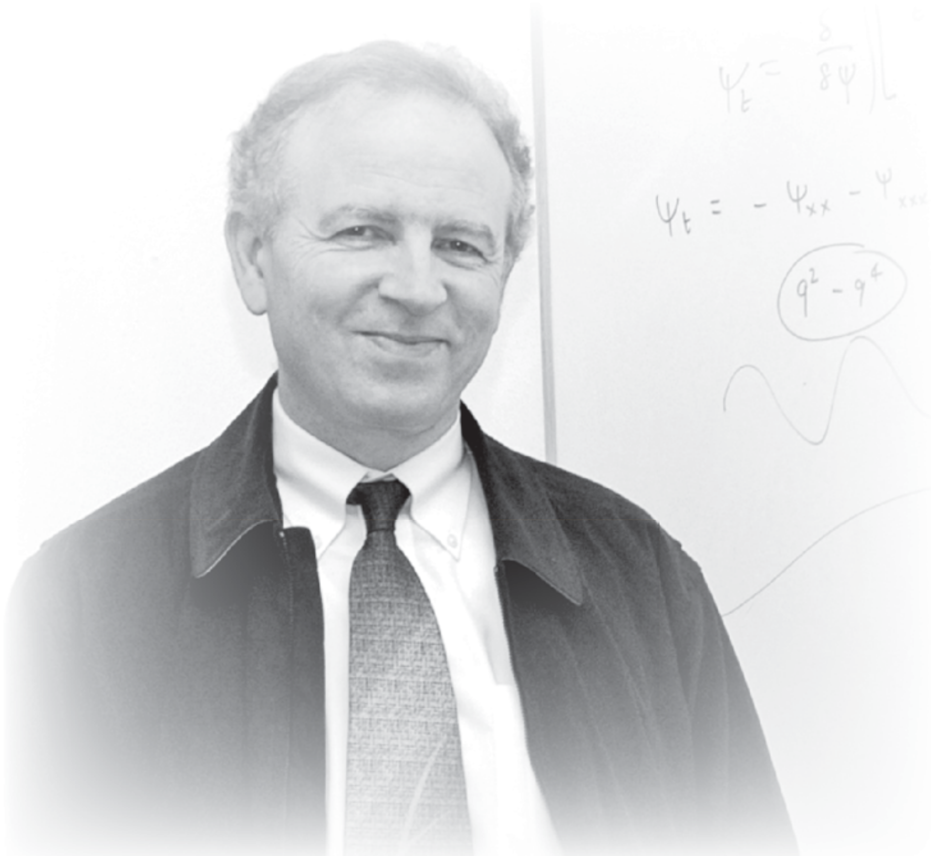

Eitan

Zen of Computational Attraction

Eitan Tadmor has made fundamental contributions to numerical analysis, the general theory of applied partial differential equations and scientific computation. His influence on applied mathematics is as deep as it is wide-ranging and as mathematical as it is organizational. His prolific research output, both personal and collaborative, must surely rank him as one of the top leading figures in his field. His direct influence may be glimpsed from an article written on the occasion of his 50th birthday and published in *Computational Methods in Applied Mathematics,* Vol. 4, No. 3 in 2004.

Tadmor's contributions span a spectrum varying from research to administration and institutional organization. He has given plenary lectures at many major scientific meetings, including an invited address at the International Congress of Mathematicians in Beijing in 2002. He was a founding co-director of the National Science Foundation (NSF) Institute for Pure and Applied Mathematics (IPAM) at the University of California, Los Angeles and is currently the Director of the Center for Scientific Computation and Mathematical Modeling (CSCAMM) of the University of Maryland, College Park. In addition to his being a professor concurrently in the Department of Mathematics, CSCAMM and the Institute for Physical Science and Technology at Maryland, he also holds the university's title of Distinguished University Professor. The list of his professional services, whether on worldwide scientific committees or editorial boards of numerous leading journals in applied mathematics is vast, and clearly shows breadth and personal commitment rarely found in a single scholar.

