SCE
Shamoon College of Engineering

Software Engineering

Final Paper

An Empirical Study of Local Properties of the Landscape of the Maximum Satisfiability Problem

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE

By
Antokolsky Leon

UNDER THE SUPERVISION OF
Yochai Twitto

August 16, 2025

Contents

Abstract

In this work, we present an empirical analysis of the structure of local optima in the landscape of the Maximum Satisfiability (MAX-SAT) problem. Focusing on parameterized families of random MAX-SAT instances, we systematically measure and compare several key properties: the proportion of local optima among all assignments, the distribution of local optima by the number of satisfied clauses , the fraction of neighbors with equal quality, and the Hamming distances between local optima at different heights.

Our results show that the ratio and distribution of local optima are significantly influenced by the problem parameters, such as the number of variables per clause and the total number of clauses. In particular, more complex families tend to have a greater diversity of high-quality solutions and larger Hamming distances between local optima. We also observe stable patterns in the distribution of local optima within families with similar parameters.

These findings provide new quantitative insights into the landscape structure of random MAX-SAT problems, which may help inform future algorithmic studies and the development of search heuristics for combinatorial optimization.

1 Introduction

This chapter introduces the Maximum Satisfiability (MAX-SAT) problem, defines the main concepts and terminology, discusses computational complexity, and highlights the significance and applications of MAX-SAT in computer science and related fields.

1.1 The Maximum Satisfiability Problem

The MAX-SAT problem is a fundamental challenge in combinatorial optimization and computational complexity theory. Given a Boolean formula in conjunctive normal form (CNF), the objective is to assign values to variables so as to maximize the number of satisfied clauses, where each clause is a logical disjunction of literals.

Formally, let F be a set of clauses over Boolean variables $\{x_1, x_2, \ldots, x_n\}$. The task is to find a truth assignment that maximizes the number of clauses in F that evaluate to true. This problem is known to be NP-hard, meaning that, unless P=NP, no polynomial-time algorithm exists for solving all instances exactly. (The well-known SAT problem is a special case of MAX-SAT.)

As a central topic in theoretical computer science, maximum satisfiability has numerous applications in artificial intelligence, hardware verification, scheduling, and bioinformatics. Moreover, many important combinatorial problems can be reduced to it, emphasizing its foundational role in the study of computational intractability and approximation algorithms.

We use the term **family** to describe a group of problems sharing the same parameters:

- Number of variables (n): the total number of Boolean variables;
- Number of clauses (m): each clause is a logical expression composed of one or more variables joined by the logical OR;
- Number of variables per clause (r): the exactly number of variables in any single clause.

An **instance** is a specific set of clauses over a fixed set of variables, fully describing the logical structure. A **configuration** is an assignment of Boolean values to all variables in the instance.

Illustrative Example: Suppose we are dealing with four variables: X_1, X_2, X_3, and X_4.

Consider the following *instance* in CNF:

$$(X_1 \vee \neg X_2 \vee X_3) \wedge (\neg X_1 \vee X_2 \vee \neg X_4)$$

This instance consists of two clauses:

- Clause 1: $X_1 \vee \neg X_2 \vee X_3$
- Clause 2: $\neg X_1 \vee X_2 \vee \neg X_4$

Given the following *configuration* (assignment of values):

- $X_1 = \text{True}, \quad X_2 = \text{False}, \quad X_3 = \text{False}, \quad X_4 = \text{True}$
- After substitution, Clause 1 becomes: True $\vee$ True $\vee$ False. Since at least one of the operands in "OR" is true, the entire clause is considered true.
- After substitution, Clause 2 becomes: False$\vee$False$\vee$False. In this case, all operands are false, so the clause is not satisfied.

This configuration satisfies one clause out of two. The goal in MAX-SAT is to find a configuration that maximizes the number of satisfied clauses.

1.2 Landscape Analysis

Landscape analysis provides a systematic approach for studying the structure of the solution space and understanding how different configurations affect the number of satisfied clauses. In this context, each configuration can be visualized as a point in a high-dimensional space, where the "height" of each point corresponds to the number of satisfied clauses — that is, the quality of the solution.

A central concept is that of a **local maximum**: a solution that is better than all its immediate neighbors (i.e., those differing in the value of a single variable), but not necessarily optimal across the entire space. Such points represent configurations from which any small change results in a decrease in the number of satisfied clauses.

Landscape analysis enables researchers to:

- **Assess problem difficulty.** The presence of many local maxima can make the search for a global optimum more challenging, especially when these maxima are widely distributed.

- **Classify problems into families.** Problems can be grouped based on landscape features such as the number and distribution of local maxima and the ruggedness of the landscape.

- **Develop effective search strategies.** By understanding the landscape, it is possible to design or select algorithms that are better suited for moving between local maxima and escaping suboptimal regions in pursuit of superior solutions.

This analysis is not only about identifying the optimal configuration but also about understanding the underlying structure that influences search dynamics. It helps explain why some problem instances are inherently more difficult and provides insights for designing algorithms capable of efficiently navigating complex solution landscapes.

1.3 Definitions

1.3.1 Hamming Distance

Let us first deal with the definition of Hamming distance. Hamming distance is a measure that shows how different two configurations are from each other. It is the number of positions at which the corresponding symbols in the two strings are different. In other words, it is the minimum number of changes needed to make one string equal to another, assuming that these changes are limited to replacing characters, not adding new ones, or deleting existing ones.

For example, consider the following two configurations of variables for a certain problem:

- Configuration 1: True, False, True

- Configuration 2: True, True, True

The Hamming distance between these configurations is 1 because only one variable (the second one) differs between the two sets. To transition from one configuration to the other, only the value of one variable needs to be changed.

1.3.2 Local Optimum

A local optimum is a particular configuration of variables where changing the value of any single variable does not lead to an increase in the number of satisfied clauses. In other words, it is a configuration that is at least as good as any of its immediate "neighbors" (i.e., those that differ by only one variable).

Consider a MAX-SAT problem with three variables, X_1, X_2, and X_3, and the following clauses:

- $X_1 \vee X_2$
- $X_1 \vee \neg X_3$
- $\neg X_2 \vee X_3$

Let us analyze the configuration C: $X_1 = \text{True}$, $X_2 = \text{False}$, $X_3 = \text{True}$. Suppose the MAX-SAT instance consists of the following three clauses:

- $X_1 \vee X_2$
- $X_1 \vee \neg X_3$
- $\neg X_2 \vee X_3$

This configuration satisfies all three clauses:

- Clause 1 is satisfied because $X_1 = \text{True}$.
- Clause 2 is satisfied since $X_3 = \text{True}$.
- Clause 3 is satisfied because $X_2 = \text{False}$ and $X_3 = \text{True}$.

As a result, configuration C satisfies all three clauses.

Now, let us consider all "neighboring" configurations C' that can be obtained by flipping a single variable in C:

- C'_1: $X_1 = \text{False},\ X_2 = \text{False},\ X_3 = \text{True}$
  - Clauses satisfied: only clause 3.
  - Number of satisfied clauses: 1.

- C_2': $X_1 = \text{True},\ X_2 = \text{True},\ X_3 = \text{True}$
  - Clauses satisfied: 1 and 3.
  - Number of satisfied clauses: 2.
- C_3': $X_1 = \text{True},\ X_2 = \text{False},\ X_3 = \text{False}$
  - Clauses satisfied: 1 and 2.
  - Number of satisfied clauses: 2.

From the analysis, it is clear that changing any variable from the original configuration C results in fewer or an equal number of satisfied clauses. Therefore, C is considered a local optimum, as none of its neighbors have more satisfied clauses than C itself.

This example illustrates how the concept of a local optimum functions in solving MAX-SAT problems. A local optimum is a configuration where any single-variable modification does not lead to a better solution, thereby making it an optimal choice within its immediate neighborhood.

1.3.3 Variability Among Different Instances

Within the same "family", where all instances share the same parameters n, m, and r, the number of local optima can still vary. This variation arises from differences in the specific structure of clauses and their interrelations in each instance. Some instances may exhibit a "smoother" landscape with fewer local optima, while others might have a "rougher" landscape with many local maxima.

The parameters n, m, and r influence both the complexity of the solution landscape and the number of local optima. As these parameters change, the landscape can become either more diverse or more structured, depending on the relationships among variables and clauses. Generally, increasing the number of variables and clauses provides a richer space of configurations, which may result in greater variability in the number and distribution of local optima. However, the precise effect of these parameters is not straightforward and can differ significantly across instances, depending on their internal structure and clause interactions.

