Effective Conditions for Convergence of the Warning Propagation Algorithm on WP-Solvable Formulas (Postprint)

Authors: Cui Li, Wang Xiaofeng, Niu Jin

Date: 2019-04-01T00:00:00+00:00

Abstract

Message passing algorithms are highly effective for solving satisfiability problems, and the Warning Propagation (WP) algorithm is the most fundamental among them. Through analysis of the mathematical principles of the WP algorithm, we find that partial assignments determined with high probability are closely related to the backbone set and backdoor set of the formula. In studying the convergence properties of the WP algorithm, we define WP-solvable formulas based on the backbone set and backdoor set, prove the convergence of the WP algorithm under the G(n, 3, m) model and the planted assignment model, and present the necessary and sufficient conditions for algorithm convergence. Finally, through numerical experimental verification on the planted assignment formula generation model, the results demonstrate that a satisfiability formula is WP-solvable if and only if the WP algorithm converges with high probability.

Full Text

Preamble

Effective Conditions for Warning Propagation Algorithm Convergence on WP-Solvable Formulas

Cui Li, Wang Xiaofeng, Niu Jin

(School of Computer Science & Engineering, North Minzu University, Yinchuan 750021, China)

Abstract: Message passing algorithms are highly effective for solving satisfiability problems, with the Warning Propagation (WP) algorithm being the most fundamental. Through mathematical analysis of the WP algorithm’ s principles, we find that variables determined with high probability are closely related to the backbone set and backdoor set of the formula. To study WP algorithm convergence, we define WP-solvable formulas based on backbone and backdoor sets, and prove convergence of the WP algorithm under both the G(n, 3, m) model and planted distribution model, providing necessary and sufficient conditions for algorithm convergence. Numerical experiments on the planted distribution formula generation model verify that a satisfiability formula is WP-solvable if and only if the WP algorithm converges with high probability.

Keywords: warning propagation algorithm; backbone; backdoor; WP-solvable formula; instance generation model

---

0 Introduction

Constraint Satisfaction Problems (CSP) constitute an important research area in artificial intelligence, with the Satisfiability (SAT) problem being a typical CSP. The NP-completeness of SAT indicates that no polynomial-time algorithm exists for solving it. However, many real-world complex problems can be encoded as SAT problems, such as robot path planning, intelligent system knowledge reasoning, and large-scale complex system control. Although SAT solvers have become increasingly intelligent, narrowing the hard region and even solving difficult instances near phase transition points in polynomial time, research remains NP-hard. As studies deepen and hardware develops, SAT research focuses on two main aspects.

First, constructing instance generation models to analyze phase transition phenomena. The most representative is the random 3-SAT instance generation model G(n, 3, m), where the clause-to-variable ratio = m/n is a crucial parameter affecting both instance satisfiability and decision difficulty. Statistical phenomena show that for random 3-SAT, there exists a satisfiability phase transition threshold _d. When < _d, instances are highly probable to be satisfiable; when > _d, they are highly probable to be unsatisfiable. This critical phenomenon between satisfiability and unsatisfiability is called a phase transition, with _d being the phase transition point. Instances near _d are particularly difficult to solve. While the exact value of _d remains unknown, research shows it is at least 3.52 and at most 4.4898. In this model, controlling parameter constructs instances, with extensive experimental studies showing maximum difficulty around 4.27. Xu et al. proposed the RB and RD models with exact phase transition points based on B and D models, addressing the trivial unsatisfiability problem in classic SAT instance generation models and widely used for constructing hard instances of SAT and CSP.

Second, designing more effective SAT decision algorithms. Modern SAT solvers exploit special hidden structures within instances, significantly improving search capability. Backbone sets and backdoor sets are the most important structures in SAT instances, playing key roles in algorithm performance. Monasson and

Williams introduced backbone and backdoor concepts when studying SAT phase transitions and complexity.

Both backbone and backdoor sets correlate with problem difficulty: larger backbone sets indicate greater difficulty, as do larger backdoor sets. For structured instances, backbone and backdoor sets have certain relationships. Many real-world structured SAT problems have small backbone and backdoor sets, which famous search algorithms exploit to solve practical problems more effectively than random SAT.

