My Current Interests in Computation

Rod Downey
Victoria University
Wellington
Isaac Newton Institute, Cambridge

February 2012

THIS LECTURE:

- I have decided to give a brief account of some of the things I am currently interested in rather than a detailed lecture. Talk to me if you are interested.
- Parameterized Complexity
- Generic Case Complexity
- Algorithmic Randomness
- Reverse Mathematics

Parameterized Complexity

- A mathematical idealization is to identify “Feasible” with P (polynomial time). (I won’t even bother looking at the problems with this.)
- With this assumption, the theory of NP-hardness is an excellent vehicle for mapping an outer boundary of intractability, for all practical purposes.
- Indeed, assuming the reasonable current working assumption that NTM acceptance is $\Omega(2^n)$, NP-hardness allows for practical lower bound for exact solution for problems.
- A very difficult practical and theoretical problem is “How can we deal with P?”.
- More importantly how can we deal with $P - FEASIBLE$, and map a further boundary of intractability.

- Lower bounds in P are really hard to come by. But this theory will alow you establish infeasibility for problems in P, under a reasonable complexity hypothesis.
- Also it will indicate to you how to attack the problem if it looks bad.
- As we soon see, sensitizing the run times to parameters allows the development of a distinctive and often useful toolkit.
- The theory equips us with both a positive and negative tool kit.

I'M DUBIOUS; EXAMPLE?

- Below is one application that points at why the completeness theory might interest you.
- The great PCP Theorem of Arora et. al. allows us to show that things don't have PTAS's on the assumption that P$\neq$NP.
- Some things actually do have PTAS's. Lets look at a couple taken from recent major conferences: STOC, FOCS, SODA etc.

- Arora 1996 gave a $O(n^{\frac{3000}{\epsilon}})$ PTAS for EUCLIDEAN TSP
- Chekuri and Khanna 2000 gave a $O(n^{12(\log(1/\epsilon)/\epsilon^8)})$ PTAS for MULTIPLE KNAPSACK
- Shamir and Tsur 1998 gave a $O(n^{2^{2^{\frac{1}{\epsilon}}}-1)})$ PTAS for MAXIMUM SUBFOREST
- Chen and Miranda 1999 gave a $O(n^{(3mm!)^{\frac{m}{\epsilon}+1}})$ PTAS for GENERAL MULTIPROCESSOR JOB SCHEDULING
- Erlebach et al. 2001 gave a $O(n^{\frac{4}{\pi}(\frac{1}{\epsilon^2}+1)^2(\frac{1}{\epsilon^2}+2)^2})$ PTAS for MAXIMUM INDEPENDENT SET for geometric graphs.

- Deng, Feng, Zhang and Zhu (2001) gave a $O(n^{5\log_{1+\epsilon}(1+(1/\epsilon))})$ PTAS for UNBOUNDED BATCH SCHEDULING.
- Shachnai and Tamir (2000) gave a $O(n^{64/\epsilon+(log(1/\epsilon)/\epsilon^8)})$ PTAS for CLASS-CONSTRAINED PACKING PROBLEM (3 cols).

Reference	Running Time for a 20% Error
Arora (Ar96)	$O(n^{15000})$
Chekuri and Khanna (CK00)	$O(n^{9,375,000})$
Shamir and Tsur (ST98)	$O(n^{958,267,391})$
Chen and Miranda (CM99)	$> O(n^{10^{60}})$ (4 Processors)
Erlebach et al. (EJS01)	$O(n^{523,804})$
Deng et. al (DFZZ01)	$O(n^{50})$
Shachnai and Tamir (ST00)	$O(n^{1021570})$

The Running Times for Some Recent PTAS's with 20% Error.

What is the problem here?

- Arora (1997) gave a PTAS running in nearly linear time for Euclidian TSP. What is the difference between this and the PTAS's in the table. Can't we simply argue that with more effort all of these will eventually have truly feasible PTAS's.
- The principal problem with the baddies is that these algorithms have a factor of $\frac{1}{\epsilon}$ (or worse) in their exponents.
- By analogy with the situation of *NP* completeness, we have some problem that has an exponential algorithm. Can't we argue that with more effort, we'll find a much better algorithm? As in Garey and Johnson's famous cartoon, we cannot seem to prove a better algorithm. BUT we prove that it is NP hard.