1.3.4 Local Optimum Heights

The concept of "height," defined as the number of satisfied clauses, serves as a key indicator of solution quality in MAX-SAT problems. Studying the distribution of local optima across height levels provides valuable insight into the overall structure of the problem space. For example, tightly clustered optima with similar values may indicate flat regions that challenge local search methods, while a broader spread suggests greater variation and potential for improvement.

These patterns tend to remain stable within problem "families" that share common parameters such as the number of variables, clauses , and variables in clauses. This consistency allows researchers to anticipate the general topography of new instances within the same family, making the search process more predictable. Even moderate adjustments to these parameters usually leave the overall characteristics of the landscape intact, highlighting a degree of structural robustness.

Ultimately, analyzing the height-based distribution of optima not only clarifies the inherent difficulty of the problem but also supports the design of more effective optimization strategies. Guides the selection of algorithms that can adapt to terrain, be it exploring broad regions or exploiting known peaks, improving both performance and reliability.

2 Literature Review

2.1 Introduction to Fitness Landscapes and Their Importance

Fitness landscapes describe how solution quality varies across the space of possible assignments in optimization problems such as MAX-SAT [1]. Each configuration is associated with a "fitness" value, typically the number of satisfied clauses. Landscape analysis enables researchers to study how the structure and distribution of these fitness values impact problem complexity and algorithm performance.

For MAX-SAT, important landscape features include the number and arrangement of local optima, the degree of ruggedness (the frequency and amplitude of fitness changes between neighbors), and the size of basins of attraction. These characteristics influence the effectiveness of local and stochastic search algorithms, as they determine how easily an algorithm can escape suboptimal solutions and explore the search space.

Probabilistic models have been used to estimate the behavior of MAX-SAT instances, especially for random formulas where each clause contains exactly r literals [2]. Such models allow the calculation of the expected number of satisfied clauses under random assignments and the variance of these numbers. Additionally, the autocorrelation of fitness values—how similar neighboring configurations are in terms of fitness—can indicate whether the landscape is "smooth" or "rugged." High autocorrelation suggests smoother landscapes, potentially making local search more effective.

The computational intractability of optimizing combinatorial measures is well-documented in recent literature. For instance, it has been shown that maximizing certain diversity scores on phylogenetic networks is NP-hard and not approximable beyond specific thresholds [3]. These complexity results highlight the theoretical challenges inherent in problems like MAX-SAT and motivate the development of specialized heuristics and approximation algorithms.

Recent advancements in neural network architectures have shown that message passing neural networks can be trained to solve SAT instances using only minimal supervision [4]. The NeuroSAT model demonstrates not only the ability to classify satisfiable and unsatisfiable formulas, but also to decode satisfying assignments from its internal activations. This opens new directions for applying machine learning methods to combinatorial optimization, especially in the context of analyzing and navigating complex solution landscapes.

Modern industrial optimization problems are frequently formulated as MAX-SAT instances, which has driven the development of efficient SAT-based MAX-SAT solvers [5]. These algorithms leverage advances in SAT solving by reducing MAX-SAT to a sequence of SAT queries, achieving significant practical success on large-scale and weighted variants. Experimental evaluations consistently show that SAT-based techniques outperform traditional approaches on real-world datasets, establishing them as the preferred standard for industrial applications.

Understanding these properties provides a basis for developing and selecting algorithms tailored to the specific challenges presented by different classes of MAX-SAT problems.

Applications in Combinatorial Optimization

Fitness landscape analysis extends beyond MAX-SAT to a range of combinatorial problems. These problems are characterized by large, discrete solution spaces where exhaustive search is often infeasible. Analyzing their landscapes enables the design of algorithms that can escape local optima and efficiently navigate the solution space.

Several canonical problems exemplify this approach:

- **Traveling Salesman Problem (TSP)**: Highlights factorial growth in solution complexity and the challenge of path optimization.

- **Graph Bi-Partitioning (GBP)**: Involves minimizing interconnecting edges while maintaining balance between partitions.

- **Quadratic Assignment Problem (QAP)**: Requires minimizing assignment costs based on flow and distance, a problem compounded by solution interdependencies.

- **NK-Landscape Model**: Simulates epistatic interactions between N components with K dependencies, offering insight into ruggedness and landscape tunability.

These problems reinforce the importance of understanding fitness landscapes as a tool for both theoretical analysis and practical algorithm design.

The patterns and structures observed across different problem types provide generalizable strategies for navigating complex search spaces.

Recent research has demonstrated the potential of continuous-time analog dynamical systems for solving hard MAX-SAT problems [6]. Unlike classical digital algorithms, these approaches utilize the dynamics of ordinary differential equations to explore the solution space efficiently, often providing near-optimal results significantly faster for large and complex instances. Such physical computing paradigms expand the algorithmic toolkit available for combinatorial optimization and have been shown to compete with, or even surpass, state-of-the-art digital solvers on certain benchmarks.

Ultimately, this foundation allows for the development of adaptive and scalable optimization methods that are informed by the geometry and topology of the underlying problem space.

2.2 Properties of Fitness Landscapes

The structural features of fitness landscapes play a critical role in determining the behavior and success of heuristic optimization methods. A fitness landscape can be characterized by several key properties that directly influence algorithm performance:

- **Ruggedness:** Refers to the degree of variability in fitness between neighboring solutions. High ruggedness—signaled by many peaks and valleys—indicates a challenging landscape filled with local optima, which can easily trap search algorithms.

- **Local Optima Distribution:** Understanding how many local optima exist and how they are spatially arranged helps gauge the problem's complexity. Dense distributions may require algorithms that can escape local traps, while sparse distributions often favor simpler strategies.

- **Basins of Attraction:** Each local optimum has a basin of attraction—a region from which local search will likely converge to that point. The size and shape of these basins determine how easily solutions can be discovered and improved during the search process.

To assess these features, researchers use quantitative metrics that provide insights into the landscape's navigability:

- **Autocorrelation:** Measures the similarity between fitness values at neighboring configurations. A high autocorrelation length typically indicates a smoother landscape with fewer abrupt changes, facilitating gradual improvement through local moves.

- **Fitness-Distance Correlation (FDC):** Evaluates the relationship between solution quality and distance to the global optimum. A high FDC implies that better solutions tend to be closer to the optimum, which is favorable for local and greedy search strategies.

- **Elementary Landscapes:** These are simplified mathematical models that capture core structural patterns in complex landscapes. They allow the computation of mean fitness, variance, and neighborhood properties with reduced computational overhead.

Together, these concepts form the foundation of *landscape theory*, which offers a systematic framework for analyzing optimization problems. By quantifying structural complexity and navigational cues, landscape theory enables the development of tailored algorithms—such as memetic or hybrid heuristics—that dynamically adapt to ruggedness, optima distribution, and basin structures. This theoretical grounding enhances both exploration and exploitation, ultimately improving optimization performance across a range of problem domains.

2.3 Advanced Analytical Measures in Landscape Theory

To address the intricacies of rugged and high-dimensional fitness landscapes, advanced analytical metrics are employed to inform algorithm design and improve optimization performance [7]. These measures provide a deeper understanding of both global structures and local dynamics within the search space:

- **Expected Fitness After Bit-Flip Mutation:** This measure estimates the average change in fitness resulting from flipping a single variable in a solution. It plays a crucial role in evaluating the sensitivity of configurations and in tuning mutation operators within evolutionary frameworks.

- **Autocorrelation and FDC (Revisited):** While previously introduced, these metrics gain additional significance in this context by offering predictive insights into algorithmic behavior. High autocorrelation suggests that fitness varies smoothly across neighbors, favoring gradual improvement. A strong Fitness-Distance Correlation (FDC) implies that better solutions are closer to the global optimum, guiding exploitation.

- **Landscape Entropy and Variability Measures:** Additional metrics such as entropy or dispersion indices help quantify the unpredictability and diversity of solution quality across the landscape. These indicators assist in understanding how broadly the algorithm should search and how much variation to expect during exploration.

Together, these analytical tools support the development of more responsive and adaptive search strategies, especially in the context of complex combinatorial optimization problems.

Memetic Algorithm Design: Integrating Global and Local Search

Memetic algorithms combine evolutionary principles with problem-specific local refinement, leveraging fitness landscape insights to maximize performance. Their effectiveness depends on the careful design of several components:

- **Solution Representation and Evaluation:** Effective encodings must capture the structural nuances of the problem, while evaluation functions should accurately reflect the fitness landscape's topology.

