

Exact Computation of the Expectation Curves for Uniform Crossover

Francisco Chicano
Departamento de Lenguajes y
Ciencias de la Computación
University of Málaga, Spain
chicano@lcc.uma.es

Darrell Whitley
Dept. of Computer Science
Colorado State University
Fort Collins CO, USA
whitley@cs.colostate.edu

Enrique Alba
Departamento de Lenguajes y
Ciencias de la Computación
University of Málaga, Spain
eat@lcc.uma.es

ABSTRACT

Uniform crossover is a popular operator used in genetic algorithms to combine two tentative solutions of a problem represented as binary strings. We use the Walsh decomposition of pseudo-Boolean functions and properties of Krawtchouk matrices to exactly compute the expected value for the fitness of a child generated by uniform crossover from two parent solutions. We prove that this expectation is a polynomial in ρ , the probability of selecting the best-parent bit. We provide efficient algorithms to compute this polynomial for ONEMAX and MAX-kSAT problems, but the results also hold for domains such as NK-Landscapes.

Categories and Subject Descriptors

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

General Terms

Theory, Algorithms

Keywords

Uniform crossover, Walsh decomposition, fitness landscapes

1. INTRODUCTION

Uniform crossover is a well-known operator in the domain of Evolutionary Computation [1]. This operator builds a new solution by randomly selecting each “allele” from one of the parent solutions. The “allele” in the best parent is selected with probability ρ , which is called the *bias*. A common value for this bias is $\rho = 0.5$, where each parent has the same probability of providing its “allele” to the offspring.

In this work we use a Walsh decomposition and provide a closed-form formula for computing the expected value of the fitness of a child generated by uniform cross from two parent solutions x and y . We also study how the expected value depends on ρ . From a theoretical point of view, the

closed-form formula could be useful in runtime analysis and it can help to understand the behaviour of uniform crossover. From a practical point of view, it could be used to compute an optimal value for the bias. However, in this case, we need the expression to be easy to compute.

The remainder of the paper is organized as follows. In the next section the mathematical tools required to understand the rest of the paper are presented. In Section 3 we present our main contribution of this work: the expected fitness value of the solution generated by uniform crossover. Section 4 provides closed-form formulas for the expression of the expected fitness value in the case of the ONEMAX and MAX-kSAT problems. In Section 5 we analyze some implications of the theoretical result and, finally, Section 6 presents the conclusions and future work.

2. BACKGROUND

In this section we present the concepts required to understand the rest of the paper. In particular, we present some background on Walsh functions [7] and the Walsh decomposition of pseudo-Boolean functions.

DEFINITION 1. *We define a pseudo-Boolean function f as a map between \mathbb{B}^n , the set of binary strings of length n and \mathbb{R} , the set of real numbers.*

DEFINITION 2. *The (non-normalized) Walsh function with parameter $w \in \mathbb{B}^n$ is a pseudo-Boolean function defined over \mathbb{B}^n as:*

$$\psi_w(x) = \prod_{i=1}^n (-1)^{w_i x_i} = (-1)^{\sum_{i=1}^n w_i x_i}, \quad (1)$$

where the subindex in w_i and x_i denotes one particular component of the binary string.

We can observe that the Walsh functions map \mathbb{B}^n to the set $\{-1, 1\}$. The Walsh functions have some properties which are useful in our math development of Section 3. We present these properties in the following without a proof. The interested reader can refer to [6] to see a proof of these properties.

Let us consider the set of all the pseudo-Boolean functions defined over \mathbb{B}^n , $\mathbb{R}^{\mathbb{B}^n}$. We can think in one pseudo-Boolean function as an array of 2^n real numbers, each one being the function evaluation of a particular binary string of \mathbb{B}^n . Each pseudo-Boolean function is, thus, a particular vector in a vector space with 2^n dimensions. Let us define the dot-product between two pseudo-Boolean functions as:

$$\langle f, g \rangle = \sum_{x \in \mathbb{B}^n} f(x)g(x). \quad (2)$$

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'12, July 7–11, 2012, Philadelphia, Pennsylvania, USA.
Copyright 2012 ACM 978-1-4503-1177-9/12/07 ...\$10.00.

In \mathbb{B}^n there are 2^n Walsh functions that form an orthogonal basis in the set of pseudo-Boolean functions. Thus,

$$\langle \psi_w, \psi_t \rangle = 2^n \delta_w^t, \quad (3)$$

where δ denotes the Kronecker delta, which is 1 if $w = t$ and 0 if $w \neq t$.

