Phase Transitions and Computational Complexity

Moshe Y. Vardi

Rice University

Perebor: Brute-Force Search

- 1950s – Early computer science in Soviet Union: some problems seem to be solvable only via brute-force search – research program ultimately *unsuccessful*.
- 1970s – Cook, Karp, Levin: explanation via reductions, *worst-case* complexity.
- 1980s – Levin, Gurevich: explanation via *average-case* NP-completeness – not too successful (few problems amenable)

"Where The Hard Poblems Are"

Cheeseman-Kanefsky-Taylor, 1991:

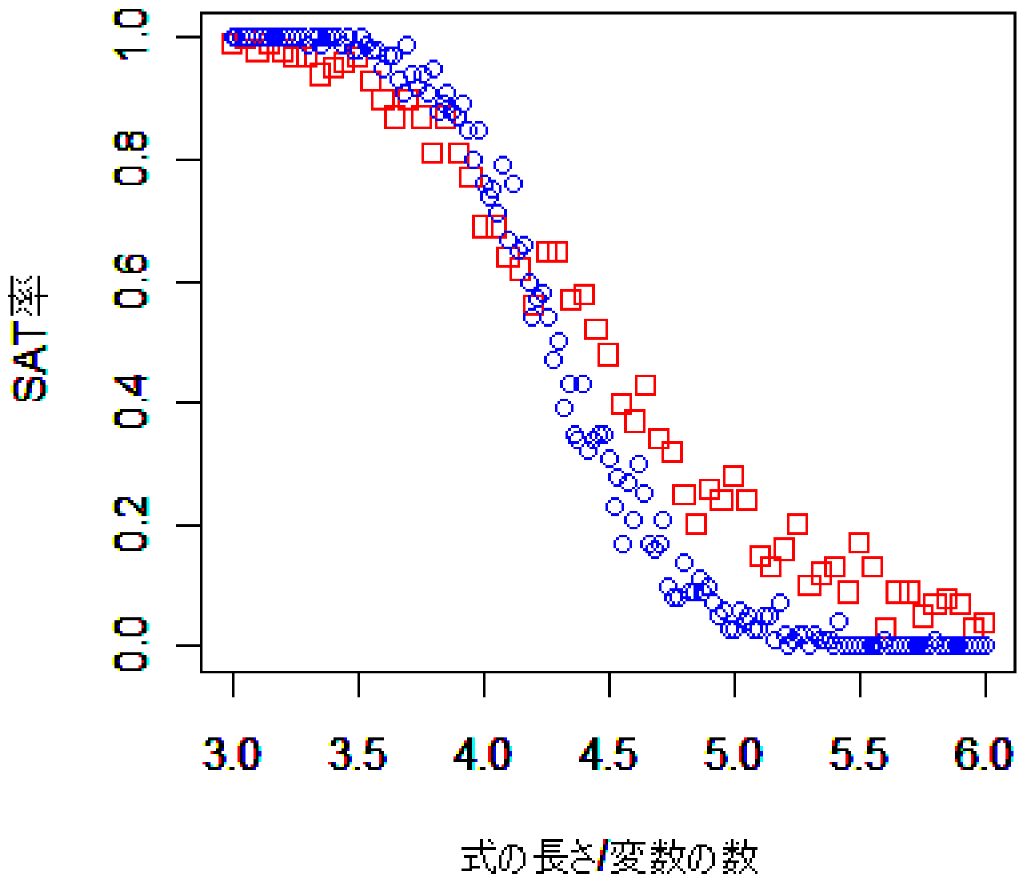
- Worst-case complexity not always useful – not a good guide for real-world performance
- Instances of NP-complete problems can be quite easy in practice, e.g., industrial SAT instances with millions of clauses!
- Decision problems can often be characterized in terms of "constrainedness"
- The hard problems are those that lie in the transition from under-constrained to over-constrained.

