

An Ising Model inspired Extension of the Product-based MP Framework for SAT

Oliver Gableske¹

Ulm University, Theoretical Computer Science, 89077 Ulm, Germany,
`oliver.gableske@uni-ulm.de`, <https://www.gableske.net>

Abstract. Message Passing (MP) has been presented in a unified and consistent notational frame in [7]. The paper explained the product-based MP framework (PMPF) and various MP heuristics (BP, SP, and several interpolations). The paper concluded, that an increased flexibility of MP heuristics (in the form of tunable parameters) leads to a tunable MP algorithm that can be helpful to solve a wide variety of different CNFs. Based on this work, the paper at hand makes three contributions. First, we extend the PMPF regarding flexibility based on theoretical insights from the Ising Model [13]. As an immediate result, this extended PMPF (ePMPF) provides new possibilities for parameter tuning. Second, the ePMPF will also allow us to uncover various theoretical connections between well-known variable and value ordering heuristics for SAT (Zero-first, One-first, occurrence-based heuristics). We show, that these heuristics can be understood as special cases of Message Passing. Third, we show that the ePMPF provides numerous possibilities to tightly integrate Message Passing with various aspects of the CDCL search paradigm.

1 Introduction

The language SAT (comprised of all satisfiable Boolean formulas) is one of the most studied languages in computer science. The interest in the SAT problem (deciding whether a given formula is in SAT) was once purely academic (due to its NP-completeness [4]). Even though there is little hope to find an algorithm that solves the SAT problem (a SAT solver) in an efficient way, ongoing research has constantly improved the performance of SAT solvers during the past decades.

In practice, SAT solvers search for an assignment to the Boolean variables of a given formula under which the formula evaluates to true (a solution). Major advances have been made w.r.t. the design of efficient data-structures for problem representation, the design of paradigms to conduct the search, as well as the design of powerful variable and value ordering heuristics to advance the search.

Our work focuses on Message Passing (MP) algorithms [7], which help to determine variable and value orderings to advance the search. The task of an MP algorithm is to provide biases for the variables of a formula. Roughly speaking, a bias estimates the marginal assignment to a variable in all solutions of a formula. Such biases can be used for both variable ordering (implied by the absolute value of the biases), and value ordering (implied by the signs of the biases) [2, 7, 8].

Numerous MP algorithms have been proposed so far. The first was Belief Propagation (BP) [14], which is based on graphical models (e.g. Bayesian networks) in order to perform inference. Later, Survey Propagation (SP) was proposed [2], which is based on the Ising Model [13] from physics. Since BP and SP might fail to provide biases (by failing to converge), their expectation maximization (EM) variants have been proposed [9–12]. They are derived with the EM algorithm [5] and are guaranteed to provide biases (by guaranteed convergence).

An MP *algorithm* is comprised of two parts: an MP *framework* and an MP *heuristic*. We suggest reading [7] in case the reader is not familiar with MP.

The MP *framework* formalizes how Message Passing works in general. It clarifies that biases are the result of an interaction of variables and clauses facilitated by an exchange of messages (δ and ω equations). This exchange might converge into an equilibrium in which all the messages do not change notably anymore. The framework clarifies that the equilibrium ω -messages are used in order to compute variable freedom values ($\mathcal{T}$ and $\mathcal{F}$ equations). These values are then used in magnetization functions (μ equations) in order to compute biases (β equations). Note, that the MP *framework* explicitly defines all but the δ and μ equations. Therefore, *it governs the overall process to compute biases*.

The MP *heuristic* formalizes how the disrespect- and magnetization-values are computed. Put differently, the MP heuristic provides the definitions for the δ and μ equations. Therefore, *it influences the overall process to compute biases*.

All together, an MP algorithm performs Message Passing according to an MP framework applying an MP heuristic in order to compute biases. As shown in [7], all of the above MP algorithms follow the same product-based MP framework (PMPF), and differ only in the applied MP heuristic. Furthermore [7] showed, that all these heuristics can be combined into a single MP heuristic called $\rho\sigma\text{PMP}^i$. The BP, SP, EMBPG, and EMSPG *heuristics* can all be represented by $\rho\sigma\text{PMP}^i$. Hence, applying $\rho\sigma\text{PMP}^i$ in the PMPF allows us to represent the BP, SP, EMBPG, and EMSPG *algorithms*. This insight will be needed later.

The “helpfulness” of biases provided by a specific MP algorithm depends on the *type of formula* (random 3-SAT, battleship, edge-matching, etc.) that the SAT solver must solve. For example, we might consider the biases computed by the SP algorithm (using the SP heuristic in the PMPF). They can be very helpful to solve large uniform random 3-SAT formulas. In contrast, they are not helpful when trying to solve hard combinatorial formulas [2, 9, 8].

Given a specific type of formula, there are three main approaches to improve the helpfulness of biases in order to increase an MP-based SAT solver’s performance. First, we can design a new MP *heuristic* that *influences* the bias computations to increase their helpfulness. Second, we can design a new MP *framework* that *alters* the bias computations to increase their helpfulness. Third, we can do *both simultaneously* by providing a new MP heuristic *and* a new MP framework.

The first approach can be realized in two ways. First, analyze the formulas in question and derive strong theoretical insights on how the MP heuristic should look like. This is basically what was done in the design of SP [2]. Here, the Ising Model was used in order to acquire theoretical insight strong enough to design

an MP heuristic for random k -SAT. However, this approach is not feasible in general. One cannot possibly analyze all types of formulas and provide a “custom made” MP heuristic for each type. This leads us to the second possibility. One might try to design an MP heuristic with increased flexibility, realized by parameters that can be tuned. Parameter tuning on different types of formulas will then lead to suitable parameter settings for the MP heuristic. This approach was already covered in [7, 8]. The fundamental insight here is that an increased *flexibility of the MP heuristic* provides a meaningful possibility to conduct parameter tuning. This allows us to influence the bias computations on the given type of formula in order to increase an MP-based SAT solver’s performance.

In analogy to the above, the second approach can be realized in two ways as well. First, try to design a specific MP framework based on a theoretical analysis of the formula type. Again, this was done in the design of SP. The Ising Model was used to design the MP framework to conduct Message Passing in SP. But again, following this purely theoretical approach to design “custom made” MP frameworks for the various formula types is not feasible in practice. In analogy to the above, one might try to increase the *flexibility of the MP framework* by introducing tunable parameters. Parameter tuning w.r.t. the current PMPF is impossible since it has no parameters. Hence, this approach has never been tried.

The third approach is basically a combination of the first two, i.e. designing a flexible MP heuristic (using parameters) and a flexible MP framework (using additional parameters). Tuning all parameters simultaneously for a given type of formula should provide the best performance achievable with the means of MP. As stated above, the MP heuristic $\rho\sigma\text{PMP}^i$ can represent all product-based MP heuristics currently available for SAT. We do not see how to improve this any further. We might, however, try to increase the flexibility of the MP *framework*.

The first contribution of this paper is to provide an extended PMPF (ePMPF) with increased flexibility. Recall, that the SP algorithm is based on the Ising Model, which provides concepts to design an MP framework. However, some of these concepts were *ignored* in the design of SP. We will explain these concepts and show, how they can be integrated into the PMPF (Sections 4 and 5).