In SAT algorithm research, hidden structures affect algorithm performance, with backbone and backdoor sets being the most important. The backbone set is a set of literals that are true in every satisfying assignment of a satisfiable formula. The backdoor set is a set of variables whose assignment simplifies the formula into an easily solvable class (e.g., Horn formulas). Backbone size correlates with difficulty: intuitively, larger backbone sets increase solution clustering, reduce flexible assignment variables, increase incorrect assignment probability, and raise difficulty for local search algorithms, explaining why formulas near the satisfiability phase transition are hard. Similarly, larger backdoor sets increase decision difficulty. Backdoor sets depend on the solving algorithm, while backbone sets are unique to each SAT instance. By definition, backbone and backdoor sets have no necessary connection, with minimal overlap.

In the 1980s, physicists proposed message-passing algorithms based on information propagation, which proved highly effective for combinatorial optimization and have been widely applied in artificial intelligence and engineering. Three message-passing algorithms for random SAT were designed: Warning Propagation (WP), Belief Propagation (BP), and Survey Propagation (SP). WP can solve random instances where the clause-to-variable ratio is below a certain threshold. However, when > 3.5, WP often fails to converge. While experiments show this phenomenon, systematic theoretical explanations are lacking, making convergence analysis crucial. Message-passing algorithms can determine partial variable assignments with high probability, simplifying formulas. Repeated operations may reduce formulas to easily solvable forms, at which point polynomial-time SAT algorithms can solve them. WP "freezes" certain variables that are forced to fixed values with probability 1, meaning some clauses' satisfiability depends entirely on these frozen variables. This paper defines WP-solvable formulas through backbone and backdoor sets, analyzes their structure, and presents effective conditions for WP algorithm convergence on such formulas.

---

1 Special Variable Sets in Propositional Formulas

Backbone: Let F be a CNF formula with variable set var(F) = $\{x , \cdots, x \}$. A variable subset S var(F) is called a backbone set if for every satisfying assignment of F, the variables in S have fixed values. Equivalently, for any satisfying assignment, the literals corresponding to variables in S are true. The literals formed by taking positive (or negative) literals from each variable in S, when conjoined, yield a formula that must be satisfied.

Backdoor Set: Let F be a CNF formula with variable set var(F) = {x , ···, x }, and C an easily solvable class (where satisfiability is polynomial-time decidable). A variable subset S var(F) is a weak backdoor set if there exists an assignment to variables in S that simplifies F to a formula in C. It is a strong backdoor set if for every assignment to variables in S, the simplified formula belongs to C. Generally, S is a weak backdoor if some assignment reduces F to an easily solvable formula, and a strong backdoor if all assignments do so.

In Warning Propagation (WP) algorithm, messages take values 0 or 1. A message of 1 from clause node c to variable node x indicates that all other variables in c are forced with probability 1, making c ' s satisfiability depend entirely on x —effectively freezing x ' s value. For WP, formulas whose factor graphs are trees (tree formulas) are easily solvable. This mechanism reveals a close relationship between backbone sets and backdoor sets relative to easily solvable classes. Larger backbone sets increase the likelihood of simplifying formulas to easily solvable forms; smaller backbone sets reduce WP' s effectiveness per iteration. WP' s difficulty on certain formulas correlates with backbone and backdoor set sizes and distributions, suggesting that WP's frozen variables relate to backbone and backdoor variables. This motivates the WP-solvable concept.

1.1 WP-Solvable Formulas

A CNF formula F is WP-solvable if there exists a variable set S var(F) satisfying: 1. S is a backbone set of F, corresponding to a partial assignment on literals over S 2. Under , the simplified formula F| has a backbone set S 3. F| is an easily solvable formula

Clearly, tree formulas are WP-solvable. A 3-CNF formula is a conjunction of clauses, each being a disjunction of three literals (a variable or its negation). A formula is satisfiable if each clause contains at least one true literal under some Boolean assignment.

