

Computing Short SAT Implicants via Ising/QUBO Encodings

Giuseppe Spallitta ✉ 

Rice University, Houston, TX, USA

Leonardo Duenas-Osorio ✉ 

Rice University, Houston, TX, USA

Moshe Y. Vardi ✉ 

Rice University, Houston, TX, USA

Abstract

Many reasoning tasks require short partial satisfying assignments (implicants), sometimes focusing on a set of important variables. SAT-to-Ising-QUBO formulations are implicitly designed so that ground states correspond to total assignments, since the Ising/QUBO model assigns a value to every spin and has no native representation of unassigned variables. We introduce an Ising/QUBO framework that incorporates “don’t-care” semantics into the quadratic model via a dual-polarity representation, enabling the retrieval of short implicants. The encoding supports implicant shrinking and projection through minor objective modifications. We provide parameter regimes under which ground states correspond to short partial satisfying assignments, achieving minimality and, when the quadratic penalty function permits, minimum-cardinality. We empirically evaluate the encoding with simulated annealing on random 3-SAT enumeration benchmarks and non-CNF formulas, showing that it leaves about one-third of variables unassigned on random 3-SAT formulas while preserving satisfiability, and that consecutive polarity-freezing rounds achieve minimality (and minimum-cardinality) with high probability.

2012 ACM Subject Classification Theory of computation → Logic and verification; Theory of computation → Automated reasoning; Mathematics of computing → Discrete optimization

Keywords and phrases SAT-to-QUBO, Ising encodings, prime implicants, minimum-cardinality implicant, partial and projected assignments

Digital Object Identifier 10.4230/LIPIcs.CP.2026.51

Supplementary Material

Software: https://github.com/giuspek/minimal_assignment_via_Ising

1 Introduction

Many reasoning and knowledge-representation tasks require *short partial satisfying assignments* (implicants), where unassigned variables are genuine don’t-cares [10]. In many applications, only a subset of variables is semantically relevant, and one seeks short partial assignments *over that subset* while allowing the remaining variables to act as hidden support. Such (projected) partial models are central to explanation and diagnosis [22], model compression [18], and enumeration pipelines that compactly represent large solution spaces via projection [16, 17, 15]. More broadly, the SAT literature has long emphasized reasoning with partial assignments in contexts such as abstraction and logic minimization [18, 22]. Dealing with partiality, however, is not trivial. For simplicity and efficiency, modern SAT solvers naturally compute total satisfying assignments, and obtaining short implicants typically requires repeated SAT calls or dedicated post-processing [7].

Interestingly, the same phenomenon appears in SAT-to-Ising/QUBO encodings, i.e., translations that map a CNF formula into a quadratic energy function whose ground states correspond to satisfying assignments. SAT and MaxSAT have been extensively studied in



© Giuseppe Spallitta, Leonardo Duenas-Osorio, and Moshe Y. Vardi;
licensed under Creative Commons License CC-BY 4.0

32nd International Conference on Principles and Practice of Constraint Programming (CP 2026).

Editor: Nicolas Beldiceanu; Article No. 51; pp. 51:1–51:21



Leibniz International Proceedings in Informatics

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

the QUBO and Ising frameworks, motivated by the evolution of quantum annealing and classical Ising heuristics [13, 2, 4, 19, 28]. Still, the logical interpretation of the optimizer output typically corresponds to a total Boolean assignment. This is inherent to the Ising setting, where optimization assigns a value to each Ising variable, thereby providing no native representation of “unassigned” values. As a result, computing short implicants is not captured within the quadratic model itself and is instead left to external algorithms.

This paper introduces a *unified optimization framework* for computing short partial assignments of CNF formulas in Ising/QUBO form, with native support for implicant shrinking and projection. The key idea is to represent, within the quadratic model, the distinction between variables set to false and unassigned variables, so that a QUBO optimizer’s output can be decoded into a partial Boolean assignment. Within this framework, implicant shrinking and projection do not require constructing a distinct optimization model: they are obtained by minor modifications to the original formulation, resulting in a flexible approach to computing partial assignments via quadratic optimization. The encoding could then be used as a drop-in replacement for shrinking subroutines in AllSAT/explanation pipelines [24].

An important theoretical question is under which weight regimes the framework correctly enforces clause satisfaction, partial-assignment semantics, and minimum-cardinality. We therefore identify conditions ensuring that the implicants decoded from ground states have minimum cardinality, and provide a simple criterion for recognizing implicants from the raw QUBO output energy.

► **Remark.** Our experiments are intended as a semantic validation of the encoding: we test whether low-energy configurations decode to short (possibly minimum) partial assignments and whether the expected behavior is visible already under a generic optimization backend.

Contributions

1. A SAT-to-Ising/QUBO encoding that represents three-valued partial assignments (true/false/unassigned) natively via dual-polarity semantics.
2. QUBO variants where implicant shrinking is expressed by polarity restrictions, and where projection penalizes only a set of variables, keeping the QUBO backbone unchanged.
3. A weight-regime analysis establishing conditions under which ground states correspond to a minimum-cardinality implicant, and under which satisfying assignments can be easily detected from just the final energy value.
4. An empirical study on random 3-SAT and non-CNF formulas evaluating the effectiveness of the QUBO encoding to minimal and minimum implicant computation.

2 Preliminaries and Related Work

2.1 The QUBO/Ising Optimization Problem

From an algorithmic viewpoint, quantum annealers and quantum-inspired heuristics can be regarded as solvers for quadratic optimization over binary variables. In particular, they seek low-energy configurations of an Ising Hamiltonian:

$$\arg \min_{\mathbf{z} \in \{-1, 1\}^{|V|}} H(\mathbf{z}), \quad \text{where} \quad H(\mathbf{z}) \stackrel{\text{def}}{=} \sum_{i \in V} h_i z_i + \sum_{(i,j) \in E} J_{i,j} z_i z_j. \quad (1)$$

Here each spin $z_i \in \{-1, 1\}$ is a binary decision variable, and $G = (V, E)$ is an interaction graph specifying which pairs of spins may couple. The coefficients h_i and $J_{i,j}$ determine the energy landscape and thus which configurations are preferred.

An equivalent representation is the *quadratic unconstrained binary optimization* (QUBO) form, which minimizes a quadratic polynomial over Boolean variables $x_i \in \{0, 1\}$:

$$\arg \min_{\mathbf{x} \in \{0,1\}^{|V|}} \mathbf{x}^\top Q \mathbf{x},$$

for some real matrix Q (typically taken upper-triangular, with diagonal terms encoding linear coefficients). The two formulations are interreducible by the affine change of variables $z_i = 2x_i - 1$ (equivalently $x_i = \frac{1}{2}(z_i + 1)$), which converts quadratic terms $z_i z_j$ into quadratic terms $x_i x_j$ plus linear and constant offsets; constants can be dropped without affecting minimization. We therefore use Ising and QUBO interchangeably as notationally convenient (see, [13] for more details).

Since QUBO/Ising minimization is NP-hard, we rely on heuristic optimization algorithms that explore the energy landscape and typically return a low-energy configuration, without a guarantee of optimality. Common classical choices include simulated annealing [12] and tabu/local-search variants [9], while quantum annealers can be used as stochastic samplers that generate many candidate configurations [11]. These methods may get trapped in local minima; their effectiveness depends on how they diversify the search (e.g., via temperature schedules [12], tabu memory [8], or in quantum annealing via reverse annealing [26]).

In practice, one runs the optimization algorithm for multiple trials (often over a small grid of hyperparameters), collects a pool of candidate configurations, and returns the lowest-energy configuration observed. Incremental refinement can further improve solutions: after each batch of samples, the current best configuration is used to bias or initialize the next run, a strategy shown to be effective for Ising/QUBO optimization [5].

2.2 Boolean Formulas, Partial Assignments, and Projection

Let F be a Boolean formula in conjunctive normal form (CNF) over a finite set of propositional variables $\mathcal{V} = \{A_1, \dots, A_n\}$. Let $\mathcal{L} = \{A_i, \neg A_i \mid A_i \in \mathcal{V}\}$ denote the set of all literals over $\mathcal{V}$.

A *total assignment* is a function $\eta : \mathcal{V} \rightarrow \{\perp, \top\}$. A *partial assignment* is a partial function $\mu : \mathcal{V}' \rightarrow \{\perp, \top\}$, with $\mathcal{V}' \subseteq \mathcal{V}$. We write $\text{vars}(\mu) \subseteq \mathcal{V}$ to indicate the variables assigned by μ . We write $\mu_1 \supseteq \mu_2$ to denote that μ_1 extends μ_2 , i.e., $\text{vars}(\mu_1) \supseteq \text{vars}(\mu_2)$ and $\mu_1(v) = \mu_2(v)$ for all $v \in \text{vars}(\mu_2)$. Throughout the paper, assignments will be represented either as sets of literals, for instance $\eta = \{A_1, \neg A_2, A_3\}$, or equivalently as conjunctions of literals, e.g. $\eta = (A_1 \wedge \neg A_2 \wedge A_3)$. The size $|\mu|$ is the number of literals fixed by μ .

