Cornell University

Survey Propagation Revisited,
or where is all the satisfaction coming from

Lukas Kroc, Ashish Sabharwal, Bart Selman

Intelligent Information Systems Institute
CORNELL

Uncertainty in Artificial Intelligence
July 2007

Talk Outline

- Introduction
  - What is Survey Propagation (SP)?
- Some details of SP
  - How is SP related to BP?
- Empirical Studies
  - Why SP works so well?

Introduction

Survey Propagation* is
a **message passing** algorithm used by
a **decimation procedure** to solve
large instances of hard **random k-SAT problems**

*[Mezard, Parisi, Zecchina ‘02]

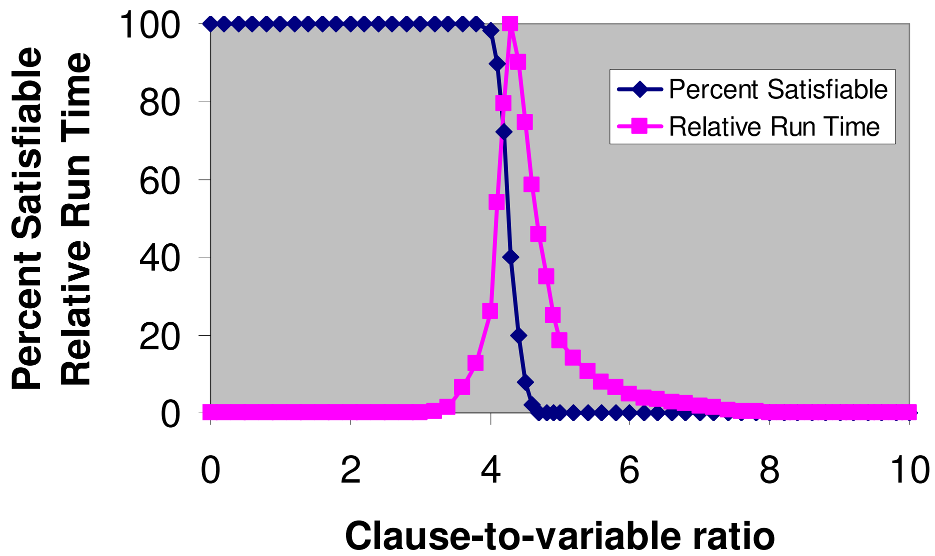
Random k-SAT Problem

- k-SAT Problem: Satisfiability of Boolean formula in CNF
  - *n* vars∈ {0,1}≡{F,T}, *m* clauses, each clause has *k* literals

$$F = \underbrace{(\neg x \vee y \vee z)}_{A} \wedge \underbrace{(x \vee \neg y \vee z)}_{B} \wedge \underbrace{(x \vee y \vee \neg z)}_{C}$$

  - Is F satisfiable? Find a solution.
    - Canonical NP-complete problem

- Random problem: clauses chosen at random
  - Phase transition, as a function of clause-to-variable ratio $\alpha = \frac{m}{n}$
  - For 3-SAT, $\alpha_{thr} \approx 4.26$



Decimation Procedure

Given some ordering of the literals

Do:

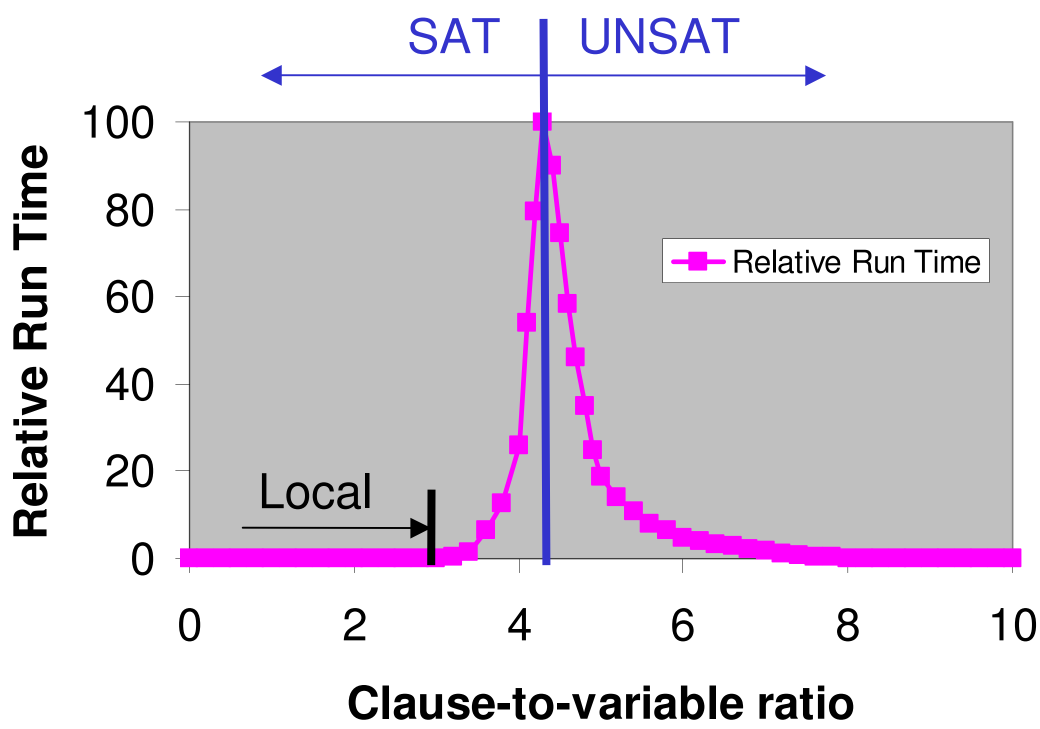
1. Assign the first literal its value
2. Simplify the formula
3. Recompute the ordering and repeat

- Very scalable!
- No repair mechanism $\Rightarrow$ the ordering must be “smart” to find a solution

Where to get “the smart ordering” from?

