On the Merits of Theory of Computing

Oded Goldreich* Avi Wigderson†

May 1996

Abstract

In this essay we provide an assessment of TOC as a fundamental scientific discipline. We focus on the important scientific role of TOC, and on its great achievements, productivity and impact (both scientific and technological) so far.

*Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, ISRAEL. E-mail: oded@wisdom.weizmann.ac.il

†Institute for Computer Science, Hebrew University, Givat Ram, Jerusalem, ISRAEL. E-mail: avi@cs.huji.ac.il

1 Introduction

The material presented here was extracted from an essay we have written in a rush so that it is ready by *STOC96*. Despite our impression that the views expressed in this essay enjoy great support in the community, we want to make it clear that we alone take full responsibility for its contents. The time pressure, as well as our personal biases and limitations, make this essay far from perfect, even in our eyes.

Organization: We start by explicitly stating our beliefs regarding science and the evaluation of scientific disciplines. These beliefs are far from being original. They are rooted in the best philosophical and scientific traditions. (For lack of time and energy, we do not provide a host of references.) Once these are stated, we proceed to the main part of this text, where we discuss the fundamental nature of the Theory of Computation (TOC) and its success so far. We then turn to discuss the impact of TOC on technology and on other sciences.

A personal comment: We consider ourselves very fortunate to have taken our first steps in the Theory of Computing in an enlightened and exciting atmosphere, very much different from the current mood. We consider this essay as a minor payment towards our duty to try and provide a similar atmosphere for the new generations of TOC students.

2 Culture, Science and Technology

The search for truth and beauty is the essence of civilization. Since the Renaissance, the search for truth takes the form of (or is called) Science. Technology is an important by-product of the scientific progress, not its *raison d'etre*. Furthermore, philosophical reasoning as well as experience show that technology is best served by a free scientific process; that is, a scientific process which evolves according to its own intrinsic logic and is not harnessed to the immediate technological needs. Such free scientific process evolves by formulating and addressing intermediate goals which are aimed at narrowing the gap between the ultimate goals of the discipline and the understanding achieved so far.

It is ironic that as the contribution of science to technology becomes wide-spread, a popular demand arises to have more. Namely, the success of science and in particular the benefits of its technological by-products causes the populace to turn against science (in the form of demands that science deliver even more consumable commodities). Still, one has to oppose these demands. Science is to maintain its *autonomy* which is correlated to its success. In the long run, this is also the best way to serve technology.

Technology evolves mostly via applied scientists and engineers who use the scientific knowledge they have acquired and their own creative forces to the development of specific applications. Contrary to popular beliefs, the most important contributions of science to technology do not stem from the harnessing of scientists to engineering tasks, but rather from the fact that scientists instruct and enrich the thinking of these engineers. The education of engineers does not reduce to the acquisition of information. Its more important features are the development of conceptualization and problem-solving abilities. The conceptual frameworks of the discipline are offered to the student and the better these frameworks are the better an engineer he/she may become. This form of education is most effective when done by good scientists who enjoy the freedom to pursue their own research interests.

It is important to note that the nature of the process by which science effects technology makes it very hard for the laymen, and sometimes even the expert, to trace a technological breakthrough to its scientific origins. Almost always these breakthroughs depend on the conceptual scientific framework and very often they utilize specific discoveries which were considered totally impractical at the time of discovery (e.g., complex numbers and electricity).

3 Evaluating (the importance and success of) scientific disciplines

The scientific disciplines are defined by the questions they address. The *importance of a discipline* is determined by the nature of its formative questions. The more fundamental these questions are the more important the discipline is. Educated laymen and certainly scientists can usually assess how fundamental major scientific questions are.

The *success of a discipline* is measured by the progress it achieves on its own formative questions. To measure the amount of progress one has to understand the questions and the state of knowledge of the discipline with respect to these questions. This usually requires the understanding of experts, but can be conveyed to scientists of other disciplines.

Neither the importance nor the success of a scientific discipline can be measured by the impact of its current discoveries on technology (or on other disciplines). If the discipline is indeed important and successful such impacts are likely to follow. However, rarely will this impact be linearly related to the scientific progress in the discipline.

Individual scientific disciplines do not exist in a vacuum. The healthy evolution of a scientific discipline is sensitive to *scientifically relevant* inputs from other disciplines as well as technological developments. We wish to stress that the influence of these inputs is determined by the disciplines internal logic and inherent goals and that such influences are vastly different from non-inherent suggestions (e.g., that in order to increase funding and/or employment opportunities the discipline should pursue alternative directions).