When Tadmor visited the Institute as an invited speaker of the program on nanoscale material interfaces, he was interviewed on 11 January 2005. The following is an edited and enhanced transcript of the interview, in which he reveals his unusually early, if not precocious, fascination with mathematical analysis, both pure and numerical, and a Zen-like attraction to things computational and algorithmic. It also gives an insight into his consuming passion for research and total commitment to the scientific community.

Imprints: Could you tell us something about your graduate training and how you became attracted to your present research interests?

Eitan Tadmor: I was attracted to mathematics during the early years of my elementary school in Tel-Aviv. At age 13, I joined a local math club which was run by a Tel-Aviv University professor, Gideon Zwas. He had a lively personality and a great ability to make very appealing presentations of mathematical ideas. I attended Zwas' π-club throughout my early years of high school. It was there that I first became interested in applied and numerical analysis. Zwas became my first mentor as I had begun taking courses at the university while still in high school. Back then, in 1970, it was the first administrative arrangement of its kind. Later on, I was pleased to see how it paved the way for more established channels of bright students who wish to study an academic curriculum during their high school years.

In 1973, I continued with my graduate studies in applied mathematics at Tel-Aviv University. Tel-Aviv had an outstanding group of numerical analysts. Those were formative years for me, with teachers and students who later became colleagues and who helped shape my interest in applied and numerical analysis. I felt very comfortable with analysis and was caught up by the interplay between the analytical and computational aspects of numerical algorithms.

Later, I continued with my postdoctorate studies at CalTech which was home to one of the top-rated groups in applied and computational mathematics. It was then that I met Professor Heinz-Otto Kreiss, who later became my mentor. I was greatly influenced by his work. Fifteen years later, we ended up as colleagues at UCLA and have remained close friends over the years.

I: Is the π-club still in existence?

T: No, unfortunately. It lived through the '70s and it was very successful in attracting many bright young mathematicians, led first by Professor Zwas and later by Professor Moshe Goldberg. The topics covered in those weekly meetings were a cross section of analysis and computation. I know that the memory of the π-club remains very vivid in the minds of those who attended it; I know it is in mine.

I: You were attracted to numerical analysis right from the beginning?

T: Yes. My interest grew out of my education. There were the analytical tools that one learns about at the early stages. I mentioned my early years at the π-club and at Tel-Aviv University. At the same time, there were the numerical algorithms that one could implement and one would like to know if they work; more important, why they do not work. The analysis part of the numerical analysis plays an important role in clarifying these "if" and "why" parts. Often, these questions cannot be addressed within the numerical universe *per se*: they cannot be divorced from the underlying mathematical model they are trying to simulate. I have always liked the interface between mathematics and numerics. I still do. Back then, mathematics was used to design more efficient numerical algorithms. Today, there is a feedback loop, when numerical algorithms impact the kind of mathematical questions being asked.

I: Do you actually use the computer to create the patterns and algorithms?

T: Well, the computer cannot replace the creative process. So, the answer is "no". If you would like to create or analyze a numerical algorithm you are on your own. But this is only part of the answer. The computer is the test bed which enables your ideas to be tested. It is the new experimental laboratory, so you are not alone but you go back and forth. At the same time, the answer is "yes"; the computer, or more precisely its output, does produce patterns. These could be just spurious numerical artifacts; but they could also reveal new phenomenon that was not observed before and drives a new mathematical inquiry. It is a partnership. Numerical algorithms are an asset of this partnership.

I: Do you actually devise the algorithms?

T: Yes. I devised, for example, algorithms for solving certain nonlinear partial differential equations. Other algorithms were constructed which enable me to "manipulate" various representation of discrete data. You often read that the computer "solves" a problem. The computer does not solve anything. It implements different algorithms to solve different problems. In my case, I am interested in developing and analyzing numerical algorithms which produce accurate solutions for differential equations, or process voice and image data.

I: Am I right to say that you are not interested in the algorithms *per se* but that you are interested in algorithms to solve partial differential equations and so on?

