

Title: Empirically Challenging NC Theory Maxims

Subtitle: Can Manycore CPUs Enhance Automated Reasoning?

A seven-orders-of-magnitude increase in transistor count since the advent of the personal computer is yet to yield effective parallelism for computing every day single tasks. A chicken-and-egg paralysis has been the main culprit: should the paradigm shift towards effective parallelism start with a developed computing platform or with developed meritorious applications?

Automated reasoning technologies (ART) are known for their insatiable appetite for computing. Established complexity theory hardships ranging from P-completeness (for Horn SAT) and NP-completeness (for SAT) to undecidability (for SMT) partially suggests that routines necessitated by these technologies are potential beneficiaries from parallel computing.

Having observed in prior work a good fit of ART with a manycore branch characterized by “fine-grained” and “irregular” parallelism begs asking: can manycores enhance ART similar to how GPUs enhanced machine learning? Specific results to be noted or covered include:

1. Demonstrated potential for significant speedups over state-of-the-art SAT solver, with J. Edwards.
2. Two surprising results for Horn-SAT whose P-Completeness is widely conjectured to prohibit poly-logarithmic time using a polynomial number of operations in the worst case:
 - 2.1 Demonstrated poly-logarithmic time using a linear number of operations. Furthermore,
 - 2.2 First-of-its kind empirical challenge for any worst case complexity theory, with A. Hari. Using the Moore, Istrate, Demopoulos and Vardi 2007 probabilistic Horn-SAT model, we demonstrate poly logarithmic time and linear work algorithms for all but an *infinitesimally small range*, compressing the non-poly logarithmic time worst case expected by NC Theory/P-completeness to a mere singularity.

Bio

Uzi Vishkin has long been a permanent member of the University of Maryland Institute for Advanced Computer Studies (UMIACS). Inspired by the possibility that transition to parallelism may be the only paradigm shift in core computing during his professional lifetime, he devoted the first four decades of his research career to two strategic goals. First, crystalize the notion of “parallel algorithms”. He introduced the work-depth framework that was adopted as the presentation framework in parallel PRAM algorithms textbooks, and contributed symmetry breaking techniques, parallel graph and string matching as well as other algorithms. His integer sorting algorithms evolved into the map reduce paradigm, and was overall recognized by his ACM Fellow for, among other things, having “played a leading role in forming and shaping what thinking in parallel has come to mean in the fundamental theory of Computer Science”. Second, the invention of the PRAM-On-Chip desktop supercomputer framework that scales to 1000 processors on chip, and beyond, and its extensive hardware and software prototyping. Notably, “irregular”, “fine-grained” and lockstep PRAM algorithms are effectively supported by a multithreaded computer system, thereby refuting “common wisdom” that effective PRAM support would be “unrealistic”. Prototypes include a 64-processor computer that has already been programmed by nearly 1000 students in a single high-school.

His current goal is establishing “killer app” propositions for general-purpose manycore CPUs. Scouting impactful applications his journey is currently considering ART. A 2019 assertion by hardware design leaders that it is “*unlikely that simple multicore scaling will provide* cost-effective path to growing performance” has only (re)energized his quest to refute it, as well.