A partial assignment μ is an *implicant* of F if it entails F (written $\mu \models F$), meaning each model of μ is a model of F . Equivalently, every total assignment $\eta \supseteq \mu$ satisfies F , i.e.,

$$\forall \eta \supseteq \mu : \quad \eta \models F.$$

Fix a set of *visible* variables $P \subseteq \mathcal{V}$ and let $H \stackrel{\text{def}}{=} \mathcal{V} \setminus P$ be the hidden variables. A *projected (partial) assignment* π is a partial assignment with $\text{vars}(\pi) \subseteq P$. Intuitively, π is a *projected implicant* if it entails the existential projection of F onto the visible variables, i.e., $\pi \models \exists H.F$. Concretely, this means that after fixing π , every completion of the remaining visible variables can still be satisfied by a suitable assignment to the hidden variables:

$$\forall \rho : (P \setminus \text{vars}(\pi)) \rightarrow \{\perp, \top\} \exists \sigma : H \rightarrow \{\perp, \top\} : (\pi \wedge \rho \wedge \sigma) \models F. \quad (2)$$

Let π_μ denote the restriction of the partial assignment μ to P .

2.3 Minimal and Minimum Partial Assignments

An implicant is *inclusion-minimal* iff no subassignment $\mu' \subset \mu$ is an implicant of F (equivalently, for every literal $\ell \in \mu$, the assignment $\mu \setminus \{\ell\}$ is not an implicant of F). In the SAT literature, inclusion-minimal implicants are also called *prime implicants* or just *minimal implicants*. Minimality is a local notion: it rules out redundant literals within μ , but does not compare μ to implicants using different literals. An implicant μ is *minimum-cardinality* if it minimizes $|\mu|$ among all implicants of F . Minimum-cardinality is a global notion, and implies inclusion-minimality, but not conversely: there may exist many prime implicants of different sizes.

In the projected setting, optimality is measured only over the projected variables P , while the hidden variables $H = \mathcal{V} \setminus P$ serve only as auxiliary support. Thus, we measure the size of a projected implicant π by $|\pi|$, i.e., the number of fixed literals over P . A projected implicant π is *projected-prime* if there is no subassignment $\pi' \subset \pi$ that is a projected implicant. A projected implicant π is *projected-minimum* if it minimizes $|\pi|$ among all projected implicants of F . Projected-minimum implies projected-prime, but the converse does not hold.

In the context of this paper, given a CNF formula, we want to construct a quadratic energy function whose ground states correspond to the shortest partial assignments. Thus, our aim is to characterize weight regimes that guarantee that minimum-cardinality implicants are ground states.

2.4 Related Work

Encodings of Boolean satisfiability into Ising models and QUBO objectives have been studied for more than a decade, starting from generic penalty-based reductions from NP-complete problems to quadratic optimization [13]. In these approaches, a CNF formula is translated into an energy function whose minimum corresponds to a satisfying assignment, while clause violations are penalized. Direct quadratic encodings for Max 2-SAT and Max 3-SAT were later proposed to avoid higher-order terms and reduce the number of auxiliary variables [4].

More recent work focused on improving SAT-to-Ising/QUBO encodings, for example, by reducing auxiliary variables, controlling coefficient ranges, or exploiting structural properties of the input formula [28, 19]. Several studies also evaluated SAT-to-Ising on real quantum annealers, analyzing penalty calibration, minor embedding, and robustness to hardware noise and limited precision [2, 1], as well as how formula structure and modular encoding choices affect success probability and solution quality, in particular dealing with prime factorization [5, 6]. Overall, this line of work treats SAT and MaxSAT primarily as optimization problems over quadratic energy functions, where the minimum energy corresponds to an assignment that satisfies the input problem, with no focus on computing partial models or on projection.

Minimum-cardinality implicants of a CNF formula are closely related to classical covering viewpoints: one could reduce the task to selecting a minimum collection of literals that “covers” all clauses, resulting in a set-cover/hitting-set style formulation, and set cover admits standard QUBO/Ising encodings [13]. The set-cover pipeline, however, changes the computational object in a way that undermines the main goals of this work. First, it replaces clause-level satisfiability semantics with a derived structure (literal-clause coverage), so implicant shrinking and projection require generating a different encoding for each assignment. Thus, unlike our approach, current set-cover-style QUBO/Ising encodings do not support shrinking or projection within a single fixed quadratic model. Second, an approach inspired by set cover would destroy a key property of our SAT-to-QUBO encodings: the clause-satisfaction core, penalty calibration, and coefficient regime would need to be reanalyzed and re-tuned

for each derived instance. In contrast, our approach keeps a fixed clause-satisfaction core and realizes partiality, projection, and shrinking by parameter-level modifications inside the same quadratic model, allowing a single weight-regime correctness analysis and calibration strategy to be reused across distinct tasks and assignments.

3 Quadratic Encoding of SAT with Partial Assignment Semantics

3.1 Polarity-Splitting Representation of Variables

Let F be a CNF formula over variables $\mathcal{V} = \{A_1, \dots, A_n\}$ ¹. For each propositional variable A_i , we introduce in the QUBO encoding two binary *polarity spins* (p_i and n_i) intended to encode the positive and negative literal, respectively.

This representation is inspired by the well-known *dual-rail* encoding used in SAT/MaxSAT reasoning, which duplicates each variable into two rails corresponding to its two polarities [3]. Concretely, we interpret each pair (p_i, n_i) as a three-valued assignment:

$$(p_i, n_i) = \begin{cases} (1, 0) & A_i = \top, \\ (0, 1) & A_i = \perp, \\ (0, 0) & A_i \text{ unassigned (don't-care)}, \\ (1, 1) & \text{inconsistent (forbidden)}. \end{cases} \quad (3)$$

To exclude inconsistent pairs, we add the *semantic consistency penalty* function:

$$E_{\text{excl}}(i) = M \cdot p_i n_i$$

where $M > 0$ is a penalty weight, high enough to guarantee assignments inconsistent with the dual rail semantic not to be valid solutions.

3.2 Clause Encoding

Consider a clause $C_k = (\ell_{k,1} \vee \ell_{k,2} \vee \ell_{k,3})$. We now map each literal $\ell_{k,j}$ to its corresponding polarity spin, i.e. we encode the clause $(\llbracket \ell_{k,1} \rrbracket \vee \llbracket \ell_{k,2} \rrbracket \vee \llbracket \ell_{k,3} \rrbracket)$ where:

$$\llbracket \ell_{k,j} \rrbracket = \begin{cases} p_i & \text{if } \ell_{k,j} = A_i, \\ n_i & \text{if } \ell_{k,j} = \neg A_i. \end{cases}$$

Let $a_k = \llbracket \ell_{k,1} \rrbracket$, $b_k = \llbracket \ell_{k,2} \rrbracket$, and $d_k = \llbracket \ell_{k,3} \rrbracket$. A clause is violated exactly when all three corresponding polarity spins are 0. A standard cubic penalty for this is $(1 - a_k)(1 - b_k)(1 - d_k)$, but this is not quadratic. We therefore follow the approach from [4] and introduce one auxiliary variable c_k per clause to capture the “*first-two-literals-false*” condition:

$$c_k = (1 - a_k)(1 - b_k) \equiv (\neg a_k \wedge \neg b_k).$$

Intuitively, $c_k = 1$ means that neither of the first two literals satisfies the clause, and satisfaction must be ensured by the third literal d_k .

We use standard quadratic penalty functions [14] enforcing $z = xy$ over $x, y, z \in \{0, 1\}$:

$$P_{\wedge}(x, y, z) = \lambda(xy - 2xz - 2yz + 3z) \quad (4)$$

¹ We assume a standard preprocessing step in which unit propagation is applied to remove unit clauses, tautological clauses are removed, and duplicate literals inside clauses are eliminated.