The ePMPF will provide new possibilities for parameter tuning. However, we will show that the ePMPF also allows us to draw additional theoretical conclusions. More precisely, we will uncover various connections between well-known heuristics like Zero-first, One-first, as well as the conflict avoiding and conflict seeking literal-occurrence based heuristics. The second contribution of this paper is to show, that all these heuristics can be understood as special cases of Message Passing using the ePMPF (Section 6).

Additionally, the ePMPF allows us to tightly integrate MP with various aspects of the CDCL search paradigm (e.g. feedback for the variable ordering based on conflict analysis, clause database maintenance, and restarts). Furthermore, we explain how such a CDCL-based MP-inspired decimation (MID) solver [7] can use a-priori knowledge about solutions of a formula to increase performance. The third contribution of this paper is to elaborate on the possibilities of integrating CDCL and MP, based on the theoretical insights mentioned above (Section 6).

2 Notational Details and Definitions

Let $\mathcal{V} = \{v_1, \dots, v_n\}$ be a set of n Boolean *variables*, and let $\mathcal{L} = \{v, \bar{v} | v \in \mathcal{V}\}$ be the set of *literals*. A clause c is a set of literals, interpreted as a disjunction. A *CNF formula* F is a set of m *clauses*, interpreted as a conjunction. We call $|c|$ the *size* of clause c . A clause c is called *unit*, iff $|c| = 1$. A clause c is called *tautological*, iff $\exists l \in c : \bar{l} \in c$. A literal l is called *pure*, iff $\nexists c \in F : \bar{l} \in c$. We call F a *k-CNF formula*, iff $\max_{c \in F} |c| = k$. Additionally, the sets C_v^+ , C_v^- , and C_v comprise the *sets of clauses* containing v as a positive, negative, or arbitrary literal, respectively. Furthermore, we denote with $L(v, c)$ the *literal* as which v occurs in c , and we denote with $V(l, c)$ the *variable* that corresponds to l in c .

For convenience, we assume that F does not contain tautological clauses or units, and that every variable occurs at least twice as positive literal and at least twice as negative literal. We will use this in Section 5 to ensure that for all $v \in \mathcal{V}$ and for all $l \in \mathcal{L}$ it is $s(l), u(l), t(v), f(v) \geq 2$ (see Definition 7 in [7]).

An *assignment* $\alpha : \mathcal{V} \rightarrow \{0, 1\}$ is called *satisfying* (or a *solution*) for F , iff $\alpha(F)$ evaluates to true (abbr. with $\alpha(F) = 1$). We call an assignment *total*, iff it assigns all variables in $\mathcal{V}$. In case there is at least one satisfying assignment for F , we call F *satisfiable*, and *unsatisfiable* otherwise.

The *language k-SAT* is defined as the set of all satisfiable k -CNF formulas. For a k -CNF F we write $F \in k\text{-SAT}$, if F is satisfiable, and $F \notin k\text{-SAT}$ otherwise.

Let $b \in \{0, 1\}$. Let $A(v, b, F)$ be the set of all total satisfying assignments for F that assign variable v to value b , i.e. $A(v, b, F) = \{\alpha | \alpha(F) = 1, \alpha(v) = b, \alpha \text{ total}\}$. We define the *solution marginal* of variable v for a formula F as the real number $\text{mar}(v, F) = (|A(v, 1, F)| - |A(v, 0, F)|) / (|A(v, 1, F)| + |A(v, 0, F)|) \in [-1, 1]$.

A *factor graph* [14] of a CNF F is an undirected and bipartite graph that contains two types of nodes and two types of edges. The nodes are partitioned into *variable nodes* (one for each $v \in \mathcal{V}$) and *clause nodes* (one for each $c \in F$). The edges are partitioned into *positive and negative edges*. A positive (negative) edge between a variable and a clause node exists, iff the variable appears as positive (negative) literal in the clause. The next section briefly covers the connections between the factor graph, the PMPF, marginals, and biases.

3 The PMPF – From Warnings to Biases

An *MP algorithm* performs Message Passing according to an *MP framework* applying an *MP heuristic* in order to compute biases. The biases are supposed to estimate the solution marginals for the variables. As explained in [7], the biases are the result of an exchange of messages between variable nodes and clause nodes in the factor graph. The PMPF governs this exchange based on *disrespect messages* δ (send from variable nodes to clause nodes) and *warning messages* ω (send from clause nodes to variable nodes). We cannot repeat all the details from [7] here, but it is important to understand that an *MP heuristic* H defines the equation for the disrespect message $\delta_H \in [0, 1]$. The *MP framework* (PMPF) then *uses* δ_H to define the warning message $\omega_H \in [0, 1]$ as well as the literal cavity freedom values $S_H, U_H \in [0, 1]$ as follows.

$${}_z^y\omega_H(c_i, v) = \prod_{l \in c_i \setminus \{L(v, c_i)\}} {}_z^y\delta_H(l, c_i) \quad (1)$$

$${}_z^yS_H(l, c_i) = \prod_{d \in C_l^-} [1 - {}_z^y\omega_H(d, V(l, c_i))] \quad {}_z^yU_H(l, c_i) = \prod_{d \in C_l \setminus \{c_i\}} [1 - {}_z^y\omega_H(d, V(l, c_i))] \quad (2)$$

In summary, an MP algorithm performs a cycle y of iterations z until the following abort condition holds: $\forall c \in F : \forall v \in c : |{}_z^y\omega_H(c, v) - {}_{z-1}^y\omega_H(c, v)| < \omega_{\max}$ (assume $\omega_{\max} = 0.01$). The messages of the iteration in which the convergence was reached are called *equilibrium messages*, denoted ${}_*\delta_H(l, c)$ and ${}_*\omega_H(c, v)$. Based on ${}_*\omega_H(c, v)$, the MP framework defines the *variable freedom* of a variable to be assigned to 1 ($\mathcal{T}$) or 0 ($\mathcal{F}$) as follows.

$${}_y\mathcal{T}_H(v) = \prod_{c \in C_v^-} [1 - {}_*\omega_H(c, v)] \quad {}_y\mathcal{F}_H(v) = \prod_{c \in C_v^+} [1 - {}_*\omega_H(c, v)] \quad (3)$$

The PMPF defines how to use these values to estimate the solution marginal of v , i.e. it defines how to compute the bias ${}_y\beta_H(v)$ with ${}_y\mathcal{T}_H(v)$ and ${}_y\mathcal{F}_H(v)$.

Eqn. 1 will be at the heart of our modifications to the PMPF. Note, that altering this equation will have an impact on the iteration process and will therefore have an impact on the resulting values from Eqn. 3. This in turn will impact the resulting biases. The ultimate purpose of altering Eqn. 1 is to provide promising possibilities for integrating MP with the search paradigm (i.e. CDCL) of the MP-based SAT solver in order to be able to increase its performance.

Arguing about *how well the modifications serve that purpose* is, however, impossible without a theoretical context. As mentioned before, the Ising Model has served as the theoretical basis in the design of the SP framework. However, various concepts from the Ising Model (*forms of agitation*) have been *ignored* in the design of SP, even though these concepts provide strong theoretical insights. The Ising Model furthermore *dictates* the modifications to Eqn. 1 in case we want to harness these insights. This will be discussed further in the next section.