Example: 3-SAT

Motivation:
- NP-complete
- Basic reasoning task
- Useful algorithmic abstraction

- **Literal**: $\pm x$
- **Clause**: $\pm x_{i_1} \vee \pm x_{i_2} \vee \pm x_{i_3}$
- **Instance**: $\bigwedge_{i=1}^{m} C_i$
- **Order**: number of variables
- **Density**: ratio between number of clauses and number of variables
- **Random 3-SAT**: fix order and density, choose clauses uniformly
- **Asymptotics**: Fix density, take order to infinity

Phase Transition of Random 3-SAT



Asymptotic Random 3-SAT

Intuition

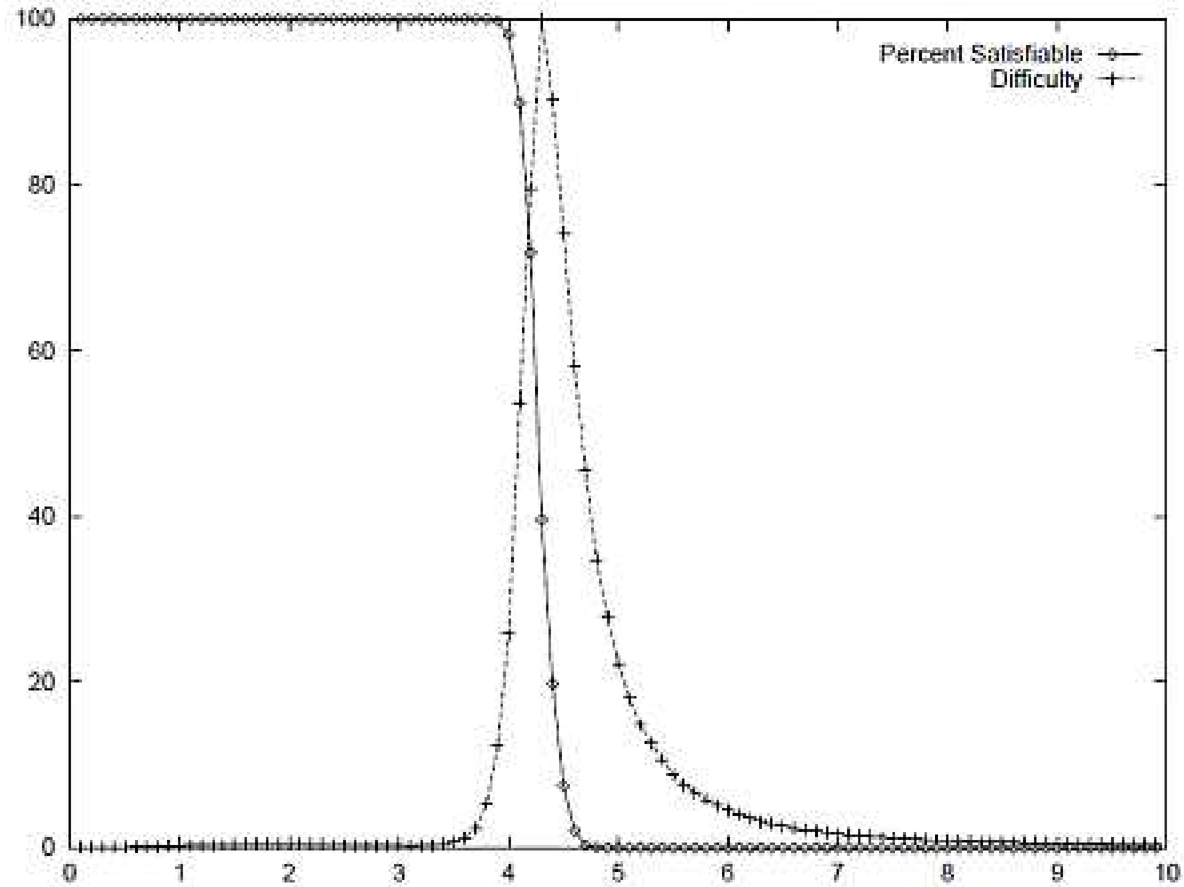
- Easy satisfiability for underconstrained problems
- Hard satisfiability for overconstrained problems

Known:

- Kaporis et al., 2006 Density < 3.52: asymptotic probability 1
- Diaz et. al, 2009: Density > 4.4898: asymptotic probability 0

Conjecture: sharp phase transition!

Phase Transition and Solver Time for Random 3-SAT



"Easy, Hard, Easy"

Mitchell–Selman–Levesque, 1992

- Constrainedness$=$Density!
- *Empirically*: phase transition from probability 1 to probability 0 at density 4.26.
- *Empirically*: DPLL solver time peaks at density 4.26.

Bottom Line:
- Low density: easy
- High density: easy
- Crossover: *hard*

Phase Transitions and Computational Complexity

Conjecture: There is a profound connection between computational complexity and phase transitions!