■ **Table 1** Truth table for the AND-penalty enforcing $c_k = (1 - a_k)(1 - b_k)$.

a_k	b_k	c_k	$P_\wedge(1 - a_k, 1 - b_k, c_k)$
0	0	0	λ
0	0	1	0
0	1	0	0
0	1	1	λ
1	0	0	0
1	0	1	λ
1	1	0	0
1	1	1	3λ

■ **Table 2** Truth table for $E_{\text{sat}}(k)$: it penalizes exactly the case $c_k = 1$ and $d_k = 0$ (i.e. all three variables in clause k are false).

c_k	d_k	$E_{\text{sat}}(k) = \lambda c_k(1 - d_k)$
0	0	0
0	1	0
1	0	λ
1	1	0

which satisfies $P_\wedge(x, y, z) = 0$ iff $z = xy$, and $P_\wedge(x, y, z) > 0$ otherwise. Here $\lambda > 0$ is the clause-penalty scale used to enforce the optimizer to satisfy the constraint. The higher the λ , the higher the penalty when this does not happen. In our setting we substitute $x = 1 - a_k$, $y = 1 - b_k$, and $z = c_k$, obtaining:

$$P_\wedge(1 - a_k, 1 - b_k, c_k) = \lambda(1 - a_k - b_k + a_k b_k - c_k + 2a_k c_k + 2b_k c_k) \quad (5)$$

Finally, we penalize the unique remaining unsatisfying configuration, namely $c_k = 1$ and $d_k = 0$ (which corresponds to $a_k = b_k = d_k = 0$), by using the penalty function:

$$E_{\text{sat}}(k) = \lambda c_k(1 - d_k)$$

The final *clause penalty* function for clause C_k is therefore:

$$E_{\text{clause}}(k) = P_\wedge(1 - a_k, 1 - b_k, c_k) + \lambda c_k(1 - d_k). \quad (6)$$

The λ in Equation 6 is the same as the one presented in Equation 4. By construction, $E_{\text{clause}}(k) = 0$ iff there exists a setting of c_k such that the clause is satisfied by at least one of the polarity spins. Tables 1 and 2 enumerate the penalty function behavior. Clauses with more than three literals are handled by iterating the same construction: auxiliary clause variables are introduced to recursively aggregate partial penalty functions $P_\wedge$, until the satisfaction of the whole clause is enforced by the final quadratic penalty $E_{\text{sat}}(k)$.

If a clause has only two literals (i.e., after transforming a non-CNF formula via Tseitin encoding), $C_k = (\ell_{k,1} \vee \ell_{k,2})$, then we need not introduce an auxiliary variable. Let $a_k = \llbracket \ell_{k,1} \rrbracket$ and $b_k = \llbracket \ell_{k,2} \rrbracket$. The clause is violated iff $a_k = b_k = 0$, hence we can enforce satisfaction with the quadratic penalty

$$E_{\text{clause}}(k) = \lambda(1 - a_k)(1 - b_k),$$

which is already quadratic and equals 0 iff $a_k \vee b_k$ is satisfied by at least one polarity spin.

3.3 Sparsity-Promoting Term

To bias solutions toward short partial assignments, we add a *sparsity penalty* function

$$E_{\text{sparse}} = \gamma \sum_{i=1}^n (p_i + n_i),$$

which penalizes active spins (i.e., polarity spins set to 1). Here $\gamma > 0$ is the sparsity weight, i.e., the cost of activating a polarity spin. Under proper weight regimes (discussed in more detail in Section 4), clause satisfaction and polarity consistency dominate sparsity, so the optimizer prefers setting $(p_i, n_i) = (0, 0)$ whenever this does not compromise satisfaction. This term is also the mechanism through which we later realize shrinking and projection: by applying sparsity weights only to a designated visible set $P \subseteq \mathcal{V}$, hidden variables may act as unpenalized support for satisfiability.

3.4 The SAT-to-Ising/QUBO Objective

Given F with clauses $C_1, \dots, C_m$ over n variables, the full quadratic energy expression is:

$$E_F = \sum_{i=1}^n M p_i n_i + \sum_{k=1}^m E_{\text{clause}}(k) + \gamma \sum_{i=1}^n (p_i + n_i) \quad (7)$$

The resulting quadratic model uses, in the worst case, $O(n + \sum_{k=1}^m |C_k|)$ binary variables: two polarity spins for each Boolean variable and then auxiliary spins for each $P_\wedge$.

4 Weight Regimes and Minimum-Cardinality Partial Assignments

The intended goal is that ground states correspond to satisfying assignments that activate a minimum number of polarity spins. The QUBO weights must be tuned to avoid ground states corresponding to assignments violating the semantics from Section 3.

► **Theorem 1** (Minimum-cardinality satisfying partial assignments). *Let F be a satisfiable CNF formula and assume the weights satisfy*

$$\lambda > n\gamma \quad \text{and} \quad M > n\gamma.$$

Then every ground state of E_F is (i) consistent, (ii) satisfies all clauses of F , and (iii) minimizes the number of spin polarities set to 1 among all consistent satisfying assignments, thus corresponding to a minimum-cardinality implicant for F . Thus, the encoding reduces the computation of minimum-cardinality implicants to exact ground-state optimization.

Proof. Let E_F be the energy function based on the double polarity spin encoding in Section 3. Let $\mathbf{z}^*$ be a ground state (minimum-energy spin configuration) of E_F , and let η^* be the induced three-valued assignment over $\mathcal{V}$ obtained from $\mathbf{z}^*$ by reverting Equation 3.

Since F is satisfiable, fix an arbitrary total satisfying assignment $\tilde{\eta}$ of F , and let $\tilde{\mathbf{z}}$ be its consistent polarity spins encoding. For this configuration, all clause penalties can be set to 0, hence $E_F(\tilde{\mathbf{z}}) = n\gamma$.

(i) Clause satisfaction. Assume for contradiction that η^* violates at least one clause of F . Consider the minimum value of the clause penalty function for a violated clause (i.e., $a_k = b_k = d_k = 0$). By construction of $E_{\text{clause}}(k)$, after minimizing over c_k the clause contributes at least λ . Therefore

$$E_F(\mathbf{z}^*) \geq \lambda.$$

Since $\lambda > n\gamma$ and $E_F(\tilde{\mathbf{z}}) \leq n\gamma$, we obtain $E_F(\mathbf{z}^*) > E_F(\tilde{\mathbf{z}})$, contradicting that $\mathbf{z}^*$ is a ground state. Hence every ground state satisfies all clauses.

(ii) **Polarity consistency.** Assume for contradiction that $\mathbf{z}^*$ is inconsistent, i.e., there exists a variable A_i with $p_i = n_i = 1$. Then the semantic consistency term contributes M , so

$$E_F(\mathbf{z}^*) \geq M.$$

Using $M > n\gamma$ and again $E_F(\tilde{\mathbf{z}}) \leq n\gamma$, we derive $E_F(\mathbf{z}^*) > E_F(\tilde{\mathbf{z}})$, contradicting that $\mathbf{z}^*$ is a ground state. Therefore every ground state is polarity-consistent.

(iii) **Minimum-cardinality among satisfying consistent assignments.** By (i) and (ii), $\mathbf{z}^*$ is consistent and satisfies all clauses, so all clause penalties and semantic consistency penalties vanish and

$$E_F(\mathbf{z}^*) = \gamma \cdot s(\mathbf{z}^*),$$

where $s(\mathbf{z})$ is the number of active polarity spins in $\mathbf{z}$. Assume for contradiction that there exists another satisfying configuration $\mathbf{z}'$ with strictly fewer active polarities: $s(\mathbf{z}') < s(\mathbf{z}^*)$. Then $E_F(\mathbf{z}') = \gamma s(\mathbf{z}') < \gamma s(\mathbf{z}^*) = E_F(\mathbf{z}^*)$, contradicting that $\mathbf{z}^*$ is a ground state.

Thus $\mathbf{z}^*$ minimizes the number of active polarities among all consistent satisfying configurations, i.e., it corresponds to a minimum-cardinality satisfying partial assignment (and hence, under our CNF notion, to a minimum-cardinality implicant). ◀

► **Corollary 2.** Assume $\lambda > n\gamma$ and $M > n\gamma$. Then any spin configuration $\mathbf{z}$ with $E_F(\mathbf{z}) \leq n\gamma$ is polarity-consistent and satisfies all clauses of F (hence induces a consistent satisfying three-valued assignment under the polarity semantics).

Proof. If $\mathbf{z}$ violates some clause, then the corresponding clause term contributes at least λ , so $E_F(\mathbf{z}) \geq \lambda > n\gamma$, a contradiction. If $\mathbf{z}$ is inconsistent, then the semantic consistency term contributes M , so $E_F(\mathbf{z}) \geq M > n\gamma$, a contradiction. ◀

A concrete admissible choice satisfying the above condition is $\gamma = 1$ and $\lambda = M = n + 1$. These coefficients are integer-valued, have magnitude $O(n)$, and can be rescaled uniformly to meet hardware-dependent bounds without affecting the ground-state semantics. By Corollary 2, under this weight regime any configuration with energy at most $n\gamma$ must be clause-satisfying and polarity-consistent.

