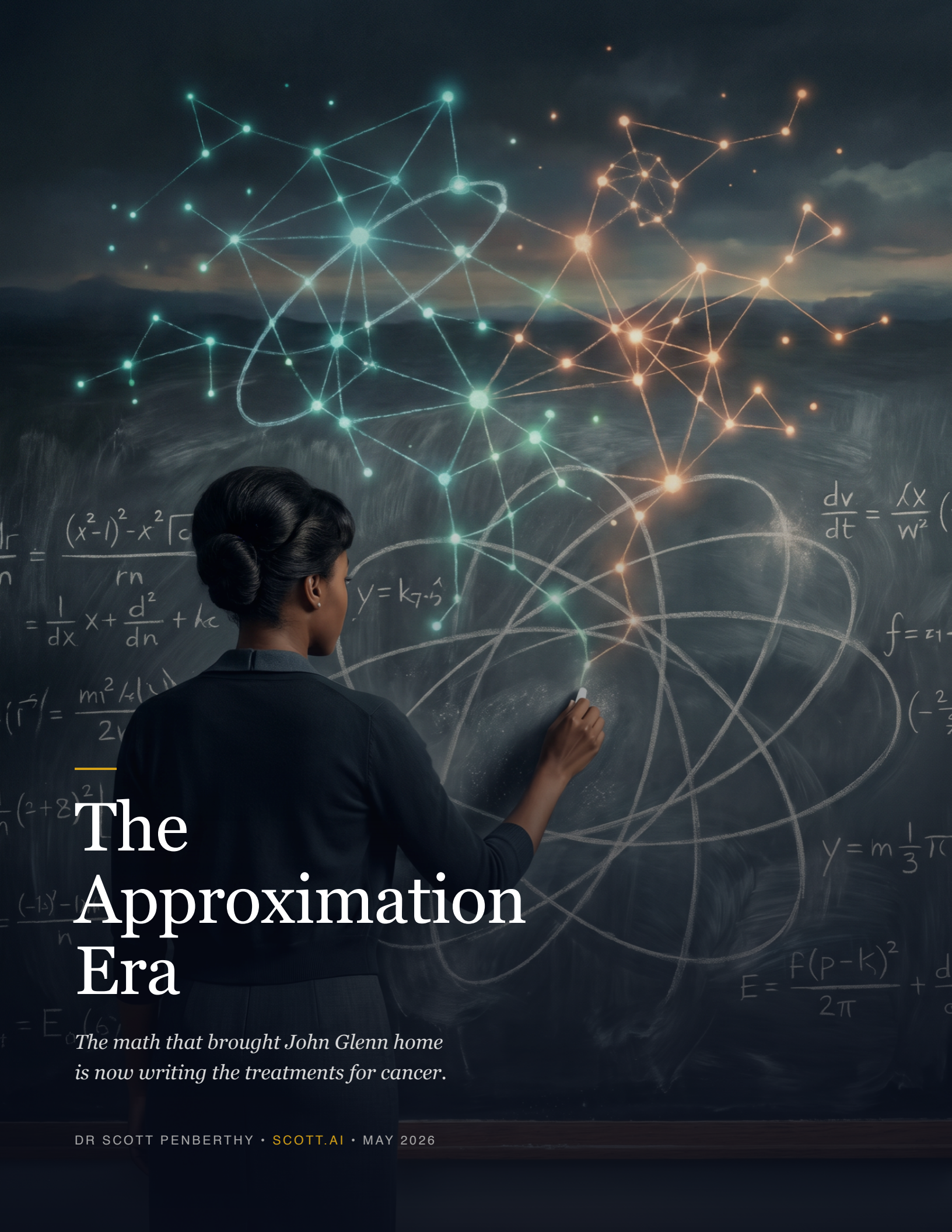


The Approximation Era

The math that brought John Glenn home is now writing the treatments for cancer.

DR SCOTT PENBERTHY • SCOTT.AI • MAY 2026

The thesis

Compute is replacing calculus. Not everybody believes this. I have been working through the thesis for about a decade. I call it the Approximation Era because that is what it actually is. Walk through it with me and make up your own mind.