Evidence: *Random-Graph Colorability*

- *Order*: Number of nodes
- *Density*: Ratio between number of edges and number of nodes
- *Random Graphs*: Fix order and density, and choose edges uniformly
- *Asymptotics*: Fix density and take order to infinity

Known:
- 2-COLOR *does not* have a phase transition (provably)
  – 2-COLOR is in PTIME.
- 3-COLOR *does* have a phase transition (empirically)
  – 3-COLOR is NP-complete.

Phase Transitions and Computational Complexity

1990s: *lots of excitement*!!!

- Physicists get involved
- Articles in *Science*
- Many invited talks
- Microsoft Research forms a Theory Group, planning to use Statisticsl Physics to solve P vs. NP!

Difficulties:
- Nice results about probability, but no concrete results about complexity.
- 2-SAT has a phase transition at density 1 [Chvatal–Reed, Goerdt, 1992]
- XOR-SAT has a phase transition at density 1 [Creignu–Daube, 2000]

HornSAT

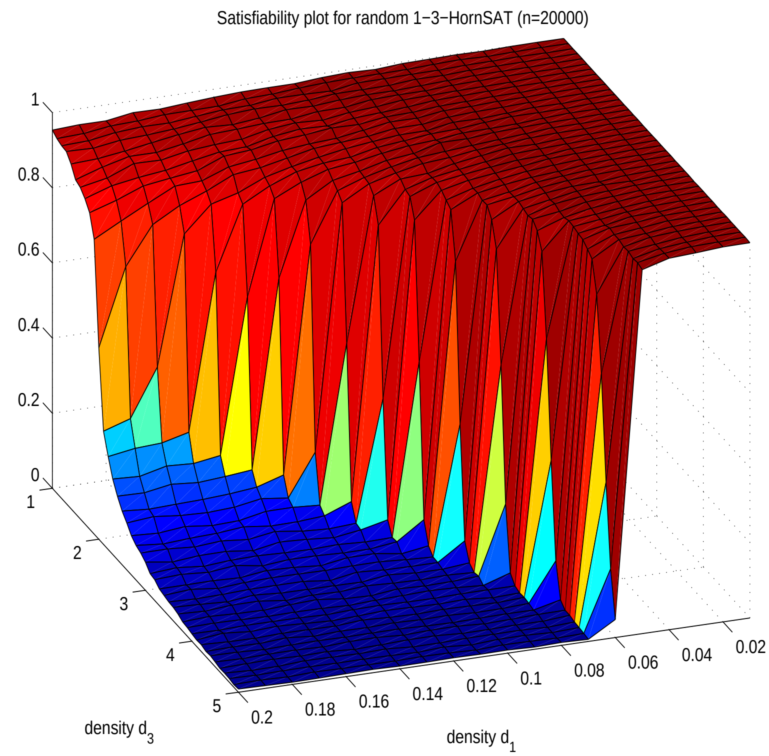
HornSAT: at most one positive literal in a clause

- Tractable – solvable in linear time [Beeri–Bernstein, 1979, Dowling–Gallier, 1984].

Demopolous-V., 2003: *Random 1-3-HornSAT*

- n variables
- 1 negative clause
- $d_1 n$ positive (unit) clauses
- $d_3 n$ implications ($x_i \wedge x_j \rightarrow x_k$)

Phase Transition for HornSAT?



Satisfiability probability of a 1-3-Horn formula of size 20,000

Istrate-Moore-Demopolous-V., 2003:
Analytical Results – (d_1, d_3)-plane
- Phase transition for asymptotic probability
- Phase transition between region with phase transition and region without phase transition

Bottom Line: Phase transitions are *not* intrinsically related to computational complexity!

What is "Easy, Hard, Easy"?

Coarfa–Demopolous–San Miguel Aguirre–Subramanian–V., 2000:

- What is meant by "easy" and "hard"?
- Where are the transitions between "easy" and "hard"?

Note: MSL's "Easy-Hard-Easy" refers to *fixed order* and *varying density*.