Local Counting Heuristic

- Used in standard DPLL solvers for SAT
- Variants of: “set the most frequently occurring variable based on majority vote”
- Fast, easy to compute, but not very powerful



SAT as an Inference Problem

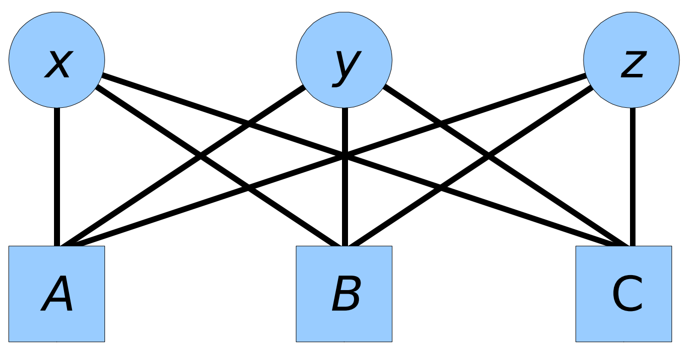
- Compute marginal probabilities:

  Pr[*x*=True | a solution]

  – Use magnetization = Pr[*x*=T|solution] - Pr[*x*=F|solution]

- Inference in a Bayesian Network:

$$F = \underbrace{(\neg x \vee y \vee z)}_{A} \wedge \underbrace{(x \vee \neg y \vee z)}_{B} \wedge \underbrace{(x \vee y \vee \neg z)}_{C}$$

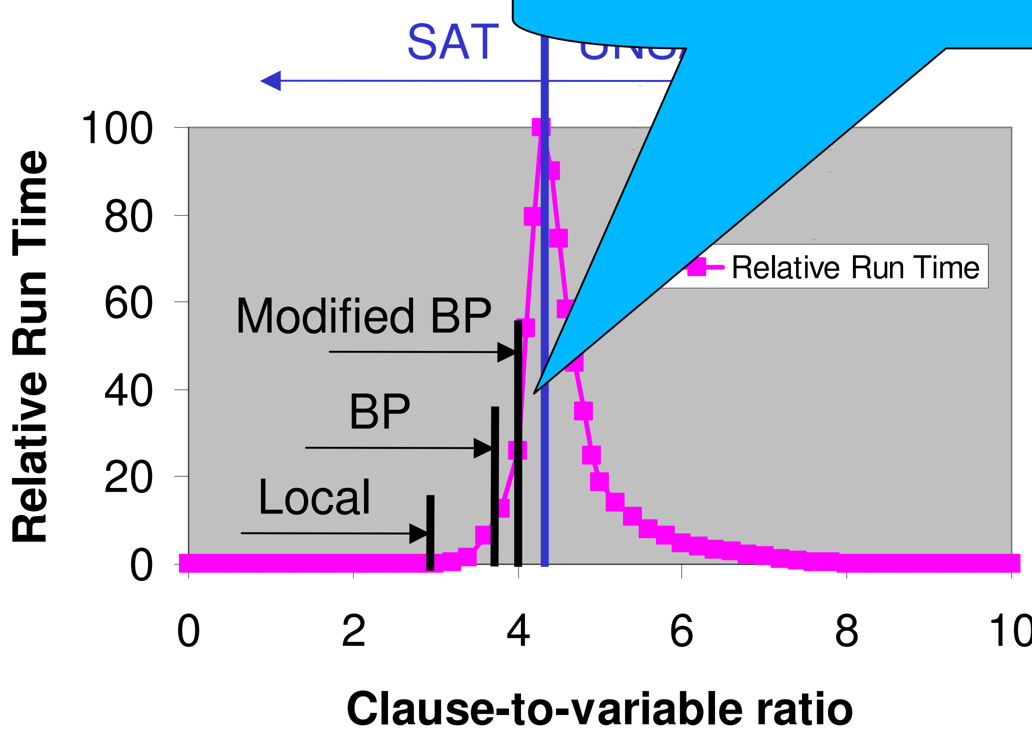


  – More difficult than SAT!

SAT as an Inference Problem

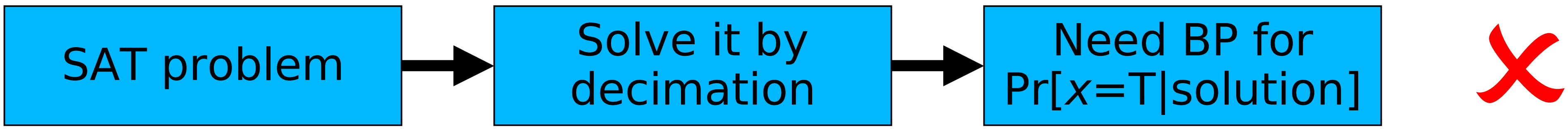
- Marginals can be approximated using Belief Propagation:
  - Message Passing: numbers for each (clause,var) pair governed by a system of equations.
  - Scalable, but does not work for hard enough problems due to loops in the factor graph
    - Both convergence and accu

Want to close the gap!

