Approximating the Distribution of Fitness over Hamming Regions

Andrew M. Sutton L. Darrell Whitley Adele E. Howe

Department of Computer Science
Colorado State University
Fort Collins, CO 80523
{sutton,whitley,howe}@cs.colostate.edu

ABSTRACT

The distribution of fitness values across a set of states sharply influences the dynamics of evolutionary processes and heuristic search in combinatorial optimization. In this paper we present a method for approximating the distribution of fitness values over Hamming regions by solving a linear programming problem that incorporates low order moments of the target function. These moments can be retrieved in polynomial time for select problems such as MAX-k-SAT using Walsh analysis. The method is applicable to any real function on binary strings that is epistatically bounded and discrete with asymptotic bounds on the cardinality of its codomain.

We perform several studies on the ONE-MAX and MAX-k-SAT domains to assess the accuracy of the approximation and its dependence on various factors. We show that the approximation can accurately predict the number of states within a Hamming region that have an improving fitness value.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search

General Terms

Theory, Algorithms

Keywords

Pseudo-Boolean functions, search space analysis

1. INTRODUCTION

Evolutionary processes and heuristic search algorithms operate by dynamically moving through a large set of states. The behavior of such algorithms strongly depends on the distribution of fitness values over this configuration space. In general, it is intractable to characterize this distribution. It can be approximated for general fitness functions by direct sampling or by biased (e.g., Metropolis) sampling [8, 1, 2]. However, when the fitness function is *epistatically bounded*, i.e., has bounded nonlinearity, it has a sparse representation in an alternate (Walsh polynomial) basis which can be invoked for the efficient calculation of low order *moments* of the distribution without resorting to sampling. In this paper we will derive an analytical approximation of the distribution of certain epistatically bounded fitness functions over volumes of Hamming space by exploiting this fact.

Heckendorn et al. [4] first showed how low order moments of the distribution of fitness values over the entire configuration space can be computed in polynomial time for epistatically bounded functions (which they called *embedded landscapes*). Later, Heckendorn [3] extended this work to show that moments of the distribution of fitness values over hyperplanes can be computed in polynomial time for epistatically bounded fitness functions.

While hyperplane statistics are important from the perspective of hyperplane sampling for selection-based genetic algorithms, processes that explore the configuration space in a *localized* manner are also strongly influenced by the distribution of fitness values in volumes of the configuration space that are in some sense "nearby". Toward that end, Sutton et al. [9] further generalized the work of Heckendorn et al. to show how low order moments of the fitness distribution over local regions (Hamming balls and spheres) can be computed in polynomial time for epistatically bounded functions.

Though low *moments* of the fitness distribution over such functions can be efficiently computed, it is still an open question exactly how these moments relate to the fitness distribution function itself. We show how the moments over Hamming regions can be used to derive an approximation of the fitness distribution over these regions when the fitness function is epistatically bounded and discrete with asymptotic bounds on the cardinality of the codomain.

Rather than resorting to sampling, we take advantage of the fact that the moments over Hamming regions are related to the distribution over that region by a system of linear equations. Though it is intractable to compute arbitrarily high moments of this distribution, we use a linear programming approach to solve an underdetermined linear system. The solution of this system approximates the distribution of fitness over any Hamming region.

Discrete, epistatically bounded fitness functions are pervasive in many combinatorial optimization domains such as

FOGA'11, January 5–9, 2011, A-6867 Schwarzenberg, Austria.

MAX-k-SAT and MAX-CUT and appear in certain models of evolutionary systems such as the quantized model of NK-landscapes [7]. The distribution of fitness over a Hamming region is an important analytical characteristic of the fitness landscape. For instance, consider a Hamming region around a particular state x. The fitness distribution over that region determines quantities such as the count of elements in the region with higher fitness than x, or even the count of best values in the region. Approximations of this distribution are directly applicable to estimating such quantities.

Given all the fitness function moments over a region, our approach gives the exact distribution of values over that region. However, we show that if $\mathsf{P} \neq \mathsf{NP}$, it is intractable to compute all the moments over general Hamming regions. The linear programming formulation allows for the incorporation of bounds on higher moments into the approximation. We also show the approach allows us to compute an exact upper and lower bound on the fitness distribution function over any Hamming region. This bound provides an exact limit on the count of states at a particular fitness value lying in the region.

To demonstrate the accuracy of the approximation we first compare the approximated fitness distribution function in the ONE-MAX domain in which we can exactly calculate true fitness distribution functions analytically. We find the accuracy improves with region size and is better toward the extrema of the fitness function. We then apply the approach to an actual combinatorial optimization domain, the MAX-k-SAT problem, and find the approximation accurately predicts the number of states in a region with higher fitness than the point at the center of the region.

This approximation is useful to practitioners in several ways. In general, the approximated distribution estimates the number of states in a region of the configuration space that lie within any range of the fitness function. Thus it may be possible to estimate the probability of a global optimum lying within a certain Hamming distance of a given state. The approximated distribution can be also used to estimate the probability of a hill-climbing search or a mutation-only evolutionary algorithm reaching a state with a given fitness value. Furthermore, the approximation makes it possible to compare two arbitrary points in the search space by estimating the relative merits of exploration near each state.

2. HAMMING REGIONS

We study fitness landscapes over length-n binary sequences $\{0,1\}^n$ where the fitness function $f : \{0,1\}^n \to A$ has the following two properties.

1. f is epistatically bounded by a constant k.
2. The codomain $A \subset \mathbb{R}$ of f is discrete with cardinality $O(n)$.

Local search algorithms and some evolutionary processes make small changes to a given state (e.g., by inverting a small number of bits). The short-term dynamics of such algorithms are influenced by the statistical properties of the states that are mutually reachable by a small number of changes. Accordingly, we are interested in the distribution of fitness values over regions that are somehow "local" to a given state. We formalize this as follows. Consider two points $x, y \in \{0,1\}^n$. The Hamming distance $\mathcal{D}(x,y)$ is the number of positions in which the strings x and y differ. The set $\{0,1\}^n$ together with the Hamming distance function form a metric space and, given a "centroid" x, we can partition $\{0,1\}^n$ into regions about x.

DEFINITION 1. *A sphere of radius r around a point $x \in \{0,1\}^n$ is defined as the set*

$$S^{(r)}(x) = \{y \in \{0,1\}^n : \mathcal{D}(x,y) = r\}.$$

Similarly, we define a union of concentric spheres as follows.

DEFINITION 2. *A ball of radius r around a point $x \in \{0,1\}^n$ is defined as the set*

$$B^{(r)}(x) = \{y \in \{0,1\}^n : \mathcal{D}(x,y) \leq r\}.$$

We will use the term *Hamming region* to denote either a sphere or ball of arbitrary radius around any arbitrary point.

2.1 The fitness statistics of Hamming regions

If $X \subseteq \{0,1\}^n$ is a Hamming region as defined above, then the c^{th} moment of a function f over X is defined as

$$\mu_c(X) = \frac{1}{|X|} \sum_{y \in X} f(y)^c \tag{1}$$

For any k-bounded pseudo-Boolean function, as long as k and c are taken to be constants, Heckendorn et al. [4] showed that the c^{th} moment of the fitness function over the entire landscape can be computed in polynomial time using Walsh analysis.

Sutton et al. [9] extended this result to Hamming regions and used a Walsh transform to construct an algorithm that computes $\mu_c(X)$ for any Hamming region X in time polynomial in n. This moment computing algorithm relies on the fact that k-bounded functions and their powers have a sparse representation in the Walsh basis and that moments of f over Hamming regions can be written as a bounded sum of weighted Walsh functions. When the epistasis of f is bounded by k, the moment algorithm of Sutton et al. has a time complexity of $O(rn^{ck})$ where r is the Hamming radius of the region. Moreover, if f can be expressed as a sum over $m \ll n^k$ subfunctions, the complexity results can be improved to $O(rm^c)$. Note that this bound holds even when $|X|$ is exponential in n (this includes the result of Heckendorn et al. [4] as a special case: when X is a Hamming ball of radius n).