4 The Ising Model and Forms of Agitation

In principle, physicists try to discover fundamental laws that describe our world. One aspect of their work is to derive models that explain properties of condensed matter. Roughly speaking, a condensed-matter object consists of a large number of *sites*. These sites have local properties that can induce a global property of the object. The Ising Model is used to describe and study the interconnections between local site-properties and global object-properties.

For example, we might use a block of iron (the object) and study the iron-atoms' electrons (the sites) w.r.t. a local property (electron spin). A *spin* can be understood as an *angular momentum* (think of a direction that the electron points to in 3D space) [18]. If the spins have a specific "alignment" (local property) the block of iron induces a magnetic field (global property).

Site agitation. The type of matter (iron) implies a connection of the sites (electrons). These connections are called *couplings*. Couplings allow sites to influence each other in their local properties (the spins of the electrons influence each other). This influence is called *site agitation*. One of the fundamental questions that physicists address when studying condensed matter is as follows. *Given a random configuration of the spins and a fixed set of couplings (implied by the matter), can the spins re-arrange (via agitation), such that a stable configuration is reached (in which no further re-arrangement takes place)?* The difficulty of answering this question (and the quality of the answer w.r.t. the real-world) depends on how sophisticated the model is in which we formalize the question.

Intriguingly, this question from physics has a strong connection to the question whether $F \in k\text{-SAT}$. The connection between these two questions can be formalized using the Ising Model [13]. Physics has produced a vast body of literature to answer the above question [2, 18]. The Ising Model allows us to translate results from physics into the domain of theoretical computer science, providing us with new techniques to answer whether a given $k\text{-CNF}$ formula is in $k\text{-SAT}$.

In the Ising Model, we understand the structure of the matter to be a d -dimensional lattice of n sites $x \in \{1, \dots, n\}$. Furthermore, the *spin* s_x of site x is $s_x \in \{+1, -1\}$ (it points either up or down). The *spin configuration* of all sites is denoted $s = (s_1, \dots, s_n) \in \{+1, -1\}^n$. Finally, site agitation is confined to neighboring sites, represented by the edges (x, y) in the lattice. The strength of the coupling associated with the edge (x, y) is given as a *constant* $J_{x,y} \in \mathbb{R}$.

In that sense, the condensed-matter object relates to a $k\text{-CNF}$ formula F , the lattice structure relates to the factor graph, the sites relate to Boolean variables, the spins to variable assignments, the spin configurations to total assignments, and the couplings relate to literal-occurrences in clauses. In addition, asking for a maximally stable spin configuration w.r.t. the condensed-matter object can be understood as asking for an assignment that satisfies the most clauses of F .

Before we can explain any results from physics (and answer the above question), we must define what “stability of a configuration” means in the Ising Model. In order to do this, physicists use a *Hamiltonian Operator*. The Hamiltonian Operator fulfills two tasks. First, it governs the site agitation, thereby describing how spins influence each other via the couplings. Second, it measures the stability of a configuration under a given set of couplings. A smaller value computed by the Hamiltonian refers to a more stable configuration. Various Hamiltonian Operators have been proposed in the literature. We will, however, focus on the *Edwards-Anderson Hamiltonian* (EAH) given as follows [18].

$$\text{EAH}(s, J, \Phi) = - \left[\sum_{(x,y)} J_{x,y} s_x \cdot s_y + \Phi \sum_x s_x \right]$$

The first sum is over all pairs of nearest neighbors in the lattice. The second sum represents the influence by a uniform external magnetic field in which the matter resides, where $\Phi \in \mathbb{R}$ gives the strength and direction of the magnetic field ($\Phi > 0$ means it “points up” and $\Phi < 0$ means it “points down”). The case $\Phi = 0$ is called the zero-field case. Keep in mind, that the $J_{x,y}$ are constants.

The stability of a spin configuration s is implied by $\text{EAH}(s, J, \Phi)$. We call s more stable than s' , iff $\text{EAH}(s, J, \Phi) < \text{EAH}(s', J, \Phi)$. In case $\Phi = 0$, the second sum is zero, and the value computed by the EAH is implied by s alone. During the design of SP, the couplings were restricted to $J_{x,y} \in \{+1, -1\}$. Under this restriction, an increased number of satisfied couplings implies an increased stability. This makes sense when relating the term of “stability” of a spin configuration to the amount of satisfied clauses of a k -CNF formula under a given assignment. However, this restriction is *not required*. Allowing $J_{x,y} \in \mathbb{R}$ as provided by the Ising Model can, as we will later see, be understood as allowing *weighted literal-occurrences* in an MP algorithm. The first extension of the PMPF will introduce literal-weights based on arbitrary coupling-strengths (see the next section).

Magnetic agitation. Note, that a strong positive (negative) magnetic field $\Phi \gg 0$ ($\Phi \ll 0$) will result in a smaller EAH-value as soon as more spins point up (down). A strong magnetic field can be understood to “overrule” the couplings. It forces the spins into the direction of the magnetic field in order for them to realize a stable configuration. We call this influence *magnetic agitation*. In relation to k -SAT, this influence relates to the ability to *prefer* a specific 0-1-ratio in an assignment. The second extension of the PMPF will introduce such 0-1-ratio preferences based on magnetic agitation (see the next section).

Thermal agitation. Given a condensed-matter object in the Ising Model (defined by the couplings J), we might ask the following question. *What happens if a spin configuration does not satisfy all couplings?* Roughly speaking, any object in the real world will try to acquire a state of maximum stability (e.g. liquids cool off, apples fall down, etc.). This, in conjunction with the site agitation governed by the EAH, implies that the spins re-arrange themselves in order to increase stability. In relation to k -SAT, this process of re-arrangement is what Message Passing facilitates by an exchange of messages – the convergence into an equilibrium relates to the re-arrangement of spins into a configuration with maximum stability. The Ising Model allows us to analyze this process in more detail. In order to do so we need to introduce a fundamental equation from physics, called the *partition function* [3].

$$Z(\Upsilon, J, \Phi, n) = \sum_{s \in \{+1, -1\}^n} e^{-T \cdot \text{EAH}(s, J, \Phi)}, \text{ with } T = \frac{1}{b \cdot \Upsilon}$$

Here, b is Boltzmann’s constant, and $\Upsilon \in (0, \infty)$ is the temperature of the matter. The partition function allows us to define the *probability function* P , that gives the probability for a specific spin configuration s to be realized by agitation [3].

$$P(s, \Upsilon, J, \Phi, n) = \frac{e^{-T \cdot \text{EAH}(s, J, \Phi)}}{Z(\Upsilon, J, \Phi, n)}$$

Note, that $\lim_{\Upsilon \rightarrow \infty} P(s, \Upsilon, J, \Phi, n) = \frac{1}{2^n}$, i.e. all configurations have an equal probability to be realized by agitation under high temperature. Let us now investigate $\lim_{\Upsilon \rightarrow 0} P(s, \Upsilon, J, \Phi, n)$. Given the spin configuration s , we want to know how probable its realization by agitation is at zero temperature. We partition the configuration space $\{+1, -1\}^n$ w.r.t. the stability of s as follows.