► **Example 3.** We illustrate the encoding of the one-clause formula $F = (x_1 \vee \neg x_2 \vee x_3)$. For this clause we have $a_1 = p_1$, $b_1 = n_2$, and $d_1 = p_3$. Introducing the auxiliary spin c_1 , the QUBO energy (for this single-clause instance) is

$$\begin{aligned} E_F = & M(p_1 n_1 + p_2 n_2 + p_3 n_3) + \gamma(p_1 + n_1 + p_2 + n_2 + p_3 + n_3) \\ & + \lambda(1 - p_1 - n_2 + p_1 n_2 - c_1 + 2p_1 c_1 + 2n_2 c_1) + \lambda c_1(1 - p_3). \end{aligned}$$

Throughout the example we assume the weight regime $\lambda > n\gamma$ and $M > n\gamma$ from Section 4, where here $n = 3$.

Case 1: inconsistent polarity. Consider the contradictory polarity assignment $\mathbf{z}_1$:

$$p_1 = n_1 = 1, \quad p_2 = n_2 = p_3 = n_3 = 0.$$

Since $p_1 = 1$, the clause is already satisfied regardless of (n_2, p_3) , hence the Ising model can choose $c_1 = 0$ to make the clause penalties vanish. However, the semantic consistency term contributes M and sparsity contributes 2γ , so

$$E_F(\mathbf{z}_1) = M + 2\gamma.$$

Because $M > 3\gamma$, this energy exceeds the sound bound 3γ for clause satisfaction and consistency.

Case 2: all polarities inactive. Let $\mathbf{z}_2$ set all polarity spins to zero:

$$p_1 = n_1 = p_2 = n_2 = p_3 = n_3 = 0.$$

Then the clause $(x_1 \vee \neg x_2 \vee x_3)$ is violated. Indeed, setting $p_1 = n_2 = p_3 = 0$ forces all three literals to be false. Minimizing over c_1 yields a clause-violation penalty of at least λ : choosing $c_1 = 1$ satisfies the penalty function $c_1 = (1 - p_1)(1 - n_2)$, but then the term $\lambda c_1(1 - p_3)$ contributes λ . Thus:

$$E_F(\mathbf{z}_2) = \lambda,$$

which is strictly larger than 3γ by the assumption $\lambda > 3\gamma$. Therefore, the configuration is not a satisfying assignment ground state.

Case 3: a total satisfying assignment. Let $\eta_1 = x_1 \wedge \neg x_2 \wedge x_3$, mapped to the polarity spin configuration $\mathbf{z}_3$ with polarity encoding:

$$(p_1, n_1) = (1, 0), \quad (p_2, n_2) = (0, 1), \quad (p_3, n_3) = (1, 0).$$

The clause is satisfied because $p_1 = 1$, so we get zero clause penalty. The configuration is consistent, and sparsity activates three polarity spins, hence:

$$E_F(\mathbf{z}_3) = 3\gamma.$$

This matches the certificate bound $n\gamma$ for $n = 3$, so it is consistent with clause satisfaction under the weight regime.

Case 4: a partial satisfying assignment. Finally, consider the partial assignment $\eta_2 = x_1$, setting the polarity spin configuration $\mathbf{z}_4$ to:

$$(p_1, n_1) = (1, 0), \quad (p_2, n_2) = (0, 0), \quad (p_3, n_3) = (0, 0).$$

The clause is satisfied because $p_1 = 1$, so again choosing $c_1 = 0$ sets the clause penalty to zero. The configuration is consistent, and sparsity activates only one polarity, hence:

$$E_F(\mathbf{z}_4) = \gamma.$$

Therefore, η_2 has strictly smaller energy than any total satisfying assignment for this instance, and is preferred by the sparsity term while still satisfying F .

5 Implicant Shrinking via Fixed Polarities

Section 3 considered minimizing E_F over the full search space to obtain minimum-cardinality implicants from scratch. In many workflows, however, a total satisfying assignment η is already available (e.g., produced by a SAT/AllSAT solver), and the goal is to shrink η into a shorter implicant by deleting literals that are not required.

Let η be a total satisfying assignment of F , written as a conjunction of literals

$$\eta = \bigwedge_{i=1}^n \ell_i, \quad \text{where } \ell_i \in \{A_i, \neg A_i\}.$$

We restrict the value of a subset of polarity spins so that we may only drop literals from η :

$$\begin{cases} \eta(A_i) = \top : & n_i = 0, \quad p_i \in \{0, 1\} \text{ free,} \\ \eta(A_i) = \perp : & p_i = 0, \quad n_i \in \{0, 1\} \text{ free.} \end{cases} \quad (8)$$

Thus, for each variable A_i , the polarity spin consistent with η remains switchable (kept if set to 1, dropped if set to 0), while the opposite polarity is forbidden to get to 1.

Under these constraints, every semantically valid polarity spin configuration decodes to a partial assignment $\mu \subseteq \eta$ obtained by deleting some literals of η . In particular, setting the remaining allowed polarity spin to 0 makes A_i unassigned, which matches the implicant semantics: an unassigned variable must be a genuine don't-care, i.e., the formula must remain satisfied for both truth values of that variable.

► **Theorem 4** (Minimum-cardinality implicants within a given assignment). *Assume F is satisfiable and the same weights configuration as in Theorem 1. Let η be a total satisfying assignment of F , and let E_F^η be the energy obtained from E_F by imposing the polarity restrictions from Equation 8. Then every ground state of E_F^η decodes to a partial assignment $\mu \subseteq \eta$ that is an implicant of F and has minimum cardinality among all implicants contained in η . Consequently, μ is also inclusion-minimal within the family $\{\mu' \subseteq \eta \mid \mu' \text{ implicant of } F\}$.*

Proof. Let $\mathbf{z}^*$ be a ground state of E_F^η and let $\mu^* \subseteq \eta$ be its decoded partial assignment.

(1) Clause satisfaction. Assume for contradiction that μ^* does not satisfy some clause anymore (i.e., removing one literal that was necessary for the satisfiability of a clause). By the construction of $E_{\text{clause}}(k)$, after minimizing over the auxiliary variable c_k a violated clause contributes at least λ , hence $E_F^\eta(\mathbf{z}^*) \geq \lambda$. On the other hand, the polarity encoding $\tilde{\mathbf{z}}$ of the original total model η is feasible for the restricted problem and satisfies all clauses, so its energy is purely sparsity: $E_F^\eta(\tilde{\mathbf{z}}) = n\gamma$. Since $\lambda > n\gamma$, we obtain $E_F^\eta(\mathbf{z}^*) > E_F^\eta(\tilde{\mathbf{z}})$, contradicting that $\mathbf{z}^*$ is a ground state. Hence μ^* is clause-satisfying.

(2) Polarity consistency. In the reduced problem E_F^η , for each variable A_i one polarity is hard-fixed to 0 (namely, $n_i = 0$ if $\eta(A_i) = \top$ and $p_i = 0$ if $\eta(A_i) = \perp$). Thus, for every feasible configuration we have $p_i n_i = 0$ for all i , i.e., feasibility already enforces polarity consistency. Thus, the semantic consistency penalties are always 0 on the feasible set.

(3) Minimum cardinality within η . By (1) and (2), all clause and semantic consistency penalties vanish on $\mathbf{z}^*$, so $E_F^\eta(\mathbf{z}^*) = \gamma \cdot s(\mathbf{z}^*)$, i.e., energy is exactly γ times the number of kept literals from η . Assume for contradiction that there exists another implicant $\mu' \subseteq \eta$ with $|\mu'| < |\mu^*|$. Its polarity encoding is feasible for the restricted problem and has energy $E_F^\eta(\mu') = \gamma|\mu'| < \gamma|\mu^*| = E_F^\eta(\mathbf{z}^*)$, contradicting that $\mathbf{z}^*$ is a ground state. Therefore, μ^* has minimum cardinality among implicants consistent with η . ◀

► **Example 5.** Consider the one-clause formula

$$F = (x_1 \vee \neg x_2 \vee x_3), \quad \eta = x_1 \wedge \neg x_2 \wedge x_3.$$

Translated to a polarity spin encoding, η corresponds to

$$(p_1, n_1) = (1, 0), \quad (p_2, n_2) = (0, 1), \quad (p_3, n_3) = (1, 0).$$

To restrict optimization to partial assignments $\mu \subseteq \eta$, we forbid the polarity spins opposite to η for each variable:

$$n_1 = 0, \quad p_2 = 0, \quad n_3 = 0,$$

leaving the compatible polarities p_1, n_2, p_3 free in $\{0, 1\}$.