- **Local Search Integration:** Embedding efficient local search heuristics enables the algorithm to rapidly converge toward local optima and refine promising regions of the search space.

- **Initialization and Diversity Management:** Well-crafted initialization methods and diversity-preserving mechanisms help maintain broad coverage of the solution space, reducing the risk of premature convergence.

- **Operator Design and Adaptation:** Dynamic adjustment of mutation and crossover rates based on real-time landscape feedback can improve convergence speed while avoiding stagnation.

By aligning these components with observed landscape characteristics—such as ruggedness, optima spacing, and mutation sensitivity—memetic algorithms can effectively adapt to a wide range of optimization challenges. Their hybrid nature allows them to balance global exploration with fine-tuned local improvement, making them especially powerful in navigating rugged, multimodal landscapes.

2.4 Practical Applications of Fitness Landscape Theory

The application of fitness landscape theory has significantly advanced the field of combinatorial optimization by enabling more informed algorithm design and analysis. Memetic algorithms and other heuristic strategies have benefited particularly from these insights, allowing them to outperform traditional approaches through a more nuanced understanding of search space dynamics.

A milestone in the evaluation of MAX-SAT solvers was set by Argelich et al. (2008), who systematically compared state-of-the-art algorithms through the first international Max-SAT evaluations. Their results provided key insights into solver strategies, benchmarking standards, and the distinction between exact and heuristic approaches, greatly advancing the field [8]. These benchmark competitions continue to inform best practices and set directions for future solver development.

Recent advancements have demonstrated the applicability of multi-objective MAX-SAT solvers in optimizing highly configurable products, especially in industrial domains such as automotive manufacturing. For instance, Bruns (2024) developed a novel framework utilizing MAX-SAT-based techniques to efficiently handle conflicting objectives across product lifecycles, including cost, market fit, and technical constraints [9]. This work highlights the growing impact of MAX-SAT methods for real-world configuration and planning problems, moving beyond traditional theoretical benchmarks.

2.4.1 Memetic Algorithms in Combinatorial Optimization

Memetic algorithms combine evolutionary mechanisms with local search techniques, effectively balancing exploration and exploitation. Their success in addressing classic combinatorial problems stems from their ability to adaptively refine solutions while maintaining diversity in the population.

- **Traveling Salesman Problem (TSP):** By leveraging genetic recombination and neighborhood search, memetic algorithms efficiently explore the exponentially large solution space to identify near-optimal routes. This hybrid strategy has consistently delivered superior results compared to standalone methods.
- **Quadratic Assignment Problem (QAP):** The structured yet interdependent nature of QAP instances makes them particularly challenging. Memetic algorithms have shown proficiency in navigating these landscapes by iteratively improving facility-location assignments through focused local refinement.
- **NK-Landscapes:** These synthetic landscapes, characterized by tunable ruggedness via the parameter K, are ideal for testing algorithm adaptability. Memetic approaches demonstrate strong performance across varying levels of complexity, successfully managing both smooth and rugged terrain.

These applications illustrate how landscape-aware algorithm design can improve convergence speed and solution quality in diverse problem settings.

2.4.2 Incomplete Algorithms for SAT and MAX-SAT

Incomplete solvers have gained prominence in SAT and MAX-SAT contexts where rapid, high-quality solutions are preferred over guaranteed completeness. These methods, grounded in fitness landscape principles, often provide competitive alternatives to complete solvers.

- **Stochastic Local Search (SLS):** SLS methods utilize heuristic-driven, iterative improvements to traverse the landscape efficiently. They have shown strong performance on large and heterogeneous SAT instances, particularly when deterministic methods struggle.
- **Greedy Search and Random Walks:** Algorithms like GSAT and Walksat blend deterministic hill-climbing with stochastic exploration, allowing them to escape local optima and maintain diversity in the search process. These techniques exemplify early successful implementations of fitness-based heuristics.

- **Survey Propagation:** This advanced method uses probabilistic inference to estimate marginal probabilities for variable assignments. Particularly effective in large, constrained SAT instances, it captures landscape structure without exhaustive enumeration, guiding search through informed approximations.

The evolution of incomplete algorithms—from purely random approaches to sophisticated hybrids—reflects growing awareness of landscape topology and its influence on algorithmic behavior.

Recent studies in deep learning and computer vision highlight that the structure of the solution space can significantly impact the performance and reliability of optimization algorithms. For example, in video super-resolution tasks, recurrent neural networks are sensitive to landscape-induced artifacts, which can propagate and degrade the final solution quality [10]. Techniques such as hidden-state attention have been proposed to specifically address these issues, illustrating the broader relevance of landscape analysis in both traditional combinatorial optimization and modern machine learning applications.

2.4.3 Open-Source Solvers and Frameworks

Beyond algorithmic design, several modular frameworks have emerged to facilitate the development, evaluation, and deployment of optimization algorithms informed by landscape theory.

Open-WBO [11] is a flexible, open-source framework that integrates SAT solver components into customizable MAX-SAT strategies. It supports the development of unsatisfiability-based techniques, linear and core-guided search methods, and other experimental solvers.

- **Architecture:** Its modular structure separates parsing, algorithm execution, and SAT solving, enabling independent improvement and testing of each component.

- **Solver Integration:** Designed to interface with any MiniSAT-compatible solver, Open-WBO benefits from state-of-the-art SAT improvements without re-engineering the entire pipeline.

- **Research and Education:** Widely used in both academia and industry, it serves as a platform for algorithm development, comparison, and teaching.

ParadisEO-MO [12] supports the design and deployment of local search algorithms for diverse optimization problems. It offers tools for analyzing fitness landscapes, developing hybrid metaheuristics, and scaling to high-performance computing environments.

- **Modularity and Templates:** The template-based design enables users to define problem-specific behavior while reusing common algorithmic components.
- **Hybridization and Parallelism:** The framework supports hybrid strategies and parallel computation, making it suitable for both academic research and large-scale industrial applications.
- **Use Cases:** ParadisEO-MO has been applied in production planning, bioinformatics, climate modeling, and more—demonstrating the practical utility of landscape-aware search strategies.

These frameworks illustrate the growing integration of theoretical landscape concepts into practical toolkits, bridging the gap between algorithm research and application deployment.

2.5 Algorithm Evaluation and Experimental Performance

This section reviews notable algorithmic approaches and key experimental results reported in the literature for the MAX-SAT problem. Emphasis is placed on recent methods that have demonstrated state-of-the-art performance on benchmark datasets and industrial instances.

Recent advances demonstrate that MAX-SAT is not only a theoretically significant problem, but also plays a critical role in practical software analysis and optimization. For instance, modern software verification and bug detection tasks can be formulated as large-scale MAX-SAT instances, where efficient solving techniques enable precise and scalable analyses [13]. These approaches balance competing tradeoffs such as soundness, precision, and computational efficiency, and highlight the importance of MAX-SAT in real-world applications.

The continuous progress in MAX-SAT solver technology is regularly assessed through international MAX-SAT Evaluations, where diverse solvers are compared on a variety of benchmarks [14]. These evaluations provide insights into current algorithmic trends, state-of-the-art techniques, and challenges faced when scaling to industrial-size problem instances.

2.5.1 CCLS: Configuration Checking with Make Heuristic

The CCLS algorithm [15] is a prominent stochastic local search method for the weighted MAX-SAT problem. It incorporates the Configuration Checking with Make (CCM) heuristic, which prioritizes variable flips based on recent changes and their impact on solution improvement.

- **Heuristic Strategy:** The CCM heuristic has been found to enhance efficiency by avoiding redundant flips and targeting high-impact variables, thereby reducing computational overhead and improving convergence speed.

- **Search Dynamics:** The integration of random walks with CCM helps maintain a balance between exploration and exploitation, allowing CCLS to escape local optima and incrementally improve solution quality.

- **Benchmark Performance:** Experimental comparisons in the literature indicate that CCLS outperforms traditional local search and complete solvers on both crafted and industrial instances, especially under tight computational budgets.

2.5.2 Hybrid Algorithm Design and Evaluation

Recent studies have proposed hybrid algorithms that combine deterministic greedy selection with stochastic local search. Such combinations enable rapid progress in the early stages of search while preserving the flexibility to escape local optima and improve solution quality over time.

- **Greedy Components:** Greedy heuristics, such as Johnson's strategy, have been employed to quickly satisfy high-gain clauses, establishing a solid starting point for subsequent refinement.

- **Stochastic Refinement:** Local search is then applied to fine-tune promising configurations, often with adaptive heuristics tailored to the local landscape.