1.2 Planted Distribution

Recent years have seen extensive research on algorithmic theory of random structures. Random 3-SAT exhibits a clear threshold in clause-to-variable ratio: below the threshold, formulas are highly probable to be satisfiable (whp); above, unsatisfiable. The exact threshold _d is unknown—bounds suggest 3.42 _d 4.5. At these ratios, most formulas are unsatisfiable, making analysis difficult. Therefore, we focus on the planted distribution model.

The planted distribution generates satisfiable instances by first selecting a random assignment uniformly from {0,1} , then including each satisfied clause with probability p. Formally, from n variables, generate all possible clauses, each containing k variables in random order. In the planted distribution model denoted planted_{n,p}, instances are generated from a random truth assignment: each satisfied clause is included independently with probability p. In this work, we consider $p = d/n^2$ where d is a sufficiently large constant, focusing on random 3-CNF formulas that are whp satisfiable.

When p is sufficiently large, planted instances whp have only one satisfying solution. In WP algorithm, messages (called warnings) indicate a clause' s satisfiability preference for variable assignments. WP' s message update equations are:

[Figure 1: see original paper]

2.1 Factor Graph

A CNF formula F with n variables {x , ···, x } can be represented by a bipartite graph G = (C X, E) called a factor graph. Variable nodes X = {1, 2, ···, n} correspond to variables; clause nodes C = {C , C , ···, C } correspond to clauses. Edges are of two types: solid edges (C , x) E if clause C contains positive literal x ; dashed edges if C contains negative literal ¬x . SAT problems convert to factor graphs where each variable node (circle) connects to function nodes (squares) when the variable appears in a clause. Solid lines represent positive literals; dashed lines represent negative literals.

2.2 WP Algorithm

A CSP is a triple (X, D, C) where X = {x , ···, x } are variables, D are domains, and C are constraints. SAT is a typical CSP: given a CNF formula F, determine if a satisfying assignment exists. The WP algorithm proceeds as:

Warning Propagation (CNF formula F) 1. Construct factor graph G(F) 2. Initialize all edge messages to 0 3. Repeat until convergence (or max iterations): - Randomly permute edges - Update messages using equation (1) - Compute partial assignment from messages - Simplify formula F using

3 Convergence Conditions for WP Algorithm

3-SAT is NP-hard; under P NP, no polynomial-time algorithm exists. Message-passing algorithms are currently the most effective for SAT, but near phase transition points they often fail to converge. Theoretical convergence analysis remains limited and difficult.

For any satisfying assignment of F, variables in backbone set S have fixed values. In backdoor sets, assignments simplify F to varying degrees, eventually reaching an easily solvable formula. The relationship between backbone and backdoor sets falls into four categories (Figure 2 [Figure 2: see original paper]): (1) no intersection; (2) partial overlap; (3) backdoor set contained in backbone set; (4) backbone set contained in backdoor set. The overlapping variables are called core variables; clauses they satisfy are supporting clauses.

Based on WP-solvable formulas, we focus on case (3). When WP converges, it yields partial assignments that whp satisfy the WP-solvable formula. Changing these assignments makes planted instances whp unsatisfiable, indicating WP' s determined variables are whp backbone variables. We therefore have:

Theorem 1: A CNF formula F is WP-solvable if and only if the WP algorithm converges with high probability.

To prove Theorem 1, we need Theorems 2 and 3:

Theorem 2: If WP converges, then CNF formula F is WP-solvable.

Proof: We analyze case (3). Satisfiability problems are NP-hard; 3-CNF formulas cannot be solved in polynomial time. WP heuristically assigns variables and simplifies formulas iteratively to find satisfying assignments. Let B be the backbone variable set. If WP converges to fixed point , then for a subset of variables, the satisfiability of certain clauses depends entirely on those variables' values. If a clause becomes unit, WP' s convergence implies the formula is satisfiable. For any satisfying assignment, literals corresponding to variables in the backbone set are critical. Thus, WP' s converged assignments whp identify backbone variables, which are contained in the satisfying assignment. Since WP-solvable formulas simplify to easily solvable formulas, convergence implies WP-solvability.