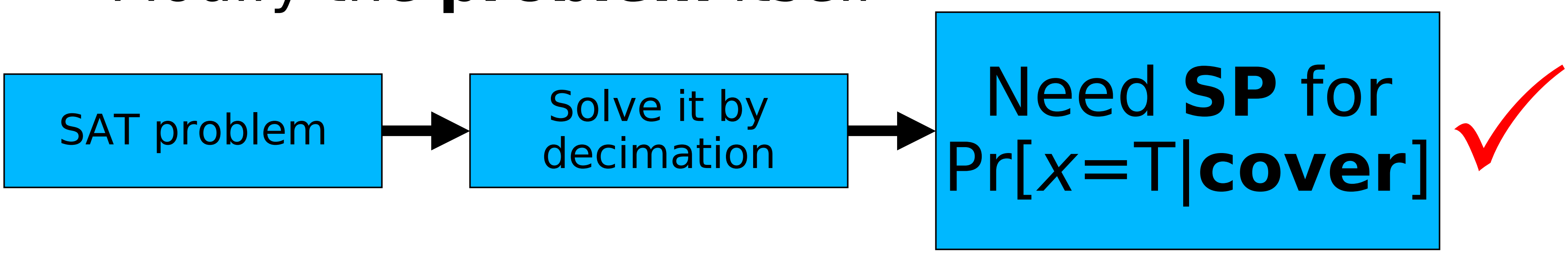


Substituting SP for BP

- BP-inspired-decimation does not work for hard random problems



- For more difficult random SAT problems, use SP-inspired-decimation
  - Modify the **problem** itself



Talk Outline

- Introduction
  - What is Survey Propagation (SP)?
- **Some details of SP**
  - Introduction to covers
  - SP, BP and covers
- Empirical Studies
  - Why SP works so well?

Introduction to Covers

- What are covers?
  - Generalized {0,1,*} assignments (* means "undecided") such that

1. Every clause has a satisfying literal or ≥ 2 *s
2. Every non-* variable has a "certifying" clause in which all other literals are false

- e.g. F has covers (***) and (000)

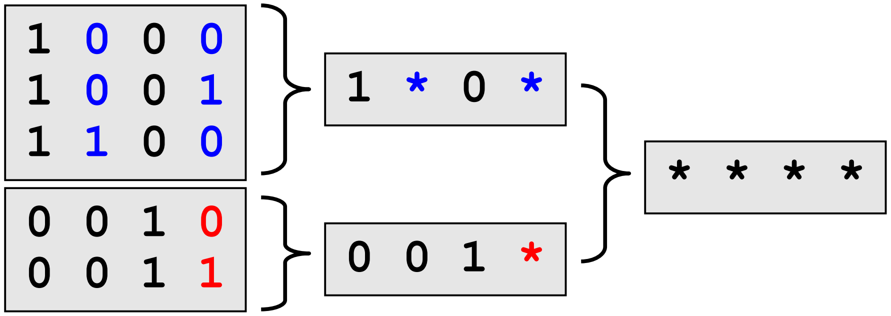
$$F = (\neg x \vee y \vee z) \wedge (x \vee \neg y \vee z) \wedge (x \vee y \vee \neg z)$$

1.	(1 0 0)	(0 1 0)	(0 0 1)
2.	x must be 0	y must be 0	z must be 0

Properties of Covers

- Every formula (sat or unsat) without unit clauses has the trivial cover, ***

- Covers represent clusters of solutions
  - * generalizes both 0 and 1
  - solutions that differ in one bit are represented by the same cover
  - Some covers may not generalize any solution



- Unlike finding solutions, finding covers is not a self-reducible problem
  - covers cannot be computed by simple decimation

SP, BP and Covers

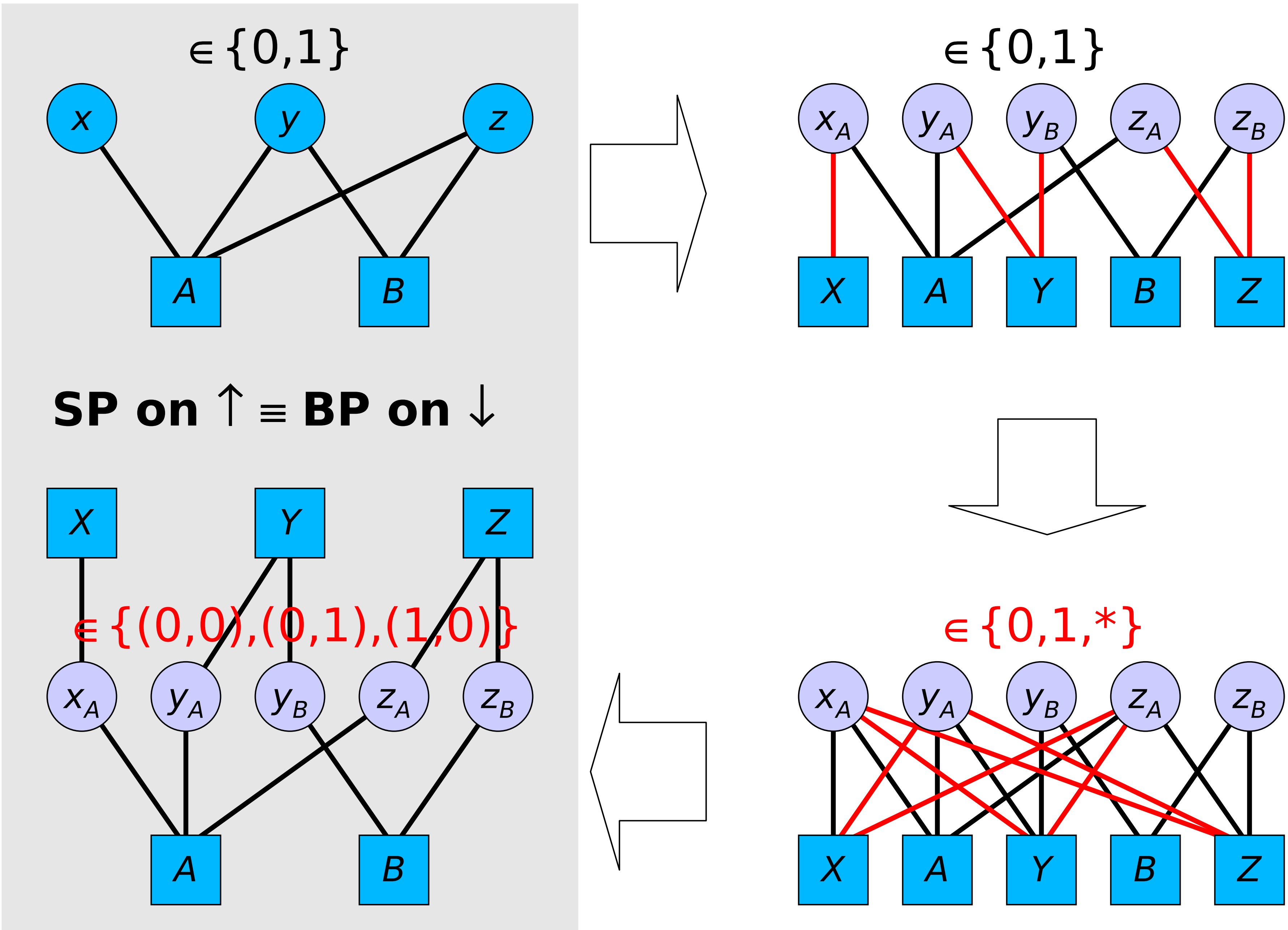
- SP is Belief Propagation on the Cover Problem [Braunstein, Zecchina '03; Maneva, Mossel, Wainwright '04]
  - Derivation via CSP modification in the paper

