On the Empirical Time Complexity of Random 3-SAT at the Phase Transition

Zongxu Mu Holger H. Hoos

Department of Computer Science
University of British Columbia
Vancouver, BC, Canada

IJCAI 2015

Complexity and SAT

Time complexity is key in theoretical CS and **practical** applications

- Scaling of running time as function of instance size

Complexity and SAT

Time complexity is key in theoretical CS and **practical** applications

- Scaling of running time as function of instance size

Propositional satisfiability problem (SAT)

- First problem proved to be $\mathscr{NP}$-complete (Cook, 1971)
- Intense academic interest & many **practical** applications
- Dramatic & sustained progress in SAT solving

Outline

1. Background & Location of Phase Transition
2. Scientific Questions & Related Work
3. Empirical Scaling Analysis – Methodology
4. Empirical Scaling Results
   - DPLL-based Solvers
   - SLS-based Solvers
5. Conclusions

Background – Phase Transition

Soluability phase transition: 50% of random instances satisfiable

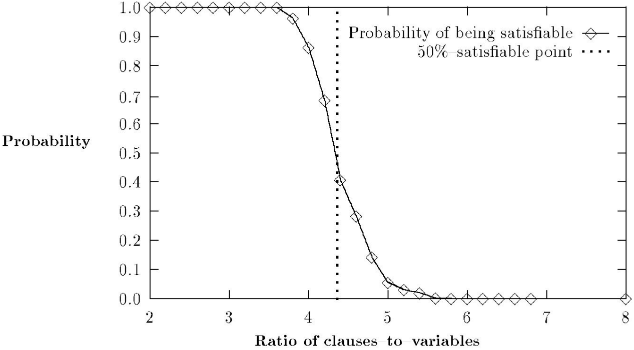


Figure from (Mitchell *et al.*, 1992)

Background – Phase Transition

Soluability phase transition: 50% of random instances satisfiable

- Phase transition is sharp (Cheeseman *et al.*, 1991)
- Believed to converge to fixed threshold

Background – Phase Transition

Soluability phase transition: 50% of random instances satisfiable

- Phase transition is sharp (Cheeseman *et al.*, 1991)
- Believed to converge to fixed threshold

Widely studied instance distribution

- Prominent model of computational hardness in SAT and beyond
  - For DPLL-based solvers (Mitchell *et al.*, 1992)
  - For SLS-based solvers (Yokoo, 1997)
  - ...

Location of Phase Transition

Best previous model (Crawford and Auton, 1996):

$$m_c = 4.258 \cdot n + 58.26 \cdot n^{-2/3}$$

Location of Phase Transition

Best previous model (Crawford and Auton, 1996):

$$m_c = 4.258 \cdot n + 58.26 \cdot n^{-2/3}$$

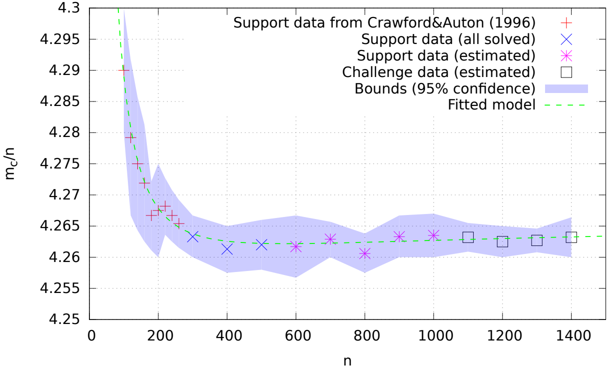
Weaknesses:

- Inconsistent with results from cavity method (Mertens *et al.*, 2006):

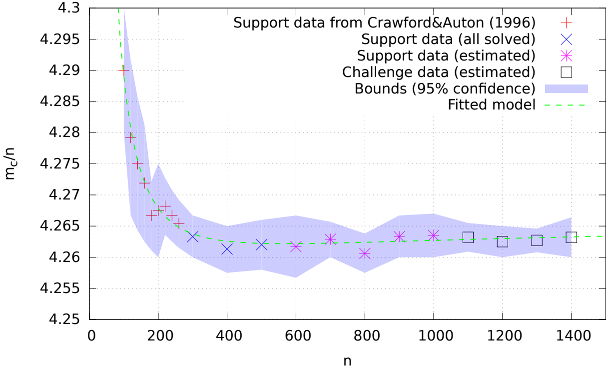
$$\lim_{n \to \infty} m_c / n = 4.26675 \pm 0.00015$$