Since we have constrained the codomain of the fitness function to a finite set with linearly bounded cardinality it must take the form

$$f : \{0,1\}^n \to A$$

where

$$A = \{a_0, a_1, \ldots, a_{q-1}\}$$

is a finite set of q elements where q is $O(n)$. This asymptotic bound on the codomain size is important to subsequent analysis since we later characterize the distribution of fitness values as the solution to q linear equations in q unknowns.

Without loss of generality, let us impose a total order on A so that $i < j \implies a_i < a_j$. Assuming maximization, let

$$a^* = \max_{x \in \{0,1\}^n} f(x)$$

Given a Hamming region X, we define the measure

$$p_X : A \to [0, 1]$$

where $p_X(a_i)$ is the probability that the element chosen *uniformly at random* from X has fitness a_i. In this case, p_X is a *probability mass function* with support A. For a given distribution, the set of q discrete values $\{p_X(a_i)\}$ are called the *impulses* of the probability mass function.

We can thus define the distribution of fitness values, or the *fitness distribution function*, over the region X as a function

$$N_X : A \to \mathbb{N}$$

where $N_X(a_i) = |X|p_X(a_i)$ is the number of states $y \in X$ such that $f(y) = a_i$.

The fitness distribution function thus exactly characterizes the allocation of fitness values to states in a region X. Under the assumption of maximization, $N_X(a^*)$ is the number of optimal solutions in X, and

$$\sum_{a_i > f(x)} N_X(a_i)$$

is the number of states in the Hamming region with improving fitness function value with respect to a point x. Thus, an approximation of this quantity can be used to estimate the number of optimal solutions in X and the number of states in in X with improving fitness.

When the fitness function has the constraints we have imposed, the moments of a region X appear in a system of equations that determine the fitness distribution N_X. We now show this.

2.2 Computing the exact fitness distribution

Consider a state drawn uniformly at random from the Hamming region X. Its fitness can be modeled as a random variable Z. Since each state $y \in X$ has a probability $\frac{1}{|X|}$ of being selected, we can write the expectation of Z raised to the c^{th} power as

$$\begin{aligned}\mathbb{E}[Z^c] &= \frac{1}{|X|} \sum_{y \in X} f(y)^c \\ &= \mu_c(X) \qquad \text{by (1).} \end{aligned} \tag{2}$$

But $\mathbb{E}[Z^c]$ is, by definition, the c^{th} moment of the distribution of the random variable Z. Note that the distribution of Z is the above defined probability mass function p_X. Hence we can write

$$\mathbb{E}[Z^c] = \sum_{i=0}^{q-1} a_i^c p_X(a_i). \tag{3}$$

Putting together Equations (2) and (3) we have the following identity:

$$\mu_c(X) = \sum_{i=0}^{q-1} a_i^c p_X(a_i). \tag{4}$$

In other words, the c^{th} moment of f is equal to the c^{th} moment of the probability mass function p_X.

Consider the lowest $|A| = q$ moments of X:

$$\{\mu_0(X), \mu_1(X), \ldots, \mu_{q-1}(X)\}.$$

Using the identity in (4) we have the following system of q equations in q unknowns.

$$\sum_{i=0}^{q-1} a_i^j p_X(a_i) = \mu_j(X) \tag{5}$$

for $j = \{0, 1, \ldots, q-1\}$. Letting

$$\boldsymbol{p} = (p_X(a_0), p_X(a_1), \ldots, p_X(a_{q-1}))^\top$$

and

$$\boldsymbol{\mu} = (\mu_0(X), \mu_1(X), \ldots, \mu_{q-1}(X))^\top,$$

if the $q \times q$ matrix

$$\boldsymbol{M}_{i,j} = \left((a_i)^j\right)^\top$$

is nonsingular, there is a unique solution $\boldsymbol{p}$ to

$$\boldsymbol{M}\boldsymbol{p} = \boldsymbol{\mu}$$

which defines the probability mass function for the Hamming region since $p_X(a_i) = \boldsymbol{p}_i$. The fitness distribution is then given by $N_X(a_i) = |X|p_X(a_i)$.

$\boldsymbol{M}$ belongs to a well-known class of matrices known as Vandermonde matrices. The determinant is

$$\det(\boldsymbol{M}) = \prod_{i<j} (a_j - a_i).$$

The matrix is nonsingular if and only if all the values of a_i are distinct. In our case, since A is a set, all elements $a_i \in A$, by definition are distinct so $\boldsymbol{M}$ always has an inverse and the above system of equations always has a unique solution. Hence, if we have q moments of the Hamming region, we can obtain exactly the probability mass function over X by solving the system. Since q is $O(n)$, the size of the linear system is polynomial in n, even if $|X|$ is exponential in n.

3. DISTRIBUTION APPROXIMATION

In the foregoing, q moments of f over X are needed to characterize N_X. Thus we must be able to retrieve moments of *arbitrary order*. If $\mathsf{P} \neq \mathsf{NP}$, this is computationally difficult, as is captured by the following theorem.

THEOREM 1. *In general, the calculation of N_X is $\#\mathsf{P}$-hard.*

PROOF. Let $\mathcal{F}$ be a propositional 3-SAT formula with n variables and m clauses. Let $f : \{0, 1\}^n \to \{0, 1, \ldots, m\}$ give the number of clauses satisfied under an assignment. Note that f satisfies the conditions we have imposed.

Let X be a ball of radius n around an arbitrary assignment x. In other words, $X = \{0, 1\}^n$ and $N_X(m)$ gives the number of satisfying assignments to $\mathcal{F}$, solving #3-SAT which is $\#\mathsf{P}$-complete [10]. □

Given all q moments, it is theoretically possible to solve the linear system in polynomial time. Thus the computational intractability must arise in the calculation of the moments themselves. Indeed, the calculation of $\mu_c(X)$ is exponential in c [4, 9]. Hence, we are interested in finding a way to approximate the distribution using only low moments of f over the region.

Given only $0 < c \ll q$ moments, we have the partial Vandermonde system where j in Equation (5) runs from 0 to $c-1$. Algebraically, let $\boldsymbol{M}'$ be the $c \times q$ *partial* Vandermonde matrix that consists of the first c rows of $\boldsymbol{M}$ and a *truncated*

moment vector $\boldsymbol{\mu}' = (\mu_0(X), \mu_1(X), \ldots, \mu_{c-1}(X))^\top$ consisting of the lowest c moments. We seek a solution $\hat{\boldsymbol{p}}$ to the linear system[1]

$$\boldsymbol{M}'\hat{\boldsymbol{p}} = \boldsymbol{\mu}'. \tag{6}$$

This system is underdetermined, so there are potentially infinite solutions. Furthermore, there is no guarantee that $\hat{\boldsymbol{p}}$ gives a valid probability mass function. A solution to (6) may contain elements that are meaningless as probabilities, i.e., lying outside of the unit interval.

However, we can encode this requirement as a set of fixed variables and bounding constraints by posing the formulation of a solution to the partial Vandermonde system in Equation (6) in terms of a linear programming problem.

$$\begin{array}{ll} \max & \boldsymbol{b}^\top \hat{\boldsymbol{p}} \\ \text{s.t.} & \boldsymbol{M}'\hat{\boldsymbol{p}} = \boldsymbol{\mu}'. \\ & 0 \leq \hat{\boldsymbol{p}} \leq 1. \end{array} \tag{7}$$

where $\boldsymbol{b}$ is a length q vector of coefficients.

A solution to this system has a number of desirable properties. First, it is a probability mass function in the sense that its elements lie between 0 and 1 because of the constraints imposed by the linear program. Moreover, the zeroth moment $\mu_0(X) = 1$ corresponding to the first row of $\boldsymbol{M}'$ ensures the elements sum to unity. Thus we can define the *approximated* probability mass over X in terms of $\hat{\boldsymbol{p}}$: $\hat{p}_X(a_i) = \hat{\boldsymbol{p}}_i$. Finally, the approximated probability mass function shares low moments with the exact solution to (5). This is captured by the following.