4 On the fundamental nature of TOC and its success so far

The Nature of Efficient Computation and its natural as well as surprising derivatives, is the formative question of the Theory of Computing (TOC). We consider this question to be one of the most fundamental scientific questions ever asked. Unfortunately, the fundamental status of this question is usually disregarded due to its immediate technological impact.

We feel that both the fundamental nature of the questions of the Theory of Computing and the success of our community in engaging these questions (up to this very day) are evident. To be on the safe side, here is some evidence.

An excellent demonstration of the the fundamental nature of TOC was provided by Papadimitriou [1] in his survey on the impact of NP-completeness on other sciences. Papadimitriou lists about 20 diverse scientific disciplines which were unsuccessfully struggling with some of their internal questions and came to recognize their intrinsic complexity when realizing that these questions are, in some form, NP-complete. According to his bibliographic search, NP-completeness is mentioned as a keyword in about 6,000 scientific articles per year. How many scientific notions have had such impact?

More generally, TOC has established a direct relationship between structural and computational complexity. Efficient algorithms are discovered almost only if tangible mathematical structure exists. This connection has already benefited mathematical progress in many areas such as Number

Theory, Algebra, Group Theory and Combinatorics, where on one hand a need for efficient algorithms existed, and on the other hand the search for them has generated structural results of independent interest.

Actually, we tend to forget the revolution in problem-solving introduced by the TOC treatment of algorithms. This revolution consists of the explicit introduction of the concept of an algorithm and the measures for its efficiency, the emphasis on data representation and organization, the general techniques for creating algorithms for classes of problems, and the notion of reductions between problems. Needless to mention the impact of all these on computer practice, but we wish to stress the impact on any kind of problem solving.

The TOC has drastically changed the perception of knowledge and information. Specifically, the TOC stresses that different representations of the same information may not be *effectively* equivalent; that is, it may be infeasible to move from one representation to the other (although a transformation does exist). In this new world, publicly available information may be unintelegible. All of Modern Cryptography is based on this Archimedes' point, and its scientific and technological impact are well known. Here we wish to suggest that this revolution applies not only to computer systems but to any aspect of human interaction in which privacy and fault-tolerance are important concerns.

The TOC has introduced totally novel ways of understanding and using randomness. The probabilistic algorithms developed within the TOC use randomness in many varied sophisticated ways. The applicability of randomized procedures for solving tasks from different domains such as number theory, optimization and distributed computing is amazing. Moreover, the growing study of derandomization has lead to derivation of better deterministic algorithms from probabilistic ones.

Combining randomness and interaction lead TOC to create and successfully investigate fascinating concepts such as interactive proofs, zero-knowledge proofs and Probabilistically Checkable Proofs (PCP). Each of these concepts introduces a deep and fruitful revolution in the understanding of the notion of a proof, one of the most fundamental notions of civilization. Furthermore, these revolutions bore fruits which reached far beyond the realm of proof systems. For example, work on PCP lead to the first breakthrough in the understanding of the hardness of approximation. This is but one incredible demonstration of the how probabilistic thinking leads (very indirectly and non-trivially) to fundamental understanding of totally non-random phenomena.

In addition, combining randomness and complexity, TOC has generated meaningful notions of pseudorandomness. Computational hardness yields pseudorandom generators: using "one-way" functions, randomness can be "stretched" in an almost unlimited way as far as efficient observations are concerned. This yields the stunning (to most scientists) conclusion that if their Monte-Carlo algorithm (estimating perhaps a numerical integral or simulating a physical process) behaved differently on sequences produced by such generator, than on genuine random sequences, then they have discovered an efficient factoring algorithm! Totally different pseudorandom generators which TOC discovered can fool any space limited algorithm. Since all standard statistical tests have such implementations, this is great news to Statisticians, Physicists, and most Social Scientists who use such tests on everyday basis. Namely, the results of all their experiments are guaranteed to hold even if they replace all their random choices by pseudorandom choices produced by from tiny random seed.

TOC has gained considerable understanding of organizing work on huge systems of many components. The study of parallel algorithms resulted in amazing ways to get around "inherently sequential" tasks. Subdividing work to smaller chunks in efficient and balanced ways is taking place not only in computer systems but in many organizations, and the insights gained by TOC are avail to them too. A different kind of parallel computing arises in settings where the information is distributed among the components of the system. TOC studies of such distributed environments resulting in models and methods of consistency, recovery, knowledge, synchrony and decision making, are relevant not only to (distributed) computer systems but also to economics and other social sciences.