- Then assuming the working hypothesis that there is basically no way to figure out if a NTM has an accepting path of length n except trying all possibilities there is no hope for an exact solution with running time significantly better than 2^n. (Or at least no polynomial time algorithm.)
- Moreover, if the PCP theorem applies,then using this basic hypothesis, there is also no PTAS.

- In the situation of the bad PTAS's the algorithms are polynomial. Polynomial lower bound are hard to come by.
- It is difficult to apply classical complexity since the classes are not very sensitive to things in P.
- Our idea in this case is to follow the NP analogy but work within P.

- What parametric complexity has to offer:
- Then assume the working hypothesis that there is basically no way to figure out if a NTM has an accepting path of length k except trying all possibilities. Note that there are $\Omega(n^k)$ possibilities. (Or at least no way to get the "k" out of the exponent or an algorithm deciding k-STEP NTM,)

- One then defines the appropriate reductions from k-STEP TURING MACHINE HALTING to the PTAS using $k = \frac{1}{\epsilon}$ as a parameter to argue that if we can "get rid" of the k from the exponent then it can only be if the working hypothesis is wrong.

EFFICIENT PTAS'S

- Even if you are only interested in "classical" problems you would welcome a methodology that allows for "practical" lower bounds in P, modulo a reasonable complexity assumption.
- An optimization problem Π has an efficient P-time approximation scheme e (EPTAS) if it can be approximated to a goodness of $(1 + \epsilon)$ of optimal in time $f(k)n^c$ where c is a constant and $k = 1/\epsilon$.

BAGZAN, CAI-CHEN

- (without even the formal definition) (Bazgan (Baz95), also Cai and Chen (CC97)) Suppose that Π_{opt} is an optimization problem, and that Π_{param} is the corresponding parameterized problem, where the parameter is the value of an optimal solution. Then Π_{param} is fixed-parameter tractable if Π_{opt} has an EPTAS.

- Parameterized complexity allows for an extended “dialog” with the problem at hand. (More on this soon).

- Others to use the hardness theory include the following
- (Alekhnovich and Razborov (AR01)) Neither resolution not tree-like resolution is automizable unless $W[P]$ is randomized FPT by a randomized algorithm with one-sided error. (More on the hypothesis later)
- Frick and Grohe showed that towers of twos obtained from general tractability results with respect to model checking can't be gotten rid of unless $W[1] = FPT$, again more later.

PARAMETERS

- Without even going into details, think of all the graphs you have given names to and each has a relevant parameter: planar, bounded genus, bounded cutwidth, pathwidth, treewidth, degree, interval, etc, etc.
- Also nature is kind in that for many practical problems the input (often designed by us) is nicely ordered.

Two Basic Examples

- VERTEX COVER
  Input: A Graph G.
  Parameter : A positive integer k.
  Question: Does G have a size k vertex cover? (Vertices cover edges.)
- DOMINATING SET
  Input: A Graph G.
  Parameter : A positive integer k.
  Question: Does G have a size k dominating set? (Vertices cover vertices.)

- VERTEX COVER is solvable by an algorithm $\mathfrak{O}$ in time $f(k)|G|$, a behaviour we call fixed parameter tractability, (Specifically $1.2745^k k^2 + c|G|$, with c a small absolute constant independent of k.)
- Whereas the only known algorithm for DOMINATING SET is complete search of the possible k-subsets, which takes time $\Omega(|G|^k)$.

Basic Definition(s)

- Setting : Languages $L \subseteq \Sigma^* \times \Sigma^*$.
- Example (Graph, Parameter).
- We say that a language L is fixed parameter tractable if there is a algorithm M, a constant C and a function f such that for all x, k,

$$(x, k) \in L \text{ iff } M(x) = 1 \text{ and}$$

$$\text{the running time of } M(x) \text{ is} f(k)|x|^C.$$

- E.g. VERTEX COVER has $C = 1$. Vertex Cover has been implemented and shown to be practical for a class of problems arizing from computational biology for k up to about 7000 and n large.
- One example: Langston et. al. 2008 Innovative computational methods for transcriptomic data analysis: A case study in the use of FPT for practical algorithm design and implementation. in *The Computer Journal*, 51(1):26–38, 2008.

Does this matter?

- The table below illustrates why this might be interesting.