THEOREM 2. *Let $\hat{\boldsymbol{p}}$ be a solution to the above linear program. Taken as probability mass functions p_X has the same j^{th} moment as $\hat{p}_X$ for $0 \leq j < c$.*

PROOF. Let $0 \leq j < c$. The j^{th} moment of $\hat{p}_X$ is

$$\begin{aligned} \sum_{i=0}^{q-1} a_i^j \hat{p}_X(a_i) &= \sum_{i=0}^{q-1} \boldsymbol{M}'_{j,i}\hat{\boldsymbol{p}}_i \\ &= \boldsymbol{\mu}'_j \qquad \text{by (7)} \\ &= \mu_j(X). \end{aligned}$$

By (4), $\mu_j(X)$ is equivalent to the j^{th} moment of p_X: the true probability mass function over X. □

In other words, since mean and variance depend only on the first and second moments, for $c > 2$, the approximated probability mass function given by solving the above linear program has the same mean and variance as the true probability mass function of the region. The fitness distribution over X is approximately

$$\hat{N}_X(a) = |X|\hat{p}_X(a). \tag{8}$$

3.1 Choosing the coefficient vector

It is not immediately clear what an appropriate choice for the coefficient vector $\boldsymbol{b}$ in the objective function for the linear program in (7) might be. One particular approach will be used in Section 3.3 to obtain an exact bound on the fitness distribution function.

We would expect impulse values occurring near the mean (that is, the values of a_i closest to $\mu_1(X)$) to be highest in the probability mass function. Hence a heuristic might be to maximize impulses near $\mu_1(X)$. Let ω be a "window size" parameter. Define also the index of the element nearest to the mean as $\zeta = \arg\min_i |a_i - \mu_1(X)|$ (recall we have imposed a total order on A). We can then define the coefficient vector as

$$\boldsymbol{b}_i = \begin{cases} 1 & \text{if } |\zeta - i| \leq \omega \\ 0 & \text{otherwise} \end{cases}$$

Maximizing $\boldsymbol{b}^\top\hat{\boldsymbol{p}}$ is akin to finding the approximated probability mass function in which impulses lying near the mean value are maximal. Determining more principled values for $\boldsymbol{b}$ remains a direction for future research.

3.2 Limiting impulse values

Since the linear program is very underconstrained, the above approach tends to result in sparse probability mass functions in which a large amount of mass is allocated to few impulses. Empirical data suggests that the nonzero impulse values tend to be "clustered" around the mean, each with a limited mass. To further refine the accuracy of the approximation we introduce an upper limit to the mass contribution of each impulse.

If $A \subset \mathbb{N}$ and $\hat{p}_X$ is reasonably well-behaved, then a suitable continuity correction would allow us to model $\hat{p}_X$ with a continuous distribution. Neglecting higher moments, we note that a normal probability distribution with variance σ^2 has a maximum of $\frac{1}{\sqrt{2\pi\sigma^2}}$. Hence we might limit the maximum value of the impulses in $\hat{p}_X$ by

$$\left(2\pi(\mu_2(X) - \mu_1(X)^2)\right)^{-1/2}$$

to mitigate the sparse distribution of mass in the above approach.

Imposing this heuristic limit does not violate the constraints of the program and hence the resulting solution is still a probability mass function with the same c moments as p_X. We find in many cases that the limit improves the accuracy of the approximated fitness distribution function.

3.3 An exact bound

One consequence of the linear programming approach is that we can use it to provide an exact bound on the impulses of the distribution function. By choosing the appropriate coefficient vector, we can ensure the resulting solution bounds a particular impulse of N_X.

THEOREM 3. *Let X be a Hamming region. Let $\hat{p}^\star_X$ be the probability mass function obtained by solution to the linear program in (7) with $\boldsymbol{b}$ being the standard j^{th} basis vector:*

$$\boldsymbol{b}_i = \delta_{ij}, \quad i = 0, 1, \ldots, q-1$$

where δ is the Kronecker delta function. Then

$$N_X(a_j) \leq |X|\hat{p}^\star_X(a_j).$$

PROOF. By definition, $N_X(a_j) = |X|p_X(a_j)$ so it is enough to prove that $p_X(a_j) \leq \hat{p}^\star_X(a_j)$.

Suppose for contradiction that $p_X(a_j) > \hat{p}^\star_X(a_j)$. In other words, we have $\boldsymbol{p}_j > \hat{\boldsymbol{p}}^\star_j$. By definition, $\hat{\boldsymbol{p}}^\star$ is the unique solution that maximizes

$$\boldsymbol{b}^\top\hat{\boldsymbol{p}}^\star = \hat{\boldsymbol{p}}^\star_j = \hat{p}^\star_X(a_j)$$

and satisfies the partial Vandermonde system $\boldsymbol{M}'\hat{\boldsymbol{p}}^\star = \boldsymbol{\mu}'$. Now, consider the full Vandermonde system $\boldsymbol{M}\boldsymbol{p} = \boldsymbol{\mu}$. Since

[1] As a notational convention we use the prime symbol ($'$) to denote truncation and the hat symbol ($\hat{}$) to denote approximation.

all equations in the partial system corresponding to $\boldsymbol{M}'$ are contained in the full system, it follows that $\boldsymbol{M}'\boldsymbol{p} = \boldsymbol{\mu}'$ is also satisfied. But $\boldsymbol{p}_j > \hat{\boldsymbol{p}}_j^\star \implies \boldsymbol{b}^\top \boldsymbol{p} > \boldsymbol{b}^\top \hat{\boldsymbol{p}}^\star$, a contradiction that $\hat{\boldsymbol{p}}^\star$ maximizes the linear program corresponding to the partial Vandermonde system. □

Iteratively maximizing the linear program using the j^{th} standard basis vector for $j = 0, 1, \ldots, q-1$ thus generates an upper bound for each of the impulses of the distribution function N_X. A lower bound can be analogously found by solving the corresponding minimization problem using j^{th} standard basis vectors.[2]

3.4 Incorporating bounds on higher moments

Another advantage to the linear programming approach is that we can incorporate bounds on higher moments into the approximation. Let c be the maximum moment degree available and d be an arbitrary increment. Bounds on moments of higher degree can be added explicitly to the linear program as doubly bounded constraints:

$$LB(\mu_{c+d}(X)) \leq \sum_{i=0}^{q-1} a_i^j p_X(a_i) \leq UB(\mu_{c+d}(X)) \qquad (9)$$

where $LB(\mu_{c+d}(X))$ and $UB(\mu_{c+d}(X))$ are lower and upper bounds (respectively) on the moment of order $c+d$. Obtaining the exact moments of higher degrees becomes computationally difficult (and is generally intractable by Theorem 1). However, if bounds on higher moments can be efficiently obtained, they may be incorporated into the approximation in this way.

We now impose some mild restrictions on the codomain A of f and calculate upper and lower bounds on higher moments. First, we will assume $\forall a_i \in A,\ a_i \geq 0$. Since A is a finite set with cardinality linear in the problem size, we can impose this condition without loss of generality since the evaluation of f can always be shifted by an appropriate constant. Before deriving upper and lower moment bounds, we prove the following preparatory lemma.

LEMMA 1. *Let X be a Hamming region. As long as there exist at least two states $x_1, x_2 \in X$ with $f(x_1) \geq 1$ and $f(x_2) \geq 1$, then, for $c, d \geq 1$,*

$$\frac{1}{|X|} \sum_{y \in X} f(y)^c \left(\sum_{z \in X \setminus \{y\}} f(z)^d \right) \geq \mu_c(X).$$

PROOF. Since either $x_1 \in X \setminus \{y\}$ or $x_2 \in X \setminus \{y\}$, we have $\sum_{z \in X \setminus \{y\}} f(z)^d \geq 1$. □

The conditions for the lemma are relatively weak since, if necessary, we can shift f without altering the total order on A. We are now ready to give an upper bound on moments of degree $c + d$.