SP must compute a loopy approximation to cover marginals

- Covers provably exist in k-SAT for k≥9 [Achlioptas, Ricci-Tersenghi '06]

Believed not to exist in random 3-SAT, because previous studies could not find them

Derivation of the Cover Problem



What is Left to Do?

- Do non-trivial covers exist in random 3-SAT?
  - Because if they do not, what is SP doing then?
- Can SP compute cover marginals?
  - Because if it does not, what good is its output?
  - Similar topology of the factor graph as the solution problem, same algorithm applied!
    - We know BP does not work on the solution problem
- How do cover marginals relate to solutions?
  - Because if they do not, why does the decimation based on covers work?

Talk Outline

- Introduction
  - What is Survey Propagation (SP)?
- Some details of SP
  - Introduction to covers
  - SP, BP and covers
- **Empirical Studies**

  1) Empirical evidence that **covers do exist** in large random 3-SAT formulas

  2) **SP computes cover marginals** remarkably well

  3) Cover marginals **correlate well with solution marginals**

Part 1

Do non-trivial covers exist in random 3-SAT?

Can SP compute cover marginals?

How do cover marginals relate to solutions?

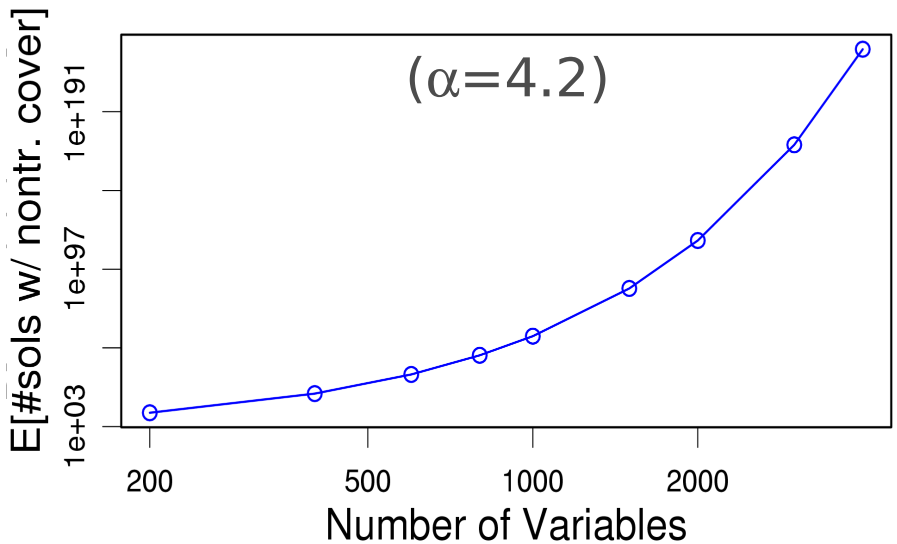
Local Search for Covers

Algorithm inspired by the "peeling-procedure" [MMW'04]:

(a) Sample a solution using SampleSat

(b) *-propagate to a cover (turn every uncertified 0 or 1 into a * until no such variable)

How often do solutions *-propagate to non-trivial covers?