The organization and availability of information was always a major part of civilization, and in particular science and technology depend on it. The models and solutions developed by TOC for such problems not only resulted in computer systems that would do it for people, but in the very way people and institutions have to think about information. The amazing new abilities to handle huge masses of data increase, rather than decrease, the human decisions on what they want to be stored, what access patterns they want to allow and disallow, what should be retrieved quickly and what can take longer, etc. The theoretical understanding enables to formalize their demands, and enable programmers (who should understand the algorithms and data structures as well) either to satisfy these demands or to explain why they are impossible to achieve.

Likewise, some of TOC's insights to performance analysis, the minimizing and balancing of several resources, are of universal applicability. One example is the notion (and techniques) of competitive analysis, whose applications range from operating systems to information compression (Lempel-Ziv) to emergency services to stock-market investments. More generally, asymptotic analysis has taught us that structure is often revealed at the limit. The adversarial point of view developed for worst case analysis (both of inputs to algorithms and behavior of distributed systems) has taught us a similar lesson: structure is often revealed under the worse circumstances and may be obscured by unjustified assumptions on "typical behavior". Such structure often leads to better (in every respect) theoretical and practical solutions.

Finally, let us mention that that many inter-disciplinary scientific activities involve and further seek the participation of TOC members. These include the different "neurocomputational" groups (encompassing brain models, learning, and neural networks, involving physicists, biologists, psychologists) and "rational behavior" groups (encompassing economy, ecology, evolution, competition, and decision making, involving economists, statisticians, psychologists and mathematicians). They want TOC to be there since they have recognized, in contrast to some members of the TOC community, the universal value of the problems TOC deals with and the understanding TOC has obtained so far, and in particular their relevance to these areas.

Clearly, lack of space, time and knowledge prevents us from going on. Still, the massive list above illustrates the fundamental nature of our endevours from the scientific point of view. But they are fundamental also from two other important viewpoints. One is the philosophical viewpoint, which has dealt with many of the notions and questions above for centuries, and which receives a fresh, radically different perspective (namely the computational one) from TOC. As an example consider the question of P vs. NP vs. CoNP. Some tend to think of it is a mere technical question and miss its deep philosophical significance: Understanding the relation between the difficulty of solving a problem to the difficulty of verifying the correctness of the solution, to the difficulty of proving that no solution exists. Additional examples are the TOC perceptions of the notion of a proof, its view of randomness, and its emphasis on the importance of specific representations. The second viewpoint is the potential contribution of TOC to the general education and enrichment of humanity. Many notions, problems and even some of the solutions TOC has produced are available for understanding (in nontrivial levels) by laymen. We have successfully tried to explain some of them to elementary school kids (and indeed we foresee some of them taught and used as teaching paradigms in grade and high school). Few sciences (which existed for many centuries) can compete on these grounds with what TOC achieved in a few decades.

To summarize, this subsection illustrated the fundamental importance of TOC as well as its success. As for the latter point, let us stress that the achievements sketched above are more or less equally spread over the last 30 years, and many are very recent. Indeed, the rate of progress done by TOC in these years is astonishing and there is no inherent reason for this progress to stop. It is thus essential to oppose the external pressures and internal moods which endanger the continuation of the fundamental and successful research in TOC.

5 On the impact of TOC on Technology

While we rejected technological impact as a measure of importance and progress of a scientific discipline, the enormous impact of TOC research on technology should not be made a secret. We are far from experts regarding this impact, still there are a few points that even we can tell. We hope and believe that a much better treatment will be given in the future by more qualified colleagues.

The most important impact of TOC on Computer Science and Technology stems from the fundamental goals of TOC. In its endevour to understand the nature of computation, TOC created general abilities to conceptualize, model, unify, solve and analyze computational mediums and problems. The effects of this understanding are present in essentially every working system and algorithm on earth. Without them the computer revolution, which has changed life on this planet in a fundamental way and will continue to effect it at increasing speed, would simply not be possible! Indeed, they are the very reason that theory courses are mandatory for all undergraduates in computer science departments. They are the reason that most applied computer science courses are not a mere collection of ad-hoc tricks and are thus suitable to be taught in universities. They are the reason that the originators of technological breakthroughs, as well as all engineers and programmers, can actually think, talk, present and evaluate their ideas. Some critics may say that these understandings were achieved long ago, and there is no need for further "refinements". This is contradicted by many technological advances which have resulted (and will continue to result) from *recent* developments of such understandings regarding, for example, parallel, distributed, interactive, secure and fault-tolerant computation. Many such developments were achieved by special interest groups within TOC, who took on to study in depth such models and algorithms. Their specialized conferences, which are a relatively recent phenomena, often enjoy the active participation of more applied scientists, who have both easy access to this knowledge as well as a forum to influence its direction.