- Under-estimates m_c for larger n

Location of Phase Transition



Location of Phase Transition



Refined model:

$$m_c = 4.26675 \cdot n + 447.884 \cdot n^{-0.0350967} - 430.232 \cdot n^{-0.0276188}$$

Questions & Experiments

Scientific questions:

Questions & Experiments

Scientific questions:

- How do running times of high-performance SAT solvers scale?

Questions & Experiments

Scientific questions:

- How do running times of high-performance SAT solvers scale?
- Is scaling difference between solvers significant?
  - Between SLS- and DPLL-based solvers?
  - Between two solvers of same kind?

Questions & Experiments

Scientific questions:

- How do running times of high-performance SAT solvers scale?
- Is scaling difference between solvers significant?
  - Between SLS- and DPLL-based solvers?
  - Between two solvers of same kind?
- How much faster are complete solvers solving satisfiable instances?

Questions & Experiments

Scientific questions:

- How do running times of high-performance SAT solvers scale?
- Is scaling difference between solvers significant?
  - Between SLS- and DPLL-based solvers?
  - Between two solvers of same kind?
- How much faster are complete solvers solving satisfiable instances?

Solvers studied:

Questions & Experiments

Scientific questions:

- How do running times of high-performance SAT solvers scale?
- Is scaling difference between solvers significant?
  - Between SLS- and DPLL-based solvers?
  - Between two solvers of same kind?
- How much faster are complete solvers solving satisfiable instances?

Solvers studied:

- DPLL-based: kcnfs, march_hi, march_br

Questions & Experiments

Scientific questions:

- How do running times of high-performance SAT solvers scale?
- Is scaling difference between solvers significant?
  - Between SLS- and DPLL-based solvers?
  - Between two solvers of same kind?
- How much faster are complete solvers solving satisfiable instances?

Solvers studied:

- DPLL-based: kcnfs, march_hi, march_br
- SLS-based: WalkSAT/SKC, BalancedZ, probSAT

Related Work

Observations on empirical scaling of:

- SLS-based solvers, e.g., Gent and Walsh (1993); Gent *et al.* (1997)
- DPLL-based solvers, e.g., Coarfa *et al.* (2003)

Related Work

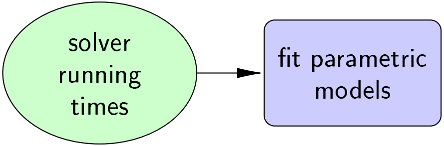
Observations on empirical scaling of:

- SLS-based solvers, e.g., Gent and Walsh (1993); Gent *et al.* (1997)
- DPLL-based solvers, e.g., Coarfa *et al.* (2003)

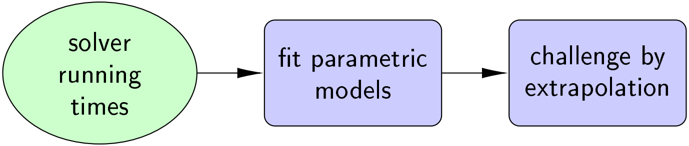
Limitations:

- # variable flips vs. actual running times,
  e.g., Gent and Walsh (1993); Gent *et al.* (1997)
- Inconclusive results, e.g., Gent and Walsh (1993)
- Simple curve fitting & vague definition of "good fit"

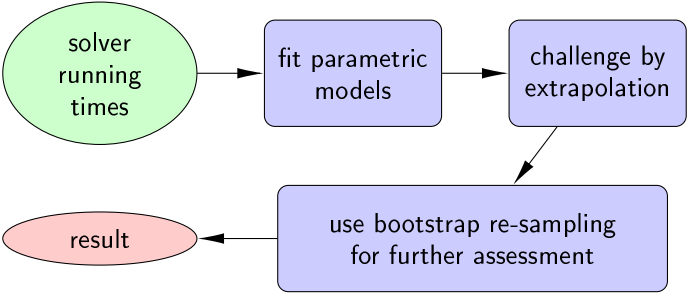
Methodology (Hoos, 2009; Hoos and Stützle, 2014)