- **Comparative Results:** According to published benchmark tests, such hybrid approaches achieve a favorable balance between runtime efficiency and solution quality, making them suitable for large-scale or time-constrained applications.

Hybrid optimization techniques that combine human expertise with machine learning have been shown to improve the efficiency and effectiveness of solving complex scheduling and resource allocation problems. Apprenticeship learning, in particular, enables algorithms to generalize expert demonstration policies to new instances, providing a promising avenue for collaborative optimization in hard combinatorial tasks [16]. These approaches complement purely algorithmic strategies and can be adapted for challenging settings such as MAX-SAT.

These literature results establish a reference point for the comparative evaluation of algorithmic performance in MAX-SAT, providing context for new developments in the field.

3 Methods

This section outlines the methodologies and strategies employed to investigate the local optima. Each method plays a crucial role in exploring the landscape of potential solutions and understanding the interactions between them.

Generating Instances

To simulate real-world scenarios, we generated a series of logical expressions, or instances. Each instance comprises various clauses with a combination of variables and logical operations. This process allows us to evaluate the performance of algorithms across a diverse set of problem configurations.

Evaluating Clause Success

Our study includes a mechanism to assess the number of clauses successfully satisfied by each configuration of variables. A clause is considered successful if at least one literal within it evaluates to true. This metric is essential for determining the effectiveness of different configurations in solving the MAX-SAT problems.

Generating Random Configurations and Neighbors

We developed a system to generate random configurations of variables. Subsequently, we produced neighboring configurations by altering a single variable at a time. This approach is fundamental in exploring the solution space and observing the effects of minor changes on the overall solution quality.

Identifying Local Optima

The core of our analysis is identifying local optima within the solution space. We define a local optimum as a configuration where no single-variable modification leads to a better solution, i.e., a configuration that satisfies more clauses than any of its immediate neighbors.

Analyzing Local Optima Heights

Local optima are analyzed not just for their quality but also for their 'height,'. This analysis helps in understanding the distribution and density of local optima across different problem instances.

Calculating Hamming Distances

To gauge the diversity among solutions, we compute the Hamming distances between pairs of local optima. This metric illustrates the number of changes required to transform one optimum into another, providing insights into the solution landscape's variability.

By employing these methods, we aim to gain a comprehensive understanding of the landscape of solutions in Max-SAT problems, which can inform the development of more effective optimization strategies.

For those interested in the detailed implementation, the source code used in our experiments is provided in the Appendix of this document. This allows readers to review the exact methods and potentially reproduce the study or apply the techniques to their own research.

4 The Ratio of Locally-Optimal Assignments Out of All Assignments

4.1 Overall Ratio Distribution

This experiment is specifically designed to determine the percentage of local optima for various numbers of instances within the solution. Our goal is to assess how the distribution of local optima varies across different instance counts, providing insights into the scalability and complexity of problem configurations under examination.

It is important to understand how local optima are identified based on the comparison criteria of "strictly greater than" and "greater than or equal to". These criteria help define the conditions under which a configuration is a local optima.

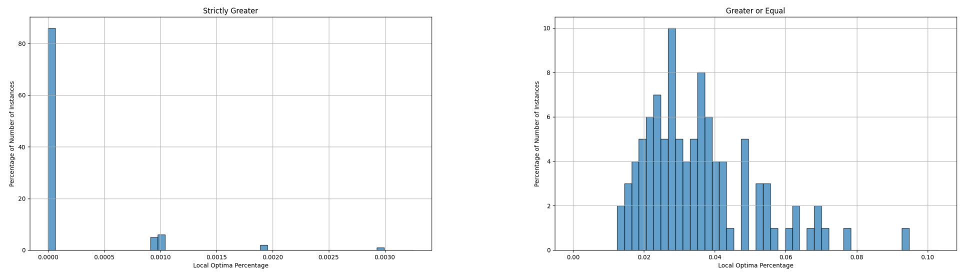
A configuration is considered a *strict local optimum* if it achieves a higher number of satisfied clauses than any of its immediate neighbors. This indicates a superior solution among the adjacent possibilities.

A configuration is deemed a *non-strict local optimum* if it meets or exceeds the number of satisfied clauses compared to any of its neighbors. This category allows a configuration to be considered optimal even if it merely matches the performance of its neighboring solutions.

Utilizing previously described functions for generating problem instances, calculating successful clauses, and determining local optima, we conducted an analysis across several problem "families" differentiated by their parameters (r, n, m). For each family, 100 instances were randomly generated, and for each instance, we explored 1024 different variable configurations to assess the presence of local optima. The results were visualized using histograms to compare the distribution of local optima ratio, where the ratio can be calculated using the following formula:

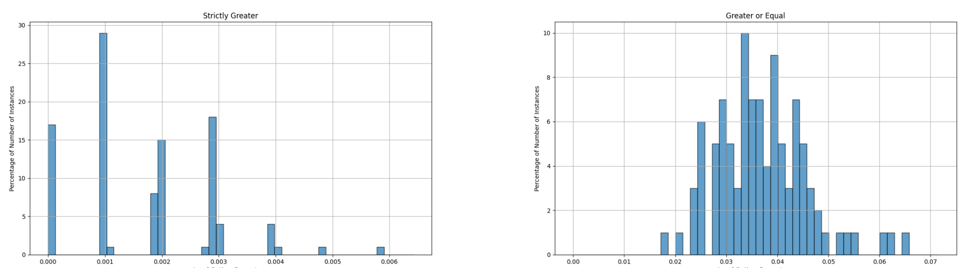
$$\text{Local Optimally Ratio(normalized)} = \frac{\text{number of local optima}}{\text{number of configurations}}$$

The experiment's outcomes were visualized through histograms, where the x-axis represented the ratio of local optima and the y-axis showed the frequency of instance across 100 instances.



(a) Strictly locally-optimal assignments (b) Greater or Equal locally-optimal assignments

Figure 1: Histograms of distribution of local optima ratios under different criteria for family: $r = 2, n = 10, m = 20$.



(a) Strictly locally-optimal assignments (b) Greater or Equal locally-optimal assignments

Figure 2: Histograms of distribution of local optima ratios under different criteria for family: $r = 3, n = 10, m = 50$.

From the histograms, we observe clear patterns in the distribution of local optima ratios under the conditions "strictly greater" and "greater or equal" for the specified problem families.

- **Narrow Distribution**:
  - **Family** $r = 2, m = 20, n = 10$: The "strictly greater" condition shows a very concentrated distribution of local optima ratios, mostly close to zero, indicating a rigorous selection criterion that few configurations meet.

- **Family** $r = 3, m = 50, n = 10$: Similarly, this family under the "strictly greater" condition also exhibits a tight distribution, but with a slight spread up to 0.0030, slightly more relaxed compared to the first family.

- **Wider Distribution**:
  - **Family** $r = 2, m = 20, n = 10$: Under the "greater or equal" condition, this family shows a broader distribution with a notable peak around 0.05, suggesting that a larger number of configurations qualify as local optima.
  - **Family** $r = 3, m = 50, n = 10$: This condition for the second family also results in a wider distribution but with a slightly less pronounced spread, peaking around 0.04.

- **Lower Ratios**:
  - Lower local optima ratios are more frequent under the "strictly greater" condition across both families, indicating the strictness of this criterion.

- **Higher Ratios**:
  - More instances exhibit higher local optima ratios under the "greater or equal" condition, showing that this criterion is more inclusive, allowing more solutions to be considered optimal.

- **Impact of Parameters**:
  - As m and n values vary between the families, it affects the distribution between the "strictly greater" and "greater or equal" conditions, highlighting how parameter settings can influence the landscape of potential solutions.

- **Consistent Patterns**:
  - The "greater or equal" condition shows consistency across different families in terms of broader distributions, suggesting it might be a more stable criterion across different problem complexities.

This detailed analysis shows how the choice of criteria for defining local optima influences the search landscape and the optimization strategies, vital for designing algorithms for MAX-SAT and similar complex optimization problems.

4.2 Ratio Distribution by Height

This experiment was designed to analyze the ratio of local optima by height within the solution landscape of MAX-SAT problems. We focused on determining the normalized frequency of local optima at different levels of clause satisfaction to identify any prevalent patterns in their distribution.

For each family, we generated 1024 random configurations from 100 instance to explore the landscape of local optima.

We selected three distinct families to compare the distribution of local optima across varying scales of complexity:

1. **Family 1**: $r = 2$, $n = 10$, $m = 20$, with an average of 43.44 satisfied clauses and with total local optimum equal 3,443,037.