Proposed Approach:

- Focus on *scalability*:
  - *Easy*: polynomial scalability
  - *Hard*: exponential scalability
- *Fix density, scale order*
  - Corresponds to analytical studies
  - Corresponds to applications

“Easy” $\neq$ Easy

- **Very low density problems are indeed easy**: Algorithms with linear median time below density 1.63 [Broder–Frieze–Upfal, 1993] and below density 3 [Frieze–Suen, 1996]

- **High density problems may be hard**: Exponential lower bound for resolution-proof length at density above 5.2 [Chvatal–Szemeredi, 1988]

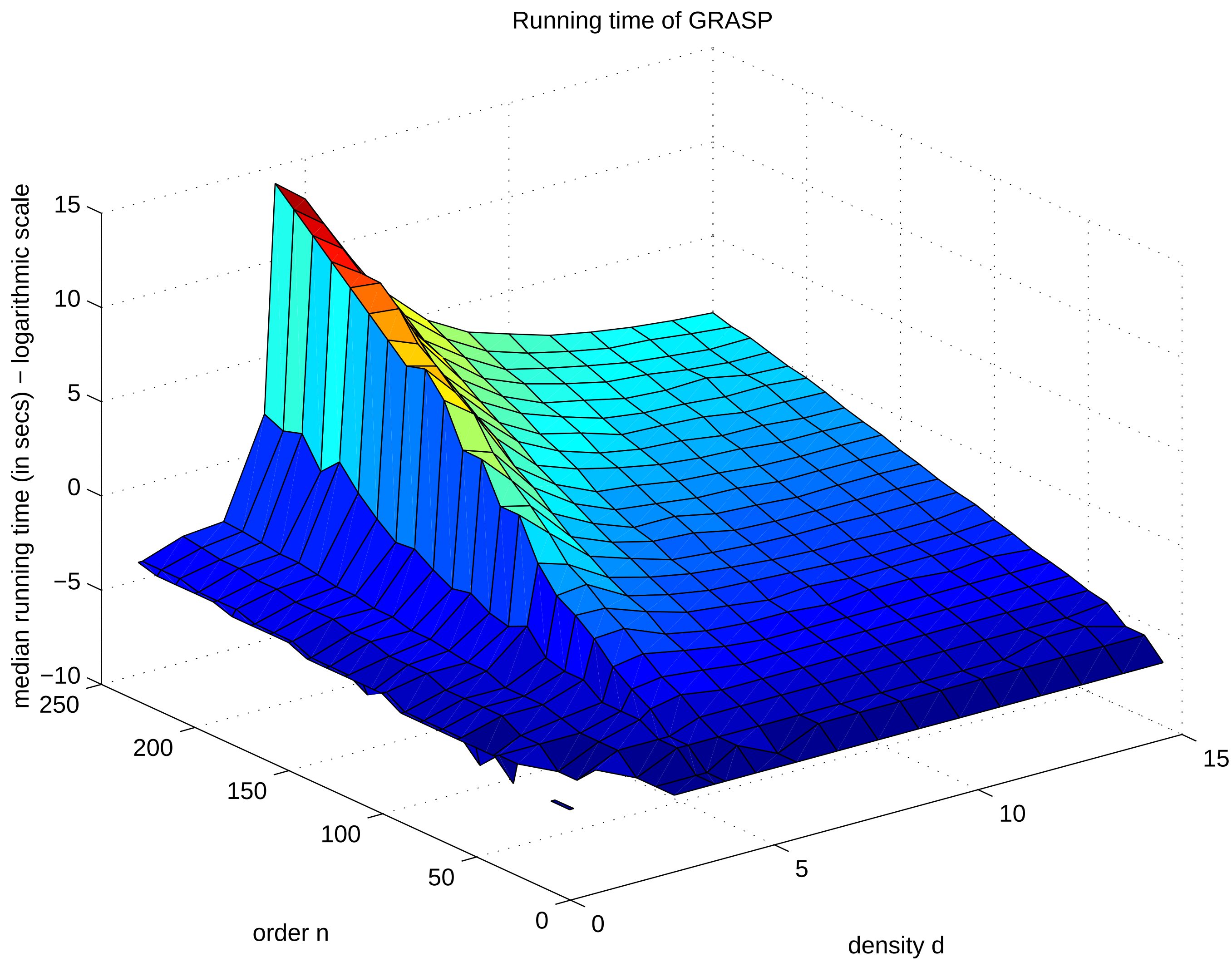
  – SAT solvers are resolution based!

- Ordered-DPLL is exponential with constant probability at density 3.81 [Achlioptas–Beame–Molloy, 2001].