$$\begin{aligned}
M^+(s) &= \{s' \in \{+1, -1\}^n \mid \text{EAH}(s, J, \Phi) > \text{EAH}(s', J, \Phi)\} \text{ (} s' \text{ is more stable)} \\
M^-(s) &= \{s' \in \{+1, -1\}^n \mid \text{EAH}(s, J, \Phi) < \text{EAH}(s', J, \Phi)\} \text{ (} s' \text{ is less stable)} \\
M^=(s) &= \{s' \in \{+1, -1\}^n \mid \text{EAH}(s, J, \Phi) = \text{EAH}(s', J, \Phi)\} \text{ (} s' \text{ is equally stable)}
\end{aligned}$$

Furthermore, and for reasons of simplicity, instead of calculating the limit for P , we will calculate the limit for the (analytically simplified) reciprocal of P , i.e.

$$\lim_{\mathcal{T} \rightarrow 0} P(s, \mathcal{T}, J, \Phi, n)^{-1} = \sum_{s' \in \{+1, -1\}^n} \lim_{\mathcal{T} \rightarrow 0} e^{[\text{EAH}(s, J, \Phi) - \text{EAH}(s', J, \Phi)]/[b \cdot \mathcal{T}]}$$

Since $\{+1, -1\}^n$ is partitioned into $M^+(s) \cup M^-(s) \cup M^=(s)$, we can separate the above sum into three disjoint parts for $s' \in M^+(s)$, $s' \in M^-(s)$, or $s' \in M^=(s)$. The three separate limits are then calculated as follows.

$$\begin{aligned}
G^+(s) &:= \sum_{s' \in M^+(s)} \lim_{\mathcal{T} \rightarrow 0} e^{[H(s, J, \Phi) - H(s', J, \Phi)]/[b \cdot \mathcal{T}]} = \begin{cases} \infty, & \text{if } M^+(s) \neq \emptyset \\ 0, & \text{otherwise.} \end{cases} \\
G^-(s) &:= \sum_{s' \in M^-(s)} \lim_{\mathcal{T} \rightarrow 0} e^{[\text{EAH}(s, J, \Phi) - \text{EAH}(s', J, \Phi)]/[b \cdot \mathcal{T}]} = 0 \quad (\text{even if } M^-(s) = \emptyset) \\
G^=(s) &:= \sum_{s' \in M^=(s)} \lim_{\mathcal{T} \rightarrow 0} e^{[\text{EAH}(s, J, \Phi) - \text{EAH}(s', J, \Phi)]/[b \cdot \mathcal{T}]} = \sum_{s' \in M^=(s)} \lim_{\mathcal{T} \rightarrow 0} e^0 = |M^=(s)|
\end{aligned}$$

Note, that $s \in M^=(s)$, so $M^=(s) \neq \emptyset$. In summary, for the original limit we get

$$\lim_{\mathcal{T} \rightarrow 0} P(s, \mathcal{T}, J, \Phi, n) = \frac{1}{G^+(s) + G^-(s) + G^=(s)} \stackrel{G^-(s)=0}{=} \frac{1}{G^+(s) + |M^=(s)|}.$$

First, assume there is a configuration s' that is more stable than s . Here, $G^+(s) = \infty$ and $\lim_{\mathcal{T} \rightarrow 0} P(s, \mathcal{T}, J, \Phi, n) = 0$, i.e. the probability for s to be realized by agitation at zero temperature is zero, as soon as a more stable configuration s' exists. Second, assume there is no configuration more stable than s . Here, $G^+(s) = 0$ and $\lim_{\mathcal{T} \rightarrow 0} P(s, \mathcal{T}, J, \Phi, n) = (|M^=(s)|)^{-1}$, i.e. all most stable configurations are equally likely to be realized by agitation at zero temperature.

Obviously, temperature plays an important role in the realization of spin configurations. It makes perfect sense to assume $\mathcal{T} \rightarrow 0$ from a physics perspective, because this will, *in theory*, result in a spin configuration with maximum stability [16]. The problem is, that the *process* that leads to a spin configuration with maximum stability is not specified. All that P does is to provide a “landscape of probabilities” for the spin configurations, which can be *altered* by $\mathcal{T}$.

In practice, we must, however, specify the process that *leads to* a satisfying assignment in the end – the SAT solver’s algorithm. Just instantiating a total assignment using the biases will most likely not yield a solution, because biases are estimations of solution marginals (for single variables), not solutions (total assignments). Therefore, a SAT solver that wants to *make use of these biases* must employ some form of search that leads towards a satisfying assignment.

Currently, the process that is supposed to lead to a satisfying assignment in MP-based SAT solvers is Message Passing Inspired Decimation (MID) [7, 2]. Here, the MP-based SAT solver uses an MP algorithm to compute biases. Then, it assigns the variable with strongest bias to the implied value. After that, it will have the MP algorithm refine the biases for the remaining variables. This MID is repeated until a satisfying assignment is found (or a conflict occurred). In any case, we can understand MID as a gradient-based approach. The biases reflect the gradient, and the search space landscape is implied by the used MP heuristic in conjunction with the PMPF.

The influence of $\mathcal{T}$ on the “landscape of probabilities” implied by P (the thermal agitation) is fundamental – the temperature can drastically change the landscape. In contrast, the landscape implied by the MP heuristic in conjunction with the PMPF, which provides the “bias gradient”, cannot be influenced at all. Put differently, the concept of thermal agitation is missing. The current PMPF assumes $\mathcal{T} \rightarrow 0$, which, as stated above, makes perfect sense from a physics perspective. In practice, altering the bias-gradient by altering the landscape implied by the application of an MP heuristic in the PMPF *might* improve the performance of an MP-based SAT solver. However, it is currently impossible to study this – due to the lack of thermal agitation in the PMPF. Therefore, the third extension of the PMPF will introduce thermal agitation.

In summary, the Ising Model applies three forms of agitation to describe how spin configurations are realized for a given set of couplings. First, site agitation (governed by the EAH) captures the ability of neighboring sites to influence each other. Second, magnetic agitation (governed by Φ) influences all sites simultaneously and might force their spins into a specific direction. Third, thermal agitation (governed by $\mathcal{T}$) influences the probability for a specific configuration to be realized. Currently, the PMPF only uses site agitation with couplings in $\{+1, -1\}$ – it assumes the zero-field case at zero temperature. In what follows, we extended the PMPF s.t. it can apply all forms of agitation described above.

5 Introducing new Forms of Agitation into the PMPF

As stated above, our goal is to extend the PMPF in such a way that it realizes site agitation with arbitrary coupling-strengths, magnetic agitation, as well as thermal agitation. It is, however, important to understand the following fact.

Agitation as described above influences *the stability of configurations*. That is, agitation must influence how the MP algorithm iterates (using δ, ω, U , and S [7]) in order to *influence the equilibrium*. Since the δ equation is defined by an MP heuristic, we cannot use this equation to extend the PMPF. Furthermore, the S and U equations are used as “helpers” to define the δ equations recursively, so we cannot use them either. The only remaining option to introduce the new forms of agitation is a modification of Eqn. 1. All other equations defined by the PMPF take effect *after* convergence and cannot influence the equilibrium.