Any arbitrary pseudo-Boolean function f can be expressed as a weighted sum of Walsh functions. We can represent f in the Walsh basis in the following way:

$$f(x) = \sum_{w \in \mathbb{B}^n} a_w \psi_w(x), \quad (4)$$

where the *Walsh coefficients* a_w are defined as:

$$a_w = \frac{1}{2^n} \langle \psi_w, f \rangle. \quad (5)$$

The previous expression is called *Walsh decomposition* of f . In the following we will denote with \underline{i} the binary string with position i set to 1 and the rest set to 0. We omit the length of the string n in the notation, but it will be clear from the context. For example, if we are considering binary strings in \mathbb{B}^4 we have $\underline{1} = 1000$ and $\underline{3} = 0010$. For a binary string $w \in \mathbb{B}^n$ we denote with $|w|$ the number of ones of w . We define the *order* of a Walsh function ψ_w as the value $|w|$. Some properties of the Walsh functions are given in the following proposition, which we present without proof.

PROPOSITION 1. *Let us consider the Walsh functions defined over \mathbb{B}^n . The following identities hold:*

$$\psi_0 = 1, \quad (6)$$

$$\psi_{w \oplus t} = \psi_w \psi_t, \quad (7)$$

$$\psi_w(x \oplus y) = \psi_w(x) \psi_w(y), \quad (8)$$

$$\psi_w(x) = \psi_x(w), \quad (9)$$

$$\psi_w^2 = 1, \quad (10)$$

$$\sum_{x \in \mathbb{B}^n} \psi_w(x) = \delta_0^{|w|} = \begin{cases} 1 & \text{if } w = 0, \\ 0 & \text{if } w \neq 0, \end{cases} \quad (11)$$

$$\psi_{\underline{i}}(x) = (-1)^{x_i} = 1 - 2x_i, \quad (12)$$

where \oplus denotes the *component-wise sum (XOR)* in \mathbb{Z}_2 .

Given a set of binary strings W and a binary string u we denote with $W \wedge u$ the set of binary strings that can be computed as the bitwise AND of a string in W and u , that is, $W \wedge u = \{w \wedge u \mid w \in W\}$. For example, $\mathbb{B}^4 \wedge 0101 = \{0000, 0001, 0100, 0101\}$.

When working with Walsh functions, it is normal to encounter integer values which are elements of the *Krawtchouk matrices*. Let $\mathcal{K}^{(n)}$ denote the n -th Krawtchouk matrix [2], which is an $(n+1) \times (n+1)$ integer matrix, whose elements are defined by the following formula:

$$\mathcal{K}_{r,j}^{(n)} = \sum_{l=0}^n (-1)^l \binom{n-j}{r-l} \binom{j}{l}, \quad (13)$$

where $0 \leq r, j \leq n$ and we assume in the previous expression that $\binom{a}{b} = 0$ if $b > a$ or $b < 0$.

The elements of the Krawtchouk matrices can also be defined with the help of the following generating function:

$$(1+x)^{n-j} (1-x)^j = \sum_{r=0}^n x^r \mathcal{K}_{r,j}^{(n)}. \quad (14)$$

PROPOSITION 2. *We have the following identity between the elements of the Krawtchouk matrices:*

$$\mathcal{K}_{n-r,j}^{(n)} = (-1)^j \mathcal{K}_{r,j}^{(n)} \quad (15)$$

PROOF. We use (13) to write:

$$\begin{aligned} \mathcal{K}_{n-r,j}^{(n)} &= \sum_{l=0}^n (-1)^l \binom{n-j}{n-r-l} \binom{j}{l} \\ &= \sum_{l=0}^n (-1)^l \binom{n-j}{n-j-n+r+l} \binom{j}{j-l} \\ &= \sum_{l=0}^n (-1)^l \binom{n-j}{r-(j-l)} \binom{j}{j-l}. \end{aligned} \quad (16)$$