⇒ Expected no. of solutions *-propagating to a non-trivial cover increases exponentially with *n*

Part 2

Covers do exist in random 3-SAT

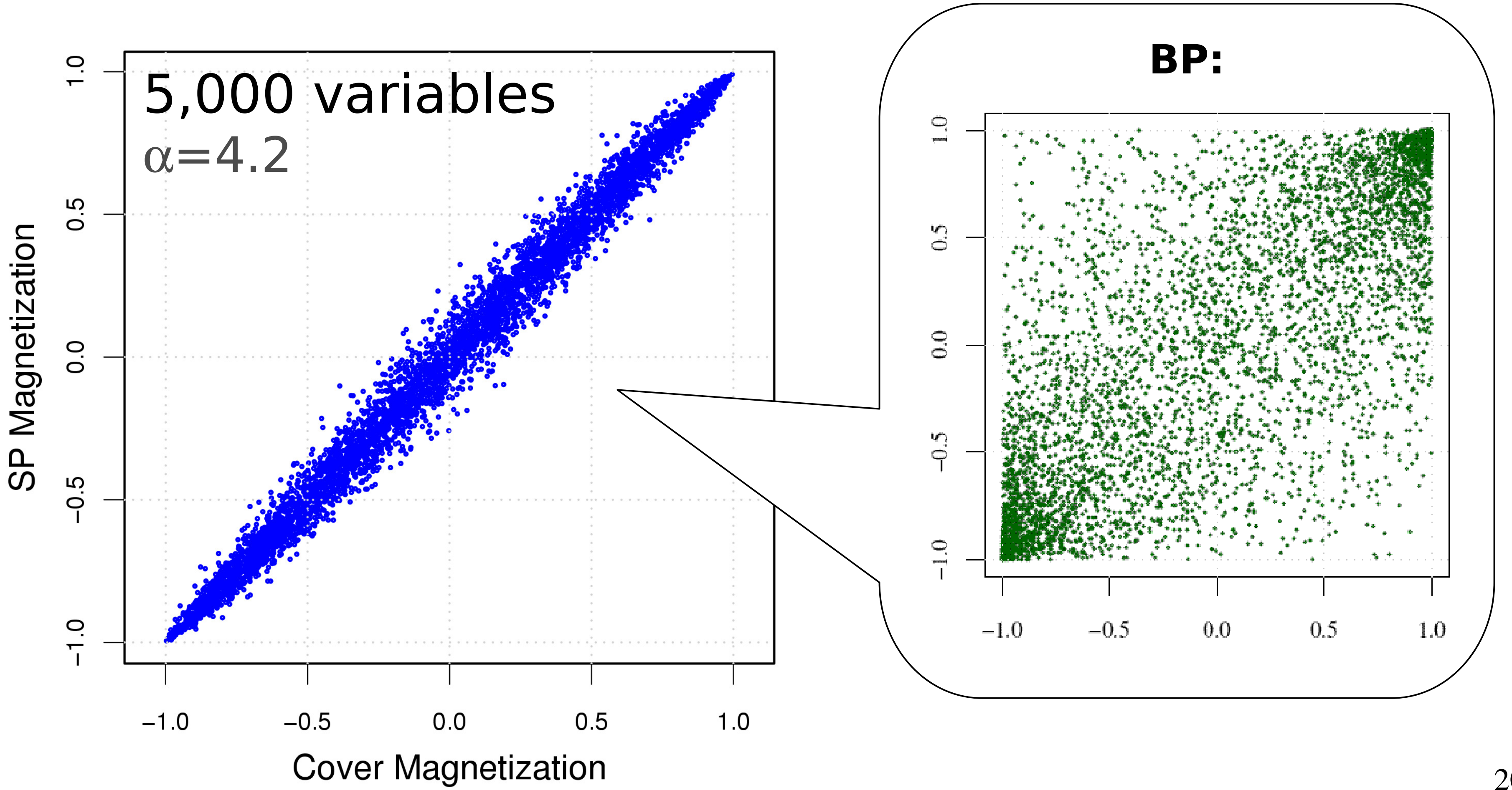
Can SP compute cover marginals?

How do cover marginals relate to solutions?

Covers vs. SP

Experiment:

1. sample many covers using local search
2. compute cover magnetization from samples (x-axis)
3. compare with SP magnetization (y-axis)