Site agitation with arbitrary coupling-strength. In principle, the coupling-strength dictates how much influence a coupling between two given sites

has on the configuration stability computed by EAH. The larger $J_{x,y}$, the larger the influence. With respect to CNF formulas we can understand the sites as variables and the couplings as literal-occurrences. A coupling-strength in the Ising Model can therefore be understood as a literal-weight in the context of SAT. We introduce the *set of weights* $\Psi = \{\psi_{i,l} | \psi_{i,l} \in [0, \infty), c_i \in F, l \in c_i\}$, i.e. weight $\psi_{i,l}$ is associated with literal l in clause c_i . Altering Eqn. 1 to

$$\omega_H(c_i, v, \Psi) = \prod_{l \in c_i \setminus \{L(v, c_i)\}} \psi_{i,l} \sqrt{\delta_H(l, c_i, \Psi)} \quad (4)$$

introduces a weight for every literal-occurrence. Recall, that $\delta_H \in [0, 1]$. In case $\psi_{i,l} > 1$ ($\psi_{i,l} < 1$), the root will increase (decrease) the factor for the product, thereby increasing (decreasing) the resulting warning. This influences the values computed in Eqn. 3, and ultimately, the biases. In case we set $\psi_{i,l} = 1$ for all literal-occurrences, Eqn. 4 is similar to Eqn. 1. How literal-weights influence an MP algorithm is discussed in the next section.

Magnetic agitation. As mentioned in the previous section, the external magnetic field (represented by Φ) has the ability to *force* spins into the direction of the field. In case $\Phi \gg 0$ (or $\Phi \ll 0$), the spin configuration will stabilize towards $s = (1, \dots, 1)$ (or $s = (-1, \dots, -1)$), as this will minimize $\text{EAH}(s, J, \Phi)$. A site-spin in the Ising Model can be understood as a variable assignment in the context of SAT (where $s_x = +1$ can be understood as $\alpha(v) = 1$). Introducing magnetic agitation into the PMPF should therefore influence the computation of biases, such that $\Phi \gg 0$ ($\Phi \ll 0$) results in ${}^y\beta_H(v) = 1.0$ (${}^y\beta_H(v) = -1.0$) for all $v \in \mathcal{V}$. Put differently, a strong positive (negative) magnetic field should force the assignment of all variables to 1 (0). Instead of $\Phi \in \mathbb{R}$, we will use $\Phi \in [-1.0, 1.0]$ in the context of SAT. Furthermore, we define the *literal-sign function* as $\text{sgn} : \mathcal{L} \rightarrow \{-1, +1\}$, $\bar{v} \mapsto -1, v \mapsto 1$. We alter Eqn. 4 as follows.

$$\omega_H(c_i, v, \Psi, \Phi) = (1 - |\Phi|) \left[\prod_{l \in c_i \setminus \{L(v, c_i)\}} \psi_{i,l} \sqrt{\delta_H(l, c_i, \Psi, \Phi)} \right] + \frac{|\Phi| + \text{sgn}(L(v, c_i)) \cdot \Phi}{2} \quad (5)$$

In the zero-field case ($\Phi = 0$) this equation is similar to Eqn. 4. Any setting $\Phi \neq 0$ can either increase a warning (if the literal and Φ have the same sign), or decrease the warning (if the literal and Φ have different signs). How the magnetic field influences the MP algorithm will be discussed in the next section.

Thermal agitation. As mentioned in the previous section, the temperature $\mathcal{T}$ influences the probability landscape of all configurations implied by P . The first extremal case is $\mathcal{T} \rightarrow 0$, which is the case used in the original PMPF. The second extremal case is $\mathcal{T} \rightarrow \infty$, which results in equal probabilities for all configurations to be realized by agitation. Based on this, we will introduce thermal agitation into the PMPF. However, instead of using $\mathcal{T} \in (0, \infty)$, we define $\mathcal{T} \in [0, 1]$ in the context of SAT. We alter Eqn. 5 to $\omega_H(c_i, v, \Psi, \Phi, \mathcal{T}) =$

$$(1 - \mathcal{T}) \left\{ (1 - |\Phi|) \left[\prod_{l \in c_i \setminus \{L(v, c_i)\}} \psi_{i,l} \sqrt{\delta_H(l, c_i, \Psi, \Phi, \mathcal{T})} \right] + \frac{|\Phi| + \text{sgn}(L(v, c_i)) \cdot \Phi}{2} \right\} \quad (6)$$

In the zero-temperature case ($\Upsilon = 0$) this equation is similar to Eqn. 5. Setting $\Upsilon = 1$ will result in $\omega_H(c_i, v, \Psi, \Phi, 1) = 0$ for *all* warning messages. How temperature influences the MP algorithm is discussed in the next section.

Replacing Eqn. 1 in the PMPF with Eqn. 6 results in the new ePMPF.

6 Theoretical and Practical Applications of the ePMPF

In this section, we will use the new forms of agitation in the ePMPF (facilitated by Ψ , Φ , and Υ) to show, that several well-known heuristics used in SAT solving (occurrence-based heuristics, Zero-first, One-first) can be understood as special cases of Message Passing. Additionally, we will use the ePMPF to provide several possibilities to tightly integrate MP with CDCL. In summary, this corroborates that the ePMPF is a useful extension of the PMPF in both theory and practice.

The influence of Ψ on Message Passing from a theoretical perspective.

We will first analyze the influence of literal-weights Ψ on the behavior of an MP algorithm with $\Phi = \Upsilon = 0$ (zero-field, zero-temperature). The extremal cases are the infinity-limit $\psi_{i,l} \rightarrow \infty$ for all occurrences (abbr. $\Psi \rightarrow \infty$), and the zero-limit $\psi_{i,l} \rightarrow 0$ for all occurrences (abbr. $\Psi \rightarrow 0$). We assume similar weights for all occurrences, i.e. for $c_i, c_j \in F$ and $l \in c_i, l' \in c_j$, it is $\psi_{i,l} = \psi_{j,l'}$ for all i, j, l, l' . For the remaining section, keep in mind that $s(l), u(l), t(v), f(v) \geq 2$ (Sec. 2).

We first investigate literal-weights in the infinity-limit. Assume, that the MP algorithm uses the MP heuristic $\rho\sigma\text{PMP}^i$ [7] with constant interpolation parameters $\rho \in [0, 1]$ and $\sigma \in (0, 1]$. Additionally, we assume non-extremal initializations of the δ -messages in iteration $z = 0$, i.e. ${}^y\delta_{\rho\sigma\text{PMP}}(l, c_i, \rho, \sigma, \Psi, 0, 0) \in (0, 1)$. Then

$$\lim_{\Psi \rightarrow \infty} {}^y\omega_{\rho\sigma\text{PMP}}(c_i, v, \rho, \sigma, \Psi, 0, 0) = \lim_{\Psi \rightarrow \infty} \prod_{l \in c_i \setminus \{L(v, c_i)\}} \psi_{i,l} \sqrt{{}^y\delta_{\rho\sigma\text{PMP}}(l, c_i, \rho, \sigma, \Psi, 0, 0)} = 1.$$