Since one polarity spin per variable is hard-fixed to 0, every feasible configuration satisfies $p_i n_i = 0$, hence the semantic consistency term vanishes on the restricted search space. Moreover, for the clause $C = (x_1 \vee \neg x_2 \vee x_3)$ we have $a_1 = p_1$, $b_1 = n_2$, $d_1 = p_3$. Minimizing the clause penalty function over c_1 yields:

$$\min_{c_1 \in \{0,1\}} E_{\text{clause}}(1) = \begin{cases} 0 & \text{if } (p_1 \vee n_2 \vee p_3) = 1, \\ \lambda & \text{if } (p_1, n_2, p_3) = (0, 0, 0). \end{cases}$$

Therefore, among satisfying restricted configurations, the objective reduces to the term

$$E_F^\eta = \gamma(p_1 + n_2 + p_3).$$

The satisfying configurations and their energies are thus:

$$\begin{aligned} (p_1, n_2, p_3) &= (1, 1, 1) && \leftrightarrow x_1 \wedge \neg x_2 \wedge x_3, && E_F^\eta = 3\gamma, \\ (p_1, n_2, p_3) &\in \{(1, 1, 0), (1, 0, 1), (0, 1, 1)\} && \leftrightarrow \text{two-literal implicants,} && E_F^\eta = 2\gamma, \\ (p_1, n_2, p_3) &\in \{(1, 0, 0), (0, 1, 0), (0, 0, 1)\} && \leftrightarrow \text{single-literal implicants,} && E_F^\eta = \gamma. \end{aligned}$$

The configuration $(0, 0, 0)$ violates the clause and has energy at least $\lambda > 3\gamma$, hence it cannot be optimal. Every ground state has energy γ and corresponds to a single-literal implicant contained in η , e.g. $(p_1, n_2, p_3) = (1, 0, 0)$ decodes to $\mu = x_1$.

6 Projected Models and Projected Implicant Shrinking

In many reasoning tasks, only a subset of variables is of interest. This situation arises, for instance, when non-CNF formulas are converted into CNF using Tseitin-style transformations [25] or Plaisted–Greenbaum encoding [21]: fresh variables introduced by the transformation are existentially quantified and should not appear in the final models. More generally, explanation and optimization tasks often focus on a set of *visible* variables, with other variables acting as auxiliary support [18].

To bias solutions toward small assignments over P , we keep the clause and semantic consistency penalties unchanged, while sparsity is restricted to variables in P only. We define the *projected sparsity* term and update the final penalty function correspondingly:

$$E_{\text{proj}} = \gamma \sum_{A_i \in P} (p_i + n_i); \quad E_F^{\text{proj}} = \sum_{i=1}^n M p_i n_i + \sum_{k=1}^m E_{\text{clause}}(k) + E_{\text{proj}}$$

Intuitively, hidden variables H may freely take whichever polarity configuration satisfies the clauses, without paying sparsity. Compared to Equation 7, only variables in P are pushed toward the unassigned state $(p_i, n_i) = (0, 0)$ unless they are necessary for clause satisfaction.

We can use the encoding to perform implicant shrinking over a set of important variables HP . In our encoding, this is obtained without changing the clause or consistency terms: we keep the objective E_F^{proj} , but fix the polarity spins of the visible variables so that each $A_i \in P$ may either keep the value prescribed by η or become unassigned, but can never flip. Concretely, for each $A_i \in P$ we impose:

$$n_i = 0 \text{ if } \eta(A_i) = \top, \quad p_i = 0 \text{ if } \eta(A_i) = \perp,$$

Hidden variables H remain unconstrained and can adapt to satisfy the clause penalty function without paying sparsity penalty.

A natural question is whether the projected sparsity encoding guarantees a projected implicant over P that is minimum (projected minimum-cardinality) or even minimal (projected-prime). In general, it does not.

► **Example 6.** Let us consider the formula

$$F = (x \vee y) \wedge (\neg x \vee \neg y), \quad P = \{x\}, \quad H = \{y\}.$$

Under the projected-implicant semantics from Section 2, the empty projected assignment is a valid projected implicant: for $\rho(x) = \top$ choose $\sigma(y) = \perp$, and for $\rho(x) = \perp$ choose $\sigma(y) = \top$. Thus $\pi = \emptyset$ is projected-minimum (it fixes no variable in P).

However, our encoding cannot produce a decoded partial assignment μ that leaves x unassigned. If x is left free, then both completions $x = \top$ and $x = \perp$ must be handled, but in F this forces y to change with x . Since μ is decoded from a single configuration, it cannot accommodate both cases at once.

Recall the standard projected-implicant semantics definition, where π is an implicant if:

$$\forall \rho : (P \setminus \text{vars}(\pi)) \rightarrow \{\perp, \top\} \exists \sigma : H \rightarrow \{\perp, \top\} : (\pi \wedge \rho \wedge \sigma) \models F.$$

In contrast, the assignment μ returned by the QUBO encoding minimization enforces a uniform witness:

$$\exists \sigma^* : H \rightarrow \{\perp, \top\} \text{ s.t. } \forall \rho : (P \setminus \text{vars}(\pi_\mu)) \rightarrow \{\perp, \top\} : (\pi_\mu \wedge \rho \wedge \sigma^*) \models F.$$

which is a different (and strictly stronger) requirement: it must hold for all completions at once, rather than allowing a hidden completion to adapt to the visible completion. Example 6 highlights the source of the mismatch: the value of the hidden variable must change with the completion of the visible one, so no single hidden assignment can witness all visible completions at once when these circular dependencies are in the CNF formulation.

This worst-case limitation does not preclude much stronger behavior in practice, in particular in real-world non-CNF instances. For non-CNF formulas converted into CNF by a Negative Normal Form (NNF) + Plaisted-Greenbaum pipeline, it is known that for every short partial assignment μ over the original variables, if μ entails the original formula, then there exists an assignment σ to the introduced auxiliary variables such that $\mu \wedge \sigma$ entails the CNF encoding [17]. Under this condition, the projected case reduces to the ordinary implicant case: every projected implicant π generated from the QUBO model admits a single hidden assignment σ such that $\pi \wedge \sigma$ entails F . Hence, the minimum-cardinality analysis extends to projected assignments as well; the formal statement is given in Appendix A.

7 Experimental Evaluation

Our experimental goal is the empirical validation of the optimization semantics induced by the QUBO model². Theorems 1 and 4 characterize exact ground states; the experiments assess whether low-energy configurations returned by a generic QUBO optimizer decode to valid implicants and how often they satisfy minimality and minimum-cardinality in practice.

7.1 Benchmarks

The first benchmark consists of satisfiable random 3-SAT instances. We generate 10 instances with n variables ($n \in \{8, 12, 16, \dots, 200\}$) at fixed clauses-to-variables density $m/n = 1.5$. The structure of the solution space is crucial: near the SAT phase transition, at a clause-to-variable ratio around 4.2, random 3-CNF formulae typically admit very few satisfying

² A direct runtime comparison against specialized SAT-side implicant procedures would primarily reflect backend engineering choices on the QUBO optimizer, which are orthogonal to the scope of this paper.

assignments. Thus, it is unlikely that a short partial assignment can cover many total satisfying assignments, since there are few models to compact in the first place. In contrast, lower-density regimes, such as $m/n = 1.5$, admit many satisfying assignments while remaining non-trivial [23], and are therefore better suited for our task.

We also consider 110 non-CNF formulas from recent work [16, 17], with number of variables $n \in \{20, 24, 28, \dots, 60\}$. These instances are Boolean formulas obtained by nesting Boolean operators up to a fixed depth, thus not directly written in CNF. We convert them to CNF using the pipeline in Masina et al. [16, 17]: first NNF, then apply the Plaisted–Greenbaum transformation [21]. The conversion introduces auxiliary variables, which are treated as the hidden set H .

The two benchmark families are chosen to test complementary properties. Random 3-SAT isolates the core minimum-cardinality implicant behavior. The non-CNF benchmarks create a realistic projected setting in which visible and hidden variables must be treated differently.

7.2 Experimental setup

For each instance, we construct the QUBO objective E_F as described in Section 3. We then optimize it using simulated annealing (SA) as a baseline quadratic optimizer, drawing 1000 samples per run and returning the lowest-energy configuration observed. Decoding the polarity spins via Equation 3 returns a partial assignment.

In addition to this one-shot SA run approach (BASIC), we also test an iterative refinement variant (ITER), inspired by Ding et al. [5]. Starting from the best sample returned by SA, we freeze its polarity pattern and re-run SA on the resulting restricted model, pushing additional variables into the don’t-care state. In iterative rounds, we use 100 samples and stop when the assignment no longer shrinks. For people interested in better understanding ITER, we provide a guided example in Appendix B.