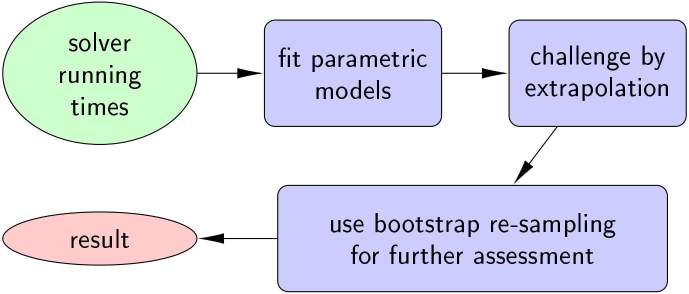
Methodology (Hoos, 2009; Hoos and Stützle, 2014)



Methodology (Hoos, 2009; Hoos and Stützle, 2014)



Methodology (Hoos, 2009; Hoos and Stützle, 2014)



Extensions made in this work:

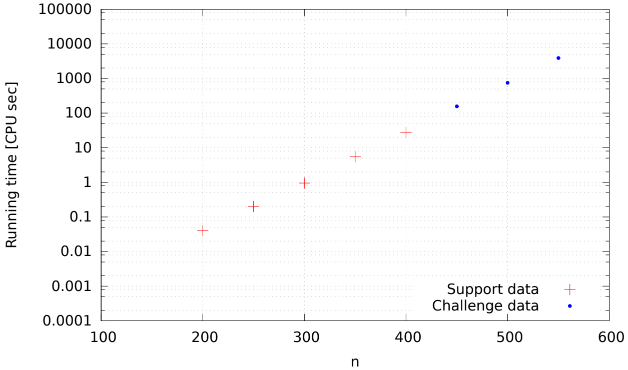
- Use confidence intervals of observed data to assess models
- Compare scaling models of two solvers based on confidence intervals of observed/predicted data

Empirical Scaling Results – DPLL-based Solvers

Divide instance sets into support and challenge:

n	200	250	300	350	400
median	0.040	0.200	0.950	5.455	27.580

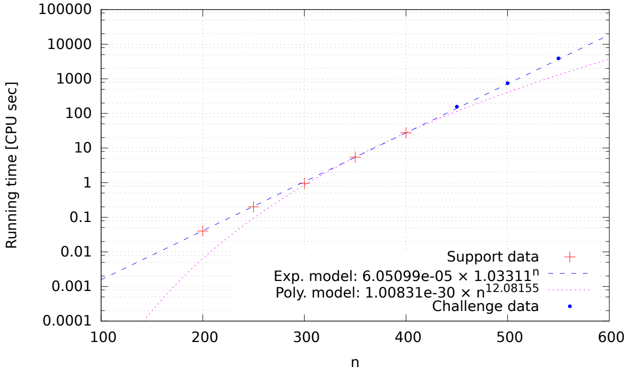
n	450	500	550
median	156.480	750.510	3896.450



Empirical Scaling Results – DPLL-based Solvers

Fit parametric models:

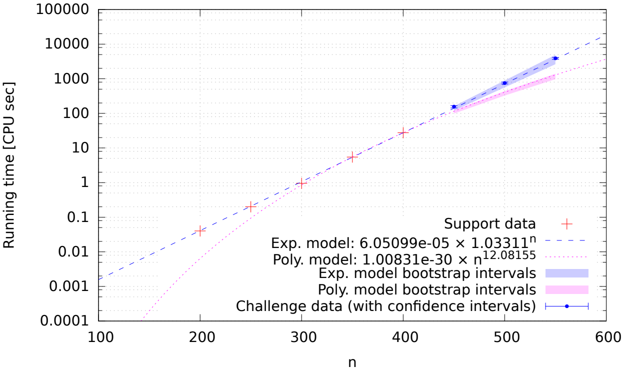
		Model	RMSE (support)	RMSE (challenge)
kcnfs	**Exp. Model**	$\mathbf{4.30400 \times 10^{-5} \times 1.03411^{n}}$	**0.05408**	**143.3**
	Poly. Model	$9.40745 \times 10^{-31} \times n^{12.1005}$	0.06822	1516