2. **Family 2**: $r = 3$, $n = 10$, $m = 50$, with an average of 215.76 satisfied clauses and with total local optimum equal 3,645,590.

The results were depicted through histograms, where:

- The x-axis represents the **Normalized Number of Satisfied Clauses**, adjusting the scale to account for different maximum possible clause satisfactions in each family.

- The y-axis indicates the **Normalized Frequency of Local Optima**, presenting a relative comparison of how frequently local optima occur at each level of clause satisfaction.

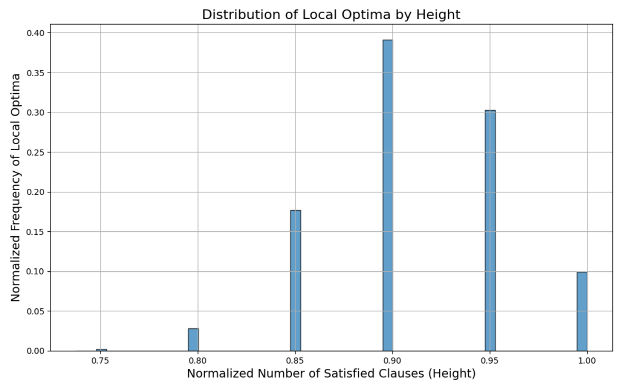


Figure 3: Histogram of the ratio of locally-optimal assignments as function of height for family $r = 2, n = 10, m = 20$

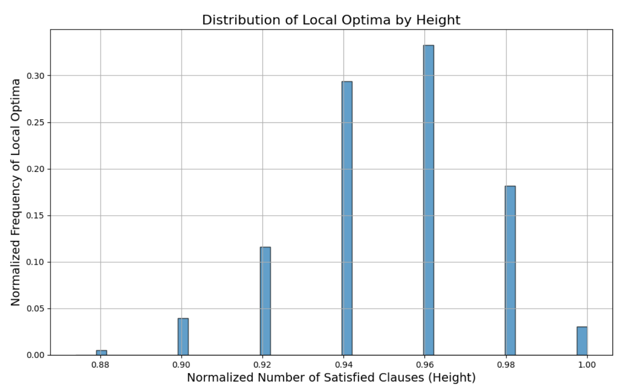


Figure 4: Histogram of the ratio of locally-optimal assignments as function of height for family $r = 3, n = 10, m = 50$

This experiment provided insights into the distribution and characteristics of local optima within the solution landscapes of various MAX-SAT problem configurations. Below we summarize our key findings:

- **Variability in Local Optima Counts:** Across the two families, the number of local optima varied despite each instance having the same number of configurations analyzed. This variability suggests that the complexity of the solution landscape and the likelihood of encountering local optima are significantly influenced by the specific parameters of each problem family.

- **Average Number of Satisfied Clauses:** While the absolute number of satisfied clauses varied due to differences in the total number of clauses (m), comparative analysis reveals that, on average, about 86% of clauses were satisfied across all families. This high percentage underscores the effectiveness of the solutions within the explored landscapes and indicates robustness in clause satisfaction relative to the problem size.

- **Increased Local Optima in Larger Configurations:** Notably, in larger problem settings, the number of local optima was approximately 5% higher compared to smaller configurations. This observation points to a denser concentration of local optima in more complex configurations, suggesting that larger problems, though more challenging, also provide more opportunities for finding locally optimal solutions.

- **Similarity Across Graphical Representations:** Despite variations in specific metrics, the overall graphical representations of the local optima distributions were remarkably similar across the two families. This similarity in distribution patterns suggests underlying commonalities in the structure of the solution landscapes, regardless of the problem scale or complexity.

These observations imply that the landscape's inherent characteristics, influenced by problem parameters, play a crucial role in shaping the search for optimal solutions.

4.3 Summary Statistics

In our research, we delve into the behavior of local optima in different configurations. Here's why we measure specific statistical metrics and what they reveal about the problem landscape:

- **Mean Heights:** Represents the average number of clauses satisfied by the local optima across various instances. A high mean height indicates that on average, the solutions are effectively solving many clauses, possibly suggesting easier problems or more effective solution strategies.

- **Standard Deviation:** This metric indicates the variability in the quality of solutions. High standard deviation suggests a wide range of solution effectiveness, which can signal the need for algorithms that are adaptable to diverse solution landscapes.

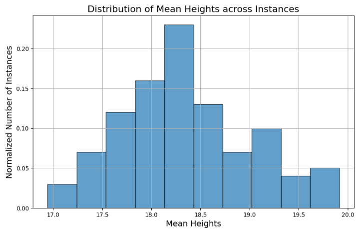
- **Median Heights:** The median gives a central value of solution quality, less affected by extreme performances. It provides a dependable measure of what a typical solution achieves, helping set realistic expectations for algorithm performance.

- **Mean Absolute Deviation (MAD):** Measures how consistently solutions perform in relation to the average. Low MAD indicates that most solutions are close to the average performance, beneficial for predicting the behavior of problem-solving approaches in typical scenarios.

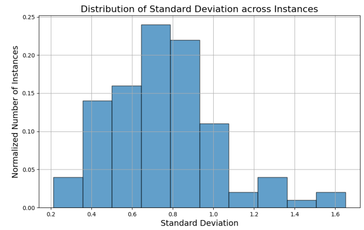
Our study involves experiments where each problem instance consists of 100 clauses to maintain consistent complexity. We analyze 1 million configurations per instance to ensure we explore the solution space comprehensively. This depth of analysis helps us uncover various behaviors of local optima under different conditions.

The significant number of configurations helps us capture a broad range of outcomes, identifying both common patterns and unique cases in the distribution of local optima.

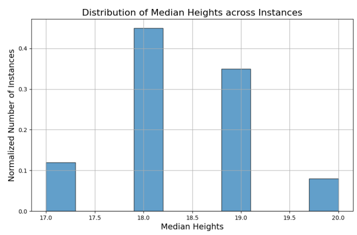
This approach ensures that our data presentation is clear and that the comparisons between different configurations are meaningful. By detailing our experimental parameters and the basis of our normalization, we aim to enhance the clarity of our findings, making it easier to interpret the complexities of MAX-SAT problem landscapes.



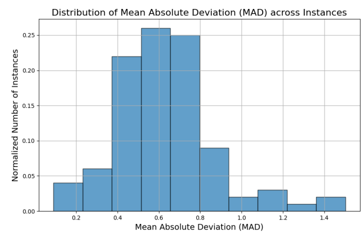
(a) Mean Heights



(b) Standard Deviation

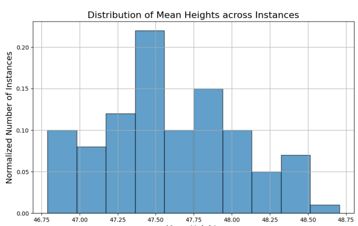


(c) Median Heights

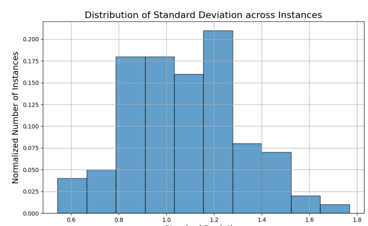


(d) Mean Absolute Deviation

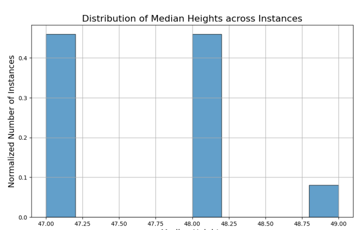
Figure 5: Histograms of metrics for family: $r = 2, n = 10, m = 20$



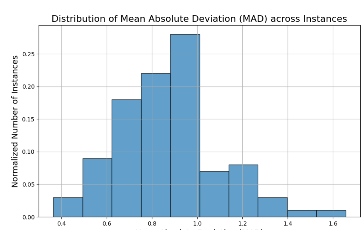
(a) Mean Heights



(b) Standard Deviation



(c) Median Heights



(d) Mean Absolute Deviation

Figure 6: Histograms of metrics for family: $r = 3, n = 10, m = 50$

Our analysis of local optima across two different families of MAX-SAT problems, based on the statistical parameters of Mean Heights, Standard Deviation, Median Heights, and Mean Absolute Deviation (MAD), yielded insightful distinctions:

- **Mean Heights:** For the first family ($r = 2$, $n = 10$, $m = 20$), the mean heights were concentrated around 18.0 to 18.5, indicating a moderate level of solution quality. Conversely, the second family ($r = 3$, $n = 10$, $m = 50$) showed mean heights more densely packed around 47.5 to 48.0, suggesting a higher consistency in achieving near-optimal solutions.