We report results under two complementary design choices. First, we consider whether optimization is performed directly on the full QUBO encoding from Section 3 (FULL) or used to *shrink* a pre-computed total satisfying assignment as in Section 5 (SHRINK). Second, we consider whether sparsity is penalized over all variables as in Sections 3-5 (STANDARD) or only over a designated visible set P as in Section 6 (PROJECTED).

7.3 Experiment 1: Compactness of partial assignments

We first measure the compactness of our encoding (i.e., the length of the short assignments produced by BASIC) on the random 3-SAT benchmarks. In standard mode, we report the ratio $|\mu|/n$, whereas in projected mode, we randomly preselect the set P (with $|P|$ being 75% of the total variables) and report the ratio $|\pi|/|P|$. Figure 1 reports mean $\pm$ std over 10 instances per n , while Figures 2 and 3 report the retrieved size per instance via scatterplots.

At fixed density, the ratios are approximately constant with respect to n . In standard mode, FULL returns smaller assignments than SHRINK, as expected: in SHRINK the polarity pattern is fixed by the initial SAT model, so the optimizer can only delete literals under that sign pattern, reducing the degrees of freedom for reaching more compact assignments. The same qualitative trend holds in projected mode.

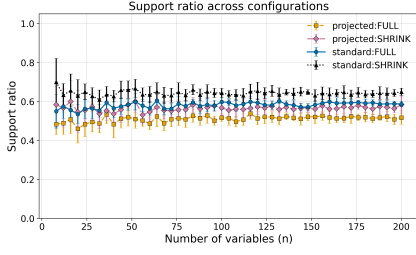


Figure 1 Ratio of assigned variables across configurations (standard: $|\mu|/n$, projected: $|\pi|/|P|$). Mean $\pm$ std over 10 instances per n .

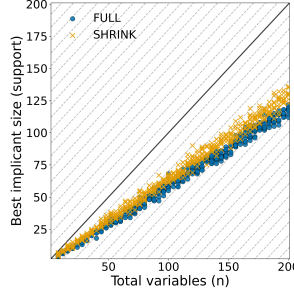


Figure 2 Standard mode: best implicant size $|\mu|$ vs n .

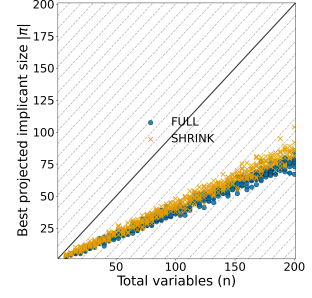


Figure 3 Projected mode: best projected size $|\pi|$ vs n .

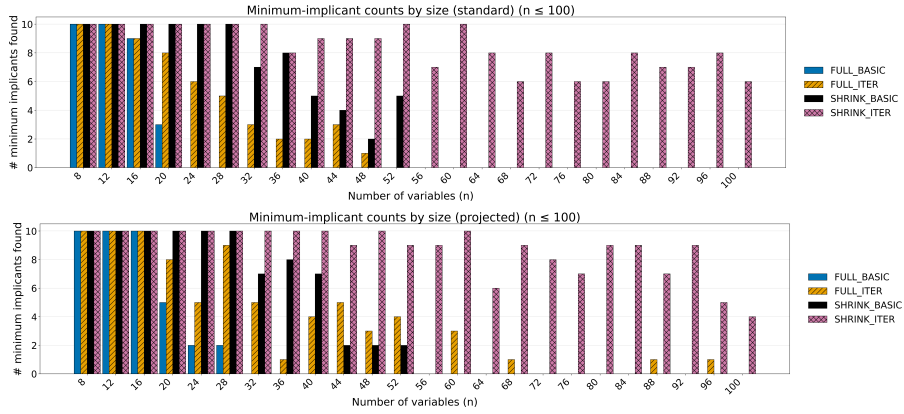


Figure 4 Minimum implicants found for $n \leq 100$. **Top:** standard. **Bottom:** projected.

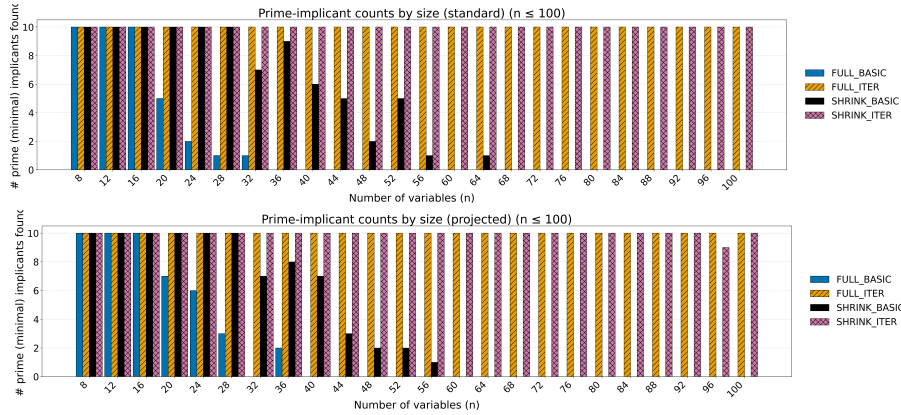
7.4 Experiment 2: Certification of minimality and minimum-cardinality

Because SA does not guarantee reaching a global optimum (as stated in Section 2.1), we explicitly post-check three properties of each returned solution: (i) implicant check, (ii) minimality check, and (iii) minimum-cardinality check. Figures 4 and 5 report, for each $n \leq 100$ (where exact minimum-cardinality verification is feasible), how many instances (out of 10) satisfy these checks under each configuration.

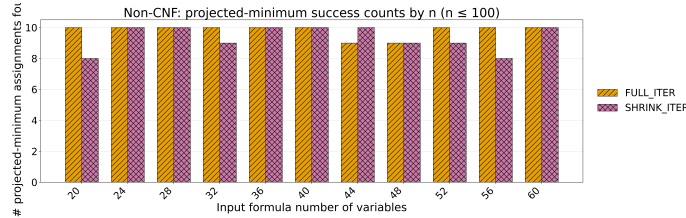
First, all assignments retrieved under both BASIC and ITER satisfy the input formula and have energy below the $n\gamma$ threshold from Corollary 2, as intended. Baseline SA often recovers minimum implicants on small-to-medium instances, but the success rate decreases with n . ITER provides a modest but consistent improvement by further shrinking the best solution found in the initial run. The observed degradation should be interpreted as a search-quality effect, not an encoding-validity effect: there always exists a ground state decoded to a minimum-cardinality assignment, but baseline SA gets stuck to a local minimum instead.

In contrast, minimality is substantially easier to obtain, and iterative refinement markedly improves it. Once the polarity pattern is fixed, the refinement rounds act as a deletion phase, pushing additional variables into the don't-care state while preserving satisfiability, often eliminating redundant literals left by the initial run. On the 3-SAT instances, this improvement typically required only 1 or 2 refinement rounds.

We complement the random 3-SAT evaluation with the non-CNF benchmarks using ITER. Figure 6 reports projected QUBO minimization on the original variables P after CNF conversion of the input problems. More refinement rounds are typically required to reach



■ **Figure 5** Prime (minimal) implicants found for $n \leq 100$. **Top:** standard. **Bottom:** projected.



■ **Figure 6** Projected minimum-cardinality assignments found for non-CNF formulae.

convergence (usually 3 to 5), plausibly due to the more complex induced energy landscape. Even with many auxiliary variables (about 65–70% of all variables per instance), the method frequently recovers minimum-cardinality assignments on P . This behavior is consistent with Section 6: for the NNF + Plaisted–Greenbaum pipeline, short assignments over the original variables admit a uniform single completion over the auxiliary variables. Thus, even with a baseline QUBO optimizer, sampled solutions often decode directly to minimum-cardinality implicants, showing that the target optimum is already practically reachable.

8 Conclusion and Future Work

We introduce a new quadratic Ising/QUBO encoding of SAT aimed at computing short satisfying partial assignments. Each propositional variable is represented by a pair of polarity spins, inducing a three-valued semantics, where setting both polarities to 0 corresponds to a don’t-care state, and a sparsity term biases solutions toward small supports. We identified a penalty regime in which ground states correspond to satisfying assignments with minimum-cardinality support, and we showed that the same encoding can be used to shrink a given SAT model by fixing its polarity pattern, achieving minimum-cardinality implicants.

Two future work directions are most immediate. First, investigating alternative Ising approaches, such as quantum annealers and specialized Ising machines, may yield shorter assignments with higher probabilities. This also requires studies of hardware-aware variants that respect real-device topologies [27]. Second, the encoding can be used as a minimization subroutine inside hybrid AllSAT workflows, replacing implicant-shrinking steps on harder instances or in recent network reliability encodings [20].