Empirical Scaling Results – DPLL-based Solvers

Bootstrap re-sampling:

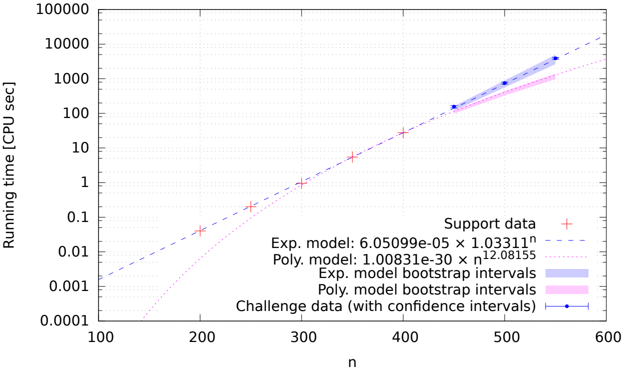
Solver	n	Predicted confidence intervals		Observed median run-time (sec)	
		Poly. model	Exp. model	Point estimates	Confidence intervals
kcnfs	450	[98.326, 122.115]	**[120.078, 161.444]**	156.480	[143.340, 166.770]
	500	[327.997, 439.089]	**[561.976, 889.428]***	750.510	[708.290, 806.130]
	550	[971.862, 1402.255]	**[2622.488, 4901.661]***	3896.450	[3633.630, 4130.915]



Empirical Scaling Results – DPLL-based Solvers

Bootstrap re-sampling:

Solver	Model	Confidence interval of a	Confidence interval of b
kcnfs	Poly.	$[3.33969 \times 10^{-31}, 4.30846 \times 10^{-29}]$	$[11.4234, 12.2674]$
	Exp.	$[3.33378 \times 10^{-5}, 1.07425 \times 10^{-4}]$	$[1.03136, 1.03476]$

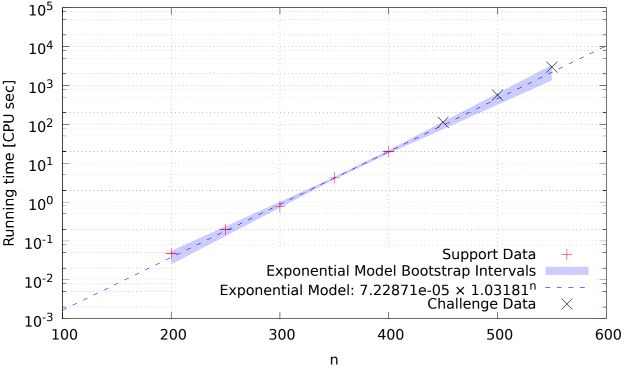


Empirical Scaling Results – DPLL-based Solvers

Compare scaling models:

- No significant difference between two march-variants
- Two march-variants scale significantly better than kcnfs

Scaling models of march_hi:

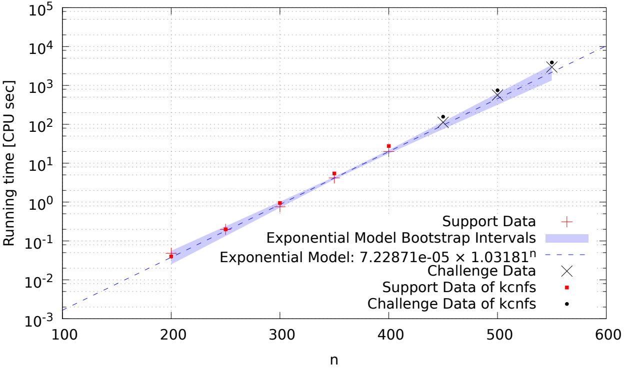


Empirical Scaling Results – DPLL-based Solvers

Compare scaling models:

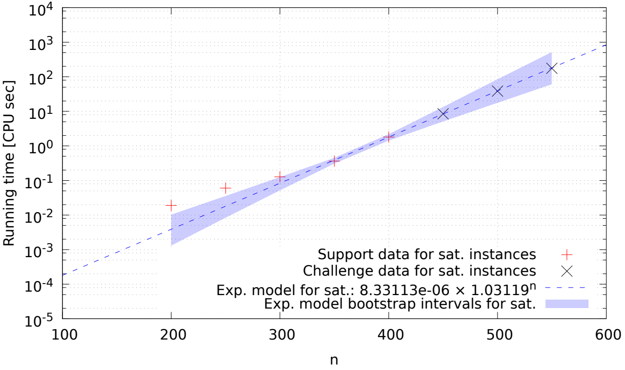
- No significant difference between two march-variants
- Two march-variants scale significantly better than kcnfs

Compare scaing models of kcnfs against march_hi:



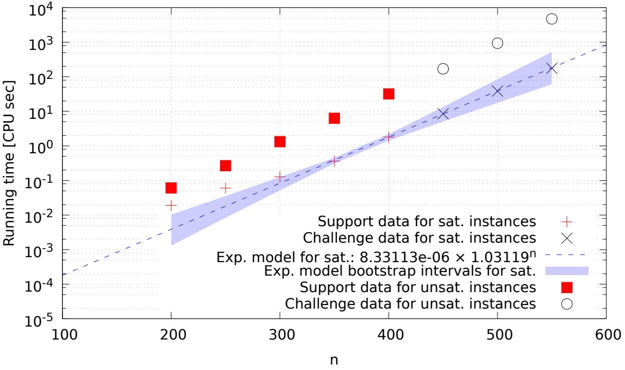
Empirical Scaling Results – DPLL-based Solvers

Difference in solving satisfiable instances and unsatisfiable instances:



Empirical Scaling Results – DPLL-based Solvers

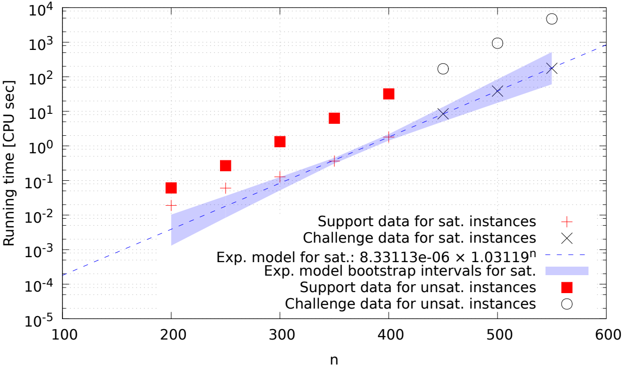
Difference in solving satisfiable instances and unsatisfiable instances:



Empirical Scaling Results – DPLL-based Solvers

Difference in solving satisfiable instances and unsatisfiable instances:

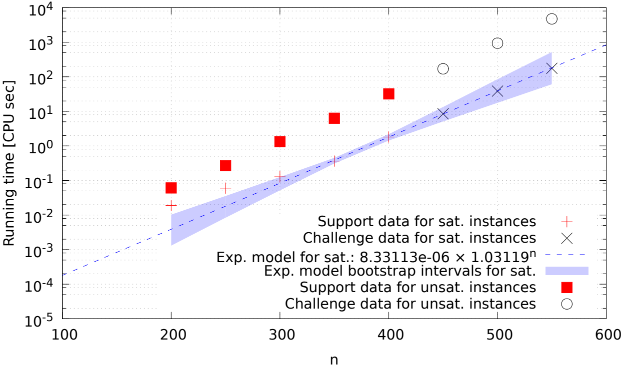
- Is the difference a constant factor?



Empirical Scaling Results – DPLL-based Solvers

Difference in solving satisfiable instances and unsatisfiable instances:

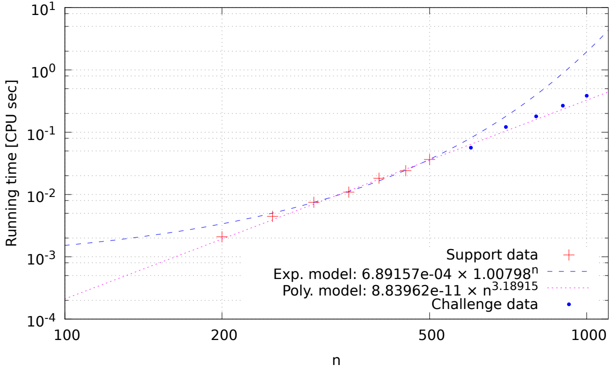
- Is the difference a constant factor?
- Fit running times of solving unsatisfiable instances with model $a \cdot b_{\text{sat}}^n$
  - Slower in solving unsatisfiable instances by constant factor only