	$n = 50$	$n = 100$	$n = 150$
$k = 2$	625	2,500	5,625
$k = 3$	15,625	125,000	421,875
$k = 5$	390,625	6,250,000	31,640,625
$k = 10$	1.9×10^{12}	9.8×10^{14}	3.7×10^{16}
$k = 20$	1.8×10^{26}	9.5×10^{31}	2.1×10^{35}

TABLE: The Ratio $\frac{n^{k+1}}{2^k n}$ for Various Values of n and k

- Note that we are using arbitarily $f(k) = 2^k$, and sometimes we can do better. (Such as the case of Vertex Cover)
- So the FPT is interesting since it works better than complete search for problems where we might be interested in small parameters but large input size.

POSITIVE TECHNIQUES

- Elementary ones
- Include Kernelization, Bounded search trees, Struction, Crown Reductions, IP Relaxation, Lenstra's IP bounded Variable, Iterative Compression.
- Colour Coding and Greedy Localization
- Graph Structure Theory Width metrics: treewidth, cutwidth d-inductive graphs etc.
- Logical metatheorems Courcelle's Theorem, Excluded Minor theorems, Bidimensionality.
- Limits

Reductions and Intractability

- Natural basic hardness class: $W[1]$. Does not matter what it is, save to say that the analog of Cook's Theorem is SHORT NONDETERMINISTIC TURING MACHINE ACCEPTANCE
  Instance: A nondeterministic Turing Machine M and a positive integer k.
  Parameter: k.
  Question: Does M have a computation path accepting the empty string in at most k steps?

- If one believes the philosophical argument that Cook's Theorem provides compelling evidence that SAT is intractible, then one surely must believe the same for the parametric intractability of SHORT NONDETERMINISTIC TURING MACHINE ACCEPTANCE.
- Moreover, recent work has shown that if SHORT NTM is fpt then n-variable 3SAT is in DTIME($2^{o(n)}$)

- Given two parameterized languages $L, \widehat{L} \subseteq \Sigma^* \times \Sigma^*$, say $L \leq_{FPT} \widehat{L}$ iff there are (computable) $f, x \mapsto x', k \mapsto k'$ and a constant c, such that for all x,

  $$(x,k) \in L \text{ iff } (x',k') \in \widehat{L},$$

  in time $f(k)|x|^c$.
- Lots of technical question still open here.

ANALOG OF COOK'S THEOREM

- Analog of Cook's Theorem: (Downey, Fellows, Cai, Chen) WEIGHTED 3SAT$\equiv_{FTP}$ SHORT NTM ACCEPTANCE.
  WEIGHTED 3SAT
  Input: A 3 CNF formula ϕ
  Parameter: k
  Question: Does ϕ has a satisfying assignment of Hamming weigth k, meaning exactly k literals made true.

W-HIERARCHY

- Think about the usual poly reduction from SAT to 3SAT. It takes a clause of size p, and turns it into many clauses of size 3. But the weight control goes awry. A weight 4 assignment could go to anything.
- We don't think WEIGHTED CNF SAT$\leq_{ftp}$WEIGHTED 3 SAT.
- Gives rise to a heirarchy:

$$W[1] \subseteq W[2] \subseteq W[3] \ldots W[SAT] \subseteq W[P] \subseteq XP.$$

- XP is quite important, it is the languages which are in DTIME$(n^{f}(k))$ with various levels of uniformity, depending on the choice of reductions.

- XP has k-Cat and Mouse Game and some other games ((DF99a)),
- $W[P]$ has Linear Inequalities, Short Satisfiability, Weighted Circuit Satisfiability ((ADF95)) and Minimum Axiom Set((DFKHW94)).
- Then there are a number of quite im portant problems from combinatorial pattern matching which are $W[t]$ hard for all t: Longest Common Subsequence ($k =$ number of seqs.,$|\Sigma|$-two parameters) ((BDFHW95)), Feasible Register Assignment, Triangulating Colored Graphs, Bandwidth, Proper Interval Graph Completion ((BFH94)), Domino Treewidth ((BE97)) and Bounded Persiste nce Pathwidth ((McC03)).
- $W[2]$ include Weighted $\{0, 1\}$ Integer Programming, Dominating Set ((DF95a)), Tournament Dominating Set ((DF95c)) Unit Length Precedence Constrained Scheduling (hard) ((BF95)), Shortest Common Supersequence (k)(hard) ((FHK95)), Maximum Likelihood Decoding (hard), Weight Distribution in Linear Codes (hard), Nearest Vector in Integer Lattices (hard) ((DFVW99)), Short Permutation Group Factorization (hard).
- $W[1]$ we have a collection including k-Step Derivation for Context Sensitive Grammars, Short NTM Computation, Short Post Correspondence, Square Tiling ((CCDF96)), Weighted q–CNF Satisfiability ((DF95b)), Vapnik–Chervonenkis Dimension ((DEF93)) Longest Common Subsequence (k, $m =$ length of common subseq.) ((BDFW95)), Clique, Independent Set ((DF95b)), and Monotone Data Complexity for Relational Databases