THEOREM 4. *Let X be a Hamming region with at least two states $x_1, x_2 \in X$ such that $f(x_1) \geq 1$ and $f(x_2) \geq 1$. Let $d \geq 1$. Then,*

$$\mu_{c+d}(X) \leq |X| \mu_c(X) \mu_d(X) - \mu_c(X).$$

[2]In practice we found this lower bound obtained with reasonable values of c to be degenerate, i.e., $N_X(a_j) \geq 0$.

PROOF.

$$\mu_c(X)\mu_d(X) = \left(\frac{1}{|X|} \sum_{y \in X} f(y)^c \right) \left(\frac{1}{|X|} \sum_{y \in X} f(y)^d \right)$$

$$= \frac{1}{|X|^2} \sum_{y \in X} f(y)^{c+d} + \frac{1}{|X|^2} \sum_{y \in X} f(y)^c \left(\sum_{z \in X \setminus \{y\}} f(z)^d \right)$$

$$= \frac{1}{|X|} \mu_{c+d}(X) + \frac{1}{|X|^2} \sum_{y \in X} f(y)^c \left(\sum_{z \in X \setminus \{y\}} f(z)^d \right)$$

Rearranging terms and multiplying by the cardinality of X we have

$$|X| \mu_c(X) \mu_d(X) - \frac{1}{|X|} \sum_{y \in X} f(y)^c \left(\sum_{z \in X \setminus \{y\}} f(z)^d \right) = \mu_{c+d}(X)$$

and by Lemma 1, $|X| \mu_c(X) \mu_d(X) - \mu_c(X) \geq \mu_{c+d}$. □

We can also derive the following trivial lower bound.

THEOREM 5. *Let X be a Hamming region such that for all $x \in X$, $f(x) = 0$ or $f(x) \geq 1$. Let $d \geq 1$. Then,*

$$\mu_{c+d}(X) \geq \mu_c(X).$$

PROOF. Since for all $x \in X$, $f(x)^{c+d} \geq f(x)^c$ we immediately have

$$\frac{1}{|X|} \sum_{y \in X} f(y)^{c+d} \geq \frac{1}{|X|} \sum_{y \in X} f(y)^c$$

which proves the claim. □

Again, the conditions for the theorem are relatively weak since: (1) domains where $A \subset \mathbb{N}$ already satisfy them, and (2) the elements of A can be appropriately shifted without changing the total order. These bounds can be added to the linear program in the manner mentioned at the beginning of the section.

4. ACCURACY MEASUREMENTS

In order to demonstrate the proficiency of the approximation derived in the foregoing sections, we perform a number of numerical measurements to compare the approximated fitness distribution with the true fitness distribution. Since we are working with discrete values, it is easier to visualize and compare distribution functions using their cumulative forms. We define the *cumulative probability distribution function* as

$$c_X(a) = \sum_{a_i \leq a} p_x(a_i)$$

and the *cumulative fitness distribution function* as

$$C_X(a) = \sum_{a_i \leq a} N_x(a_i)$$

We define $\hat{c}_X$ and $\hat{C}_X$ analogously as the approximated cumulative distribution functions simply by replacing the probability and fitness distribution functions in the above definitions with their approximated form. In order to compare

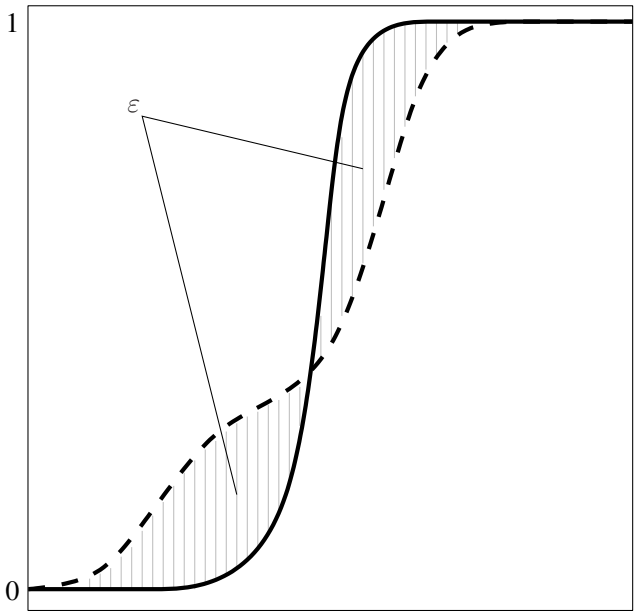

Figure 1: Illustration of the ε measure for two hypothetical cumulative distribution functions. ε measures the shaded area: the extent to which two distribution functions disagree.

how well an approximated distribution fits the true distribution, we define the (normalized) measure of absolute error as

$$\varepsilon = \frac{1}{q}\sum_{i=0}^{q-1} |c_X(a_i) - \hat{c}_X(a_i)|.$$

Note that $0 \leq \varepsilon \leq 1$ and measures the extent to which the two cumulative probability distribution functions disagree (see Figure 1). The ε metric has a loose similarity to the Kolmogorov-Smirnov statistic which measures the maximum deviation between two (continuous) cumulative distributions.

4.1 ONE-MAX

According to the result of Theorem 1, for the general class of epistatically bounded functions, it is #P-hard to compute the true fitness distribution function for all possible Hamming regions. However, if we restrict ourselves to a special class of functions, we can take advantage of its properties to compute the true distribution efficiently. It then becomes straightforward to test the accuracy of the approximation, even over intractably large regions of the space.

Given a binary sequence of length n, the ONE-MAX fitness function simply counts the number of elements in the sequence that are equal to one.

$$f : \{0,1\}^n \to \{0,1,\ldots,n\}; \quad f(x) = |\{i : x[i] = 1\}|, \quad (10)$$

where $x[i]$ denotes the i^{th} element in the sequence. This class of functions satisfies the constraints laid out in Section 2 since f is bitwise additively separable (and thus has maximum epistasis $k = 1$) and the cardinality of its codomain is clearly linearly bounded in n.

The appeal of using the ONE-MAX fitness function is that it becomes possible to derive an analytical expression for the true fitness distribution function.

4.1.1 Analytical formula

The distribution of ONE-MAX fitness values over a sphere of radius r can be written in closed form. We begin by defining the following quantity.

DEFINITION 3. *Let $x, y \in \{0,1\}^n$. We define the quantity*

$$\beta(x,y) = |\{j : x[j] = 1 \text{ and } y[j] = 0\}|$$

to be the number of 1-bits in x that are inverted to produce y (equivalently, the number of 1-bits in $x \wedge \bar{y}$ where $\wedge$ and $\bar{y}$ represent bitwise conjunction and complementation, respectively).

We claim a number of trivial constraints on $\beta(x,y)$ that will become useful in the proof of Theorem 6.

CLAIM 1. *For any $x, y \in \{0,1\}^n$, the following constraints hold for $\beta(x,y)$.*

1. *Since $\beta(x,y)$ is a count, $\beta(x,y) \in \mathbb{N}_0$.*
2. *Since there are $f(x)$ 1-bits in x, $\beta(x,y) \leq f(x)$.*
3. *Since there are $n - f(x)$ 0-bits in x, $0 \leq \mathcal{D}(x,y) - \beta(x,y) \leq n - f(x)$.*

The analytical form of the fitness distribution function for ONE-MAX is given by the following theorem.

THEOREM 6. *Let f be the ONE-MAX fitness function defined in (10). Let $x \in \{0,1\}^n$. The count of states y in a sphere of radius r around x with $f(y) = a$ can be written in the following closed form. Let $\xi = \frac{1}{2}(f(x) + r - a)$.*
Consider the following three conditions for ξ.