T: You are right. I am more interested in the mathematical aspects of such algorithms. Having said that — numerical algorithms are not just a thought experiment; computers enable us to actually implement the algorithms we devise. This is the partnership I was talking about and the various aspects of implementation, therefore, become an integral part of the overall computational task.

I: Computer scientists study algorithms too. Are you interested in computer science?

T: The various aspects of implementation I was talking about are traditionally a main focus of computer science. But computer science can be viewed as the totality of what computer scientists are interested in, and there are many who study the algorithmic aspects combined with the mathematical aspects. There is no clear borderline. It is more a question of different emphasis. My emphasis is more on the mathematical aspects and less with the algorithm *per se*.

I: Do you consider yourself to be more of an applied mathematician than a pure mathematician?

T: I am an applied mathematician. But considering my work on the theoretical aspects of partial differential equations and their counterpart in numerical algorithms, some applied mathematicians would classify me on the pure side. It is difficult to decide where pure mathematics ends and where applied mathematics begins. The branches of mathematics I am involved with are primarily analysis and scientific computation, which, in the great vague divide of classifying pure and applied math, is categorized on the applied side.

I: Do you use the algorithms to actually solve the differential equations?

T: I develop algorithms for accurate solution of differential equations and I test them on a host of model problems. These then become tools that are applied to solve a host of problems in various fields. In some cases, the problems and the methodology for solving them could be very specific. For other cases, we developed a family of "black box" solvers which are portable enough to solve differential equations from a great variety of different applications.

I: What about the Navier–Stokes equations?

T: The set of Navier–Stokes equations governs the dynamics of flows at the human scale. That is, everything from air flow around airplane wings to the water flow in your bath tub. One might be surprised, maybe even worried, that just one simple set of equations is sufficient (or supposed) to describe so many different phenomena. Well, the Navier–Stokes equations are essentially one set of equations, worked out by the giants of the past. But they are not as simple as they appear to be. It is not totally surprising, therefore, that our mathematical understanding of these equations is incomplete. Indeed, there is a $1 million Clay [Institute] Prize for successfully clarifying part of the puzzle surrounding the mathematical quantities governed by these equations: what properties do they have? But even without the full understanding of their mathematical properties, we are developing numerical algorithms for the approximate solution of these equations. Practitioners compute the numerical solutions without necessarily waiting for their full mathematical understanding. At the same time, it excites a lot of research, a lot of ingenuity, and a lot of numerical experiments which try to complement each other in gaining insight into the mathematical properties of the Navier–Stokes solutions. A large component of the weather system, for example, is also governed by the Navier–Stokes equations. Here, interactions occur across several scales which are still human scales, say, larger than atomic scales and smaller than cosmological scales. The enormous complexity cannot be contained between the purely analytical walls, but it requires modeling and numerical simulations.

I: Can we say that in some sense, numerical analysis depends on the ability to design more and more powerful computers?

T: This depends on what your meaning of the word "depends" is (excuse the cliché). The mathematical models we are trying to simulate are independent of computers and to a large extent, so are the numerical algorithms which perform these simulations. The numerical analysis of such algorithms is intimately connected with the properties of the underlying model. In this sense, numerical analysis is independent of the computer running these algorithms.

At the same time, more powerful computers alter the kind of questions we may ask about our numerical algorithms, and lead to different notions of what optimal algorithms are. If in the past, it took 48–72 hours to simulate tomorrow's weather, then naturally, numerical analysis turned its focus on developing much faster algorithms. Over the years, the speed-up in computer power accelerated by Moore's law, doubling itself every 18 months. Everyone is familiar with it from his or her PC, but this doubling factor also applies to the new and improved numerical algorithms that were developed over the years. So nowadays, when it is feasible to compute tomorrow's weather in less than 24 hours, the demand arises to include more realistic models, or to develop new algorithms to include much better visualization. In this sense, numerical analysis depends on more powerful computers. Moreover, if computers become powerful enough, they can run different algorithms that communicate across different scales, and thus, instead of using mathematics to model the ensemble of small scales we can think of numerical algorithms performing the modeling "on the fly". Clearly, this requires the development of new numerical algorithms which are by-products of more powerful computers. A more powerful computer will enable us to reach new territories that have not been reached before. Still, you need the creative process of developing and analyzing new numerics to conquer these territories, and this is independent of how powerful the computers are.