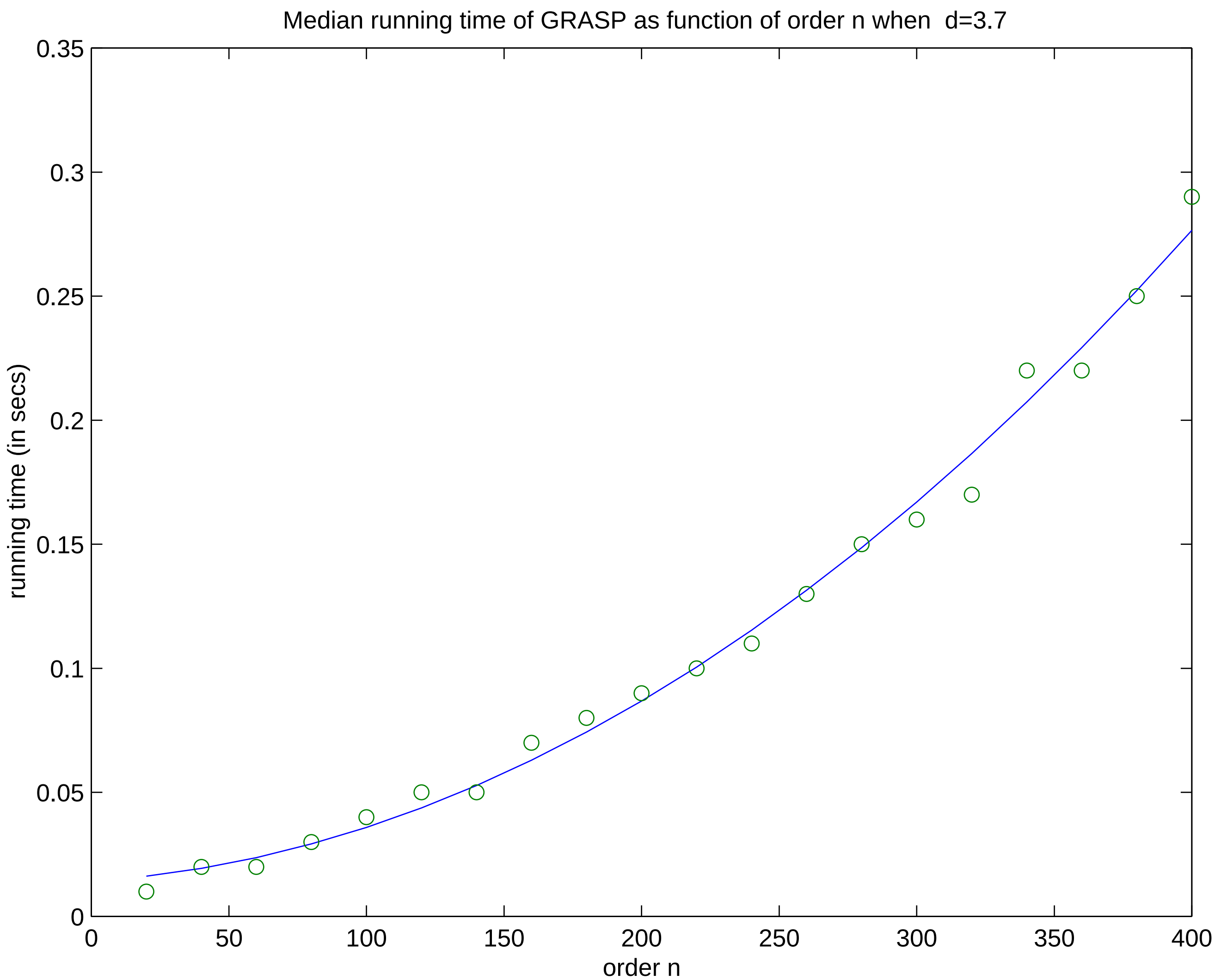
Conclusions:
- “Easy-Hard-Easy” picture is *patently wrong*.
- Algorithmic behavior is algorithm dependent!