1. $\xi \in \mathbb{N}_0$
2. $\xi \leq f(x)$
3. $0 \leq r - \xi \leq n - f(x)$

then,

$$N_{S^{(r)}(x)}(a) = \begin{cases} \binom{f(x)}{\xi}\binom{n-f(x)}{r-\xi} & \text{if conditions on } \xi \text{ hold;} \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

PROOF. Consider any $y \in S^{(r)}(x)$. Since $\mathcal{D}(x,y) = r$, we must invert exactly r elements of x to transform it into y. According to Definition 3, there are $\beta(x,y)$ 1-bits in x that are inverted to produce y and $r - \beta(x,y)$ 0-bits in x are inverted to produce y. Since there are $f(x)$ 1-bits in x, the number of 1-bits in y is

$$f(y) = (f(x) - \beta(x,y)) + (r - \beta(x,y)). \quad (12)$$

We can write

$$N_{S^{(r)}(x)}(a) = |\{y \in S^{(r)}(x) : f(y) = a\}|$$

and by Equation (12),

$$= |\{y \in S^{(r)}(x) : a = f(x) + r - 2\beta(x,y)\}|$$

and, solving for $\beta(x, y)$ in terms of a,

$$= |\{y \in S^{(r)}(x) : \beta(x, y) = \frac{1}{2}(f(x) + r - a)\}|$$

Letting $\xi = \frac{1}{2}(f(x) + r - a)$,

$$= |\{y \in S^{(r)}(x) : \beta(x, y) = \xi)\}|$$

Thus we have equated the quantity $N_{S^{(r)}(x)}(a)$ to the number elements y in the sphere where $\beta(x, y) = \xi$. Intuitively, this is the number of ways of inverting exactly ξ 1-bits in x and exactly $r - \xi$ 0-bits in x.

If ξ satisfies the three constraints in Claim 1 above, then there are

$$\binom{f(x)}{\xi}\binom{n - f(x)}{r - \xi}$$

ways to choose ξ 1-bits in x and $r - \xi$ 0-bits in x.

On the other hand, if ξ does not satisfy the three constraints on $\beta(x, y)$ given in Claim 1, then it must be the case that for any $y \in S^{(r)}(x)$, $\xi \neq \beta(x, y)$. If so, then vacuously,

$$\{y \in S^{(r)}(x) : \beta(x, y) = \xi)\} = \emptyset$$

and $N_{S^{(r)}(x)}(a) = 0$. □

Since the spheres around a state x are mutually disjoint we always have the following identity

$$N_{B^{(r)}(x)}(a) = \sum_{u=0}^{r} N_{S^{(u)}(x)}(a).$$

Substituting the terms on the r.h.s. with (11) gives the analytical expression for the true ONE-MAX fitness distribution over the ball of radius r around a state x.

These formulas permit the direct calculation of the true fitness distribution for ONE-MAX. We use this in our first approach to assessing the accuracy of the approximation.

4.1.2 ONE-MAX approximation accuracy

Figure 2 plots the actual and estimated cumulative distributions for the $n = 50$ ONE-MAX fitness function. The solid line gives the true cumulative distribution of fitness values obtained by the explicit formula in (11) over a Hamming ball X of radius $r = 10$ in the space of length-50 bitstrings. The centroid of the ball has a fitness of 5. The broken line shows the approximation obtained by solving Equation (7) with the truncated moment vector

$$\boldsymbol{\mu}' = (\mu_0(X), \ldots, \mu_5(X))^\top$$

containing six moments obtained by the algorithm in [9]. To solve Equation (7) we used the GNU Linear Programming Kit (GLPK) using a simplex-based LP solver [5].

The window length used was $\omega = 10$. We used these settings for all ONE-MAX distribution approximations reported in this section.

The analytical formula in Equation (11) allows us to compute the true distribution function without explicitly enumerating the region. Thus we can measure the accuracy of the approximation over arbitrarily large regions of Hamming space. A close examination of the analytical formula reveals that the fitness distribution function over a Hamming ball is uniquely determined by the fitness of its centroid and the radius of the ball. Hence there are exactly $(51)^2 = 2601$ uniquely determined ONE-MAX fitness distribution functions over $\{0, 1\}^{50}$. We can calculate each of these explicitly along with the corresponding approximation and determine the dependence of the accuracy measure (ε) on radius or centroid fitness.

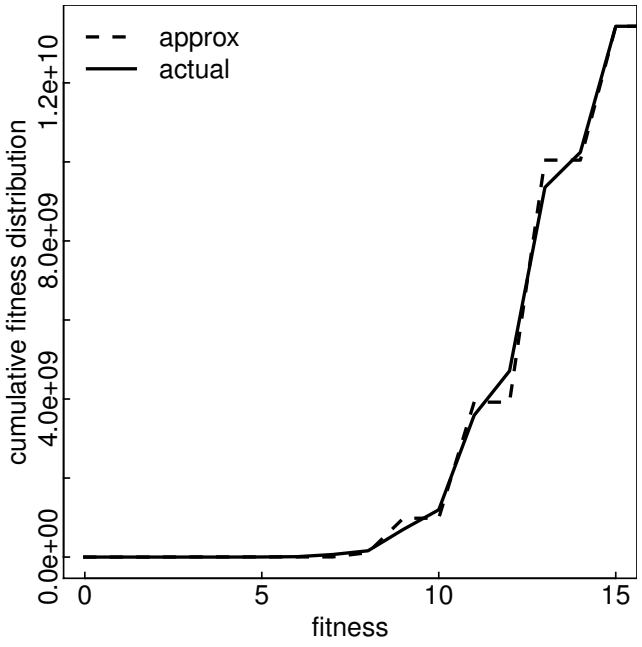

Figure 2: Cumulative fitness distribution on a single $n = 50$ ONE-MAX instance: actual vs. approximated over Hamming ball of radius 10 with centroid fitness 5.

In Figure 3, the solid line shows the mean ε over all unique ONE-MAX fitness distribution functions over $\{0, 1\}^{50}$ as a function of radius. The broken lines bracket the extremal values above and below the mean. At radius 0 (a Hamming ball containing only one point), the accuracy, of course, is always perfect since the true fitness distribution contains a single impulse whose location is entirely determined by the first two moments $\mu_0(X)$ and $\mu_1(X)$. A similar phenomenon occurs at radius 1. Subsequently, as radius increases, the accuracy diminishes until the radius reaches values in the interval $[3, 6]$. After this point, the accuracy begins to steadily improve as a function of radius.

Figure 4 plots the accuracy (ε) over all unique ONE-MAX fitness distribution functions over $\{0, 1\}^{50}$ as a function of centroid fitness. The broken line shows the maximum values (the minimum value is always $\varepsilon = 0$, occurring at low radii, c.f., Figure 3). For ONE-MAX, the accuracy is highest when the fitness of the centroid lies at extremal boundaries of the fitness function and decays as the centroid fitness approaches $n/2$: the average fitness over $\{0, 1\}^n$.

We remark that ε remains very low in all cases. This corresponds to a high approximation accuracy.

4.2 MAX-k-SAT

The results in the last section are somewhat academic since the approximation is unnecessary when an exact expression is available by Equation (11). In this section we apply the approximation to the maximum k-satisfiability

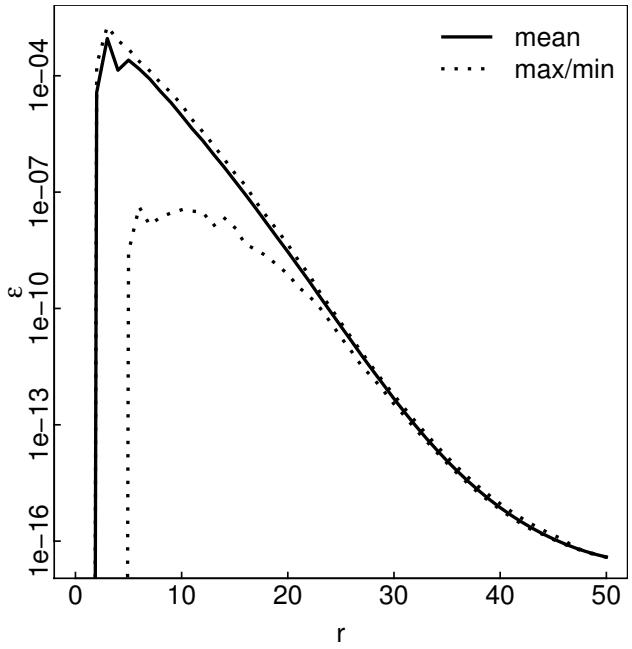

Figure 3: Dependence of approximation accuracy on ball radius for the ONE-MAX domain. The y-axis is on a logarithmic scale.