References

- 1 Z. Bian, F. Chudak, W. Macready, A. Roy, R. Sebastiani, and S. Varotti. Solving SAT (and MaxSAT) with a quantum annealer: Foundations, encodings, and preliminary results. *Information and Computation*, 275:104609, 2020. doi:10.1016/j.ic.2020.104609.
- 2 Z. Bian, F. A. Chudak, W. G. Macready, A. Roy, R. Sebastiani, and S. Varotti. Solving SAT and MaxSAT with a quantum annealer: Foundations and a preliminary report. In C. Dixon and M. Finger, editors, *Frontiers of Combining Systems – 11th International Symposium, FroCoS 2017, Brasília, Brazil, September 27-29, 2017, Proceedings*, Lecture Notes in Computer Science, pages 153–171. Springer, 2017. doi:10.1007/978-3-319-66167-4_9.
- 3 M. L. Bonet, S. Buss, A. Ignatiev, J. Marques-Silva, and A. Morgado. Maxsat resolution with the dual rail encoding. In S. A. McIlraith and K. Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 6565–6572. AAAI Press, 2018. doi:10.1609/AAAI.V32I1.12204.
- 4 N. Chancellor, S. Zohren, P. A Warburton, S. C Benjamin, and S. Roberts. A direct mapping of max k-SAT and high order parity checks to a chimera graph. *Scientific reports*, 6(1):37107, 2016. doi:10.1038/srep37107.
- 5 J. Ding, G. Spallitta, and R. Sebastiani. Effective prime factorization via quantum annealing by modular locally-structured embedding. *Scientific Reports*, 14(1):3518, 2024. doi:10.1038/s41598-024-53708-7.
- 6 J. Ding, G. Spallitta, and R. Sebastiani. Experimenting with d-wave quantum annealers on prime factorization problems. *Frontiers Comput. Sci.*, 6, 2024. doi:10.3389/FCOMP.2024.1335369.
- 7 D. Déharbe, P. Fontaine, D. Le Berre, and B. Mazure. Computing prime implicants. In *2013 Formal Methods in Computer-Aided Design*, pages 46–52, 2013. doi:10.1109/FMCAD.2013.6679390.
- 8 F. Glover. Tabu search – Part II. *ORSA Journal on computing*, 2(1):4–32, 1990. doi:10.1287/ijoc.2.1.4.
- 9 F. Glover. Tabu search: A tutorial. *Interfaces*, 20(4):74–94, 1990. doi:10.1287/inte.20.4.74.
- 10 A. Ignatiev, A. Previti, and J. Marques-Silva. On finding minimum satisfying assignments. In M. Rueher, editor, *Principles and Practice of Constraint Programming – 22nd International Conference, CP 2016, Toulouse, France, September 5-9, 2016, Proceedings*, Lecture Notes in Computer Science, pages 287–297. Springer, 2016. doi:10.1007/978-3-319-44953-1_19.
- 11 T. Kadowaki and H. Nishimori. Quantum annealing in the transverse ising model. *Phys. Rev. E*, 58:5355–5363, November 1998. doi:10.1103/PhysRevE.58.5355.
- 12 S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983. doi:10.1126/science.220.4598.671.
- 13 A. Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, Volume 2 – 2014, 2014. doi:10.3389/fphy.2014.00005.
- 14 A. Mandal, A. Roy, S. Upadhyay, and H. Ushijima-Mwesigwa. Compressed quadratization of higher order binary optimization problems. In M. Palesi, G. Palermo, C. Graves, and E. Arima, editors, *Proceedings of the 17th ACM International Conference on Computing Frontiers, CF 2020, Catania, Sicily, Italy, May 11-13, 2020*, pages 126–131. ACM, 2020. doi:10.1145/3387902.3392627.
- 15 G. Masina and R. Sebastiani. Exploiting partial-assignment enumeration in optimization modulo theories. In R. Thiemann and C. Weidenbach, editors, *Frontiers of Combining Systems – 15th International Symposium, FroCoS 2025, Reykjavik, Iceland, September 29 – October 1, 2025, Proceedings*, Lecture Notes in Computer Science, pages 117–134. Springer, 2025. doi:10.1007/978-3-032-04167-8_7.

- 16 G. Masina, G. Spallitta, and R. Sebastiani. On CNF conversion for disjoint SAT enumeration. In M. Mahajan and F. Slivovsky, editors, *26th International Conference on Theory and Applications of Satisfiability Testing, SAT 2023, Alghero, Italy, July 4-8, 2023*, LIPIcs, pages 15:1–15:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.SAT.2023.15.
- 17 G. Masina, G. Spallitta, and R. Sebastiani. On CNF conversion for SAT and SMT enumeration. *J. Artif. Intell. Res.*, 83, 2025. doi:10.1613/JAIR.1.16870.
- 18 S. Möhle, R. Sebastiani, and A. Biere. On enumerating short projected models. *Discret. Appl. Math.*, 361:412–439, 2025. doi:10.1016/J.DAM.2024.10.021.
- 19 J. Nüßlein, S. Zielinski, T. Gabor, C. Linnhoff-Popien, and S. Feld. Solving (Max) 3-SAT via Quadratic Unconstrained Binary Optimization. In J. Mikyška, C. de Mulatier, M. Paszynski, V/ V. Krzhizhanovskaya, J. J. Dongarra, and P. M.A. Sloot, editors, *Computational Science – ICCS 2023*, pages 34–47, Cham, 2023. Springer Nature Switzerland. doi:10.1007/978-3-031-36030-5_3.
- 20 R.. Paredes, L. Dueñas-Osorio, K. S. Meel, and M. Y. Vardi. Principled network reliability approximation: A counting-based approach. *Reliab. Eng. Syst. Saf.*, 191, 2019. doi:10.1016/J.RESS.2019.04.025.
- 21 D. A. Plaisted and S. Greenbaum. A structure-preserving clause form translation. *Journal of Symbolic Computation*, 2(3):293–304, 1986. doi:10.1016/S0747-7171(86)80028-1.
- 22 K. Ravi and F. Somenzi. Minimal assignments for bounded model checking. In K. Jensen and A. Podelski, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 31–45, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. doi:10.1007/978-3-540-24730-2_3.
- 23 G. Spallitta, R. Sebastiani, and A. Biere. Disjoint partial enumeration without blocking clauses. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence, AAAI’24/IAAI’24/EAAI’24*. AAAI Press, 2024. doi:10.1609/aaai.v38i8.28652.
- 24 G. Spallitta, R. Sebastiani, and A. Biere. Disjoint projected enumeration for SAT and SMT without blocking clauses. *Artif. Intell.*, 345:104346, 2025. doi:10.1016/J.ARTINT.2025.104346.
- 25 G. S. Tseitin. *On the Complexity of Derivation in Propositional Calculus*, pages 466–483. Springer Berlin Heidelberg, Berlin, Heidelberg, 1983. doi:10.1007/978-3-642-81955-1_28.
- 26 Y. Yamashiro, M. Ohkuwa, H. Nishimori, and D. A. Lidar. Dynamics of reverse annealing for the fully connected p -spin model. *Phys. Rev. A*, 100:052321, November 2019. doi:10.1103/PhysRevA.100.052321.
- 27 S. Zbinden, A. Bärtschi, H. N. Djidjev, and S. J. Eidenbenz. Embedding algorithms for quantum annealers with chimera and pegasus connection topologies. In P. Sadayappan, B. L. Chamberlain, G. Juckeland, and H. Ltaief, editors, *High Performance Computing – 35th International Conference, ISC High Performance 2020, Frankfurt/Main, Germany, June 22-25, 2020, Proceedings*, Lecture Notes in Computer Science, pages 187–206. Springer, 2020. doi:10.1007/978-3-030-50743-5_10.
- 28 S. Zielinski. Pattern QUBOs: Algorithmic construction of 3SAT-to-QUBO transformations. *Electronics*, 12(16):3492, 2023. doi:10.3390/electronics12163492.

A

 Minimum Implicant Projection Proof

► **Theorem 7** (Minimum-cardinality projected satisfying partial assignments). *Let F be a satisfiable CNF formula over variables $\mathcal{V} = P \cup H$, where P is the set of visible variables and H the set of hidden variables. Assume there exists at least one projected implicant over P , and assume the weights satisfy*

$$\lambda > |P|\gamma \quad \text{and} \quad M > |P|\gamma.$$

Assume moreover that the formula F is structured such that for every partial assignment π over P ,

$$\pi \models \exists H. F \iff \exists \sigma \text{ over } H \text{ such that } \pi \wedge \sigma \models F.$$

Then every ground state of the projected energy is (i) consistent, (ii) corresponds to a projected implicant of F under P , and (iii) minimizes the number of active visible polarities among all projected implicants over P . Thus, under the above witness-completion assumption, the encoding reduces the computation of minimum-cardinality projected implicants to exact ground-state optimization.