The core claim is simple. A large enough neural network is a universal approximator of any relationship in the observable universe. A large enough generative model is a universal generator of any sequence in that universe. Together, they let us trade differential equations for GPUs. We can compute our way to answers that used to require closed-form solutions, across creativity, science, biology, and physics, all with the same underlying machinery.

If that is true, then AGI and ASI are not the destination. They are checkpoints on a much longer route, where every discipline becomes computable through observation rather than derivation.

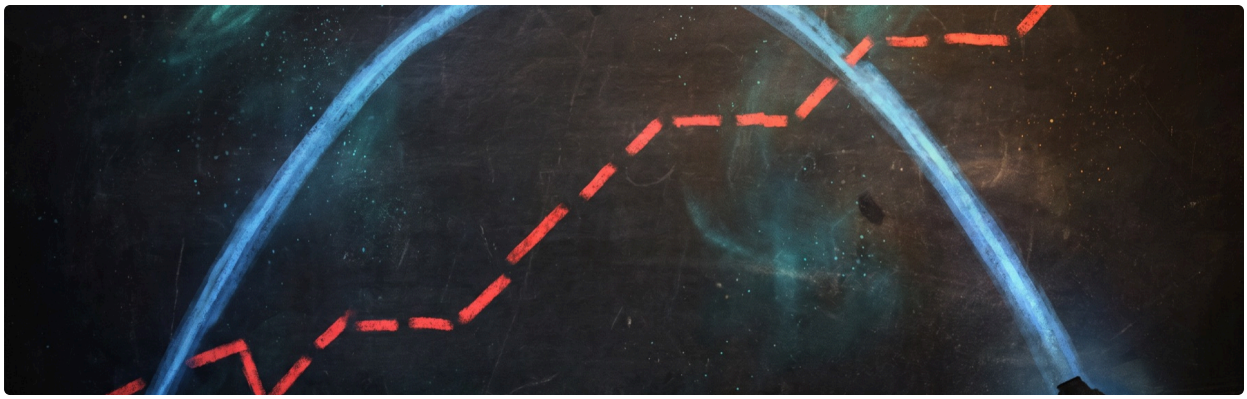
Katherine Johnson and the Euler method

The pattern starts in 1960, with a human computer named Katherine Johnson trying to keep John Glenn from burning up on reentry.

Nobody could find a closed-form solution for the path from orbit through the atmosphere. The calculus was beyond us. Johnson reached for an old technique, the Euler method. She did not know the equation either. Instead, every second or so, she figured out where the capsule was, adjusted, and stepped forward. Zero, one, two, three. Each step a small straight line approximating a curve she could not write down.

The real equation lived in four dimensions: three of space plus time. Her approximation was piecewise and linear. Wrong in the sense that Newtonian physics is wrong. Right in the sense that John Glenn came home. The technique she developed brought Gemini home too, and was carried forward into Apollo. I can imagine John asking, before he climbed into the capsule, whether Katherine had run the numbers.

Hold that image. Compute and small linear pieces, replacing a closed-form solution we could not derive. That is the whole story. Everything since is scale.



Euler's method. The true curve is unknowable in closed form. The piecewise approximation is computable. John Glenn came home on the second one.

Three strange properties of the observable universe

The reason approximation works at all is that the universe we can see is unreasonably friendly to it. Three properties make this true.

Low dimensionality. This should not work, but it does. Energy is mass times c squared, not mass to the hundredth power. Gravity falls off as the inverse square of distance. Fluids and rockets and trips to