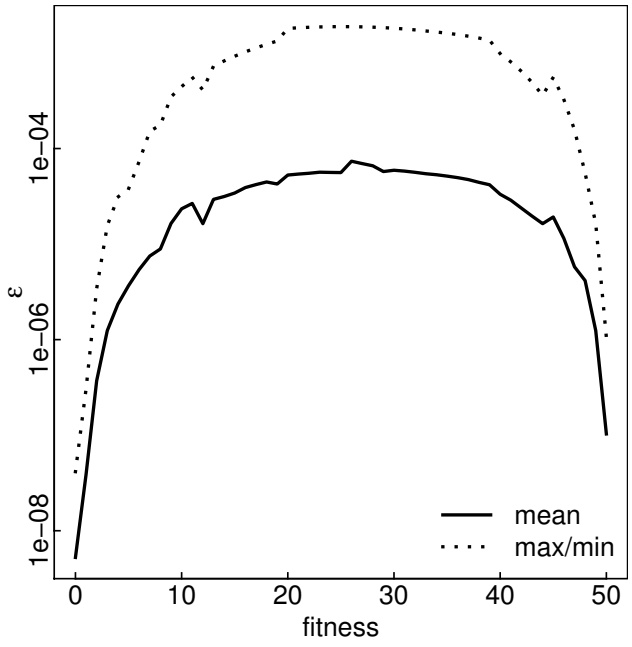

Figure 4: Dependence of approximation accuracy on centroid fitness for the ONE-MAX domain. The y-axis is on a logarithmic scale.

problem (MAX-k-SAT), an important NP-hard combinatorial problem. In this case, unless P = NP, it is intractable to generate the true fitness distribution over all Hamming regions since such a quantity yields a solution to the decision problem.

Therefore, given a Hamming region, we construct the true

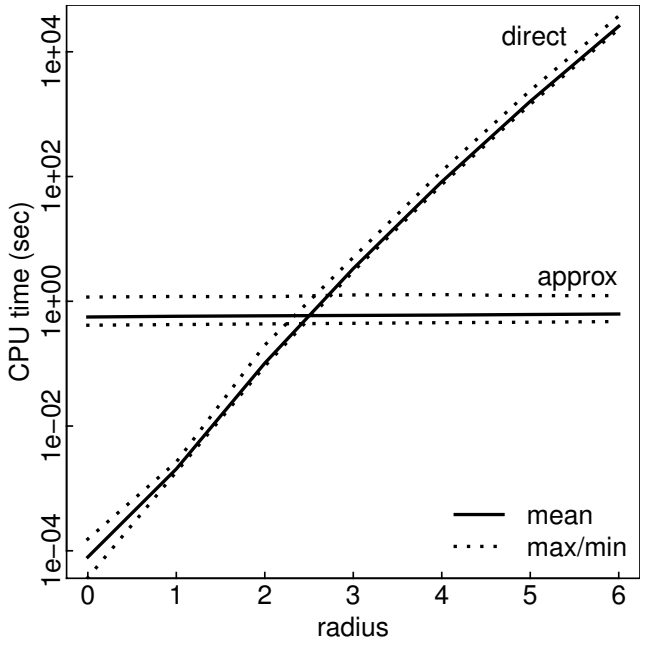

Figure 5: A comparison of time (in seconds) to exhaustively compute true distribution and time to perform LP approximation as a function of ball radius. The y-axis is on a logarithmic scale.

fitness distribution by a direct count of states at each fitness value in the region and compare it with the approximated distribution. Of course this limits the comparison to computationally manageable regions. Figure 5 illustrates this with a logarithmic plot of CPU time in seconds necessary to compute the true distribution as a function of Hamming ball radius on a 100 variable MAX-k-SAT instance. The required time is directly proportional to the cardinality of the Hamming ball which is exponential in the radius. As a comparison, we also plot in Figure 5 the time required to perform the LP approximation of the distribution. While the time to compute the true distribution increases to over 20 minutes for each Hamming region, the time to perform the LP approximation remains less than a second on average. This means it becomes intractable to compare the approximation accuracy for all radius values on nontrivial instances. However, we conjecture that the approximation accuracy remains stable with increasing radius, or possibly improves as it does in the case of ONE-MAX.

An instance of MAX-k-SAT consists of a Boolean formula with n variables and a set of m clauses. Each clause is composed of at most k literals in logical disjunction (a literal is an instance of a variable or its negation). The objective is to find a variable assignment that maximizes the number of satisfied clauses. The fitness function $f : \{0,1\}^n \to \{0,\ldots,m\}$ maps a variable assignment represented by a length-n binary string to the number of clauses satisfied under that assignment.

The function f is a sum over m subfunctions of length at most k, hence it is k-bounded. Furthermore, its codomain is the set $A = \{0,1,\ldots,m\}$, so we have exactly the type of specialized function described in Section 2. For any Hamming

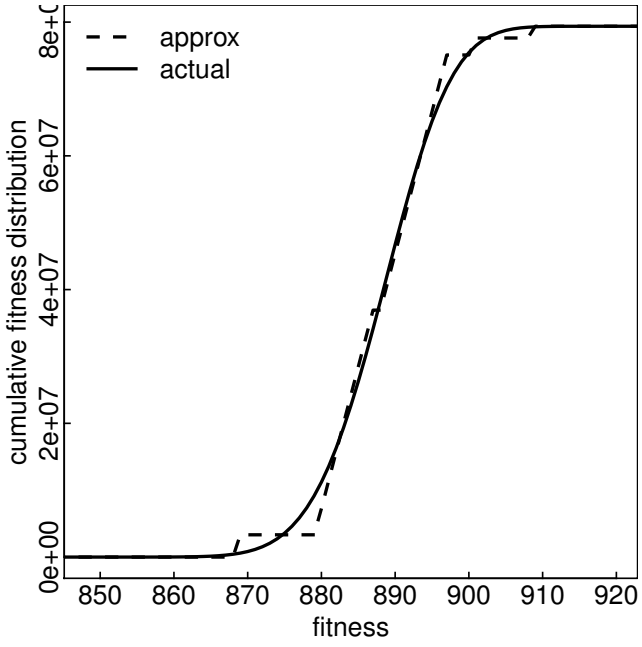

Figure 6: Cumulative fitness distribution on a single MAX-2-SAT instance: actual vs. approximated over region of radius 5.