Theorem 3: If CNF formula F is WP-solvable, then WP converges with high probability.

Proof: Analyzing case (3) from Figure 2. In the planted distribution model, when d is sufficiently large, a satisfying assignment whp exists. By WP-solvable definition, backbone variables are whp contained in . In planted_{n,p} with random uniform selection, when d is large enough, the set of core variables whp includes all variables. Since core variables belong to the backbone set, they share features with backbone variables in the distribution. Tree formulas are clearly WP-solvable. For WP-solvable F, the partial assignment under the backbone set simplifies F to an easily solvable formula. In planted_{n,p}, when d is large enough, there exists satisfying F. WP convergence whp identifies backbone variables contained in . Literature [24] analyzed WP convergence on planted formulas, giving sufficient conditions: for random 3-CNF with $p = d/n^2$ and sufficiently large d, WP converges in O(log n) iterations, producing partial assignment where assigned variables A are backbone variables and unassigned variables U form a simple formula solvable in polynomial time. This holds for random instance generation models. Therefore, WP-solvability implies WP convergence whp.

Thus, Theorem 1 is proved.

4 Numerical Experiments

The number of clauses grows exponentially with variables (O(2)), so we test n = 3,4,5,6,7,8,9,10 to examine all possible clause counts and whp clause counts under planted distribution (Figure 3 [Figure 3: see original paper], Table 1).

Figures 4-6 show convergence for n = 10, 20, 30; Figures 7-9 show correct decision probabilities for n = 10, 20, 30.

As n increases, clause counts increase. With growing variables, algorithm convergence approaches 1, while non-convergence decreases. Correct decision probability shows discrete distribution: high for small iteration counts, decreasing as iterations increase.

After 1,000 WP iterations, clauses in satisfiable formulas are contained whp within clauses generated by planted distribution. This proves that when WP converges, identified variables are whp backbone variables contained in the satisfying assignment, supporting Theorem 1.

5 Conclusion

This paper introduces WP-solvable formulas based on backbone-backdoor relationships: randomly generated F is WP-solvable iff WP converges whp. Future research directions include: (a) structural properties of WP-solvable formulas and relationship between hard formulas and WP-unsolvability; (b) probabilistic convergence of message-passing algorithms; (c) structural properties of WP-solvable (3,4)-CNF formulas and convergence analysis.

References

[1] Dyer M, Frieze A, Molloy M. A probabilistic analysis of randomly generated binary constraint satisfaction [J]. Theoretical Computer Science, 2003, 290(3): 1815-1828.

[2] Creignou N, Daudé H. The SAT–UNSAT transition for random constraint satisfaction problems [J]. Discrete Mathematics, 2009, 309(8): 2085-2099.

[3] Gaspers S, Papadimitriou C, Sæther S H, et al. On satisfiability problems with a linear structure [J]. arXiv preprint arXiv:1602.07876, 2016.

[4] Doerr B, Neumann F, Sutton A M. Time complexity analysis of evolutionary algorithms on random satisfiable k-CNF formulas [J]. Algorithmica, 2017, 78(2): 561-586.

[5] Sosa-Ascencio A, Ochoa G, Terashima-Marin H, et al. Grammar-based generation of variable-selection heuristics for constraint satisfaction problems [J]. Genetic Programming and Evolvable Machines, 2016, 17(2): 119-144.

[6] Abbe E, Montanari A. Conditional random fields, planted constraint satisfaction entropy concentration [M]// Approximation, Randomization, and Com-

binatorial Optimization. Algorithms and Techniques. Berlin: Springer, 2013: 332-346.

[7] Alfredo B, Luca D, Guilhem S, et al. The large deviations of the whitening process in random constraint satisfaction problems [J]. Journal of Statistical Mechanics: Theory and Experiment, 2016, 2016(5): 053401.

[8] Kirkpatrick S, Selman B. Critical behavior in the satisfiability of random boolean expressions [J]. Science, 1994, 264(5163): 1297-1301.