- **Standard Deviation:** The first family displayed a broader range of standard deviation, spanning 0.6 to 1.4, reflecting significant variability in the quality of solutions. The second family exhibited a narrower range of standard deviation, mostly between 0.6 and 0.8, indicating a more consistent level of solution quality across instances.

- **Median Heights:** Median heights for the first family ranged broadly from 17.5 to 19.0, matching the spread observed in mean heights. For the second family, the median heights were tightly clustered around 47.5, reinforcing the higher consistency in solution quality.

- **Mean Absolute Deviation (MAD):** The MAD for the first family varied more widely from 0.4 to 1.2, indicating fluctuating deviations from the mean. In the second family, MAD values were generally lower and less variable, ranging from 0.4 to 0.6, which supports the observation of a more stable solution environment.

These findings suggest that the problem settings with increased complexity (r, n, and m) not only elevate the quality of solutions (as seen in higher mean and median heights) but also reduce variability among outcomes, leading to a more predictable and stable landscape of solutions. The implications of these results are critical for designing optimization strategies, as they reveal that larger, more complex problem settings might yield more consistent and potentially higher-quality solutions, though with reduced diversity among the optima.

5 The Ratio of Same-Height Neighbors Out of All Neighbors of Locally-Optimal Assignments

In this experiment and subsequent, it is important to note that our analysis will focus exclusively on two sets of parameters. These sets were selected to illustrate how varying the number of variables per clause and the total number of clauses can impact the neighborhood characteristics of local optima:

- Set 1: $r = 2$, $n = 40$, $m = 80$, $i = 100$ (number of instances), $a = 10000$ (number of configurations).
- Set 2: $r = 3$, $n = 40$, $m = 200$, $i = 100$, $a = 10000$.

In this experiment, we investigate the neighborhood dynamics of local optima within the solution landscape. Our focus is specifically on quantifying the percentage of neighbors that share the same performance level, or "height," as their corresponding local optimum.

For each local optimum, we calculate the proportion of its neighbors that match its height. For example, if a local optimum satisfies 5 clauses and out of its 6 neighbors, 3 have a height of 5, then 50% of its neighbors share the same height. This metric provides insight into the clustering of solutions around local optima, indicating how frequently solutions with similar success levels occur in close proximity.

Across 100 instances, each with 10,000 configurations, this approach allows us to analyze up to 1 million local optima. The resulting data are visualized on histograms, where the y-axis represents the normalized number of local optima, and the x-axis shows the normalized percentage of neighbors at the same height. This analysis helps us understand the density and distribution of local optima across different configurations of the SAT problem, shedding light on the structural complexity of its solution landscape.

For each set of parameters, we generated a large number of problem instances and solved them to identify local optima and their neighbors with identical heights.

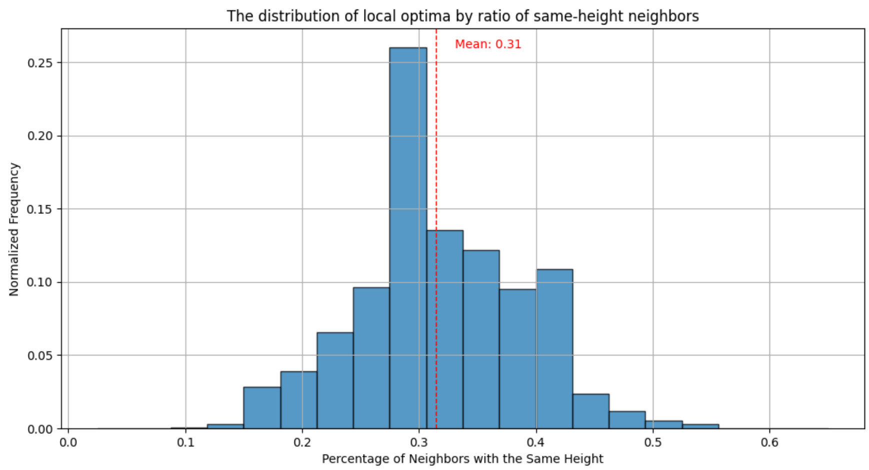


Figure 7: The distribution of local optima by ratio of same-height neighbors, for family $r = 2, n = 40, m = 80$

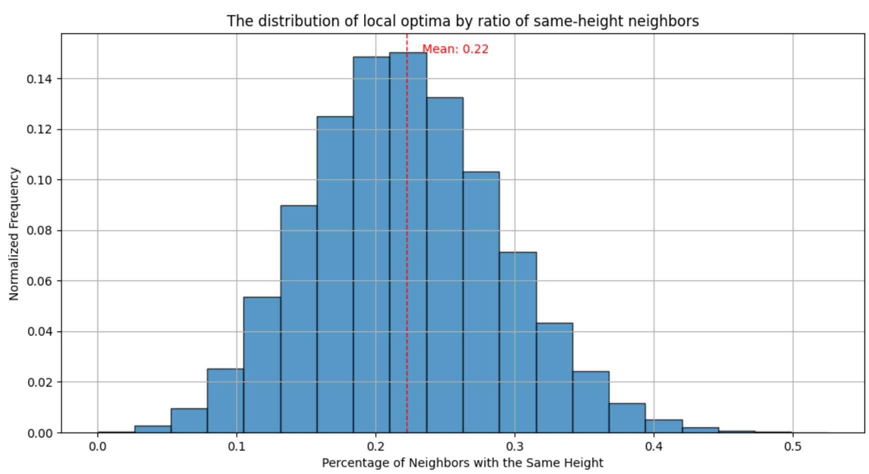


Figure 8: The distribution of local optima by ratio of same-height neighbors, for family $r = 3, n = 40, m = 200$

Our experiment was designed to investigate how the structural parameters of MAX-SAT problems influence the clustering and distribution of local optima. By analyzing two families of problems, each with distinct settings for the number of variables per clause and total clauses, we aimed to understand how these settings impact the solution landscape.

Observations and Analysis:

- **Decrease in Neighbor Similarity:** Family 1 showed a higher average percentage of neighbors with the same height (0.31) compared to Family 2 (0.22). This suggests that having fewer variables per clause allows more solutions to be similarly effective, leading to a denser cluster of local optima. In contrast, increasing the number of variables per clause, as in Family 2, results in a wider variety of solution qualities, reducing the chance that neighboring solutions are equally optimal.

- **Local Optima Density:** The maximum normalized count of local optima was 5% higher in Family 1, indicating a more crowded field of solutions. This implies that simpler clause structures (fewer variables per clause) might create a more navigable landscape where many solutions are close to optimal.

- **Peak Neighbor Count:** The higher peak count of neighbors sharing the same height in Family 1 (0.29 vs. 0.15) further confirms that solutions are more clustered in simpler clause configurations.

- **Maximum Similarity:** The highest proportion of neighbors sharing the same height was greater in Family 1 (0.5 vs. 0.4 in Family 2), illustrating that simpler problems tend to have more solutions that are not only local optima but also closely match each other in effectiveness.

These trends show that more complex problem structures tend to spread out the quality of solutions, making it less likely to encounter clusters of similarly high-quality solutions. This can make it harder for optimization algorithms to pinpoint the absolute best solution, as the landscape does not funnel towards a single peak but rather presents multiple viable peaks.

6 The Hamming-Distance Between Locally-Optimal Assignments

6.1 Analysis of Hamming Distances Between Local Optima

The main goal is to understand the diversity and variability in the configurations of local optima by examining how much they differ from each other in binary terms. This provides insights into the ruggedness of the solution landscape and the distribution of solutions within it.

We start by selecting 100 local optima from various simulated instances of the Max-SAT problem, ensuring these optima are indeed local by verifying no single flip of a variable can improve the solution. We then calculate the normalized Hamming distance between each pair of these optima. This normalization adjusts the raw Hamming distance to a relative scale, where a distance of 0.5 means that half of the bits differ between the two configurations, reflecting significant variation.

- **Mean of Hamming Distances:** This metric provides the average normalized distance across all pairs of local optima, offering a general measure of how diverse the optima are from one another across the entire landscape.

- **Standard Deviation of Hamming Distances:** This metric assesses the variability in the Hamming distances, indicating the consistency of the distances between optima. A higher standard deviation suggests a greater spread in the similarity of configurations, implying a more complex landscape.

- **Median of Hamming Distances:** The median provides a robust measure of central tendency, less affected by outliers than the mean. This metric helps us understand the typical level of difference between the configurations, highlighting the commonality or disparity within the optima.