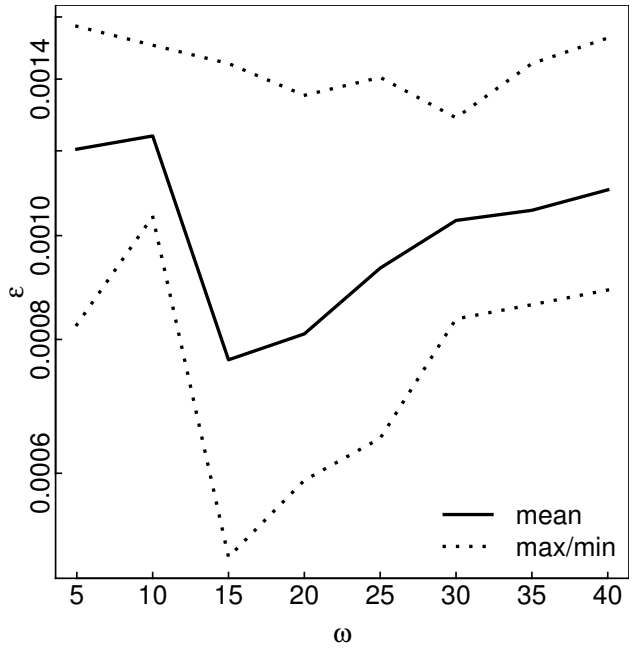

Figure 7: Dependence of approximation accuracy on window size for MAX-k-SAT benchmark set `s2v100c1200`. The y-axis is on a logarithmic scale.

region X, if we are given the $m+1$ moments

$$\{\mu_0(X), \mu_1(X), \ldots, \mu_m(X)\}$$

we could solve the $(m+1)\times(m+1)$ linear system (e.g., using a specialized algorithm for pure Vandermonde systems [6]) to obtain N_X. Due to Theorem 1, it is NP-hard to construct all these moments in general. Since f is k-bounded, lower moments can be found in polynomial time, even when $|X|$ is superpolynomial [9].

4.2.1 MAX-k-SAT approximation accuracy

In this section we report accuracy results for the approximation on the MAX-k-SAT domain. As a test set, we use the 10 instance `s2v100c1200` MAX-2-SAT benchmark set from the MAXSAT-2009 competition.[3] Each instance contains 100 variables and 1200 clauses. Each fitness distribution is evaluated over Hamming balls of fixed radius $r = 5$. Thus the calculations are over regions containing 79375496 states.

We plot the actual vs. approximated cumulative distribution function in Figure 6 for a radius 5 Hamming ball around a random point sampled from a particular instance from the benchmark set (the results are consistent across instances). The approximation is calculated using a truncated moment vector of the first four moments of the region

$$\boldsymbol{\mu}' = (\mu_0(X), \mu_1(X), \mu_2(X), \mu_3(X))^\top,$$

each generated using the algorithm in [9]. The approximation reported here also incorporates the upper and lower bounds on moments $\mu_4(X)$, $\mu_5(X)$, and $\mu_6(X)$, as in Section 3.4, and uses the heuristic impulse limit based on the second moment (Section 3.2). The window was set to $\omega = 20$. The measured ε value is approximately 7.47×10^{-5}.

[3] `http://www.maxsat.udl.cat/09/`

To determine the dependence of approximation accuracy on window size, we varied the window size from

$$\omega = 5, 10, \ldots, 40.$$

For each unique ω value, we sampled 10 states from each of the 10 instances. For each state we compute the ε for the approximation (using the current ω value) with respect to the actual fitness distribution (obtained exhaustively). Figure 7 shows that the accuracy as a function of window size appears to tend toward a minimum at $\omega = 15$.

To determine the dependence of approximation accuracy on the length of the moment vector, added bounds on higher moments (Section 3.4), and heuristic impulse limiting (Section 3.2), we repeat the experiment, holding the window size at 15 and varying the number of moments used (1 to 4), and the bounds on higher moments. We performed the experiments with and without heuristic impulse limiting. The results are given in Figure 8. As expected, the more moments, the more accurate the approximation. The higher moment bounds, however, do not appear to produce a strong effect. Clearly, the heuristic impulse limit improves the approximation accuracy in this case.

The results for ONE-MAX suggest that the approximation accuracy depends on the fitness of the centroid point (Figure 4). We also find this phenomenon occurs to some degree in the MAX-k-SAT domain. To show this, we select a representative instance (`s2v100c1200-1`) and measure the approximation accuracy for a number of different centroid states at varying fitness levels.

Since arbitrarily low fitness values are somewhat extraneous in the MAX-k-SAT domain (at least from the perspective of optimization), we limit our investigation to a range of fitness values that run from the average fitness of the instance to near-optimal fitness values. Consider a MAX-k-

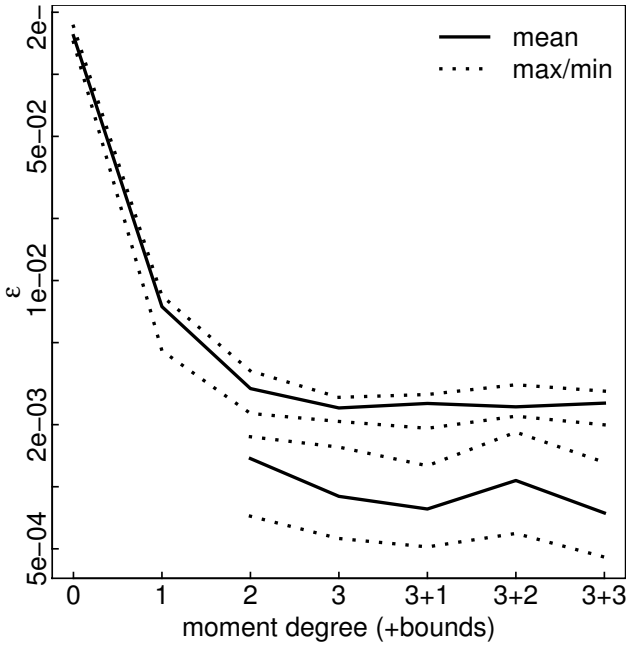

Figure 8: Dependence of approximation accuracy on moment degree (and bounds on higher moments) for MAX-k-SAT benchmark set `s2v100c1200`. Top lines are *without* heuristic impulse limit, bottom lines are *with* heuristic impulse limit (note the heuristic impulse limit requires the second moment). The y-axis is on a logarithmic scale.

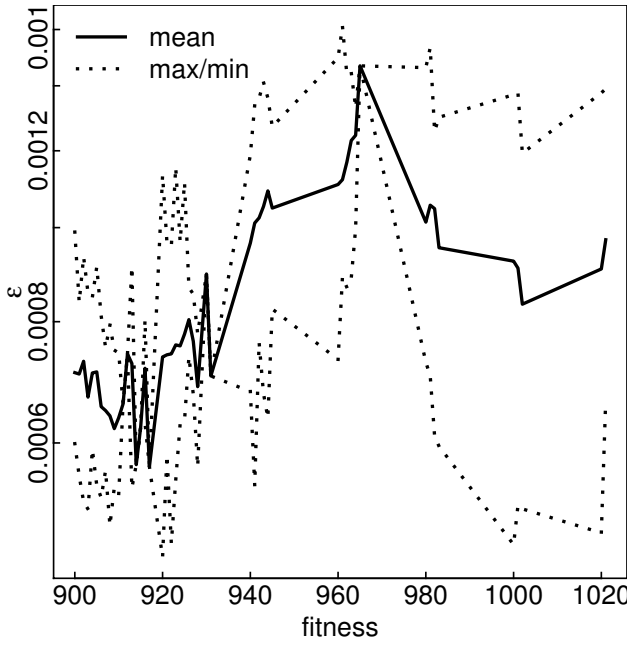

Figure 9: Dependence of approximation accuracy on centroid fitness for MAX-k-SAT instance `s2v100c1200-1`. Expected fitness of a random solution 900, best fitness 1031. The y-axis is on a logarithmic scale.

SAT formula with n variables and m clauses such that each clause contains exactly k literals. Given a particular literal, a random assignment satisfies that literal with probability $\frac{1}{2}$. Each clause is satisfied by $2^k - 1$ of the 2^k possible assignments of the literals they contain. Hence, each clause is satisfied under a random assignment with probability $\frac{2^k-1}{2^k}$. Depending on whether it is satisfied or not, a clause contributes a one or a zero to the fitness function. By linearity of expectation, the expected fitness under a random assignment is thus $\left(\frac{2^k-1}{2^k}\right) \times m$. For the 1200 clause MAX-2-SAT instance `s2v100c1200-1`, the expected fitness of a random state is $\frac{3}{4} \times 1200 = 900$. The optimal fitness level (found by a complete solver) of this particular instance is 1031.