Coarfa et al., 2000: *empirical investigation*

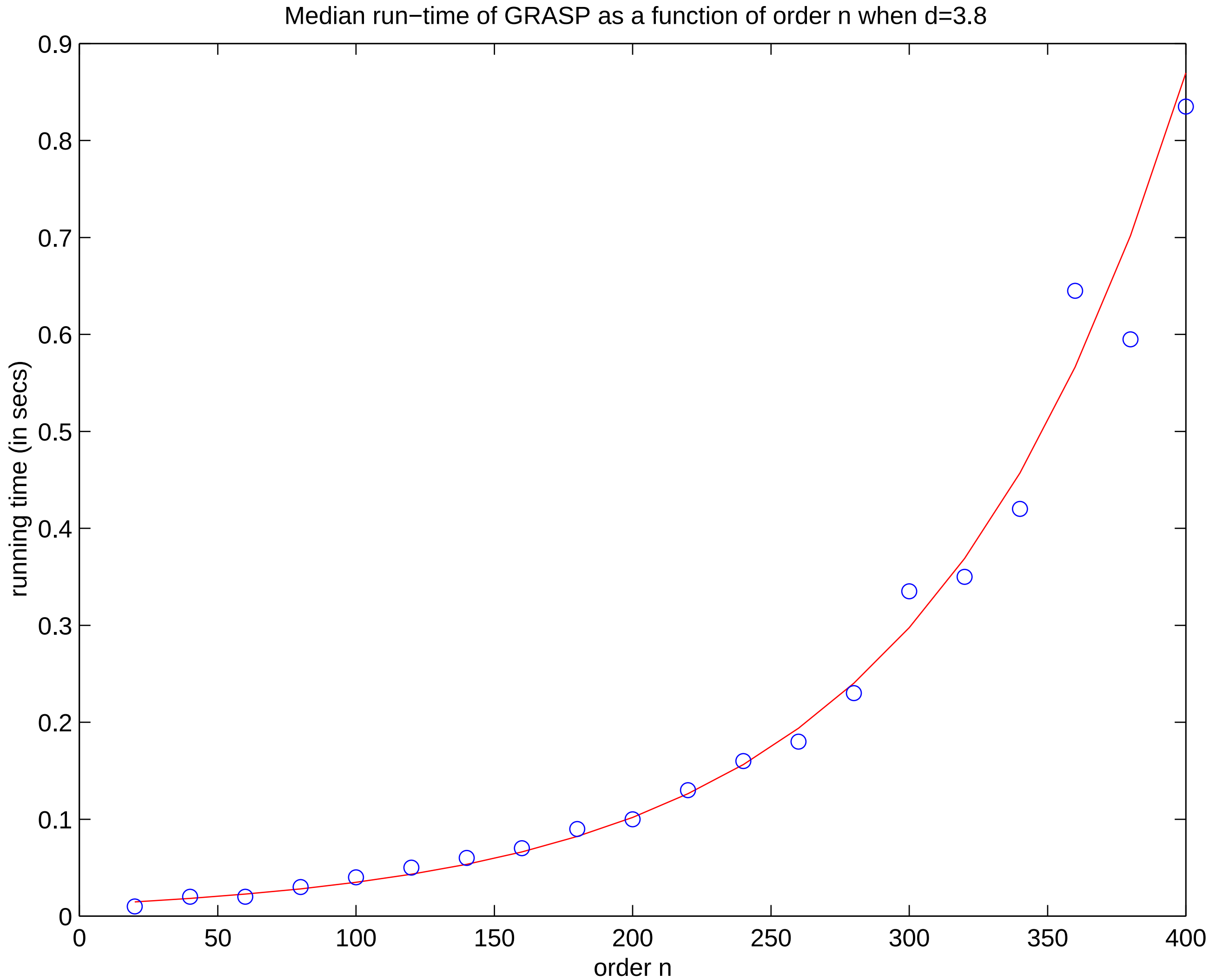
3-D plot of median running time



Polynomial scalability at density 3.7



Exponential scalability at density 3.8



Easy, Harder, Less Hard

New View:

- Density below 3.8: polynomial scalability – "easy"
- Density between 3.8 and 4.26: exponential scalabilty, increasing exponent
- Density larger than 4.26: exponential scalability, decreasing exponent

Single Transition $\longrightarrow$ **Multiple Transitions**:

- Density 1.63: algorithmic correctness transition (analytical)
- Density 3.0: algorithmic correctness transition (analytical)
- Density 3.8: scalability transition (empirical)
- Density 4.26: probability transition (empirical)

Question: Any theory?

Algorithmic-Barrier Theory

Solution-Space Geometry

- *Solution Space*: $sat(f) = \{\alpha \in \{0,1\}^n : \alpha \models f\}$
- *Adjacency*: Hamming distance 1 between solutions
- *Cluster*: a connected component of $sat(f)$

Achlioptas–Coja-Oghlan, 2008:

- For density < 1.63: $sat(f)$ has a single cluster.
- There is a density in the satisfiabe region where $sat(f)$ *shatters* into exponentially many small clusters.

Intuition: algorithms stop being effective after the shattering point.