Now we can make a variable change and introduce $h = j - l$:

$$\begin{aligned} \mathcal{K}_{n-r,j}^{(n)} &= \sum_{l=0}^n (-1)^l \binom{n-j}{r-(j-l)} \binom{j}{j-l} \\ &= \sum_{h=j-n}^j (-1)^{j-h} \binom{n-j}{r-h} \binom{j}{h} \\ &= (-1)^j \sum_{h=j-n}^j (-1)^h \binom{n-j}{r-h} \binom{j}{h}, \end{aligned} \quad (17)$$

where we restrict the lower limit of the sum from $h_{lb} = j - n$ to $h_{lb} = 0$ and the upper limit from $h_{ub} = j$ to $h_{ub} = n$, since the new terms added and removed are all 0. Finally:

$$\mathcal{K}_{n-r,j}^{(n)} = (-1)^j \sum_{h=0}^n (-1)^h \binom{n-j}{r-h} \binom{j}{h} = (-1)^j \mathcal{K}_{r,j}^{(n)}.$$

□

From (14) we deduce that $\mathcal{K}_{0,r}^{(n)} = 1$. Observe that $\mathcal{K}_{0,r}^{(n)}$ is the constant coefficient in the polynomial. Krawtchouk matrices have an important role when we sum an exponential number of Walsh functions. The following proposition provides an important result in this line.

PROPOSITION 3. *Let $t \in \mathbb{B}^n$ be a binary string and $0 \leq r \leq n$. Then the following two identities hold for the sum of Walsh functions:*

$$\sum_{\substack{w \in \mathbb{B}^n \wedge t \\ |w|=r}} \psi_w(x) = \mathcal{K}_{r,|x \wedge t|}^{(|t|)} \quad (18)$$

$$\sum_{w \in \mathbb{B}^n \wedge t} \psi_w(x) = 2^{|t|} \delta_0^{|x \wedge t|} \quad (19)$$

PROOF. Let us develop the left hand side of (18):

$$\sum_{\substack{w \in \mathbb{B}^n \wedge t \\ |w|=r}} \psi_w(x) = \sum_{\substack{w \in \mathbb{B}^n \wedge t \\ |w|=r}} \prod_{\substack{j=1 \\ w_j=1}}^n \psi_{\underline{j}}(x) \quad \text{by (1)}. \quad (20)$$

Now we can identify the second member of the previous expression with the coefficient of a polynomial. Let us con-

sider the polynomial $Q_x^{(t)}(z)$ defined as:

$$\begin{aligned} Q_x^{(t)}(z) &= \prod_{\substack{j=1 \\ t_j=1}}^n (z + \psi_{\bar{j}}(x)) = \sum_{l=0}^{|t|} z^l \left(\sum_{\substack{w \in \mathbb{B}^n \wedge t \\ |w|=|t|-l}} \prod_{\substack{j=1 \\ w_j=1}}^n \psi_{\bar{j}}(x) \right) \\ &= \sum_{l=0}^{|t|} q_l z^l. \end{aligned} \quad (21)$$

From (21) we conclude that the summation in (20) is the coefficient of $z^{|t|-r}$ in the polynomial $Q_x^{(t)}(z)$, that is, $q_{|t|-r}$. According to (12) and (21) we can write $Q_x^{(t)}(z) = (z+1)^{|\bar{x} \wedge t|} (z-1)^{|x \wedge t|}$. Obviously, $|\bar{x} \wedge t| + |x \wedge t| = |t|$. According to (14), the polynomials $Q_x^{(t)}(z)$ are related to the Krawtchouk matrices by $Q_x^{(t)}(z) = (-1)^{|x \wedge t|} \sum_{l=0}^{|t|} \mathcal{K}_{l, |x \wedge t|}^{(|t|)} z^l$ and we can write $q_l = (-1)^{|x \wedge t|} \mathcal{K}_{l, |x \wedge t|}^{(|t|)}$. Replacing l by $|t|-r$ and applying Proposition 2 we obtain (18).

The expression (19) can be obtained in the following way:

$$\begin{aligned} \sum_{w \in \mathbb{B}^n \wedge t} \psi_w(x) &= \sum_{r=0}^{|t|} \sum_{\substack{w \in \mathbb{B}^n \wedge t \\ |w|=r}} \psi_w(x) = \sum_{r=0}^{|t|} q_{|t|-r} = Q_x^{(t)}(1) \\ &= 2^{|\bar{x} \wedge t|} \delta_0^{|\bar{x} \wedge t|}, \end{aligned} \quad (22)$$

since the factor $(z-1)^{|x \wedge t|}$ in the polynomial $Q_x^{(t)}(z)$ is 1 only if $|x \wedge t| = 0$ and zero otherwise. Now we can replace $|\bar{x} \wedge t|$ by $|t|$, since if $|\bar{x} \wedge t| < |t|$, then we have $|x \wedge t| > 0$ and the previous expression is 0. \square