In order to focus on pertinent levels of the fitness function for this instance, we considered a set of seven target fitness levels: 900, 920, 940, 960, 980, 1000, and 1020, which range from the random expectation value to near-optimal. For each target fitness level, we performed 100 episodes of a local hill-climbing search to generate solutions at or above the target level. Each resulting solution was then used as a centroid in a Hamming ball of radius 5, the true and approximated fitness distributions were subsequently calculated, and the resultant ε was computed (see Figure 9). Due to statistical noise, the MAX-k-SAT results are somewhat harder to interpret than the exact ONE-MAX results. However, we do note that accuracy has a stronger trend toward the boundary values of the target value range.

Again, in all cases, we note the very small ε values. We can thus conclude that the approximation is substantially accurate.

4.2.2 *Estimating the number of improving states*

To evaluate how well the model predicts the number of improving states in a region, we generated 100 random states on each of the 10 instances (1000 states total). For each generated state x, we counted the actual number of states with improving fitness that lie in the Hamming ball of radius $r = 5$ about x:

$$|\{y \in B^{(r)}(x) : f(y) > f(x)\}| = |B^{(r)}(x)| - C_{B^{(r)}(x)}(f(x)).$$

We then computed our approximation of this quantity using $\hat{N}_X$ defined in (8). We plot the actual number of improving states vs. the number predicted in Figure 10. Using the above settings, the approximation tends to slightly overpredict for lower values.

To evaluate the approximation for high-fitness states, we sampled, using hill-climbing local search, 700 states from instance `s2v100s1200-1` whose fitness values lie in the interval $[900, 1020]$ (recall the global optimum is at 1031). Using each of these states as centroids, we enumerated a radius 5 Hamming ball and counted the number of states lying in the ball with fitness at least 90% of optimal. We compare this with the corresponding count predicted by the approximation in Figure 11. In both cases we find a tight correlation between the estimate and the true count.

5. CONCLUSION

In this paper we have introduced a method for approximating the distribution of fitness values over regions of the fitness landscape. Our method is applicable to epistatically bounded fitness functions that map binary strings into a set with bounded cardinality. Such fitness functions are often found in hard combinatorial optimization problems such as

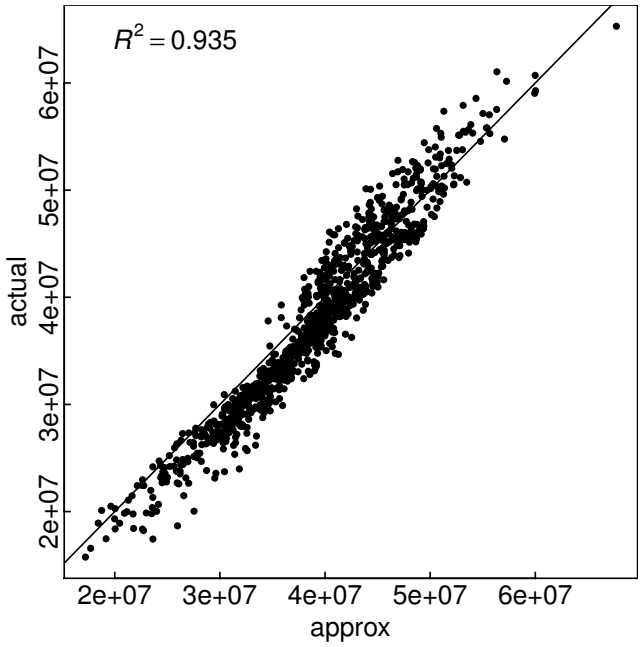

Figure 10: Number of actual improving states vs. number predicted in 1000 random regions of radius 5 over `s2v100c1200` MAX-k-SAT benchmark set.

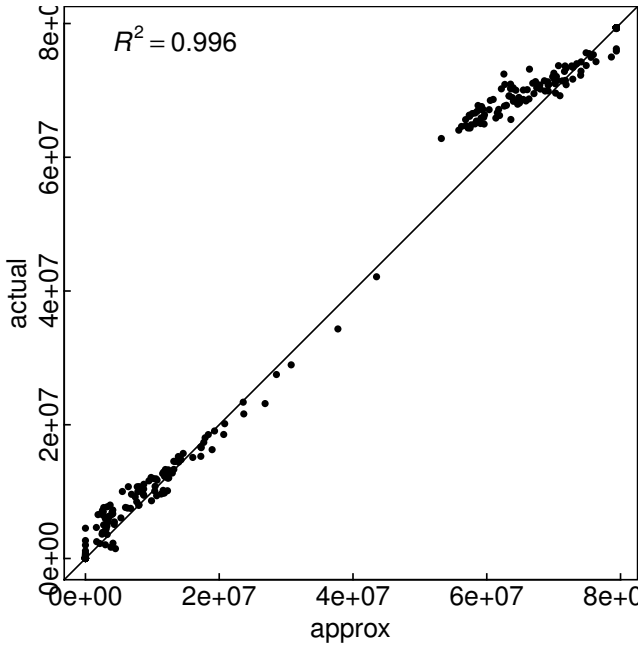

Figure 11: Number of states within 90% of optimal vs. number predicted in 700 high-fitness regions of radius 5 on the `s2v100c1200-1` instance taken from the MAX-k-SAT benchmark set.

MAX-k-SAT, spin models, and MAX-CUT as well as certain models of evolution, e.g., quantized NK-landscapes.

In tests on two domains, we found our method to be highly accurate at approximating the distributions. The accuracy of the approximation depends on the size of the moment vector used, as well as the fitness of the centroid. Moreover, the distribution in its cumulative form can be used to accurately predict the number of improving states in a large Hamming region.

Accurate predictions of the distribution of fitness values over states in local regions can impact both evolutionary and local search processes. These predictions can be used to estimate the number of states in a region that lie in a certain fitness range (such as closer to the optimal), or to compare two arbitrary states to select the one more likely to have improving states nearby (i.e., lying within a given Hamming radius). Moreover, this information can be obtained with reasonable computational effort and without resorting to sampling.

Acknowledgements

This research was sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number FA9550-08-1-0422. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon.

6. REFERENCES

[1] Torsten Asselmeyer, Werner Ebeling, and Helge Rose. Smoothing representation of fitness landscapes – the genotype-phenotype map of evolution. *BioSystems*, 39(1):63–76, 1996.

[2] Stefano Ermon, Carla Gomes, and Bart Selman. Computing the density of states of Boolean formulas. In *Proceedings of the 16th International Conference on Principles and Practice of Constraint Programming*, 2010.

[3] Robert B. Heckendorn. Embedded landscapes. *Evolutionary Computation*, 10(4):345–369, 2002.

[4] Robert B. Heckendorn, Soraya Rana, and Darrell Whitley. Polynomial time summary statistics for a generalization of MAXSAT. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 281–288, 1999.

[5] Andrew O. Makhorin. GLPK: GNU Linear Programming Kit [computer software]. Available from `http://www.gnu.org/software/glpk/`, 2000–2008.

[6] Miroslav Morhác. An iterative error-free algorithm to solve Vandermonde systems. *Applied Mathematics and Computation*, 117(1):45 – 54, 2001.

[7] Mark E. J. Newman and Robin Engelhardt. Effect of neutral selection on the evolution of molecular species. In *Proc. R. Soc. London B.*, volume 256, pages 1333–1338, 1998.

[8] Helge Rose, Werner Ebeling, and Torsten Asselmeyer. The density of states-a measure of the difficulty of optimisation problems. In *Proceedings of the 4th International Conference on Parallel Problem Solving from Nature*, pages 208–217. Springer Verlag, 1996.

[9] Andrew M. Sutton, L. Darrell Whitley, and Adele E. Howe. Computing the moments of k-bounded pseudo-Boolean functions over Hamming spheres of arbitrary radius in polynomial time. *Theoretical Computer Science*, 2011, doi:10.1016/j.tcs.2011.02.006

[10] Leslie G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189 – 201, 1979.