The $P \neq NP$ Claim

On August 6, 2010, Vinay Deolalikar announced a proof (100-page manuscript) that $P \neq NP$.

- Aug. 6: Manuscript sent to 22 people and put on web page
- Aug. 7: First blog post [Greg Baker]
- Aug. 8: Second blog post [Richard Lipton], Slashdot, extensive commentary
- Aug. 9: Wikipedia article about V.D. (deleted later)
- Aug. 10: Wiki for technical discussion established
  - hundreds of edits
  - Fields medalists involved
- Aug. 15: CACM blogpost by Lipton
- Aug.16: New York Times article

10 Days of Fame for Finite-Model Theory

Richard Lipton, Blog, Aug. 8, 2010:

> "At the highest level he is using the characterization of polynomial time via finite-model theory. His proof uses the beautiful result of Moshe Vardi (1982) and Neil Immerman (1986)."

Theorem: On ordered structures, a relation is defined by a first-order formula plus the Least Fixed Point (LFP) operator if and only if it is computable in polynomial time.

Essence of V.D.'s Proof

Crux: 9-SAT can not be in P!

- If 9-SAT is in P, then it can be expressed in FO+LFP, by the Immerman-V. Theorem.
- But, the FO+LFP normal form is inconsistent with shattering.

XOR-SAT

XOR-SAT: "and of xors"

Example:

$$(\neg x_1 \oplus x_2 \oplus x_3) \wedge (\neg x_2 \oplus \neg x_3 \oplus x_4) \wedge (x_3 \oplus x_1 \oplus x_4)$$

In essence: Linear equations modulo 2

- Solve using Gaussian Elimination

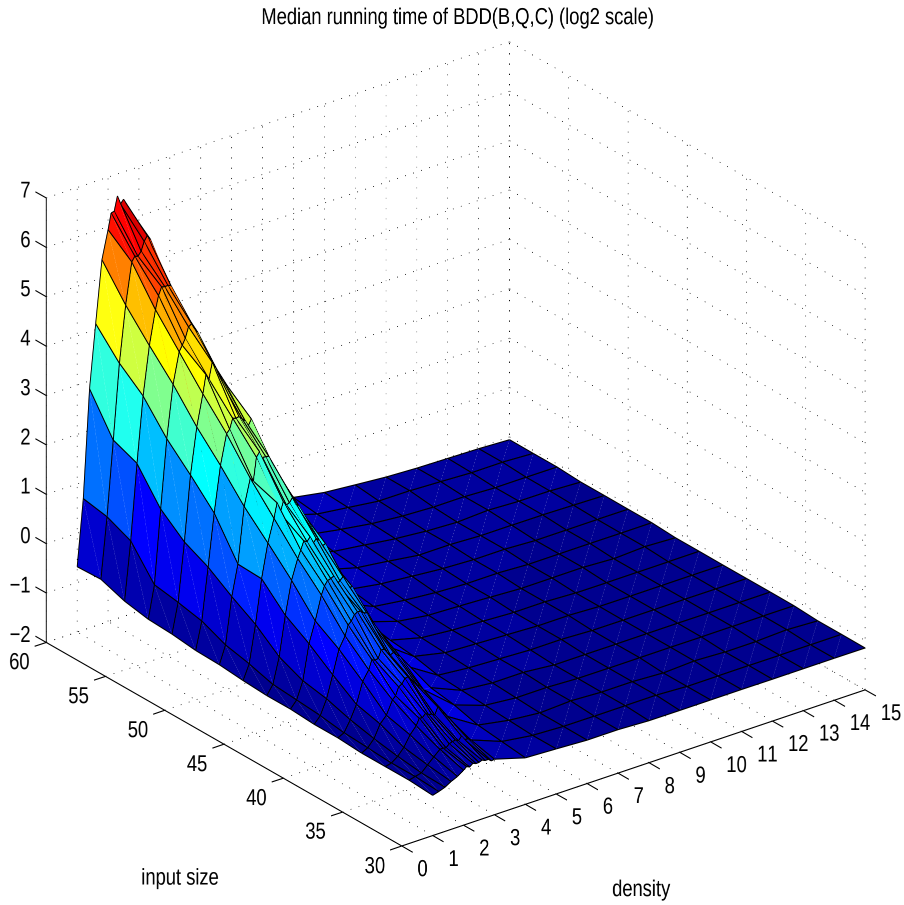
But: The solution space of XOR-SAT shatters! [Mezard–Ricci-Tersenghi–Zecchina, 2003]

Refuting Deolalikar's Proof: XOR-SAT is in P and it shatters!