Proof. Let E_F^{proj} be the projected energy function, where the sparsity term counts only active polarity spins on the visible variables P . Let $\mathbf{z}^*$ be a ground state (minimum-energy spin configuration) of E_F^{proj} , and let π^* be the induced partial assignment over P obtained from $\mathbf{z}^*$ by reverting Equation 3 on the visible variables.

By assumption, fix an arbitrary projected implicant $\tilde{\pi}$ over P , and by assumption let $\tilde{\sigma}$ be a hidden assignment over H such that

$$\tilde{\pi} \wedge \tilde{\sigma} \models F.$$

Let $\tilde{\mathbf{z}}$ be the corresponding consistent polarity-spin encoding. For this configuration, all clause penalties can be set to 0, hence

$$E_F^{\text{proj}}(\tilde{\mathbf{z}}) = \gamma \cdot s_P(\tilde{\mathbf{z}}),$$

where $s_P(\mathbf{z})$ denotes the number of active *visible* polarity spins in $\mathbf{z}$.

(i) Clause satisfaction / projected implication. Assume for contradiction that π^* is not a projected implicant of F . Then

$$\pi^* \not\models \exists H. F.$$

Hence, there exists no assignment σ over H such that

$$\pi^* \wedge \sigma \models F.$$

Equivalently, no assignment to the hidden variables completes π^* to an ordinary implicant of F . By the semantics of the clause-penalty construction, this means that no full consistent spin configuration realizing π^* can make all clause penalties vanish. Therefore, any such configuration incurs at least one clause penalty, and hence its energy is at least λ . Thus

$$E_F^{\text{proj}}(\mathbf{z}^*) \geq \lambda.$$

On the other hand, for the witness-completed projected implicant $\tilde{\pi} \wedge \tilde{\sigma}$ we have

$$E_F^{\text{proj}}(\tilde{\mathbf{z}}) = \gamma \cdot s_P(\tilde{\mathbf{z}}) \leq |P|\gamma.$$

Since $\lambda > |P|\gamma$, then:

$$E_F^{\text{proj}}(\mathbf{z}^\star) > E_F^{\text{proj}}(\tilde{\mathbf{z}}),$$

contradicting that $\mathbf{z}^\star$ is a ground state. Hence, every ground state corresponds on P to a projected implicant of F .

(ii) Polarity consistency. Assume for contradiction that $\mathbf{z}^\star$ is inconsistent, i.e., there exists a variable A_i with both polarity spins active. Then the semantic consistency term contributes M , so

$$E_F^{\text{proj}}(\mathbf{z}^\star) \geq M.$$

Using $M > |P|\gamma$ and again

$$E_F^{\text{proj}}(\tilde{\mathbf{z}}) \leq |P|\gamma,$$

we derive

$$E_F^{\text{proj}}(\mathbf{z}^\star) > E_F^{\text{proj}}(\tilde{\mathbf{z}}),$$

contradicting that $\mathbf{z}^\star$ is a ground state. Therefore, every ground state is polarity-consistent.

(iii) Minimum-cardinality among projected implicants. By (i) and (ii), $\mathbf{z}^\star$ is consistent, and its visible part $\pi^\star$ is a projected implicant of F . Hence, all clause penalties and semantic consistency penalties vanish, and

$$E_F^{\text{proj}}(\mathbf{z}^\star) = \gamma \cdot s_P(\mathbf{z}^\star),$$

where $s_P(\mathbf{z})$ is the number of active visible polarity spins in $\mathbf{z}$.

Assume for contradiction that there exists another projected implicant π' over P with strictly fewer active visible polarities than $\pi^\star$. By the witness-completion assumption, there exists an assignment σ' over H such that

$$\pi' \wedge \sigma' \models F.$$

Let $\mathbf{z}'$ be the corresponding consistent polarity-spin encoding. Then all clause penalties and consistency penalties vanish, so

$$E_F^{\text{proj}}(\mathbf{z}') = \gamma \cdot s_P(\mathbf{z}').$$

Since π' has strictly fewer active visible polarities than $\pi^\star$, we obtain

$$s_P(\mathbf{z}') < s_P(\mathbf{z}^\star),$$

and therefore

$$E_F^{\text{proj}}(\mathbf{z}') < E_F^{\text{proj}}(\mathbf{z}^\star),$$

contradicting that $\mathbf{z}^\star$ is a ground state.

Thus $\mathbf{z}^\star$ minimizes the number of active visible polarities among all projected implicants over P , i.e., it corresponds to a minimum-cardinality projected satisfying partial assignment. $\blacktriangleleft$

B Example of the Iterative Shrinking Procedure

We illustrate the iterative shrinking procedure on the formula

$$F = (x_1 \vee \neg x_2 \vee x_3),$$

starting from the total satisfying assignment

$$\eta^{(0)} = x_1 \wedge \neg x_2 \wedge x_3.$$

In the polarity-splitting representation, $\eta^{(0)}$ is encoded as

$$(p_1, n_1) = (1, 0), \quad (p_2, n_2) = (0, 1), \quad (p_3, n_3) = (1, 0).$$

To shrink $\eta^{(0)}$, we restrict the QUBO so that only literals consistent with $\eta^{(0)}$ can be kept. Thus, the falsified polarities are fixed to zero:

$$n_1 = 0, \quad p_2 = 0, \quad n_3 = 0,$$

while the satisfied polarities p_1, n_2, p_3 remain free. Setting one of these spins to 1 keeps the corresponding literal; setting it to 0 drops the literal, leaving the variable unassigned.

Suppose that the first run of the Ising/QUBO optimizer returns

$$p_1 = 1, \quad n_2 = 1, \quad p_3 = 0.$$

This decodes to the partial assignment

$$\eta^{(1)} = x_1 \wedge \neg x_2.$$

The literal x_3 has been removed, since $(p_3, n_3) = (0, 0)$ now represents that x_3 is unassigned. The assignment $\eta^{(1)}$ still satisfies F , because the clause is already satisfied by x_1 and by $\neg x_2$, and has a lower energy than the initial total assignment.

Since the optimizer is heuristic, this first result is not necessarily a fixpoint. We therefore run the shrinking step again, now starting from

$$\eta^{(1)} = x_1 \wedge \neg x_2.$$

The variable x_3 remains unassigned by fixing both p_3 and n_3 to 0, and the remaining compatible polarities are again allowed to be switched off:

$$n_1 = 0, \quad p_2 = 0, \quad p_1, n_2 \in \{0, 1\}.$$

A second run will be more prone to return a shorter assignment, e.g.:

$$p_1 = 1, \quad n_2 = 0,$$

which decodes to

$$\eta^{(2)} = x_1.$$

This example shows how iterative shrinking works in the QUBO encoding. Polarity spins are progressively switched from 1 to 0, thereby turning fixed literals into unassigned variables. At each iteration, the optimizer may keep pruning active polarities; however, pruning too aggressively would falsify some clauses and incur a large increase in the objective value due to the clause penalty term. Since Ising annealers' goal is to minimize the energy function, in proper weight regimes, we are guaranteed to move to shorter and shorter partial assignments.

Notice that this behavior is effective for encodings that explicitly represent the unassigned state. In a standard SAT-to-Ising/QUBO encoding, each Boolean variable is represented by a single spin, e.g., $z_i = 1$ for True and $z_i = 0$ for False. Hence, every optimizer output is necessarily a total assignment, and re-running the optimizer only produces the same total satisfying assignments, not a smaller implicant.

For example, the formula F could be encoded by a penalty such as

$$(1 - z_1)z_2(1 - z_3),$$

which is zero whenever the clause is satisfied. If $z_1 = 1$, the values of z_2 and z_3 are apparently irrelevant to the satisfaction of the clause. However, since the annealer must assign a value to every spin, the resulting Boolean assignment is necessarily total: in particular, concrete values are still returned for z_2 and z_3 . Thus, although the formula has logical don't-cares, the encoding does not represent them as unassigned variables. Adding a sparsity penalty on the single-spin encoding does not solve this issue: it can only bias variables toward one Boolean value, which differs from setting a variable to “unassigned”. Consequently, an iterative shrinking procedure has no partial state to preserve across rounds: re-optimizing can only produce another total Boolean assignment, not progressively delete literals. This is why repeated optimization can shrink assignments in our polarity-splitting encoding, but not in standard two-valued encodings.