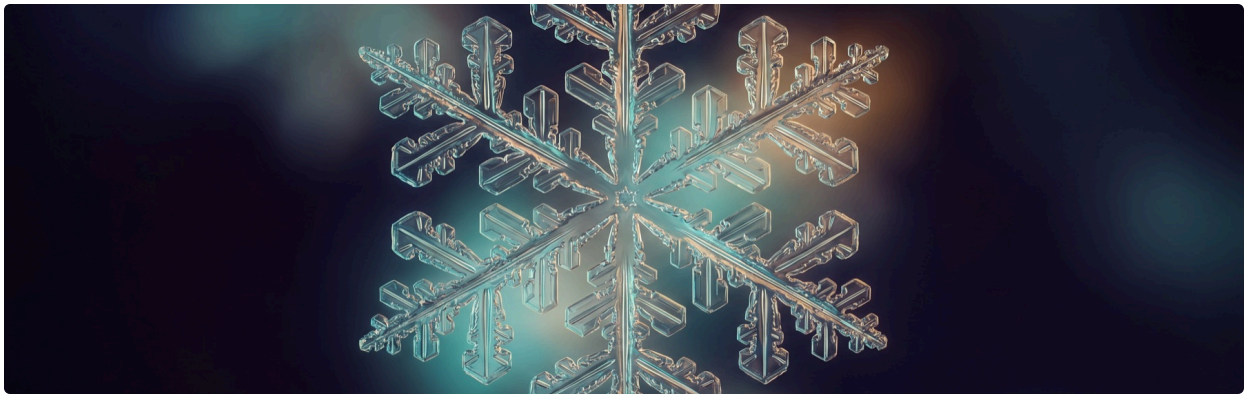
Mars are governed by Navier-Stokes, third degree. Most of what built the industrial revolution is third degree or lower. Not correct, but good enough.

Locality of reference. We can reason about Palo Alto without worrying about a butterfly in Japan or a rock on the moon. Quantum mechanics says everything is connected. For what we observe, we can ignore that and still get along.

Composability. We can reason about nucleotides, then DNA, then RNA, then cells, then organs, then bodies, then populations, then disease maps. We can do the same for parts, cars, fleets. The universe stacks cleanly across scales.

Low-dimensional, local, composable. A universe built for approximation.

There are corners where this breaks. Turbulence resists low-dimensional compression. High-energy physics climbs back into very high dimensions. Some combinatorial problems refuse to be smooth. The thesis here is not that approximation works on every problem in the universe. It is that approximation works on enough of the universe to remake civilization. That is a smaller claim and a more defensible one.



Recursion, locality, low dimensionality. The universe stacks cleanly across scales, and that is why the engine works.

The universal approximator

In 2010, Ilya Sutskever joined Geoffrey Hinton and told him he was not thinking big enough. Not thousands of neurons. Millions.

The first big result was the convolutional neural network. Take a photo of a dog in a pile of laundry. Pixels feed a grid. The grid feeds a smaller grid, then smaller, then smaller, like layers of the optic nerve. The connections start as noise. At the top, two idiot lights: cat, dog. Show the photo. The cat light comes on. Wrong. Nudge the weights backward through the network. Do it a few million times. The cat function and the dog function are now in the weights.

That is more than a clever classifier. That is a universal approximator. Any relationship in the observable universe, the theory says, can be approximated by a sufficiently deep network. Not in four dimensions like Johnson. Try twelve thousand. Each neuron is a small nonlinear piece. The same job, at a scale and dimensionality Johnson could not have imagined. The work won a Nobel Prize.

The universal approximator. Same technique as Katherine Johnson's chalk, executed in twelve thousand dimensions instead of four.

The universal generator

A single shot of cat or dog is one thing. Language, video, biology, an organism through its life, all of these change over time. Two contributions made sequences computable.

The attention head is a lookup mechanism. Based on what the network is observing right now, it selects which weights to apply in the next step of the computation. Smell pumpkin spice, look up the weights for fall. The same input activates different downstream paths depending on context. Correlations that move through time, captured directly through context-conditioned weight selection.

The diffusion model came from the physics community. Take an image. Add a little Bayesian noise. Do it a thousand times until the image is gone. Then learn to play the tape backwards, like a Beatles album in reverse. If you can reverse the noise, you can generate anything from noise.

Two sides of the same coin. Together they form a universal generator. Given enough observation, any sequence becomes producible.

We now have both pieces. A universal approximator and a universal generator. That is the engine of the Approximation Era.

The engine is not one fixed architecture. Each new domain has demanded its own innovations. Reinforcement learning from human feedback for language. Equivariant networks for proteins. Latent diffusion for images. Joint action-frame models for robotics. What is universal is the technique, not the wiring. Gradient descent on parameterized non-linear stacks, scaled until it works, applied to the data of the problem in front of you. The wiring is engineering. The technique is the era.

