Introduction

I would like to build machines which utilize nature's untapped potential for computation, data storage, and data movement. Nature abounds in computational processes, evolved systems with behavior that can be described algorithmically. Nature also offers repositories for information, and mechanisms for moving that information *en masse*. The similarity to computers is tantalizing. If we can affect the inputs to these systems, comprehend the computations performed on those inputs and ingest the outputs, then we can deploy these evolved, special-purpose computers on problems of interest to humans. This is not biomimicry, as the objective is not to replicate nature. Neither is it bio- or geo-engineering, since these natural systems will not be altered. Instead, the aim is to make nature a *subsystem* in a high-performance "natural computer." Success in this endeavor will increase global compute capacity by unlocking the latent computational power available in nature. It may also help solve the environmental crisis. Tapping these natural computational processes with even a fraction of the efficiency with which we've tapped natural power-generating processes will radically affect technology's relationship with nature. If machines that exploit or abuse nature got us into this crisis, then machines for which healthy natural systems are a critical component may help us out of it.

Natural computation

Development for "natural computers" will proceed similarly to the development of conventional computers. Like the original ENIAC and its predecessors, the first natural computers will be special-purpose. High-performance computing often makes use of application-specific hardware which interfaces with a general-purpose computer and accelerates a particularly expensive algorithm or operation. These are "hardware accelerators." I propose to build *naturally* accelerated computers, which are similar to hardware-accelerated computers except that they interface with a natural system rather than application-specific hardware. The algorithms and operations best suited for natural acceleration are those which are difficult for computers and easy for nature. Random number generation and optimization problems are two such examples.

Natural memory

Nature includes many high-entropy environments with degrees of freedom that are effectively invisible to the organisms inhabiting that environment. These invisible degrees of freedom can be used for information storage without adversely affecting any life or natural processes. As an illustrative example, consider one cubic-meter section of beach. One could fully specify the state of that natural system in one very large vector. Limiting ourselves to non-quantum states, this vector would include 6 degrees of freedom for *each* grain of sand to specify its position and orientation. If a grain of sand is approximately 0.1mm across, we'd expect to find approximately 1000 billion in one cubic meter of beach. That's 6000 billion degrees of freedom. Life is a matrix which acts on that vector, and it's a matrix with a tremendously large null space. We are free to store data (a lot of data) in that null space without having any effect on local life. It is worth noting that *every* degree of freedom is available for information storage in environments that contain no life, like the Moon and asteroids. Excitingly, some of these repositories for data *move*.

Natural data transfer

Nature excels at moving tremendous quantities of matter over tremendous distances. Examples include animal migrations, ocean currents, water and carbon cycles, and the movement of celestial objects. Humanity has a long history of using this moving matter to generate energy, but has underutilized these processes for data transfer. There is some precedent for doing so in the form of homing pigeons and sailing vessels (among other examples), but we've yet to approach the channel capacity for these systems. There exist both near-term opportunities and long-term possibilities for injecting information into these epic movements of matter.

In the near-term, we can add matter in the form of conventional data storage devices to these systems, such that they are swept along from origin to destination. This method for data transfer makes the communication channel "bursty" in the sense that the information arrives all at once rather than bit by bit, but it enables shockingly high average data transfer rate. It's worth pondering, for instance, that the data rate for a single humble homing pigeon carrying 1TB SD cards from New York City to Boston is approximately 3 GB/s. And there exist natural migrations of matter with far more capacity for excess mass than pigeons (e.g. ocean currents, trade winds, and whales). In the long term, it may be possible to embed the information in the matter itself, by looking for and modifying the non-volatile null space degrees of freedom discussed in the previous section.

Getting there from here

Developing natural computers will occur in phases, the first of which resurrects the naturalist of previous centuries. Like those naturalists, the natural computing researcher will go into the wilderness to look for new things. Rather than searching for new plants and animals, this person seeks natural computational processes, natural repositories for data, and natural migrations of matter. These processes will be studied and modeled to gain an understanding of their algorithmic qualities, and then the researcher will design and build devices which make these evolved systems components of a larger machine. The person conducting this research requires a depth of knowledge in computing and hardware accelerators, a breadth of knowledge that includes mechanical engineering and the natural sciences, and practical experience designing, building, and debugging devices that interface with the natural world. My background makes me well qualified to make foundational contributions to natural computing.