Part 3

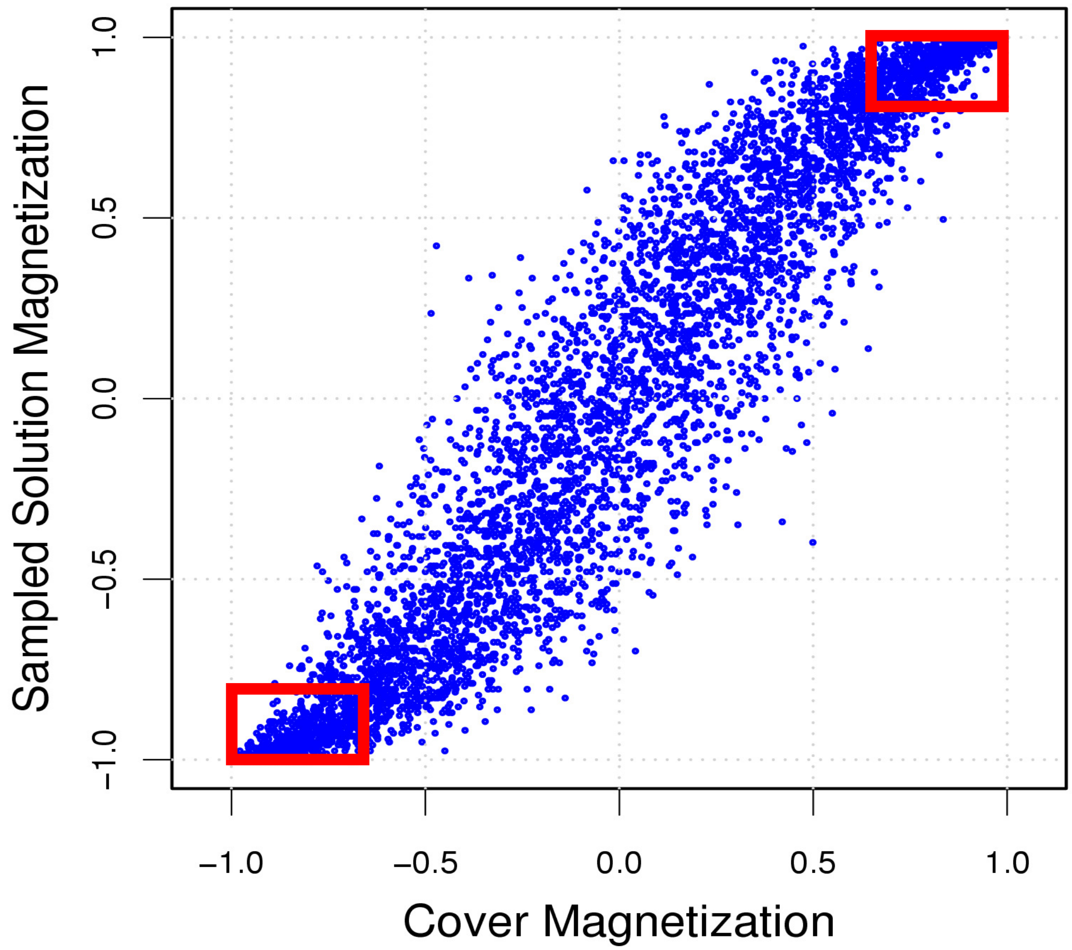
Covers do exist in random 3-SAT

SP is good at computing cover marginals

How do cover marginals relate to solutions?

Covers vs. Solutions

5,000 variables (α=4.2)



Cover marginals appear to be more conservative than (sampled) solution marginals

Covers do exist in random 3-SAT

SP is good at computing cover marginals

Cover marginals correlate well with solutions

Conclusions

- Survey Propagation-inspired decimation relies on inference about **covers** of a formula
  - Which marginals it computes very well, and they are even safer to use than solution marginals
- **Modifying the problem** is more successful than modifying the inference algorithm for the random SAT problem
- Open issues:
  - Why is BP on covers so good while BP on solutions is not?
  - Can the problem-modification be applied to non-random problems?
  - .....

Properties of Covers II

- Unlike finding solutions, finding covers is not a self-reducible problem

  $\Rightarrow$ covers cannot be computed by simple decimation

e.g. if we guess that in some cover x=0, and use decimation:

$$F = (\neg x \vee y \vee z) \wedge (x \vee \neg y \vee z) \wedge (x \vee y \vee \neg z)$$

$$F' = (\neg y \vee z) \wedge (y \vee \neg z)$$

(11) is a cover for F'

but (011) is not a cover for F

Searching for Covers

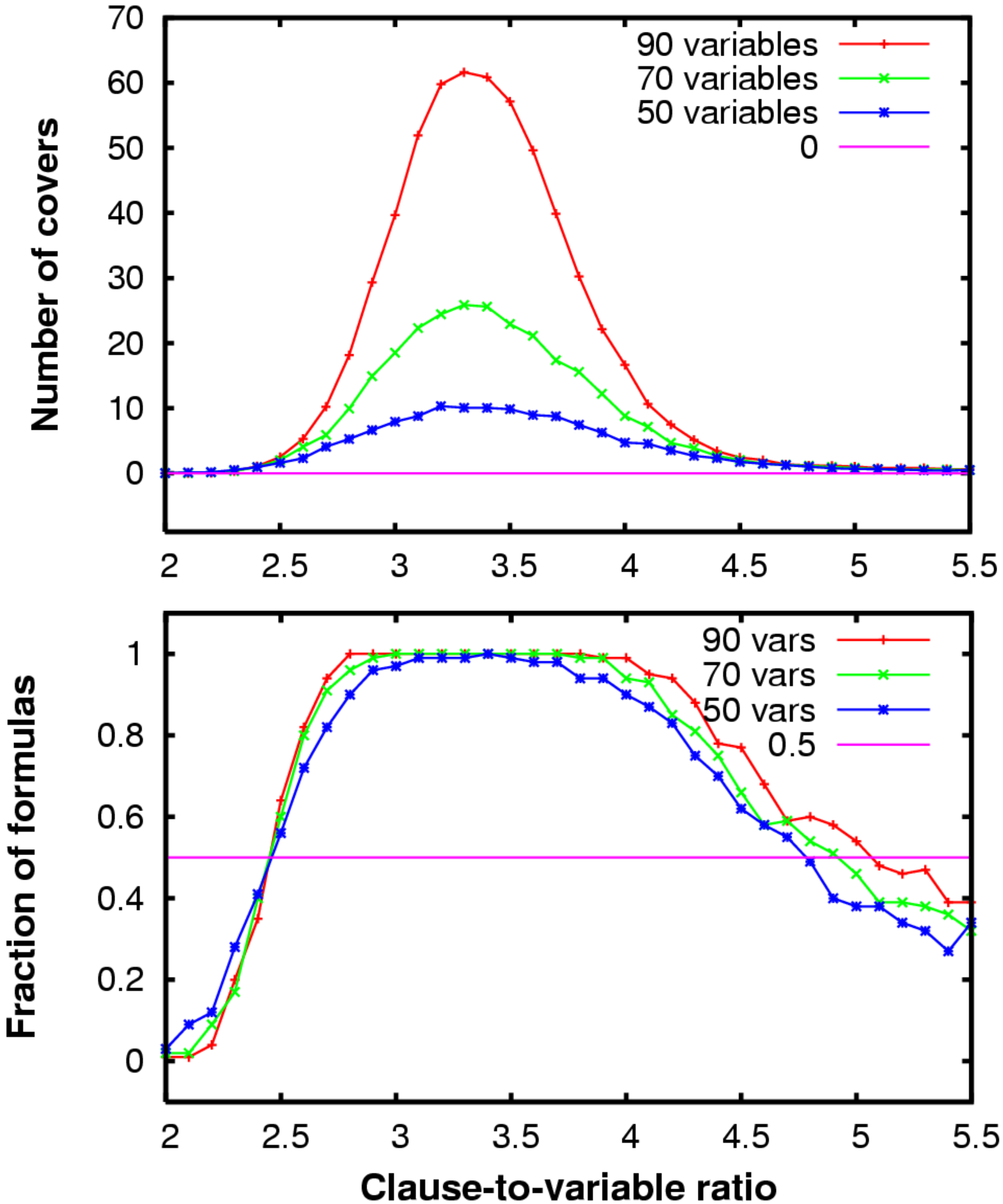
- Using an appropriate SAT encoding
  - Create a new formula whose solutions represent covers of the original formula
  - Can enumerate all covers
  - Not scalable (up to $n \sim 100$ variables)