3. ANALYSIS OF CROSSOVER

In evolutionary computation, a crossover operator is a procedure which takes two tentative solutions to a problem x and y , called parents, and computes one or two solutions, called children, based on the features of x and y . Let us denote with $C(x, y)$ the random variable giving one child of the crossover operator. If the crossover gives two children, we can imagine that it deterministically returns one. What we want to compute is the expected fitness value of this child of the crossover. That is, $\mathbb{E}\{f(C(x, y))\}$ for a crossover operator represented by the probability distribution $C(x, y)$. We focus on combinatorial optimization problems using binary strings for the solution representation. We can write the expectation as:

$$\mathbb{E}\{f(C(x, y))\} = \sum_{z \in \mathbb{B}^n} f(z) \Pr\{C(x, y) = z\},$$

and using the Walsh decomposition of f we can rewrite the previous expression in

$$\begin{aligned} \mathbb{E}\{f(C(x, y))\} &= \sum_{z \in \mathbb{B}^n} \left(\sum_{w \in \mathbb{B}^n} a_w \psi_w(z) \right) \Pr\{C(x, y) = z\} \\ &= \sum_{w \in \mathbb{B}^n} a_w \left(\sum_{z \in \mathbb{B}^n} \psi_w(z) \Pr\{C(x, y) = z\} \right) \\ &= 2^n \sum_{w \in \mathbb{B}^n} a_w b_w(x, y), \end{aligned} \quad (23)$$

where $b_w(x, y)$ denotes the Walsh coefficient of the probability function $\Pr\{C(x, y) = z\}$ with respect to z .

To calculate the desired expectation we will assume the use of the uniform crossover operator. As we will see, the Walsh coefficients $b_w(x, y)$ for the uniform crossover can be easily computed with the help of the Walsh functions. For other crossover operators, like the one point or the two point crossover, it is not yet clear if Walsh analysis can be used to obtain an efficient formula for the expectation.

We will denote uniform crossover by UX. Let $x, y \in \mathbb{B}^n$ be the parent solutions. For each position (bit) of the child binary string z , UX selects the bit in x with probability ρ and the bit in y with probability $1 - \rho$, where $\rho \in [0, 1]$ is called the *bias*. In most cases the bias is $\rho = 0.5$. We will replace the notation $C(x, y)$ used to represent a generic random variable representing the child of a crossover by a new notation including the parameter ρ of UX: $U_\rho(x, y)$.

In UX each position of the binary string is treated independently. Thus, the probability distribution of $U_\rho(x, y)$ can be written as a product of simpler probability distributions related to each bit. Let us denote with $B_\rho(x_i, y_i)$ the random variable with range in \mathbb{B} that represents the bit selected to be at position i of the child if the parent bits at this position are x_i and y_i in UX with bias ρ . The probability distribution of $B_\rho(x_i, y_i)$ is:

$$\Pr\{B_\rho(x_i, y_i) = z_i\} = \begin{cases} 0 & \text{if } x_i = y_i \text{ and } x_i \neq z_i, \\ 1 & \text{if } x_i = y_i = z_i, \\ \rho & \text{if } x_i = z_i \text{ and } y_i \neq z_i, \\ 1 - \rho & \text{if } y_i = z_i \text{ and } x_i \neq z_i. \end{cases} \quad (24)$$

The probability distribution of UX is:

$$\Pr\{U_\rho(x, y) = z\} = \prod_{i=1}^n \Pr\{B_\rho(x_i, y_i) = z_i\}. \quad (25)$$

The following lemma provides the Walsh decomposition of $\Pr\{U_\rho(x, y) = z\}$. We decorate the Walsh coefficients with ρ to highlight the dependence of the coefficient with ρ .