Empirical Scaling Results – SLS-based Solvers

Fit parametric models:

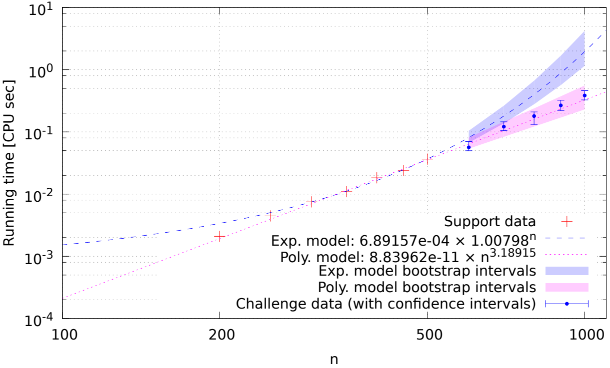
		Model	RMSE (support)	RMSE (challenge)
WalkSAT/SKC	Exp. Model	$6.89157 \times 10^{-4} \times 1.00798^{n}$	0.0008564	0.7600
	Poly. Model	$\mathbf{8.83962 \times 10^{-11} \times n^{3.18915}}$	**0.0007433**	**0.03142**



Empirical Scaling Results – SLS-based Solvers

Bootstrap re-sampling:

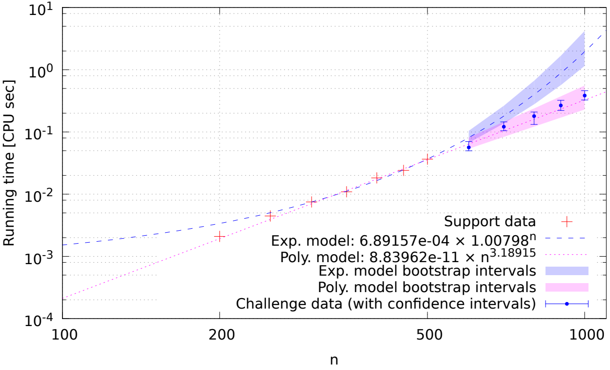
Solver	n	Predicted confidence intervals		Observed median run-time (sec)	
		Poly. model	Exp. model	Point estimates	Confidence intervals
WalkSAT/SKC	600	**[0.054, 0.081]**	[0.067, 0.104]	0.056	[0.050, 0.070]
	⋮	⋮	⋮	⋮	⋮
	1000	**[0.229, 0.557]***	[1.151, 4.200]	0.385	[0.327, 0.461]



Empirical Scaling Results – SLS-based Solvers

Bootstrap re-sampling:

Solver	Model	Confidence interval of a	Confidence interval of b
WalkSAT/SKC	Exp.	$[4.05064 \times 10^{-4}, 1.00662 \times 10^{-3}]$	$[1.00709, 1.00924]$
	Poly.	$[2.58600 \times 10^{-12}, 8.63869 \times 10^{-10}]$	$[2.80816, 3.76751]$



Empirical Scaling Results – SLS-based Solvers

No significant difference among scaling models for WalkSAT/SKC, BalancedZ & probSAT

Empirical Scaling Results – SLS-based Solvers

No significant difference among scaling models for WalkSAT/SKC, BalancedZ & probSAT

Higher quantiles:

- Scaling of 0.75- and 0.9-quantile of running times still consistent with polynomial model

Empirical Scaling Results – SLS-based Solvers

No significant difference among scaling models for WalkSAT/SKC, BalancedZ & probSAT

Higher quantiles:

- Scaling of 0.75- and 0.9-quantile of running times still consistent with polynomial model

Even larger instances:

- Limited experiments on instances of $n \in \{1500, 2000, 5000\}$
- Data consistent with polynomial models

Conclusions

Refined model for location of 3-SAT phase transition

Conclusions

Refined model for location of 3-SAT phase transition

Empirical scaling results for phase-transition random 3-SAT:

- How do running times of high-performance SAT solvers scale?
  - DPLL-based solvers: exponentially; SLS-based solvers: polynomially

Conclusions

Refined model for location of 3-SAT phase transition

Empirical scaling results for phase-transition random 3-SAT:

- How do running times of high-performance SAT solvers scale?
  - DPLL-based solvers: exponentially; SLS-based solvers: polynomially
- Is scaling difference between solvers significant?
  - March-variants scale significantly better than kcnfs
  - No significant differences between SLS-based solvers

Conclusions

Refined model for location of 3-SAT phase transition

Empirical scaling results for phase-transition random 3-SAT:

- How do running times of high-performance SAT solvers scale?
  - DPLL-based solvers: exponentially; SLS-based solvers: polynomially
- Is scaling difference between solvers significant?
  - March-variants scale significantly better than kcnfs
  - No significant differences between SLS-based solvers
- How much faster are complete solvers solving satisfiable instances?
  - Constant factor only

Conclusions

Refined model for location of 3-SAT phase transition

Empirical scaling results for phase-transition random 3-SAT:

- How do running times of high-performance SAT solvers scale?
  - DPLL-based solvers: exponentially; SLS-based solvers: polynomially
- Is scaling difference between solvers significant?
  - March-variants scale significantly better than kcnfs
  - No significant differences between SLS-based solvers
- How much faster are complete solvers solving satisfiable instances?
  - Constant factor only

Methodology applicable to other algorithms, instances and problems

Conclusions

Refined model for location of 3-SAT phase transition

Empirical scaling results for phase-transition random 3-SAT:

- How do running times of high-performance SAT solvers scale?
  - DPLL-based solvers: exponentially; SLS-based solvers: polynomially
- Is scaling difference between solvers significant?
  - March-variants scale significantly better than kcnfs
  - No significant differences between SLS-based solvers
- How much faster are complete solvers solving satisfiable instances?
  - Constant factor only

Methodology applicable to other algorithms, instances and problems

- ESA: automated tool for scaling analysis (Mu and Hoos, 2015)
  `www.cs.ubc.ca/labs/beta/Projects/ESA/esa-online.html`

References

P. Cheeseman, B. Kanefsky, and W. M. Taylor. Where the really hard problems are. In *IJCAI*, pages 331–337, 1991.

C. Coarfa, D. D. Demopoulos, A. San Miguel Aguirre, D. Subramanian, and M. Y. Vardi. Random 3-SAT: The plot thickens. *Constraints*, 8(3):243–261, 2003.

S. A. Cook. The complexity of theorem-proving procedures. In *STOC*, pages 151–158. ACM, 1971.

J. M. Crawford and L. D. Auton. Experimental results on the crossover point in random 3-SAT. *Artificial Intelligence*, 81(1):31–57, 1996.

I. P. Gent and T. Walsh. Towards an understanding of hill-climbing procedures for SAT. In *AAAI*, pages 28–33, 1993.

I. P. Gent, E. MacIntyre, P. Prosser, and T. Walsh. The scaling of search cost. In *AAAI*, pages 315–320, 1997.

H. H. Hoos and T. Stützle. On the empirical scaling of run-time for finding optimal solutions to the travelling salesman problem. *European Journal of Operational Research*, 238(1):87–94, 2014.

H. H. Hoos. A bootstrap approach to analysing the scaling of empirical run-time data with problem size. Technical report, TR-2009-16, Dept. of Computer Science, Univ. of British Columbia, 2009.

S. Mertens, M. Mézard, and R. Zecchina. Threshold values of random k-SAT from the cavity method. *Random Structures and Algorithms*, 28(3):340–373, 2006.

D. Mitchell, B. Selman, and H. Levesque. Hard and easy distributions of SAT problems. In *AAAI*, pages 459–465, 1992.

Z. Mu and H. H. Hoos. Empirical scaling analyser: An automated system for empirical analysis of performance scaling. In *GECCO Companion '15*, pages 771–772, 2015.

M. Yokoo. Why adding more constraints makes a problem easier for hill-climbing algorithms: Analyzing landscapes of CSPs. In *CP*, pages 356–370. Springer, 1997.