I: Have the powerful computers raised any new issues and brought about new developments in numerical analysis?

T: Absolutely. The canonical example is parallel computers. Parallel computers completely changed the scope of what algorithms can do. What might be impractical or even impossible to do with computers based on a single processor becomes possible to do in parallel processing. Quantum computing could be the next and perhaps ultimate leap.

I: So the limit exists?

T: I think it does. One might say that if we just have powerful enough computers, many times more powerful than what we are having today, then we will be able to solve *everything*, to simulate a complete dynamical ensemble, perhaps even from quantum scales all the way up to the human scales. I do not know about the technological barriers here. But I will argue that even if the technological difficulties will be resolved, say in the next 100 years, still the smaller it gets, the slower the clock gets. That is, one needs to sacrifice a certain level of detailed information for having a computational algorithm to run at a finite time, that is, finite in human scales. And as powerful as computers can get by miniaturization in space, they will slow down in time. In this sense, limits exist and there is room for developing numerical algorithms for mathematical models which will bridge this gap of space and time.

I: Could you give us some examples of successful numerical algorithms?

T: The Gauss elimination method for solving N linear equations with N unknowns is perhaps the most ubiquitous numerical algorithm of all. It was always out there. But the Fast Fourier Transform was not and its discovery in the mid 1960s has had a lasting impact. It computes the periodic building blocks of general waves based on N samples and it reduces the computational work by order N. This is Huge. If N has the reasonable size, say, of ten thousands, then this is equivalent, by Moore's law, to 10 years of hardware speedup. These are exact algorithms. Their success is based on clever rearrangement of the computed quantities to achieve the final result in a fraction of the time it requires for a straightforward computation. In other cases, only approximate solutions are sought. This is the case with the solution for a host of partial differential equations drawn from various branches of science; the equations are just too complicated to be solved exactly. Approximate solutions are satisfactory. Here, there is a trade-off between how accurate the computed solution is versus how fast it can be computed. Many modern numerical algorithms are successful in making this trade-off. These numerical algorithms are successful in being very efficient.

I: There is an old perception that a mathematician's job is done once the model is formulated and that the rest is the job of the mathematical technician. How much has the role of the mathematician in modeling changed over the years?

T: On the contrary, the job of a mathematician just starts when the model is formulated. The modeling I am referring to is not necessarily mathematical modeling. Before the genome, there was the double helix and it was more descriptive than quantitative. Before Kepler's laws, you had Copernicus and his concept was not as quantitative but has had much more impact than Kepler's. Mathematics seems to be the most successful language to translate our qualitative concepts about the physical world around us into a set of quantitative statements. Today, more than ever before, there is a large effort to duplicate this success with quantitative biology. Still, the mathematical modeling is not left to mathematicians but to the interaction between scientists from different disciplines with mathematics. Once a model having its roots in biology, nano-science, chemistry or astronomy has been quantified, mathematicians study the interconnections, trying to fit the mathematical model as part of a greater puzzle. Often, modeling is a much more laborious and less glamorous task than the formulation of "$E=mc^2$". It involves experiments, measurements, statistical evidence and numerical experiments. More than before, pure and applied mathematicians are involved in all those aspects. This is particularly true with regard to the computational aspects. In the past, there was one critical reality check for a scientific theory, namely, that its predictions can be proved or disproved by experiment. Nowadays, computations provide another reality check for developing theories. This is the interplay I was talking about before, of numerical algorithms simulating mathematical models from different scientific disciplines. This is the intersection called "scientific computation".

I: Is there a coherent theory of scientific computation or is scientific computation nothing more than a collection of *ad hoc* methods and clever techniques?