LEMMA 1. *Let $x, y, w \in \mathbb{B}^n$ and $\rho \in [0, 1]$. The following identity holds for the Walsh coefficient $b_{w, \rho}(x, y)$ of the probability function $\Pr\{U_\rho(x, y) = z\}$*

$$b_{w, \rho}(x, y) = \frac{1}{2^n} \psi_w(y) (1 - 2\rho)^{|(x \oplus y) \wedge w|}. \quad (26)$$

PROOF. From (5) the Walsh coefficient $b_{w, \rho}(x, y)$ is:

$$\begin{aligned} b_{w, \rho}(x, y) &= \frac{1}{2^n} \sum_{z \in \mathbb{B}^n} \psi_w(z) \Pr\{U_\rho(x, y) = z\} \\ &= \frac{1}{2^n} \sum_{z \in \mathbb{B}^n} \psi_w(z) \prod_{i=1}^n \Pr\{B_\rho(x_i, y_i) = z_i\} \\ &= \frac{1}{2^n} \sum_{z \in \mathbb{B}^n} \left(\prod_{i=1}^n (-1)^{w_i z_i} \right) \prod_{i=1}^n \Pr\{B_\rho(x_i, y_i) = z_i\} \\ &= \frac{1}{2^n} \sum_{z \in \mathbb{B}^n} \prod_{i=1}^n (-1)^{w_i z_i} \Pr\{B_\rho(x_i, y_i) = z_i\} \\ &= \frac{1}{2^n} \prod_{i=1}^n \sum_{z_i \in \mathbb{B}} (-1)^{w_i z_i} \Pr\{B_\rho(x_i, y_i) = z_i\}. \end{aligned} \quad (27)$$

For the inner sum we can write

$$\begin{aligned} & \sum_{z_i \in \mathbb{B}} (-1)^{w_i z_i} \Pr\{B_\rho(x_i, y_i) = z_i\} \\ &= \Pr\{B_\rho(x_i, y_i) = 0\} + (-1)^{w_i} \Pr\{B_\rho(x_i, y_i) = 1\} \\ &= 1 - 2\delta_1^{w_i} \Pr\{B_\rho(x_i, y_i) = 1\}, \end{aligned} \quad (28)$$

where we exploit the fact that we must get 0 or 1 in a bit after the crossover, that is:

$$\Pr\{B_\rho(x_i, y_i) = 0\} + \Pr\{B_\rho(x_i, y_i) = 1\} = 1.$$

Including this result in (27) we have

$$\begin{aligned} b_{w,\rho}(x, y) &= \frac{1}{2^n} \prod_{i=1}^n (1 - 2\delta_1^{w_i} \Pr\{B_\rho(x_i, y_i) = 1\}) \\ &= \frac{1}{2^n} \prod_{\substack{i=1 \\ w_i=1}}^n (1 - 2\Pr\{B_\rho(x_i, y_i) = 1\}). \end{aligned} \quad (29)$$

Using the definition of $\Pr\{B_\rho(x_i, y_i) = z_i\}$ in (24):

$$\Pr\{B_\rho(x_i, y_i) = 1\} = \begin{cases} 0 & \text{if } x_i = y_i = 0, \\ 1 & \text{if } x_i = y_i = 1, \\ \rho & \text{if } x_i = 1 \text{ and } y_i = 0, \\ 1 - \rho & \text{if } x_i = 0 \text{ and } y_i = 1. \end{cases} \quad (30)$$

And the factor inside (29) is

$$(1 - 2\Pr\{B_\rho(x_i, y_i) = 1\}) = (-1)^{y_i} (1 - 2\rho + 2\rho\delta_{x_i}^{y_i}). \quad (31)$$

We can develop (29) in the following way:

$$\begin{aligned} b_{w,\rho}(x, y) &= \frac{1}{2^n} \prod_{\substack{i=1 \\ w_i=1}}^n (-1)^{y_i} (1 - 2\rho + 2\rho\delta_{x_i}^{y_i}) \\ &= \frac{1}{2^n} \left(\prod_{\substack{i=1 \\ w_i=1}}^n (-1)^{y_i} \right) \prod_{i=1}^n (1 - 2\rho + 2\rho\delta_{x_i}^{y_i}) \\ &= \frac{1}{2^n} \psi_w(y) \prod_{\substack{i=1 \\ w_i=1}}^n (1 - 2\rho + 2\rho\delta_{x_i}^{y_i}). \end{aligned} \quad (32)$$

The expression $(1 - 2\rho + 2\rho\delta_{x_i}^{y_i})$ takes only two values: 1 if $y_i = x_i$ and $1 - 2\rho$ when $x_i \neq y_i$. A factor $1 - 2\rho$ is included in the product for all the positions i in which $x_i \neq y_i$ and $w_i = 1$. Then the product in (32) becomes $(1 - 2\rho)^{|(x \oplus y) \wedge w|}$ and we obtain (26). \square

Now we are ready to present the main result of this work.