It is crucial to recognize and communicate the fact that most of this understanding resulted not from attempts of solve a concrete problem under particular technological constraints. Rather, it came from generalizing the problems and abstracting away unnecessary technological details to the point that enables finding structures and connections to other knowledge. Only then could applied scientists and engineers, who had both the theoretical understanding as well as the mastership of the specifics of the technological task, fuse them together to a successful practical object. The value of this approach has many examples, and we discuss only one.

- **NC and the PRAM model**. As an example, we choose on purpose the class NC and the PRAM model, a common bashing target of "practical people" (as well as of the TOC self-destructive fashions). While the direct applicability of this model (and algorithms for it) may still be controversial, several parallel systems builders we have talked to have totally changed their attitude towards it. Technological changes have made it much closer to reality then realized 15 years ago, which teaches us a moral regarding fine tuning of theoretical models to current technology in a field in which the latter is changing at such rapid speed. But this retrospective fact is not the source of the value of PRAM. Its value stems from the fact that it is a good framework for developing an understanding of the paradigm of parallel computing. Specifically, the answer to the bashers should have always been that a very fast practical parallel algorithm (for, say, sorting or matching), on a particular architecture (say Connection Machine), is almost necessarily also an NC algorithm on PRAM. If we cannot find the second, how can we develop the first. Moreover, while a PRAM algorithm may never be implemented as is, the algorithmic techniques, communication paradigms and data structures developed in its study, have strongly influenced many different practical systems.

In general, one should advocate the value of abstractions which address some fundamental aspects of an important problem (even if they seem not address all aspects), and warn against the short-sightedness captured by dismissing such abstractions as irrelevant. The study of such an abstraction is more likely to yield fundamental insights than the study of the "real problem" (assuming such a creature exists – actually there is never one real problem but rather many different related real problems and what these have in common may well be the dismissed abstraction). Only later will people, with a concrete application and technology in mind, be able to fine-tune the theoretical understanding to their needs. (This in itself may require significant research and implementation, that was and is taking place by computer scientists and engineers, and which resulted in so many successful technological developments.)

It is equally important to recognize and communicate that it was the freedom and time given to TOC researchers to pursue these general directions, in real attempt to understand novel computational media, that resulted in such progress – quite often in surprising and unexpected ways.

One can illustrate the point above by numerous examples. We prefer to give two very recent examples whose technological and practical effects are imminent and yet to come. So far their "practicality" is demonstrated by a major leap in the algorithmic understanding of major problems. This leap is rooted in developments of complexity theory which, at first and for a long time, seemed totally irrelevant to the latter or any other algorithmic task. Such leaps are frequent in our field, and are due to the freedom of pursuing scientific intuition, as well as to the strong communication and information exchange between the various subareas of our field.

- **The Euclidean TSP Algorithm.** A few weeks ago Sanjeev Arora announced a polynomial time approximation scheme for the Traveling Salesman Problem (and a host of other combinatorial optimization problems) in the plane. The problem itself was a major object of study in our field for decades. The failed attempts to find such approximation scheme resulted in fundamental contributions to NP-completeness, probabilistic analysis, approximation algorithms and mathematical programming. It also resulted in enormous efforts to understand the relative power of various heuristics.

  The techniques present in the algorithm of Arora were available decades ago! Why was it only found now? While this is a source of speculations, Arora himself tells how he came about it. The algorithm arose from his attempts to generalize the inapproximability results of metric TSP to Euclidean TSP, attempts which revealed to him the extra structures of the Euclidean case. These attempts were based on the surprising connection of PCP proofs to hardness of approximation. In turn, these "mysterious" proofs arised from abstract results like MIP=NEXP (relating "clearly impractical" complexity classes). Moreover, the MIP model of multi-prover interactive proofs was suggested by Shafi Goldwasser as a generalization of interactive proofs (themselves the outcome of amazing developments). Needless to say that Goldwasser did not think of approximation algorithms when she suggested the new model.

- **Efficient Error Correction.** It was only a year ago that Dan Spielman discovered a *linear-rate* code which has asymptotically optimal (i.e., linear time) encoding and decoding algorithms. This central problem of communication, that originated with Shannon half a century ago, has attracted the best minds in Information Theory, Mathematics, Electrical Engineering and Computer Science, and has resulted in beautiful and important theory. Still, this major problem, resolved by Spielman, was beyond reach.