- Using local search on the original formula
  - Scales well (can find true covers for n=20K)
  - Algorithm inspired by the "peeling-procedure" [Maneva, Mossel, Wainwright '04] :

    (a) Sample a solution using SampleSat

    (b) *-propagate to a cover (turn every uncertified 0 or 1 into a * until no such variable)

SAT Encoding of Covers



- Number of covers grows with *n*
- Covers are relatively few
  e.g. ~10 covers vs.
  150K solutions
  for N=90 at α=4.2
- Phase transition near α = 2.5
- For larger *n*, covers exist for a broader range of α

Part 4

Covers exist in random 3-SAT

SP is good at computing cover marginals

Cover marginals correlate well with solutions

Can BP/SP be used on non-random instances?

BP/SP on Non-random Formulas

- SAT solving by decimation relies heavily on marginals
  - Mistakes can be fatal
  - SP does not work on anything but random formulas

$\Rightarrow$ More natural application:

Counting number of solutions

Counting With BP

- BPcount = marginal estimation + solution search
  - Quality of marginals $\propto$ Quality of the count
  - (damped) BP gives reasonable estimates
- Results

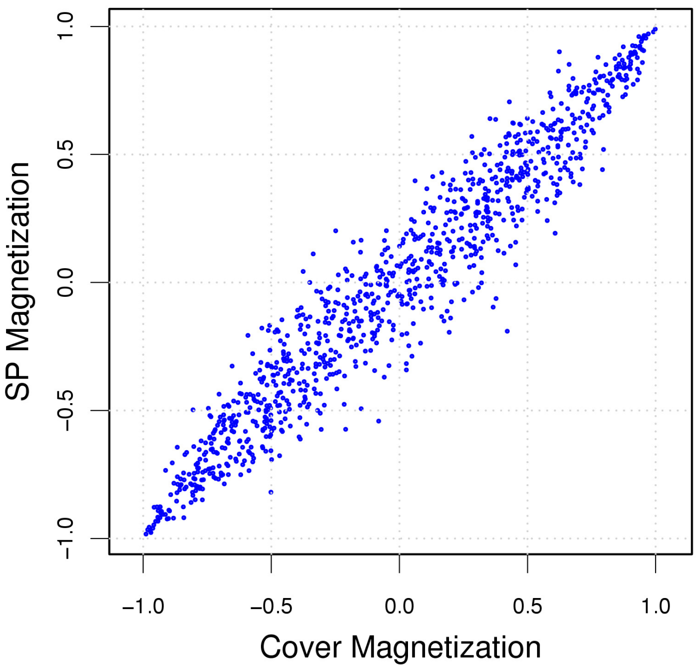
Problem	Exact Count	**BPcount**	Random margs.
2bitmax	10^29	**10^28**	10^26
LatinSquare8	10^11	**10^11**	10^7
Langford15	10^7	**10^6**	10^3

$\Rightarrow$ BP provides useful info about marginals

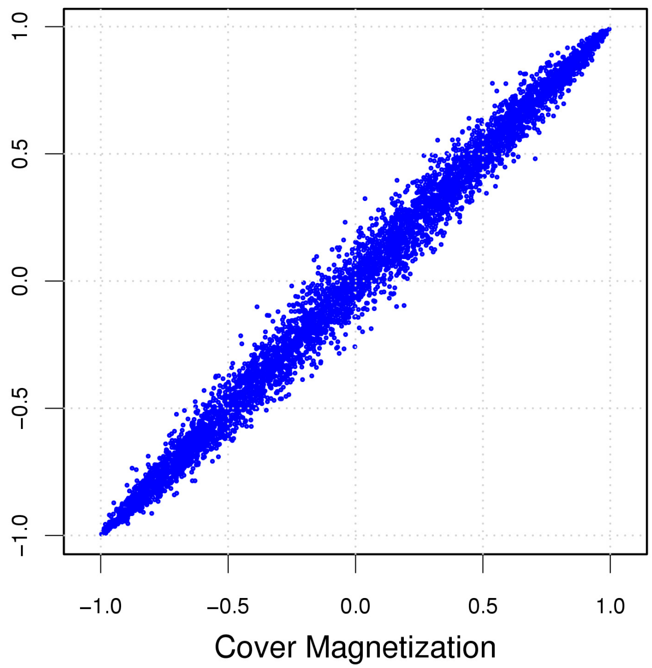
Covers vs. SP

(α=4.2)