Diffusion in a single still. Order rising from chaos, learned by reversing the destruction.

Logging the sky

The thesis above is what I think the math is doing. A scanner I built is how I check.

Like Copernicus with a notebook, I started a daily log. One verified AI deployment per day. No plan for the data, just a willingness to write down what was actually happening as it formed. The notebook would tell me what was real eventually.

I ran out of hand-curated examples in three months.

So I built a scanner that reads every earnings call and every SEC filing across the S&P 500, MidCap 400, SmallCap 600, Russell 2000, and major international indexes. Every AI mention extracted, validated against independent sources, scored for how real it is. It now tracks 5,752 verified AI deployments across 2,385 public companies. The quarterly analysis is published as *Where AI Creates Value* at scott.ai.

What the scanner shows lines up with what the thesis predicts. Vertical AI wins. Physical AI wins. The places that touch atoms, regulation, time-locked data, costly failure, and scarce expertise are where the returns concentrate. The companies running approximators against problems the universe is structured to allow are the ones the market is rewarding.

I do not yet know where the next year of logging takes us. I know it will be epic.

From cat videos to cancer

In 2020, when the thesis was just forming in my head, I met my mentor Lee Hood. He was stacking measurements through time: phenotype data, DNA, mitochondrial RNA, copy number variants, methylated DNA, RNA sequence. Looking for correlations across the stack as it evolved.

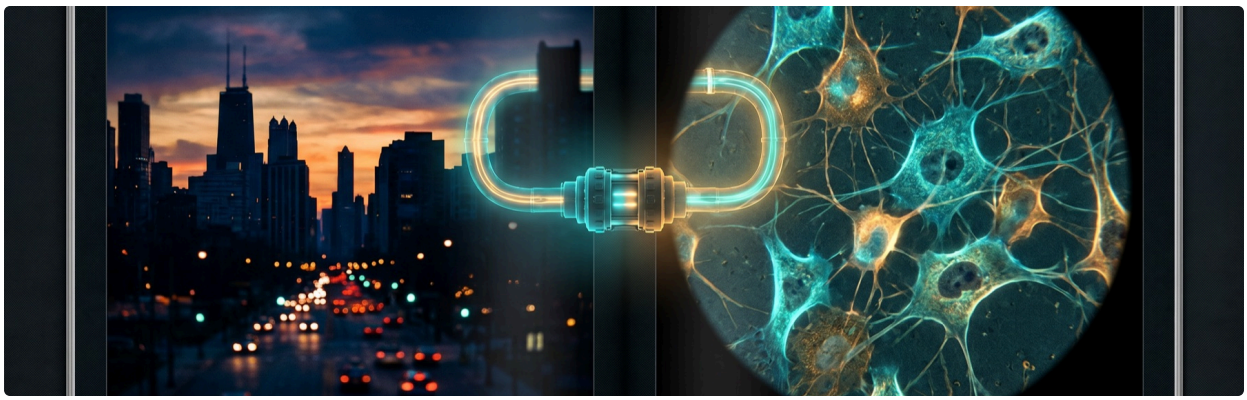
It hit me. That is video.

Video stacks channels: images, audio, dialogue, sound effects, characters, all aligned through time. If our models could learn physics by watching video, the same architecture would learn biology by watching cells. Lee called it phenomics. Break the body down to nanoscale, treat each measurement like a pixel, and apply the models we built for movies.

When I pitched a version of this to a well-known CEO, I got a polite version of "that's theoretical, you're academics." A few years later I bet I could cross the uncanny valley in generated video. The video game community said no chance. We've been at this for years, your technique will not work.

So I built a proof of concept to test my conviction in the Approximation Era. The premise: take one ad for a car and personalize it nine thousand ways in five minutes. Three hundred cities, two genders, three age groups, five personal interests. I met a wall of resistance. Not plausible. Nobody is asking for that. Nobody will.

Nobody is laughing now. The product ships. You cannot tell the personalized videos are synthetic.



