "Oh, we should run a workshop on this. This would be a good idea." And then I was a bit too enthusiastic about this idea, and so I got volunteered to actually run it. So I did with a bunch of other people, Kevin Buzzard and Jordan Ellenberg and a bunch of other people, and it wasn't a nice success. We pulled together a bunch of mathematicians and computer scientists and other people, and we got up to speed on state of the yard, and it was really interesting developments that most mathematicians didn't know was going on, lots of nice proofs of concept, just hints of what was going to happen. This was just before ChatGPT, but even then there was one talk about language models and the potential capability of those in the future.

(03:04:11) So that got me excited about the subject. So I started giving talks about this is something more of us should start looking at, now that I had arranged, run this conference. And then ChatGPT came out and suddenly AI was everywhere. And so I got interviewed a lot about this topic and in particular, the interaction between AI and [inaudible 03:04:33]. I said, "Yeah, they should be combined. This is perfect synergy to happen here." And at some point I realized that I have to actually not just talk the talk, but walk the walk. I don't work in machine

learning and I don't work in proof formalisation, and there's a limit to how much I can just rely on authority and say, "I'm a mathematician. Just trust me when I say that this is going to change mathematics," and I don't do any of it myself. So I felt like I had to actually justify it.

(03:05:03) A lot of what I get into, actually, I don't quite see in advance as how much time I'm going to spend on it, and it's only after I'm sort of waist deep in a project that I realize, but at that point, I'm committed.

Lex Fridman

(03:05:15) Well, that's deeply admirable that you're willing to go into the fray, be in some small way a beginner, or have some of the challenges that a beginner would, right?

Terence Tao

(03:05:27) Yeah.

Lex Fridman

(03:05:27) New concepts, new ways of thinking, also sucking at a thing that others... I think in that talk, you could be a Fields Medal-winning mathematician and an undergrad knows something better than you.

Terence Tao

(03:05:42) Yeah, I think mathematics inherently, mathematics is so huge these days that nobody knows all of modern mathematics. And inevitably, we make mistakes and you can't cover up your mistakes with just bravado because people will ask for your proofs, and if you don't have the proofs, you don't have the proofs.

Lex Fridman

(03:06:03) I love math.

Terence Tao

(03:06:04) Yeah, so it does keep us honest. It's not a perfect panacea, but I think we do have more of a culture of admitting error because we're forced to all the time.

The greatest mathematician of all time

Lex Fridman

(03:06:17) Big ridiculous question. I'm sorry for it once again. Who is the greatest mathematician of all time, maybe one who's no longer with us? Who are the candidates? Euler, Gauss, Newton, Ramanujan, Hilbert?

Terence Tao

(03:06:32) So first of all, as mentioned before, there's some time dependence.

Lex Fridman

(03:06:37) On the day.

Terence Tao

(03:06:38) Yeah. Like if you plot cumulatively over time, for example, Euclid is one of the leading contenders, and then maybe some unnamed anonymous mathematicians before that, whoever came up with the concept of numbers.

Lex Fridman

(03:06:53) Do mathematicians today still feel the impact of Hilbert, just-

Terence Tao

(03:06:57) Oh, yeah.

Lex Fridman

(03:06:58) Directly of what? Everything that's happened in the 20th century?

Terence Tao

(03:07:00) Yeah, Hilbert spaces, we have lots of things that are named after him of course. Just the arrangement of mathematics and just the introduction of certain concepts, 23 problems have been extremely influential.

Lex Fridman

(03:07:12) There's some strange power to the declaring which problems are hard to solve, the statement of the open problems.

Terence Tao

(03:07:19) Yeah, this is bystander effect everywhere. If no one says you should do X, everyone just mills around waiting for somebody else to do something, and nothing gets done. And the one thing that actually you have to teach undergraduates in mathematics is that you should always try something. So you see a lot of paralysis in an undergraduate trying a math problem. If they recognize that there's a certain technique that can be applied, they will try it. But there are problems which they see and none of their standard techniques obviously applies and the common reaction is than just paralysis, I don't know what to do. I think there's a quote from the Simpsons, "I've tried nothing and I'm all out of ideas." So the next step then is to try anything no matter how stupid and in fact almost the stupider, the better, which tech-

nically is almost guaranteed to fail, but the way it fails is going to be instructive. It fails 'cause you are not at all taking into account this hypothesis. Oh, this hypothesis must be useful. That's a clue.

Lex Fridman

(03:08:26) I think you also suggested somewhere this fascinating approach, which really stuck with me as they're using it, and it really works, I think you said it's called structured procrastination.

Terence Tao

(03:08:36) No, yes.