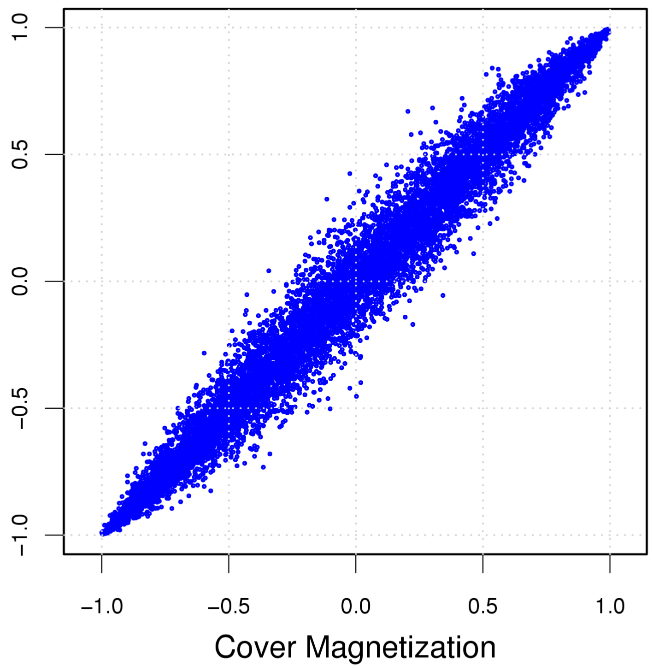
1,000 variables



5,000 variables



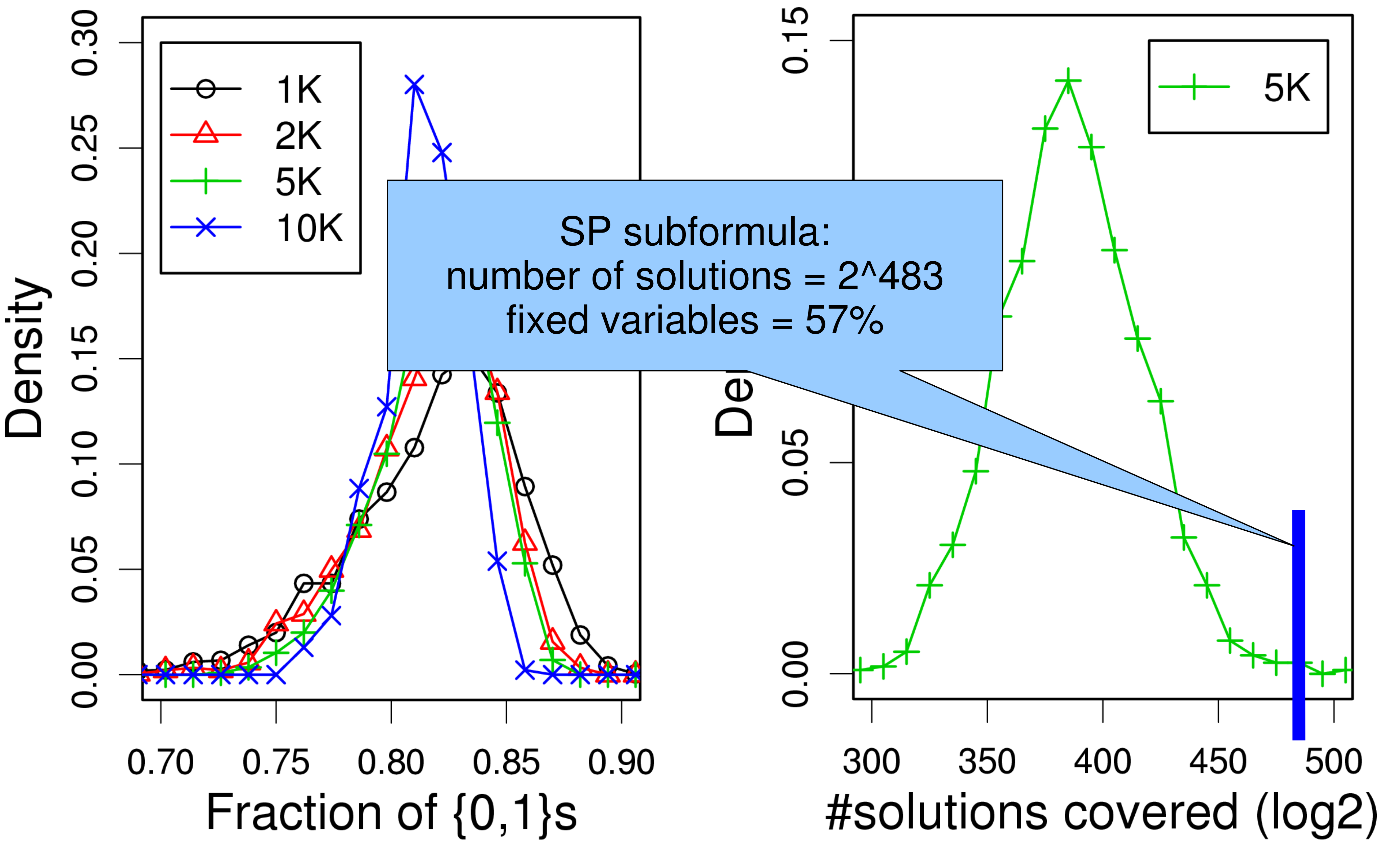
10,000 variables



500 covers per plot, took up to 1200 CPU hrs

SP is surprisingly accurate at computing cover marginals!

Covers Are Long
Clusters Are Smalls



Remark About Covers

- Why BP on covers (SP) works, while BP on solutions does not?
  - Covers are far apart in Hamming distance
  - Covers are (relatively) few
- Modified problem:
  - Adding an XOR constraint that requires all solutions to have even number of 1s
  - Creating many covers close to each other prevents SP from converging

SAT Encoding of Covers