Same machine, different inputs. From cat memes to cancer.

World models from three bits

While we were doing that, another team took the video model and added one input: a joystick. Up, down, left, right, fire. Three bits. Then they trained on video games instead of general video, asking the model to guess both the next frame and the next button press.

That became Genie. Rewire your joystick into the model, and now you are exploring a world the model is imagining frame by frame, consistent with both your inputs and the physics it has learned. Three bits per second of control, and a coherent world appears.

Now imagine that with kilobits or megabits of control input instead of three. We are just getting started.

Three bits of joystick input. A coherent world rendering itself frame by frame in response.

Programming the body

The same machinery that learns cat-or-not is now learning to read DNA and program cells.

Molecules have faces, called antigens. Your immune system looks for them. Take a sample of cancerous tissue. Use AI to find the DNA sequence that codes for the antigen on the surface of the cancer cell. Wrap that code in a lipid chassis. Inject it into the arm. The code drains into the lymph nodes, gets expressed on the surface of lymph cells, and the immune system attacks. T cells learn the face. They hunt the cancer. This is in trials for HER2-positive disease right now. That is what an mRNA vaccine actually is. Programmable defense.

For patients with weakened immune systems, the same technique reprograms the slingshot itself. CAR-T therapy attaches a custom antibody to a target cell and flips inside. Same approximation. Different lever.

An AI engineer in Australia ran this play when his dog was dying of cancer. Sequenced the dog's DNA. Sequenced the tumor. Found the difference. Targeted it. The dog is in remission.

David Fajgenbaum took it further. Diagnosed with Castleman disease and given a short prognosis, he refused to wait the seven years it takes to approve a new drug. He went looking for an existing drug with the right shape. He found it. He injected himself. He treated his own disease. He now runs EveryCure, scanning for shape similarity in tensor space between known drugs and orphan diseases. He called me once to say he thought lidocaine might help with breast cancer. Our AI co-scientist found the pathway to validate the hypothesis. He has 77 million candidate drug-disease pairs to evaluate. He will publish this year.

AlphaFold 3 fused the diffusion model with the transformer and asked: will this protein bind to this ligand? One frame of life. What used to be a five-year PhD thesis runs in two days, across two hundred thousand proteins. A million years of human work, published and given away. Another Nobel Prize, the same techniques.

A single frame is the current achievement. The bet that follows is much larger. If we can approximate one moment of a biological interaction in tensor space, the next step is to play it forward. Frame after frame.

The full 3D folding process. Cells dividing. Proteins binding and unbinding. Drugs traversing membranes. Tumors growing or shrinking under treatment. The whole movie of life, computed instead of observed. If that lands, the wet lab becomes optional for entire classes of experiment. That is the future bet. It is the same bet as Stardust, scaled to biology. Most of the people in the room will say it is not plausible.

Programmable defense. The same machinery that learns cat-or-not is now reading DNA and designing the molecules that hunt cancer.

Cheap sensors, custom silicon

The standard objection is cost. Nanoscale data is expensive. Inference is expensive. Neither holds up.

Jen Dionne showed me a sensor on a glass slide, smaller than her thumbnail. She describes it as a hundred thousand pairs of sunglasses on a tiny square. Each cell on the chip is coated to resonate with a specific magnetic frequency. DNA molecules have resonant signatures, like a tuning fork. Flow a sample across the chip in infrared light, and each cell that sees its target rings. One chip, a hundred thousand molecules at once. At scale, the chip costs less than a penny. Reagents bring the total to about five dollars per run.

On the inference side, a generation of startups is doing for LLMs what FPGAs did for cell towers. Build a custom chip per model release. Bake the layer structure and the weights into silicon at the last moment. One company is hitting fifteen to twenty thousand tokens per second on a small four-billion-parameter Llama as a proof of principle, where best in class on general hardware is around two thousand. If the same approach scales to frontier models, the cost per inference is about to plummet.

Tokens and watts

The constraint is not algorithms. It is power.