Lex Fridman

(03:08:37) It's when you really don't want to do a thing that you imagine a thing you don't want to do more that's worse than that and then in that way, you procrastinate by not doing the thing that's worse. It's a nice hack, it actually works.

Terence Tao

(03:08:51) Yeah, yeah. With anything, psychology is really important. You talk to athletes like marathon runners and so forth and they talk about what's the most important thing, is it the training regimen or the diet and so forth? So much of it is psychology, just tricking yourself to think that the problem is feasible so that you're motivated to do it.

Lex Fridman

(03:09:15) Is there something our human mind will never be able to comprehend?

Terence Tao

(03:09:21) Well, as a mathematician, [inaudible 03:09:23]. There must be some large number that you can't understand. That was the first thing that came to mind.

Lex Fridman

(03:09:31) So that, but even broadly, is there something about our mind that we're going to be limited even with the help of mathematics?

Terence Tao

(03:09:41) Well, okay, how much augmentation are you willing. Like for example, if I didn't even have a pen and paper, if I had no technology whatsoever, so I've not allowed black-board, pen and paper-

Lex Fridman

(03:09:52) You're already much more limited than you would be.

Terence Tao

(03:09:55) ... Incredibly limited. Even language, the English language is a technology. It's one that's been very internalized.

Lex Fridman

(03:10:03) So you're right, the formulation of the problem is incorrect 'cause there really is no longer just a solo human already augmented in extremely complicated intricate ways, right?

Terence Tao

(03:10:17) Yeah. Yeah.

Lex Fridman

(03:10:18) So like a collective intelligence?

Terence Tao

(03:10:20) Yes. Yeah, I guess, so humanity plural has much more intelligence in principle on its good days than the individual humans put together. It can have less, but yeah, so the mathematical community plural is incredibly super intelligent entity that no single human mathematician can come closer to replicating. You see it a little bit on these question analysis sites. So this math overflow, which is the math version of stackable flow, sometimes you get this very quick response to very difficult questions from the community, and it's a pleasure to watch actually, as an expert.

Lex Fridman

(03:11:01) I'm a fan spectator of that site, just seeing the brilliance of the different people, the depth and knowledge that people have. And the willingness to engage in the rigor and the nuance of the particular question, it's pretty cool to watch. It's almost like just fun to watch. What gives you hope about this whole thing we have going on with human civilization?

Terence Tao

(03:11:25) I think the younger generation is always really creative and enthusiastic and inventive. It's a pleasure working with young students. The progress of science tells us that the problems that used to be really difficult can become trivial to solve. Like navigation, just knowing where you work on the planet was this horrendous problem. People died or lost fortunes because they couldn't navigate. And we have devices in our pockets that do this automatically for us, like it is a completely solved problem. So things that seem unfeasible for us now, could be maybe just homework exercises.

Lex Fridman

(03:12:13) Yeah. One of the things I find really sad about the finiteness of life is that I won't get to see all the cool things we create as a civilization because in the next 100 years, 200 years, just imagine showing up in 200 years.

Terence Tao

(03:12:27) Yeah, well, already plenty has happened. If you could go back in time and talk to your teenage self or something, the internet and now AI, again, they're getting to internalize and yeah, of course, AI can understand our voice and give reasonable slightly incorrect answers to any question. But yeah, this was mind-blowing even two years ago.

Lex Fridman

(03:12:50) And in the moment, it's hilarious to watch on the internet and so on, the drama, people take everything for granted very quickly, and then we humans seem to entertain ourselves with drama. Out of anything that's created, somebody needs to take one opinion, another person needs to take an opposite opinion, argue with each other about it. But when you look at the arc of things, just even in the progress of robotics, just to take a step back and be like, "Wow, this is beautiful, that we humans are able to create this."

Terence Tao

(03:13:19) When the infrastructure and the culture is healthy, the community of humans can be so much more intelligent and mature and rational than the individuals within it.

Lex Fridman

(03:13:31) Well, one place I can always count on rationality is the Comment section of your blog, which I'm a big fan of. There's a lot of really smart people there. And thank you, of course, for putting those ideas out on the blog. And I can't tell you how honored I am that you would spend your time with me today. I was looking forward to this for a long time. Terry, I'm a huge fan. You inspire me, you inspire millions of people. Thank you so much for time.

Terence Tao

(03:13:58) Thank you. It was a pleasure.

Lex Fridman

(03:14:00) Thanks for listening to this conversation with Terrence Tao. To support this podcast, please check out our sponsors in the description or at lexfridman.com/sponsors. And

now, let me leave you with some words from Galileo Galilei, “Mathematics is a language with which God has written the universe.”

[\(03:14:21\)](#) Thank you for listening and hope to see you next time.