Conclusion: Shattering is not a good algorithmic-barrier theory.

BDD-Based Algorithms

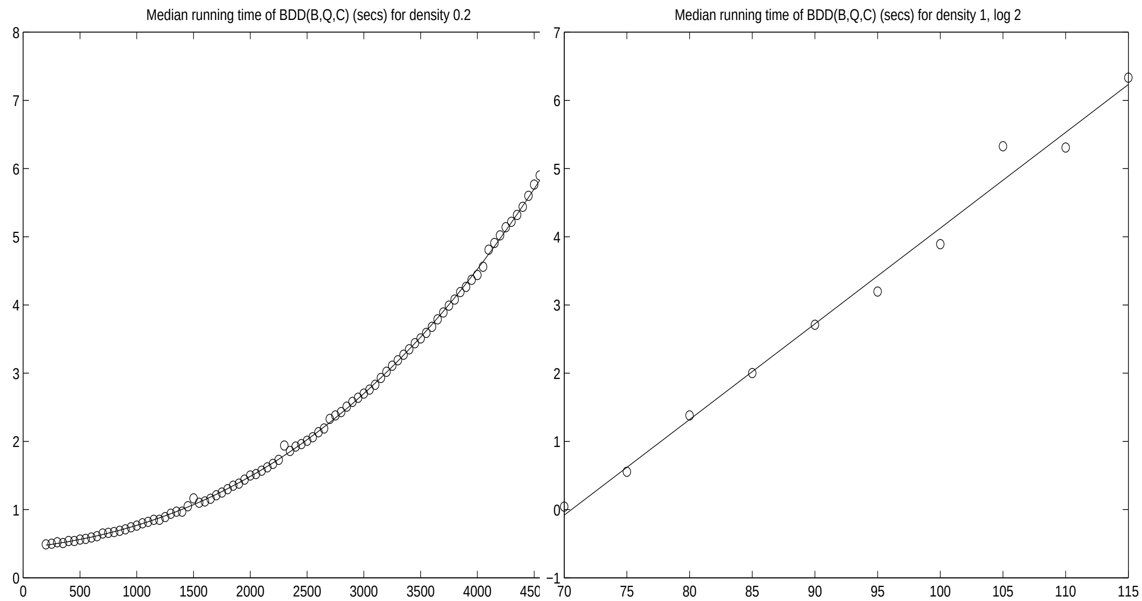
Coarfa et al., 2000, San Miguel Aguirre–V., 2001]: study BDD-based algorithms (can solve XOR-SAT efficiently [AKV'05]).



3-D Plot of median running time

Peak: Density 2.3!

BDDs: Polynomial to Exponential Transition



median running time for (left) density 0.2 and (right) density 1.0

Poly2Exp Phase Transition:

- Density 0.2: quadratic running time
- Density 1.0: exponential running time

Pattern: Easy-Harder-Less Hard

Conclusions

- "Easy-Hard-Easy" pattern – plain wrong
- "Easy-Harder-Less Hard" – more accurate pattern
  - *fundamental pattern?*
- Algorithm-dependent behavior
- Low-density problems are *indeed* easy.
- Where are the hard problems?
  - No evidence of intrinsic hardness!

Bottom Line: phase transition and computational complexity – it is a great story, but the evidence is simply not there!

Postscript: Counting Independent Sets

Weighted Independent Sets: parameter λ

- $wgt(I) = \lambda^{|I|}$

- *Input*: Undirected graph with maximum degree d
- *Output*: Sum of $wgt(I)$, over independent sets I

Results:

- Weitz, 2006: FPTAS for $\lambda < \lambda_c$
- Sly, 2012: No FPRAS for λ just above λ_c (assuming NP$\neq$RP)

"Critical fugacity of the Hardcore Model":
$\lambda_c = (d-1)^{d-1}/(d-2)^d$

Hardcore Model

Hardcore Model:

- Choose independent sets at random according to their (normalized) weights.
- Check if a given node v is in selected independent set I.

Question: How does $Prob(v \in I)$ depends on some other selected nodes V' belonging to I?

Answer: Influence of V' declines as function of distance from v iff $\lambda \leq \lambda_c$.

Atserias: It's a connection between physics and complexity!

V.: So what? The setup is completely different! It says nothing about "constraindness" and complexity!