- **Median Absolute Deviation (MAD) of Hamming Distances:** MAD offers insights into the variability around the median distance. It is particularly useful for identifying the spread of the data around the median, providing a clearer view of variability in landscapes that may have outlier data points.

We visualized our findings using histograms for each metric. These histograms provide a graphical representation of the distribution and central tendency of the normalized Hamming distances, enabling a visual assessment of how spread out the local optima are in terms of their binary configurations.

By conducting this analysis across 100 instances per family and evaluating 1 million configurations for each instance, we were able to ensure a high degree of accuracy in finding 100 local optima per instance.

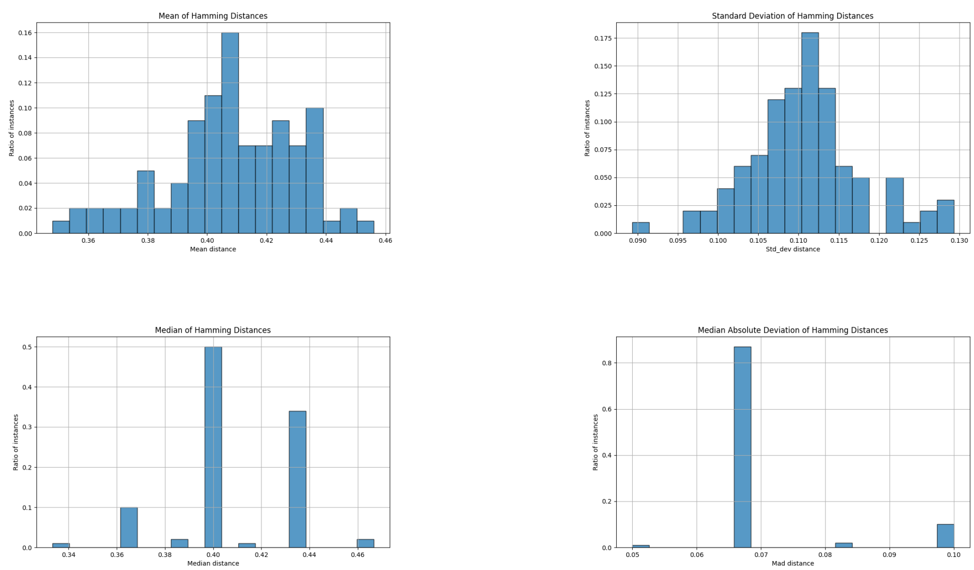


Figure 9: Histograms of distribution of the metrics hamming distance between a random pair of local optima for family: $r = 2, n = 30, m = 60$

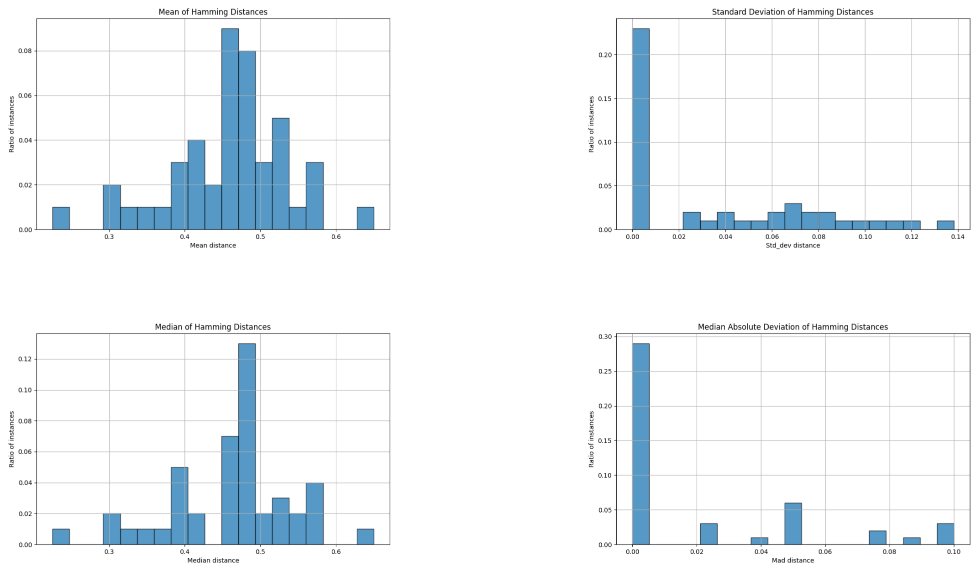


Figure 10: Histograms of distribution of the metrics hamming distance between a random pair of local optima for family: $r = 3, n = 40, m = 200$

In this analysis, we looked at the Hamming distances between local optima for two different problem families:

- Family 1: $r = 2$, $n = 30$, $m = 60$
- Family 2: $r = 3$, $n = 40$, $m = 200$

1. Mean Hamming Distance:
For Family 1 (Figure 12), the mean Hamming distances are around 0.40, meaning that, on average, about 40% of the variables are different between local optima. For Family 2 (Figure 13), the mean Hamming distances are higher, between 0.45 and 0.50, which shows that the local optima are more different from each other. This suggests that as the problem becomes more complex, the differences between local optima increase.

2. Standard Deviation of Hamming Distances:
Family 1 shows standard deviations mostly around 0.11, indicating that the variation in distances between local optima is moderate. For Family 2, the standard deviations are spread out more, with peaks around 0.02, showing

that some local optima are very similar, while others are very different. This means there is a wider range of differences between solutions.

3. Median Hamming Distance:
In Family 1, the median Hamming distance is around 0.40, which is similar to the mean, showing a balanced distribution. In Family 2, the median is a bit higher, around 0.50, meaning the typical difference between local optima is greater in more complex problems.

4. Median Absolute Deviation (MAD):
For Family 1, the MAD is centered around 0.067, showing that most local optima pairs have similar differences, indicating a more consistent solution space. In Family 2, the MAD is spread out, with peaks around 0.0 and 0.1, suggesting that the variation in the differences between local optima is greater. Some local optima are very close, while others are far apart.

As the problem complexity increases from Family 1 to Family 2, we see that the differences between local optima also increase. The greater spread in Hamming distances shows that the solution landscape becomes more diverse and harder to navigate. In more complex problems, local optima are more spread out, which makes it more difficult to move from one solution to another to find the best possible solution. The wider range in the standard deviation and MAD also suggests that the local optima are not evenly distributed, but instead form groups of solutions that are both similar and very different from each other.

6.2 Correlation Between Hamming Distances and Heights of Local Optima

In this experiment, we aimed to investigate the relationship between the average Hamming distances and the heights of local optima. The primary objective was to examine how the complexity of the SAT problem influences the diversity of solutions that perform equally well, as measured by their height. Specifically, we wanted to understand whether more complex problems lead to greater variability among configurations that satisfy the same number of clauses.

To conduct this study, we generated a comprehensive dataset comprising one million distinct configurations for each problem, across 100 different instances. This extensive sample size was deliberately chosen to ensure a high degree of accuracy in identifying 100 local optima at each height level.

By analyzing such a large number of configurations, we aimed to capture a representative distribution of local optima, allowing for a robust analysis of the Hamming distances between them.

Once the local optima were identified, we calculated the Hamming distances between each pair of these optima. To provide a clearer understanding, we normalized these distances, adjusting them based on the total length of the binary strings.

The purpose of focusing on local optima at the same height was to isolate and measure the degree of similarity or difference between configurations that are equally successful in satisfying the SAT problem's clauses. By doing so, we could assess how much variety exists among the solutions that perform at the same level, which in turn provides insights into the structure of the solution space. This analysis helps us understand how the complexity of the SAT problem affects the diversity of configurations that reach the same height, revealing patterns in how similar or distinct the local optima are from one another.