Why am I the right person to do this?

I have spent my career straddling the boundary between the natural world and the computational world, and making excursions into each. Every excursion into one has led me back to the other. The first of these occurred during my PhD with Mason Peck, while I was designing, building, and deploying gram-scale spacecraft and the infrastructure for communicating with them. These tiny spacecraft (affectionately dubbed "Monarchs") forego conventional risk mitigation strategies for radiation exposure, extreme temperatures, and nearly all other threats to the healthy operation of a spacecraft in space or on the surface of a celestial body. Instead, they are designed to be capable of mass production and deployment on orbit. This means that standard mission assurance strategies for computing probability of mission success do not apply to the Monarchs. I learned how best to plan missions involving Monarchs by studying sea turtles, rabbits, and other "R-selected" species on Earth. These creatures deploy a very similar "quantity over quality" approach to their own much-higher-stakes mission of perpetuating their species from one generation to the next. The calculus of sea turtle reproduction maps remarkably well to that of mission assurance with gram-scale spacecraft. I recently became one of the first external partner payloads for SpinLaunch, a space launch company that aims to replace the first stage of orbital rockets with a centrifuge. We shot two of these Monarchs toward space at Mach 1.5, and confirmed that they survive the stresses of a centrifugal suborbital launch. Hopefully, millions will follow.

And I've had many excursions since. I created a distributed sensing system for measuring environmental conditions in vineyards and orchards in collaboration with plant scientist Justine Vanden Heuvel, which has led since to an investigation of the stochastic processes which occur in these locations and the extent to which they can be integrated into such systems. If we can recruit the birds, bees, and wind to enforce our low-power transmitter's sleep/wake cycles, we may be able to decrease the device's average power consumption *and* make nature a subsystem in our machine. I built a similar system for gathering data from dairy cattle with veterinarian Francisco Yeal Lepes, which has led to similar lines of inquiry. I've also collaborated with Andrew Weislogel, a curator at the Johnson Museum of Art, and Frederic Gleach, curator of the Cornell Anthropology Collection, on projects at the intersection of art, engineering, and history. I collaborated with Amit Lal, an ultrasonic MEMS specialist that built a camera which can image some of nature's smallest creatures in realtime. I constructed the embedded systems infrastructure for controlling the camera and extracting the data that it generates. Most recently, I've been collaborating with David Erickson of mechanical engineering and Mario Herrero Acosta of animal science on a Bezos Earth Fund project to engineer a system for fenceless cattle management. This has been a wonderful opportunity to learn about the algorithmic properties of herding behavior.

In fact, the interplay between the natural and computational worlds has become my obsession, and I'm quite certain that it will remain so for the rest of my days. In both my microcontroller and FPGA design courses, we explore the reciprocal relationship between nature and computing. The students build cricket synthesizers (and synchronizers!), birdsong synthesizers, flocking animators, drum simulations, and Mandelbrot Set generators. My Masters of Engineering students are building systems that generate random numbers with cosmic rays, use waterways for data movement, and store data in tree rings and ice cores. The more of these natural computational processes, repositories for data, and mechanisms for data transfer that I discover, the more filled with wonder I become.

Conclusion

Engineering is my preferred mechanism for learning about the natural world. In addition to reading books and articles, watching documentaries, and listening to podcasts about nature, I enjoy building devices that allow for exploration of its algorithmic qualities. This practice has left me with an information-based outlook on the natural world. In the forests, oceans, and mountains I see data being stored, computed upon, and moved. I see the pieces of new kinds of computers and machines with healthy natural systems as critical components. I look forward to a career devoted to assembling these computers piece by piece. It will not be easy, but it is worth doing. Natural computing has the potential to increase global compute capacity and to generate a symbiotic relationship between technology and nature.