Using Eqn. 2 gives ${}^yS_{\rho\sigma\text{PMP}}(l, c_i, \rho, \sigma, \Psi, 0, 0) = {}^yU_{\rho\sigma\text{PMP}}(l, c_i, \rho, \sigma, \Psi, 0, 0) = 0$ ($\Psi \rightarrow \infty$) for all literals and clauses in F . These values give the δ -messages for the next iteration $z = 1$. Since $s(l), u(l), t(v), f(v) \geq 2$ we get

$$\lim_{\Psi \rightarrow \infty} {}^y\delta_{\rho\sigma\text{PMP}}(l, c_i, \rho, \sigma, \Psi, 0, 0) = \frac{s(l)}{s(l) + u(l)} > 0. \quad (7)$$

Therefore, $\lim_{\Psi \rightarrow \infty} {}^y\omega_{\rho\sigma\text{PMP}}(c_i, v, \rho, \sigma, \Psi, 0, 0) = 1$ also holds, i.e. the MP algorithm converges in iteration $z = 1$. With Eqn. 3 we get ${}^y\mathcal{T}_{\rho\sigma\text{PMP}}(v, \rho, \sigma, \Psi, 0, 0) = {}^y\mathcal{F}_{\rho\sigma\text{PMP}}(v, \rho, \sigma, \Psi, 0, 0) = 0$ ($\Psi \rightarrow \infty$). The biases in the infinity-limit are

$$\lim_{\Psi \rightarrow \infty} {}^y\beta_{\rho\sigma\text{PMP}}(v, \rho, \sigma, \Psi, 0, 0) = \frac{f(v) - t(v)}{f(v) + t(v)} = \frac{|C_v^+| - |C_v^-|}{|C_v|} \quad \rho \in [0, 1], \sigma \in (0, 1].$$

In case $\sigma > 0$, the MP algorithm (using $\rho\sigma\text{PMP}^i$) computes the “standard” literal occurrence heuristic in the infinity-limit for Ψ . The variable ordering prefers variables with stronger occurrence-imbalances. The value ordering prefers the value that satisfies more clauses (it acts *conflict avoiding*).

In case $\sigma = 0$, the above approach fails (Eqn. 7 does not hold). Here, the $\rho\sigma\text{PMP}^i$ heuristic collapses to ρSP^i [7] and it is unclear what the result of $\lim_{\Psi \rightarrow \infty} {}^y\delta_{\rho\text{SP}}(l, c_i, \rho, \Psi, 0, 0)$ is. Calculating the behavior of ρSP^i (including BP and SP) in the infinity-limit for literal-weights remains an open problem.

In contrast, with $\Psi \rightarrow 0$ we get $\lim_{\Psi \rightarrow 0} {}^y\omega_{\rho\sigma\text{PMP}}(c_i, v, \rho, \sigma, \Psi, 0, 0) = 0$. This results in ${}_0S_{\rho\sigma\text{PMP}}(l, c_i, \rho, \sigma, \Psi, 0, 0) = {}^yU_{\rho\sigma\text{PMP}}(l, c_i, \rho, \sigma, \Psi, 0, 0) = 1$ ($\Psi \rightarrow 0$). Since $u(l), s(l), t(v), f(v) \geq 2$, we can use arbitrary $\rho, \sigma \in [0, 1]$ to show that

$$\lim_{\Psi \rightarrow 0} {}^y\delta_{\rho\sigma\text{PMP}}(l, c_i, \rho, \sigma, \Psi, 0, 0) = \frac{1 - \rho + \sigma[u(l) + s(l) - 1 - \rho(s(l) - 1)]}{(1 - \sigma)(2 - \rho) + 2\sigma[u(l) + s(l)]} < 1.$$

Therefore, $\lim_{\Psi \rightarrow 0} {}^y\omega_{\rho\sigma\text{PMP}}(c_i, v, \rho, \sigma, \Psi, 0, 0) = 0$ also holds, i.e. the MP algorithm converges in iteration $z = 1$. This in turn gives ${}^y\mathcal{T}_{\rho\sigma\text{PMP}}(v, \rho, \sigma, \Psi, 0, 0) = {}^y\mathcal{F}_{\rho\sigma\text{PMP}}(v, \rho, \sigma, \Psi, 0, 0) = 1$ ($\Psi \rightarrow 0$), and $\lim_{\Psi \rightarrow 0} {}^y\beta_{\rho\sigma\text{PMP}}(v, \rho, \sigma, \Psi, 0, 0) =$

$$\frac{\rho\sigma[t(v) - f(v)]}{(1 - \sigma)(2 - \rho) + 2(t(v) + f(v))} \stackrel{\rho=1}{\stackrel{\sigma=1}{\equiv}} \frac{|C_v^-| - |C_v^+|}{2|C_v|} \stackrel{(*)}{\approx} \frac{|C_v^-| - |C_v^+|}{|C_v|}$$

In case $\rho = \sigma = 1$ and after re-scaling all biases by a factor of 2 ($\star$), the MP algorithm (using $\rho\sigma\text{PMP}^i$) computes a literal occurrence heuristic in the zero-limit for Ψ as well. Again, the variable ordering prefers variables with stronger occurrence-imbalances. In contrast to the infinity-limit case, the value ordering now prefers values that satisfy *less* clauses (it acts *conflict seeking*).

In summary, the extremal cases for Ψ result in biases that reflect the “standard” literal occurrence heuristics (conflict avoiding with $\sigma > 0$ and $\Psi \rightarrow \infty$, and conflict seeking with $\rho = \sigma = 1$ and $\Psi \rightarrow 0$). Note, that intermediate settings to Ψ allow for a *seamless adjustment* of the MP algorithm’s behavior. Roughly speaking, Ψ allows us to seamlessly control whether we want to avoid or seek conflicts during MID in the search of an MP-based SAT solver.

The influence of Ψ in CDCL-based MID from a practical perspective.

In practice, CDCL solvers [15] use heuristics to influence how search advances, but search might also *provide feedback to the heuristic*. For example, the VSIDS heuristic [17] uses variable activities to compute a variable ordering according to which the variables are assigned during search. However, if the CDCL runs into a conflict it performs conflict analysis which will (among other things) *alter* the variable activities. Hence, advancing the search also influences VSIDS. Such bidirectional influence is crucial for the performance of CDCL solvers. How strong the feedback is (e.g. for VSIDS) is controlled by a parameter of the solver.

In contrast, the search of a CDCL-based MID solver, that uses the original PMPF, *cannot provide feedback to the MP algorithm* – the PMPF lacks the necessary flexibility. However, the ePMPF allows the search to provide feedback by altering the literal-weights in Ψ . In what follows, we will briefly discuss how such feedback can be used to integrate MP with various aspects of the CDCL search.

Assume, that the formula F is satisfiable and that we use a CDCL-based MID solver (using the ePMPF) to solve it. Ideally, following the bias-gradient during MID (extending the decision stack) results in a satisfying assignment α .

However, in practice, the CDCL will run into conflicts, i.e. $\exists c_i \in F : \alpha(c_i) = 0$ during search. Conflict analysis (CA) resolves the conflict and adds a new learned clause d to F (implying a back-jump) [19]. However, adding a single (possibly very long) clause will most likely have no influence on the biases. Arguably, the impact that c_i had on the bias computation (that led to α) was not strong enough. This implies, that we must increase the weights $\psi_{i,l}$ of literals in c_i for future computations. Additionally, we can also increase the weights of literals in clauses that act as reasons in the conflict graph (similar to the activity increase of variables in VSIDS). Note, that we can understand clause learning (adding d) as empowering unit propagation (UP) to avoid previously discovered conflicts. Similarly, we can understand weight increases as empowering MP to provide biases that avoid previously discovered conflicts. The CDCL-based MID solver can do both, which constitutes a tight integration of CA, UP, and MP via Ψ .