Algorithms in nature

- One of my own research agendas has been to understand why algorithms work better (or worse) than we expect. This is where parameterized complexity came from. (Exploiting the fact that almost all data from "real life" has parameters bounded in some way. This can yield a lot of good algorithmics.)
- One aspect of this came from group theory through the work of Schupp, Myasnakov, and others on "generic case complexity."

References

- Asymptotic Density for c.e. Sets (with Jockusch and Schupp) in preparation.
- Generic Computability, Turing Degrees and Asymptotic Density (Jockusch and Schupp), to appear, JLMS.
- Generic case complexity, decision problems in group theory and random walks, (Kapovich, Miasnikov, Schupp and Shpilrain) J. Algebra, (2003)
- Genericity, the Arshantseva-Ol'shanskii technique and the isomorphism problem for one relator groups, (Kapovich and Schupp) Math Ann (2005)

Background

- Classical complexity, P, NP etc seems often the wrong model for actual behaviour of problems.
- E.g Simplex Algo, Polynomial Identity Testing etc.
- Other models: Parameterized complexity (Downey-Fellows), average case complexity (Gurevich-Levin), smoothed analysis (Spielman-modern version of average case)
- The first does not always explain things it seems, and the last two are hard to apply (distributions etc)
- New method suggested by Kapovich, Miasnikov, Schupp and Shpilrain in 2003.

Asymptotic density

- A finite alphabet Σ
- Let S be a subset of Σ^*. For every $n \geq 0$ let $S\lceil n$ denote the set of all words in S of length at most n.
- Let

$$\rho_n(S) = \frac{|S\lceil n|}{|\Sigma^*\lceil n|}$$

- Upper density (Borel)

$$\overline{\rho}(S) := \limsup_{n\to\infty} \rho_n(S)$$

- Similarly, Lower density
- (asymptotic) density If the actual limit

$$\rho(S) = lim_{n\to\infty}\rho_n(S) \text{ exists}$$

Generic Case Complexity

- A subset S of Σ^* is generic if $\rho(S) = 1$ and S is negligible if $\rho(S) = 0$
- exponentially fast Exist $0 \leq \sigma < 1$ and $C > 0$ such that for every $n \geq 1$ we have $1 - \rho_n(S) \leq C\sigma^n$. In this case we say that S is strongly generic.
- A (partial) $\Phi \mid \Sigma^* \to \{0, 1\}$ is a generic description of S if $\Phi(x) \downarrow \to \Phi(x) = S(x)$ and the domain of Φ is generic.
- A set S is called generically computable if there exists a *partial computable* function Φ which is a generic description of S.

Other examples

- Using what is called is called the quotient method and can be used for any $G = \langle X, R\rangle$ subgroup of K of finite index for which there is an epimorphism $K \to H$ hyperbolic and not virtually cyclic, to show generically solvable word problem.
- Applies also to 1-relator groups with ≥ 3 generators similarly (no bound for Magnus' solution), plus isomorphism problem; and braid groups, and automorphism problems for free groups etc.
- Boone's group also, unknown if there is a one without a generically solvable word problem. (See also Gilman, Miasnikov and Osin for the strong case)
- See the papers by Schupp, Kapovich etc.

LOTS OF QUESTIONS

- Understand this better.
- What about other structures.
- Generic case model theory and coarse model theory.
- How does this relate to classical complexity, etc.

ALGORITHMIC RANDOMNESS

- Have heard talks about how to use computation theory to understand randomness via things like effective null sets, effective betting.
- Recall α is ML-random iff α is not $\cap_n U_n$ where U_n is a c.e. open set of measure $\leq 2^{-n}$.
- Also recall A is K-trivial iff $K(A \upharpoonright n) \leq^K (n)$ all n iff $K^A =^+ K$.
- Have been looking at exact pairs for K-trivials, and integer valued randomness, especially with Barmpalias and with Nies.