T: Yes, there is a coherent theory of scientific computation. There are the fundamental concepts, technical tools, hierarchy of knowledge. But like every other area in mathematics, there are many isolated islands. Scientific computation, more than most of the other areas in mathematics (statistics might be the exception), is at the forefront of interaction with the other sciences and therefore, it has a constant flood of new input from the "outside". It absorbs clever new tricks, ingenious mathematical techniques and algorithms we do not always understand why they work so well. But over time, some coherence emerges. Sometimes, there are *ad hoc* methods pulling the theory forward. Other times, the theory identifies the danger zones where the numerics will not work, or even worse, will work out the wrong solution. There is a healthy tension between the hierarchy of knowledge and the collection of *ad hoc* methods. Over time, they merge into a coherent theory of scientific computation. It is a relatively young area of 20th century mathematics and it is a very lively one.

I: Do you consider mathematical modeling a science or an art? Is there some quality beyond mathematical expertise that one need to possess in order to be successful in mathematical modeling?

T: Mathematical modeling is an art, expressing itself in a quantitative language. Well, there is no recipe of how to make a mathematical model. It requires creativity, curiosity, ingenuity, imagination in addition to understanding the science behind the model.

Let me mention the example of modeling images. Images are all around us and nowadays our world is going digital. Digital images are collections of many pixels. My digital camera has five mega pixels. And each pixel has its own, very local grayscale (or color scale). We do not see these individual pixels, but instead, we see their collection as an image. In the last decades, many models were developed in order to manipulate the collection of pixels as images so that we can transmit, compress and in general, manipulate digital images. There are many mathematical models but there is still no final word about the one way that we should interpret a collection of pixels as an image.

I: Is the model independent of what goes on in the brain? Are the images not affected by the processes in the brain?

T: This is the reason I mentioned this example of image processing. Modeling digital images lives outside the mathematical universe. There is the mechanical part of the human eye. There is the conceptual part of the brain which puts together an ensemble of small pixels and gives them sense of what we understand to be an image. Once again, mathematical modeling seeks to match the world we see around us on the human scale, to a world made up from basic elements on a much smaller scale.

I: What is the greatest satisfaction you have had in your research career?

T: There are the moments you understand the answer to a mathematical question that bothered you for a long time. You know when you unlock the puzzle. These moments are very rewarding. It is a peak of a creative process. Another rewarding aspect is the development of numerical algorithms. It is rewarding to see a numerical algorithm that you have thought about, realized on the computer. And there is a great satisfaction in learning new ideas that are born into mathematics. There is a constant process of renewal, a generation of new ideas. There is still part of me which remains as excited about mathematics as I was during my days in the π-club. This is very satisfying. I feel blessed.

I: I believe that our Institute (IMS) is modeled in part on your Center for Scientific Computation and Mathematical Modeling. While IMS spreads its programs over diverse fields, your Center is extremely focused. Could you tell us something about your Center?

T: The Director of IMS, Louis Chen visited us in 2001 when I was the Director of the NSF Institute of Pure and Applied Mathematics (IPAM) at UCLA. This visit took place just before the IMS was launched. Certain aspects of the IMS, like its scope, covering a wide spectrum of mathematical areas, are modeled after the national NSF institutes such as IPAM, MSRI and IMA. In this sense, the IMS serves the purpose of appealing to a wider spectrum of the pure to the applied crowds. In 2002, I was recruited by the University of Maryland to serve as a Director of its Center for Scientific Computation and Mathematical Modeling (CSCAMM). Our Center is not a national center. It is a major initiative which is completely funded by university, devoted primarily to scientific computation and mathematical modeling and their interaction with other scientific disciplines. I elaborated before on my views as to the role of mathematics in this critical junction. A main part of our visitors' program centers around CSCAMM workshops. We organize several workshops each year and they have already achieved a considerable success in increasing the visibility of the outstanding faculty and activities in Maryland. Our center is an independent arm of the College of Physical Sciences in Maryland, with the mission of increasing the interaction between the different units through their common interface of scientific computation. Thus, the flavor of CSCAMM is somewhat different from the national institutes and it has a different strategic direction, as it is trying to lower the barriers between faculty inside and outside the university through the more focused platform of scientific computation and mathematical modeling.