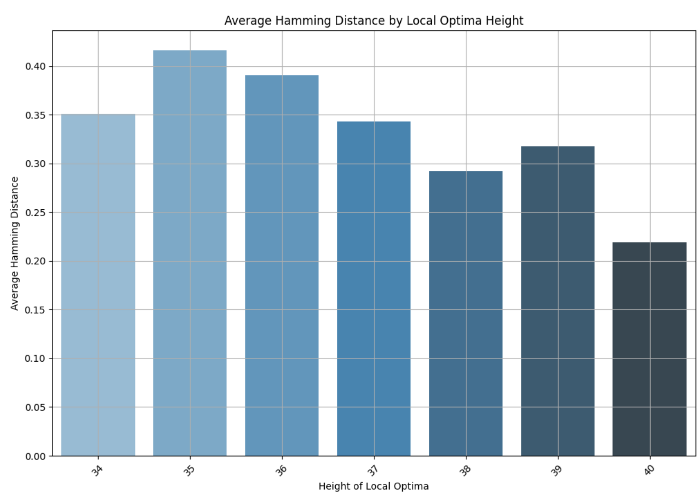


Figure 11: Histogram of average hamming distance between 100 local optima at each height for family: $r = 2, n = 20, m = 40$

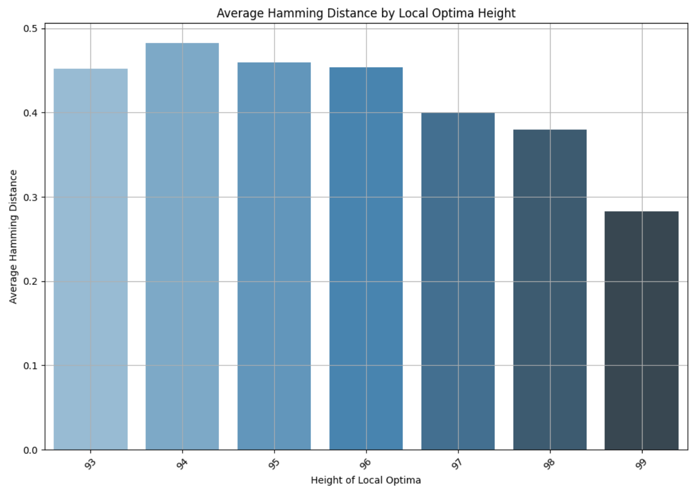


Figure 12: Histogram of average hamming distance between 100 local optima at each height for family: $r = 3, n = 20, m = 100$

Based on the analysis of the provided bar charts, several key observations can be made:

- **Variation in Hamming Distance Across Heights:** For the first family with $r = 2$, $n = 20$, and $m = 40$, the average Hamming distance generally decreases as the height of local optima increases. This suggests that as the number of satisfied clauses increases, the diversity between the configurations of local optima decreases. This might indicate that the solutions are converging to a more similar set of configurations as they achieve higher success in satisfying clauses. For the second family with $r = 3$, $n = 20$, and $m = 100$, a similar trend is observed, but with slightly higher Hamming distances on average. This could be due to the increased complexity introduced by the higher number of clauses ($m = 100$), leading to greater variability in the configurations even among those achieving high success in satisfying clauses.
- **Comparison of Hamming Distances:** The second family shows higher Hamming distances across the heights compared to the first family. This implies that the introduction of more clauses leads to greater diversity among successful configurations, even at similar heights. In

other words, the increased problem complexity results in local optima that are more distinct from each other, indicating a more rugged landscape with potentially more challenging optimization paths.

- **Implications of Height Dependency:** The dependency of the height on the number of clauses (with height directly correlating to the number of clauses satisfied) is clearly illustrated by the spread in the average Hamming distances. In the family with $m = 100$ clauses, the height range is higher, and so are the Hamming distances, suggesting that as the problem's difficulty increases, local optima become more diverse in their configurations.

- **General Conclusion:** The observed trend indicates that as the number of clauses in a problem increases, not only does the height of local optima rise, but the configuration space also becomes more spread out. This increased spread could present additional challenges in identifying global optima, as the search space becomes more complex and diverse.

These results suggest that as the complexity of the problem increases, the solution space becomes more varied, requiring more sophisticated search strategies to effectively navigate through the more rugged landscape of local optima.

7 Conclusion

Our experiments have revealed key structural features of local optima in MAX-SAT problems, offering valuable insights for the development of more effective optimization strategies.

- **Effect of Complexity:** As the number of variables and clauses increases, the landscape becomes denser with local optima. This clustering complicates the search for the global optimum but also suggests a richer space for exploration.

- **Stability Across Instances:** Despite variations in problem parameters, core patterns in the distribution and behavior of local optima remain consistent. This robustness supports the generalization of algorithmic strategies across similar problem families.

- **Diversity Among High-Quality Solutions:** We observed that optimal or near-optimal configurations often differ significantly from one another. This diversity highlights the need for algorithms that balance focused exploitation with broad exploration.

These findings underline the importance of designing search techniques that avoid premature convergence and can adapt to the varying contours of the solution landscape. Knowing that multiple, structurally distinct yet high-performing solutions exist empowers researchers to build more resilient heuristic frameworks.

Future work may include:

- **Broader analysis of problem families to generalize results across other types of combinatorial optimization.**

- **Development of adaptive heuristics that respond dynamically to landscape features such as ruggedness or basin size.**

- **Comparative studies with varying clause-to-variable ratios to test the scalability of current approaches.**

- **Deeper exploration of attraction basins and topological structures in the search space.**

Overall, this study contributes to a deeper understanding of MAX-SAT problem structure and offers a foundation for more advanced and adaptable solving techniques.

A Appendix: Computational Details

All computational experiments and data analyses in this thesis were performed using Python. The implementation included custom scripts for instance generation, evaluation of configurations, and statistical analysis. Experimental parameters such as the number of variables, clauses, and clause sizes were systematically varied in accordance with the problem families studied.

Detailed code and specific parameter settings can be provided upon request.

References

[1] Francisco Chicano, Gabriel Luque, and Enrique Alba. Problem understanding through landscape theory. *University of Málaga*, 2013.

[2] Daniel Berend and Yochai Twitto. Probabilistic characterization of random max r-sat. *Discrete Optimization*, 2021. Available online 5 March 2021.

[3] Magnus Bordewich, Charles Semple, and Kristina Wicke. On the complexity of optimising variants of phylogenetic diversity on phylogenetic networks. *Theoretical Computer Science*, 917:66–80, 2022.

[4] Daniel Selsam, Matthew Lamm, Benedikt Bünz, Percy Liang, David L. Dill, and Leonardo de Moura. Learning a sat solver from single-bit supervision. In *International Conference on Learning Representations (ICLR)*, 2019.

[5] Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. Sat-based maxsat algorithms. *Artificial Intelligence*, 196:77–105, 2013.

[6] Botond Molnár, Ferenc Molnár, Melinda Varga, Zoltán Toroczkai, and Mária Ercsey-Ravasz. A continuous-time maxsat solver with high analog performance. *Nature Communications*, 9(1):4864, 2018.

[7] Peter Merz and Bernd Freisleben. Fitness landscapes and memetic algorithm design. *Department of Electrical Engineering and Computer Science*, 1999.

[8] Josep Argelich, Chu-Min Li, Felip Manyà, and Jordi Planes. The first and second max-sat evaluations. *Journal on Satisfiability, Boolean Modeling and Computation*, 4:251–278, 2008.

[9] Noah Frederik Bruns. Application of multi-objective maxsat-solvers for optimizing highly configurable products. Master's thesis, TU Wien, 2024.

[10] Liangbin Xie, Xintao Wang, Shuwei Shi, Jinjin Gu, Chao Dong, and Ying Shan. Mitigating artifacts in real-world video super-resolution models. *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)*, 2023.

[11] Ruben Martins, Vasco Manquinho, and Inês Lynce. Open-wbo: a modular maxsat solver. *University of Oxford, Department of Computer Science, United Kingdom; INESC-ID / Instituto Superior Técnico, Universidade de Lisboa, Portugal*, 2014.

[12] Jérémie Humeau and Sébastien Verel Arnaud Liefooghe, El-Ghazali Talbi. Paradiseo-mo: From fitness landscape analysis to efficient local search algorithms. *Research Report RR-7871*, 2013. Submitted on 4 June 2013, INRIA.

[13] Xujie Si, Xin Zhang, Radu Grigore, and Mayur Naik. Maximum satisfiability in software analysis: Applications and techniques. In *International Conference on Computer Aided Verification*, pages 447–467. Springer, 2017.

[14] Fahiem Bacchus, Matti Järvisalo, and Ruben Martins. Maxsat evaluation 2018: New developments and detailed results. *Journal on Satisfiability, Boolean Modeling and Computation*, 11:99–131, 2019.

[15] Chuan Luo, Shaowei Cai, Wei Wu, Zhong Jie, and Kaile Su. Ccls: An efficient local search algorithm for weighted maximum satisfiability. *IEEE Transactions on Computers*, 64(7):1830–1843, 2015.

[16] Matthew Gombolay, Reed Jensen, Jessica Stigile, Toni Golen, Neel Shah, Sung-Hyun Son, and Julie Shah. Human-machine collaborative optimization via apprenticeship scheduling. *Journal of Artificial Intelligence Research*, 63:1–49, 2018.