Run the math. Assume an AGI can be served by an eight-way H100 box. That is sixty kilograms, the kind of thing they are now talking about putting in satellites. Assume one for every human and one for every robot roaming a warehouse. A billion of each, two billion total. Each chip pulls about seven hundred watts, the same as a vacuum cleaner. Eight chips per box is roughly fifty-six hundred watts. Two billion users at fifty-six hundred watts is around ten terawatts.

The planet generates about 9.8.

This is not theoretical. Inside Google, working on the video personalization project, I could not get enough TPU capacity. I was hitting daily limits by ten or eleven in the morning, then bartering for tokens with other teams to keep going. I ended up borrowing cloud credits from research budgets to run experiments on the public cloud. We are already against the wall.

The grid is the bottleneck. Tokens and watts are the currency of the next twenty years.

The era's hard limit. Algorithms are not the bottleneck. Power is.

What this means

ERA	TOOL	LIMIT
Newtonian	Closed-form equations	What humans can derive
Approximation	High-dimensional piecewise nonlinear approximation	Tokens and watts

The Approximation Era is not a more accurate physics. Neither was Newton. It is a better tool, because we are trading calculus for compute, and compute scales.

The same architecture learns cat or dog. It learns to render a personalized car ad in three hundred cities. It learns to imagine a playable world from three bits of joystick input. It learns to design an mRNA vaccine. It learns to dock a protein and a ligand. It learns to find an existing drug for a disease that has no treatment. One technique. From cat memes to cancer.

Creativity, science, mathematics, biology, all becoming computable through observation rather than derivation. That is the era we are in. Plenty of smart people do not believe it. They might be right. I am a huge fan.

Building toward the treatments

The Approximation Era reframes the next twenty years.

Katherine Johnson sat at a chalkboard in 1960 and approximated her way to the moon. Her tools were a slide rule, a pencil, and the most powerful piece of intelligence in the room, which was her. The technique she used to recover Apollo is the same technique we now use to recover lost dogs from cancer, to put faces on tumors so the immune system can see them, to find old drugs that solve new diseases.

The hardware has scaled by twelve orders of magnitude. The math is the same.

If you are an executive, the implication is that nothing about your industry is safe from the engine. Every domain that exhibits low dimensionality, locality, and composability is now a candidate for approximation. That includes your operations, your supply chain, your customer experience, your R&D pipeline, your products, and the diseases your employees and their families carry. The companies that figure this out first will run a long time.

If you are a researcher, the implication is that the bottleneck moved. The mathematics is settled. The compute is available. The data is collectable. What remains is the imagination to apply the engine to the problem in front of you, and the will to keep pushing through the people who will tell you it is not plausible. They told Sutskever the networks were too small. They told the diffusion crowd it would never con-

verge. They told the AlphaFold team biology was too messy. They were wrong every time, and they will be wrong on the next one.

If you are a patient, or you love one, the implication is the most important. We are not promising cures. We are promising treatments. The distinction matters. Diabetes is not cured. It is managed, and a person with diabetes today lives a full life because of accumulated incremental wins over a hundred years of science. Cancer is on the same trajectory, only faster, because the engine driving it is no longer the bench scientist alone. The engine is the bench scientist plus the approximator. mRNA programmable defense. Drug repurposing at planetary scale. Custom antibodies designed in tensor space. Each of these is a thread. Pull on enough of them and many cancers stop being lethal even when they are not eliminated. The arc bends from terminal to chronic to managed.

The science being possible is not the same as the treatment being delivered. Translation through clinical trials, regulatory review, manufacturing, payer adoption, and physician practice takes years on its own clock. The era moves the science. The institutions still have to do their work. I am building toward both.

Lee Hood and I have been circling the idea of an effort to thwart cancer, grounded in data and this thesis. If what we are seeing in the scanner and in the lab keeps pointing where it is pointing, that effort becomes the work I most want to do next.

Katherine Johnson did the math by hand because there was no other way. We have other ways now. The work is the same. Show up. Take the next step. Keep stepping until John gets home.

Two eras, same act. The chalk is gone. The work is the same.

About this work

The thesis is mine. The Nobel Prizes belong to the people who proved the math. The work ahead belongs to whoever can keep the lights on.

→ scott.ai