The construction of Spielman closely mimics the construction of a superconcentrator. This object was not available to most scientists working on this problem, and Spielman learned about it from Complexity Theory. The superconcentrator was invented in TOC, by Valiant, in his attempts at one of the quinticential impractical problems – proving circuit lower bounds. Failing to do that, Valiant turned to an even more impractical problem – to show that this particular attempts will necessarily fail! Here he was successful. He (noconstructively) exhibited the existence of expanders, and used them as building blocks of linear size superconcentrators. A deep and beautiful mathematical theory developed, motivated by the explicit and efficient construction of expanders, which effected diverse areas of TOC. More to the point of this subsection, indirectly and through much further work, derivatives of the study of expanders became extremely relevant to technological development concerning communication networks and protocols for a variety of parallel and distributed architectures.

It is our opinion that the amazing scientific consequences and the surprising practical implications which sprouted (and will continue to grow) from the totally abstract and impractical proposals of Goldwasser and Valiant in the examples above, are *alone* well worth the meager investment so far of the world in TOC. This may serve as a waring against dangerouds attitudes by which proposal of the above nature are likely to be rejected on the basis of "plac[ing] excessive weight on mathematical depth and elegance, and attach[ing] too little importance to genuine links between theory and concrete applications", especially if written by junior people.

6 On the impact of TOC on other sciences

In the short time of its existence, TOC has had an unprecedented effect on other sciences. This has taken at least three forms.

- **Algorithms.** Many sciences use heavy computation for their research, mainly for simulation and analysis. The advances in fundamental algorithms in TOC, on data structures and general techniques are essential for them to understand, so as to optimize their computational resources. The impact of these on the rate of progress in these sciences cannot be underestimated. Moreover, sometimes such disciplines generate a particular type of problems for which the general algorithmic knowledge does not suffice. In some cases where these problems raised sufficient scientific interest (perhaps luckily timed with internal developments), TOC was quick to pick up and study its natural computational structure. Two such superb examples are the great advances TOC has made in understanding and analyzing random walks, so often at the base of simulations in Physics, and its contributions to number theoretic and algebraic algorithms.

- **Natural Computational Models.** Nature computes! While this was observed long before computer science existed, TOC supplied the mechanisms to model, discuss and explain these phenomena. A recent challenge directed by TOC towards Physics is whether a Quantum Computer can be built? But even without the demonstration of the excessive power of the Quantum Computer model (e.g., Shor's polynomial-time Quantum algorithm for factoring), we speculate that complexity may be the right way of thinking about decoherence of a quantum mechanical system. The brain is another computational device whose understanding seems to be extremely far, but to which our unique contributions in neural networks and computational learning are providing important stimulation. Another natural source of (biological) computation, based on progress in molecular biology, was discovered by TOC and is studied with at least some interesting potential.

- **Universality of TOC notions.** As pointed out in Section 4, the unique computational point of view of TOC and its conceptual derivatives, has resulted in surprising impact on intrinsic studies of other disciplines. NP completeness, discovered over 20 years ago, has had a sweeping effect. But our view on other notions such as randomness, pseudorandomness, interaction and approximation is only beginning to take effect.

It should be reiterated that the discoveries above has made a fundamental impact on these sciences, and have lead them to reassess their points of view on some basic intrinsic questions and pursue novel research directions. We wish to stress that, having sound tradition and self esteem, these sciences were not (and could not have been) forced to pursue these novel directions by TOC or anyone else. Their choice was based on their scientific understanding of their intrinsic goals. Similarly, the interest of TOC in these problems arose from the understanding of TOC researchers that these problems are relevance to the goal of understanding computation. The amazing success of this impact and the high and growing regard to TOC in these sciences, again, stems from the intellectual freedom in which these interactions arose. Again, even a small fraction of these effects justified the investment so far in TOC.

7 On the future of TOC

We believe that the notion of efficient computation will further revolutionize the way people think about problems and in particular the way scientists think about basic problems in their disciplines. It is hard to imagine the effect that a *deep* understanding of efficient computation may have on the thoughts of people in the future. To have even more impact on the sciences, TOC has to get a better understanding of the nature of efficient computation, and the other sciences have to further discover the relevance of these notions and results. When this will happen, these sciences will seek insight to computation and if TOC will not commit suicide in the meanwhile it will be there to provide it.

References

[1] C.H. Papadimitriou, lecture in the workshop in honor of Karp's 60th Birthday, *FOCS95*.