Additionally, CDCL solvers perform clause database maintenance (CDBM) that removes learned clauses from the formula. Various heuristics for CDBM exist, e.g. the activity-based CDBM heuristic that associates an activity value with every learned clause [6]. During conflict analysis, the activity of learned clauses is increased in case they help to resolve the conflict. Then, maintenance will remove clauses that have a relatively small activity. This in turn increases the performance of UP (less clauses must be checked). In that sense, activity-based CDBM can be understood as a tool to improve the performance of UP. Instead of using activity values for the clauses, we might rely on the $\psi_{i,l}$ values to perform the same scheme of CDBM. Here, removing learned clauses with small literal-weights can be understood as improving the performance of UP *and* MP. This constitutes a tight integration of CDBM with UP and MP via Ψ .

Altering literal-weights during search will, eventually, result in biases (using the updated Ψ) that strongly differ from the partial assignment on the decision stack of the CDCL-based MID solver. Arguably, reverting all decisions and restarting search with the new biases increases the chances for MID to find a solution. In that sense, we can also integrate restarts with MP via Ψ .

The influence of Φ on Message Passing from a theoretical perspective.

We now analyze the influence of the magnetic field parameter Φ on the behavior of an MP algorithm with $\mathcal{T} = 0$ and $\Psi = 1$ (zero-temperature, unit-weights). The first extremal case is $\Phi = -1$. Using Eqn. 6 in iteration $z = 0$ results in two possible warnings. First, ${}^y\omega_{\rho\sigma\text{PMP}}(c_i, v, \rho, \sigma, 1, -1, 0) = 0$ iff $l = v$. Second, ${}^y\omega_{\rho\sigma\text{PMP}}(c_i, v, \rho, \sigma, 1, -1, 0) = 1$ iff $l = \bar{v}$. Note, that these results are *independent* of the δ messages. Hence, the same warnings will be computed in iteration $z = 1$ and the MP algorithm converges. Using Eqn. 3 gives us ${}^y\mathcal{T}_{\rho\sigma\text{PMP}}(v, \rho, \sigma, 1, -1, 0) = 0$ and ${}^y\mathcal{F}_{\rho\sigma\text{PMP}}(v, \rho, \sigma, 1, -1, 0) = 1$. Therefore, ${}^y\beta_{\rho\sigma\text{PMP}}(v, \rho, \sigma, 1, -1, 0) = -1.0 \forall v \in \mathcal{V}$. Put differently, the MP algorithm provides biases that reflect the Zero-first heuristic in case $\Phi = -1$. Similarly, we can show that the MP algorithm provides biases that reflect the One-first value ordering heuristic with $\Phi = +1$. In analogy to the Ising Model, it is possible to use $\Phi \in [-1, +1]$ in order to force the variable biases into a specific direction. This allows us to *seamlessly* influence the MP algorithm to prefer variable biases

with a specific value, i.e. we can “ask” for biases that reflect a specific 0-1-ratio.

The influence of Φ in CDCL-based MID from a practical perspective.

Assume, that F is satisfiable and that we use a CDCL-based MID solver (that performs MP according to the ePMPF) to solve it. Assume furthermore, that we know the 0-1-ratio of solutions in advance.

The assumption, that we know the 0-1-ratios of solutions in advance might seem very artificial. However, there are problems that, when encoded to SAT, allow us to make such assumptions. For example, consider the Graph Isomorphism (GI) problem of two graphs with $n \geq 1$ nodes. Translating the problem to SAT can be achieved in a trivial way by representing a mapping with n^2 variables, where $v_{i,j} = 1$ iff node i is mapped to node j . The formula can be constructed s.t. it is satisfiable iff an isomorphism between the two graphs exists. In this case, the solution must assign exactly n variables to 1, and $n^2 - n$ variable to 0, respectively. Therefore, we know the 0-1-ratio of solutions in case they exist. We do, however, not know which variable must be assigned to what value – which is precisely what the CDCL-based MID solver must determine.

The setting to the field parameter Φ , that is implied by the observations above, is $\Phi = (2/n) - 1 \in [-1, +1]$. Using this will force the MP algorithm into an equilibrium that prefers biases with the desired 0-1-ratio. Arguably, these biases will be more helpful to find a satisfying assignment with the CDCL-based MID solver. Note, that $\lim_{n \rightarrow \infty} (2/n) - 1 = -1$, i.e. the Zero-first heuristic should be used in case the graphs become very large in this trivial encoding.

The influence of Υ on Message Passing from a theoretical perspective.

We will now analyze the influence of the temperature parameter Υ on the behavior of an MP algorithm with $\Phi = 0$ and $\Psi = 1$ (zero-field, unit-weights). First, note that $\Upsilon = 1$ results in ${}^y\omega_{\rho\sigma\text{PMP}}(c_i, v, \rho, \sigma, 1, 0, 1) = 0$ for all literals and clauses. Therefore, the MP algorithm converges in iteration $z = 1$ with ${}^y\mathcal{T}_{\rho\sigma\text{PMP}}(v, \rho, \sigma, 1, 0, 1) = {}^y\mathcal{F}_{\rho\sigma\text{PMP}}(v, \rho, \sigma, 1, 0, 1) = 1$. Finally, we get $\forall v \in \mathcal{V} : {}^y\beta_{\rho\sigma\text{PMP}}(v, \rho, \sigma, 1, 0, 1) = 0.0$. This means, that under high temperature, all variables are unbiased and all assignments are estimated to be satisfying with equal probability. This is what $\lim_{\Upsilon \rightarrow \infty} P(s, \Upsilon, J, \Phi, n) = \frac{1}{2^n}$ means in physics.

A setting to $\Upsilon \in (0, 1]$ allows us to enforce the upper boundary $1 - \Upsilon \in [0, 1]$ for the warnings. This in turn enforces that $\mathcal{T}, \mathcal{F} > 0$. Therefore, we can alter the biases, and ultimately, the bias-gradient used by MID solvers.

Let $\gamma \in (0, 1]$ s.t. $\gamma_{max} = \gamma^{-1} \in \mathbb{N}$ (e.g. $\gamma = 0.1$ which gives $\gamma_{max} = 10$). We can use γ to determine a sequence of temperatures $\Gamma = (0, \gamma, 2\gamma, \dots, 1) = (t\gamma)_{t=0}^{\gamma_{max}}$. We can perform MP sequentially using $\Upsilon \in \{t\gamma | t\gamma \in \Gamma\}$. Due to the above, we know that in general $\Upsilon = 0\gamma = 0$ gives ${}^y\beta_{\rho\sigma\text{PMP}}(v, \rho, \sigma, 1, 0, 0) \neq 0 \forall v \in \mathcal{V}$. We also know, that $\Upsilon = \gamma_{max}\gamma = 1$ gives ${}^y\beta_{\rho\sigma\text{PMP}}(v, \rho, \sigma, 1, 0, 1) = 0 \forall v \in \mathcal{V}$. Put differently, the biases decay when we increase the temperature. However, note that biases do not necessarily decay equally fast. Some variables will have a bias even for large Υ , while others will be unbiased. More formally, let $\epsilon, \gamma \in [0, 1]$. We define the $\epsilon\gamma$ -heat resistance of variable v as $\max t\gamma$ s.t. $|{}^y\beta_{\text{H}}(v, \rho, \sigma, 1, 0, t\gamma)| > \epsilon$ when running MP. Put differently, the heat resistance is the largest temperature for v , s.t. its bias has an absolute value larger ϵ (i.e.

it is still “strongly” biased w.r.t. ϵ). Arguably, the larger the heat resistance of a variable, the more stability is implied for its bias. This can be used as follows.