[9] Kaporis A C, Kirousis L M, Lalas E G. The probabilistic analysis of a greedy satisfiability algorithm [J]. Random Structures & Algorithms, 2006, 28(4): 444-480.

[10] Dubois O, Boufkhad Y, Mandler J. Typical random 3-SAT formulae and the satisfiability threshold [J]. arXiv preprint cs/0211036, 2002.

[11] Xu Ke, Boussemart F, Hemery F, et al. Random constraint satisfaction: easy generation of hard satisfiable instances [J]. Artificial Intelligence, 2007, 171(8): 514-534.

[12] Xu Ke, Wei Li. Many hard examples in exact phase transitions [J]. Theoretical Computer Science, 2006, 355(3): 291-302.

[13] Coja-Oghlan A, Panagiotou K. The asymptotic k-SAT threshold [J]. Advances in Mathematics, 2016, 288: 985-1068.

[14] Williams R, Gomes C P, Selman B. Backdoors to typical case complexity [C]// Proc of the 18th International Joint Conference on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann Publishers Inc, 2003: 1173-1178.

[15] Kilby P, Slaney J, Thiébaux S, et al. Backbones and backdoors in satisfiability [C]// Proc of the 20th National Conference on Artificial Intelligence. Pennsylvania: AAAI Press, 2005: 1368-1373.

[16] Hemaspaandra L A, Narváez D E. Search versus decision: the opacity of backbones and backdoors under a weak assumption [J]. arXiv preprint arXiv:1706.04582, 2017.

[17] Janota M, Lynce I, Marques-Silva J. Algorithms for computing backbones of propositional formulae [J]. AI Communications, 2015, 28(2): 161-177.

[18] Gregory P, Fox M, Long D. A new empirical study of weak backdoors [C]// Proc of the 14th International Conference on Principles and Practice of Constraint Programming. Berlin: Springer, 2008: 618-623.

[19] Marques-Silva J, Janota M, Lynce I. On computing backbones of propositional theories [C]// Proc of the 19th European Conference on Artificial Intelligence. 2010: 15-20.

[20] Meier A, Ordyniak S, Ramanujan M S, et al. Backdoors for linear temporal logic [J]. Algorithmica, 2018, 81(2): 476-496.

[21] Zhou Huan, Xu Shouzhi, Ren Dong, et al. Analysis of event-driven warning message propagation in vehicular ad hoc networks [J]. Ad hoc Networks, 2017, 55: 87-96.

[22] Braunstein A, Mezard M, Zecchina R. Survey propagation: an algorithm for satisfiability [J]. Random Structures & Algorithms, 2005, 27(2): 201-226.

[23] Krivelevich M, Vilenchik D. Solving random satisfiable 3CNF formulas in expected polynomial time [C]// Proc of the 17th annual ACM-SIAM symposium on Discrete algorithm. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2006: 454-463.

[24] Feige U, Mossel E, Vilenchik D. Complete convergence of message passing algorithms satisfiability problems [M]// Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques. Berlin: Springer, 2006: 339-350.

[25] Abbe E, Montanari A. Conditional random fields, planted constraint satisfaction entropy concentration [M]// Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques. Berlin: Springer, 2013: 332-346.

[26] Wang Xiaofeng, Jiang Jiulei. Warning propagation algorithm for the MAX-3-SAT problem [J]. IEEE Trans on Emerging Topics in Computing, http://doi.org/10.1109/tetc.2017.2736504, 2017.

[27] Wang Xiaofeng, Xu Daoyun. Sufficient conditions for warning propagation algorithm convergence [J]. Journal of Software, 2016, 27(12): 3003-3013.

[28] Sutton A M, Neumann F. Runtime analysis of evolutionary algorithms on randomly constructed high-density satisfiable 3-CNF formulas [C]// Proc of International Conference on Parallel Problem Solving from Nature. Springer International Publishing, 2014: 942-951.

[29] Wang Xiaofeng, Xu Daoyun, Qin Yongbin. Information propagation algorithm for solving formula key word sets [J]. Journal of Shandong University: Engineering Edition, 2011, 41(3): 1-6.

Note: Figure translations are in progress. See original paper for figures.