- For example, BD classified the c.e. Turing degrees containing IVR's as the aray noncomputables.
- To wit: a martingale is played on $2^{<\omega}$ and has $f(\sigma) = \frac{f(\sigma 0) = f(\sigma 1)}{2}$. (Fairness)
- α is ML-random iff no left c.e. martingale succeeds on it meaning that $\limsup_n f(\alpha \upharpoonright n) \to \infty$.
- α is integer valued random iff no integer valued martingale succeeds. (Think of a "real" casino.)
- Also I think we can prove that the K-trivials have an exact pair (BDN) (reasonably longstanding technical question)

- (BD) have also studied reals with $K(\alpha \upharpoonright n) \geq^+ K(\alpha \upharpoonright f(n))$ for each computable order f. (weakly K-resolute)
- Such c.e. sets can be Turing complete, not all c.e. degrees have them, and there is a non-K-trivial completely K-resolute degree.
- Also talking with Ted Slaman's student Ian Herbert about mutual information for reals, akin to symmetry of information a 'la Levin.

SPECULATIVE, PHYSICS

- There are obvious things like Brownian motion.
- The Asarin-Fouché-Kjos-Hanssen-Nerode approach is to look (as usual) at the space of continuous $f[0,1] \to \mathbb{R}$ with the uniform metric $d(f,g) = sup|f(x) - g(x)|$ and Wiener measure. Then you can classify the notion of an individual random Brownain motion using Kolmogorov complexity.
- The question is what does this say about "real" Brownain motion?
- Classical physics treats space-time as a manifold. So most processes are pde's and presumably they are "computable" in the sense that if I closely approximate the input the same is true of the output. Thus in that context, the above would make some sense. As does algorithmic randomness.

Physics continued

- There is a nice research programme and lots of interesting questions here.
- Can such a system generate randomness? or even incomputability (Pour-El, Richards)?
- According to my friendly physicist (Matt Visser) and most of my reading (e.g. Speakable and unspeakable in quantum physics, Bell's Theorem, the recent essay competition in FQXi), quantum phyics could be hard to reconcile with the manifold interpretation.
- At the heart of quantum physics at the Planck level, things seem highly non-continuous. When, for example, we observe spin, it (with some degree of randomness) chooses.
- How to interpret this. There are at least 4-6 interpretations and one could speculate that algorithmic randomness might have a show at sheding light on this subject.

- It could be that at at some level *nothing* is computable yet (like looking at a TV from afar) it all looks smooth and computable.
- I am aware of no framework at all which can be used to computationally represent this, and it would seem a very interesting project to do this.
- I like it as I would need to learn some physics and could charm a physicist, maybe with logic.
- Also the work of Shannon→Lee Ruebel on GPAC if the world is a manifold.

Biology

- What about the processes of biology?
- At the speculative level, the sizes of computation occuring in cell walls is nano-scale and hence *must* experience some quantum effects. Thus it is tied to the above.
- There is a recent programme begun by Winfree in 1998 representing DNA as tile self assemply. The idea is you have tiles with rules and a nondeterministic bonding according to the rules.
- This seems a natural model to try to model real DNA and I think also you could add a randomness mutation to the rules. This has not been explored, but the Lutz-Adleman group have looked at this and it seems very promising.
- What about doing this in reverse, maybe proving such randomness is necessary for viability of a system.

- Of course there are older ideas *using* K-complexity.
- There is no accepted notion of good similarity between genertic information, (e.g. music also), e.g. maximum parsimony, maximum likelihood etc.
- Idea: why not use K-complexity. So the common information is measured by an approximation to Kolmogorov complexity.
- As compression algorithms improve, this seem to work better.

Reverse mathematics

- Looking at e.g. FIP. Every infinite family has a maximal subfamily where every finite subset has nonempty finite intersection.
- As Sets this is equivalent to ACA_0 over RCA_0.
- (With Diamondstone, Greenberg, Turetsky) As families the FIP (those that can compute solutions to all computable families) degrees below $\mathbf{0}'$ are exactly the degrees bounding $1-$generic ones.
- (DDGT) But there is a minimal one. Note Damir Dzharfarov has shown that the degrees are all hyperimmune.

Thank You