The influence of $\mathcal{V}$ in CDCL-based MID from a practical perspective.

Empirical observations (e.g. in [8]) suggest, that the CDCL search paradigm cannot be used to solve large uniform random 3-SAT formulas with clauses-to-variable ratios above 4.2. However, [8] showed that it is possible to use a CDCL-based MID solver to do this. It was observed that, on the one hand, the solver can find a satisfying assignment without running into a single conflict. On the other hand, the solver will time-out in case it runs into a conflict. The case in which the CDCL-based MID solver resolves a conflict and finds a solution afterwards never occurred in practice. In summary we can conclude, that, when solving uniform random 3-SAT formulas with a CDCL-based MID solver, it is beneficial to use more computational time in order to make *correct decisions*, instead of using more computational time in order to *fix mistakes*.

Using $\mathcal{V}$ to compute the $\epsilon\gamma$ -heat resistance for variables (which can be costly for small γ and ϵ) gives an indication which variables should be assigned first (those with largest heat resistance) and into what direction we should assign them (implied by the bias at the given temperature). Arguably, performing MID by following the most stable biases should enable the CDCL-based MID solver to avoid conflicts. In the above scenario this should help the solver to avoid the case in which it times out – even though advancing the search will be slower.

7 Conclusions and Future Work

The first contribution of this paper was to extend the product-based MP framework (PMPF) for SAT. The extensions, resulting in the ePMPF, are dictated by the Ising Model and facilitate forms of agitation motivated by condensed-matter physics. The ePMPF allows us to draw analytical conclusions and provide suggestions for the integration of MP algorithms with the CDCL search paradigm.

More precisely, we showed that several well-known heuristics used in SAT solving (e.g. Zero-First, One-first, conflict avoiding/seeking occurrence heuristics) can be understood as special cases of an MP algorithm using the ePMPF.

Additionally, the ePMPF also allows us to provide (theoretically motivated) possibilities to integrate an MP algorithm with CDCL search. More precisely, we argued that the search of a CDCL-based MID solver (using the ePMPF) can provide feedback to MP with the information gathered during search (by using literal-weights in Ψ). Additionally, it can harness a-priori knowledge about 0-1-ratio of solutions (by preferring bias directions with Φ). Even further, it can determine the most stable variable biases (by computing the $\epsilon\gamma$ -heat resistance using $\mathcal{V}$) in order to focus on making *correct decisions* instead of *fixing mistakes*.

Future work must now investigate the performance of such a CDCL-based MID solver empirically. We need to assess the feasibility of (the theoretically motivated) possibilities to integrate MP with CDCL (using Ψ), the possibilities to harness a-priori knowledge about 0-1-ratios of solutions (using Φ), as well as the option to compute the most heat-resistant variables during MP (using $\mathcal{V}$).

References

1. D. Battaglia, M. Kolář, R. Zecchina, *Minimizing energy below the glass thresholds*, Physical Review E 70, 036107, American Physical Society, 2004.
2. A. Braunstein, M. Mézard, R. Zecchina, *Survey Propagation: An Algorithm for Satisfiability*, Journal of Rand. Struct. and Algo., pp. 201, Wiley and Sons, 2005.
3. B. A. Cipra: *An Introduction to the Ising Model*, The American Mathematical Monthly, Vol. 94, No. 10. pp. 937, 1987.
4. S. Cook: *The complexity of theorem-proving procedures*, Symposium on the Theory of Computing (STOC'71), New York, pp. 151, ACM, New York, 1971.
5. A. Dempster, N. Laird, D. Rubin, *Maximum Likelihood from Incomplete Data via the EM Algorithm*, Jrn. o. t. Royal Statistical Society, Series B 39-1, pp. 1, 1977.
6. N. Eén, N. Sörensson, *An extensible SAT solver*, Theory and Application of Satisfiability Testing (SAT'04), LNCS 2919, pp. 502, 2004.
7. O. Gableske, *On the Interpolation between Product-Based Message Passing Heuristics for SAT*, Theory and Application of Satisfiability Testing (SAT'13), LNCS 7962, pp. 293, Springer Berlin, 2013.
8. O. Gableske, S. Müelich, D. Diepold, *On the Performance of CDCL based Message Passing Inspired Decimation using $\rho\sigma\text{PMP}^i$* , Pragmatics of SAT (POS'13), 2013.
9. E. Hsu, S. A. McIlraith, *VARSAT: Integrating Novel Probabilistic Inference Techniques with DPLL Search*, Theory and Application of Satisfiability Testing (SAT'09), LNCS 5584 pp. 377, Springer Berlin, 2009.
10. E. Hsu, S. McIlraith, *Characterizing Propagation Methods for Boolean Satisfiability*, Theory and Application of Satisfiability Testing (SAT'06), LNCS 4121, pp. 325, Springer Berlin, 2006.
11. E. Hsu, C. J. Muise, C. Beck, S. A. McIlraith, *Probabilistically Estimating Backbones and Variable Bias*, Principles and Practice of Constraint Programming (CP'08), LNCS 5202, pp. 613, Springer Berlin, 2008.
12. E. Hsu, M. Kitching, F. Bacchus, S. McIlraith, *Using Expectation Maximization to Find Likely Assignments for Solving CSPs*, Conference on Artificial Intelligence (AAAI'07), pp. 224, AAAI Press, 2007.
13. E. Ising, *Beitrag zur Theorie des Ferromagnetismus*, Zeitschrift für Physik, Nummer 31, pp. 253, 1925.
14. K. Schischang, B. Frey, H. Loeliger, *Factor Graphs and the sum-product algorithm*, IEEE Transactions On Information Theory (IT'01), Vol. 47, pp. 498, IEEE, 2002.
15. J. P. Marques-Silva, K. A. Sakallah, *GRASP: A Search Algorithm for Propositional Satisfiability*, IEEE Transactions On Computers (C'99), Vol. 48, pp. 506, 1999.
16. M. Mézard, G. Parisi, *The cavity method at zero temperature*, Journal of Statistical Physics, Vol. 111, Nos. 1/2, April 2003.
17. M. Moskewicz, C. Madigan, Y. Zhao, L. Zhang, S. Malik, *Chaff: Engineering an Efficient SAT Solver*, Design Automation Conf. (DAC'01), pp. 530, ACM, 2001.
18. D. Stein, C. Newman, *Spin Glasses and Complexity*, ISBN 978-0-691-14733-8, Princeton Press, 2013.
19. L. Zhang, C. Madigan, M. Moskewicz, S. Malik, *Efficient Conflict Driven Learning in a Boolean Satisfiability Solver*, International Conference on Computer Aided Design (ICCAD'01), pp. 279, 2001.