THEOREM 1. *Let f be a pseudo-Boolean function defined over \mathbb{B}^n and a_w with $w \in \mathbb{B}^n$ its Walsh coefficients. The following identity holds for $\mathbb{E}\{f(U_\rho(x, y))\}$:*

$$\mathbb{E}\{f(U_\rho(x, y))\} = \sum_{r=0}^n A_{x,y}^{(r)} (1 - 2\rho)^r, \quad (33)$$

where the coefficients $A_{x,y}^{(r)}$ are defined in the following way:

$$A_{x,y}^{(r)} = \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = r}} a_w \psi_w(y). \quad (34)$$

PROOF. According to (23) and (26) we can write

$$\begin{aligned} \mathbb{E}\{f(U_\rho(x, y))\} &= 2^n \sum_{w \in \mathbb{B}^n} a_w b_{w,\rho}(x, y) \\ &= \sum_{w \in \mathbb{B}^n} a_w \psi_w(y) (1 - 2\rho)^{|(x \oplus y) \wedge w|} \\ &= \sum_{r=0}^n \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = r}} a_w \psi_w(y) (1 - 2\rho)^{|(x \oplus y) \wedge w|} \\ &= \sum_{r=0}^n (1 - 2\rho)^r \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = r}} a_w \psi_w(y), \end{aligned} \quad (35)$$

and we get (33). \square

Note that the expression for the expected fitness after applying UX is a polynomial in $(1 - 2\rho)$. The degree of this polynomial depends on the Hamming distance between the parent solutions, $|x \oplus y|$, and the maximum order of the Walsh decomposition, p_{max} . The degree of the polynomial will be the minimum between these two numbers, since $|(x \oplus y) \wedge w| < |w|$ and $|(x \oplus y) \wedge w| < |x \oplus y|$. This means that the maximum degree of the polynomial is $r_{max} = \min(p_{max}, |x \oplus y|)$.

PROPOSITION 4. *Let $A_{x,y}^{(r)}$ be the polynomial coefficients for f and $B_{x,y}^{(r)}$ the polynomial coefficients for g . Then, the polynomial coefficients for $h = f + g$ are $C_{x,y}^{(r)} = A_{x,y}^{(r)} + B_{x,y}^{(r)}$.*

PROOF. Let a_w with $w \in \mathbb{B}^n$ be the Walsh coefficients of f and b_w the Walsh coefficients of g . Then, the Walsh coefficients of $h = f + g$ are $c_w = a_w + b_w$. Therefore:

$$\begin{aligned} C_{x,y}^{(r)} &= \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = r}} c_w \psi_w(y) = \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = r}} (a_w + b_w) \psi_w(y) \\ &= \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = r}} a_w \psi_w(y) + \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = r}} b_w \psi_w(y) \\ &= A_{x,y}^{(r)} + B_{x,y}^{(r)}. \end{aligned}$$

\square

When UX is used in the literature a common value for ρ is $1/2$. In this case, the expression for the expected fitness value is a simple coefficient, as the following corollary proves.

COROLLARY 1. *Let f be a pseudo-Boolean function defined over \mathbb{B}^n and a_w with $w \in \mathbb{B}^n$ its Walsh coefficients. The expected value of the fitness function after applying UX to solutions x and y with bias $\rho = 1/2$ is:*

$$\mathbb{E}\{f(U_{1/2}(x, y))\} = A_{x,y}^{(0)} = \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = 0}} a_w \psi_w(y). \quad (36)$$

PROOF. If we set $\rho = 1/2$ in the polynomial (33) all the terms $(1 - 2\rho)^r$ with $r > 0$ vanish and the expected fitness value is $A_{x,y}^{(0)}$. \square

4. TWO EXAMPLES

The result of Theorem 1 allows one to compute the expected fitness after UX is applied if we know the Walsh decomposition of the objective function f . One can argue that the computation of the coefficients of the polynomial

(33) can be costly. However, we can restrict the cost to be polynomial only considering k -bounded pseudo-Boolean functions. This class of problems includes MAX-kSAT and NK-Landscapes, as well as all linear pseudo-Boolean functions such as ONEMAX. In order to illustrate that this computation can be efficient, we provide expressions for the coefficients $A_{x,y}^{(r)}$ in the case of two well-known problems in combinatorial optimization: ONEMAX and MAX-kSAT.

4.1 ONEMAX

ONEMAX is a toy combinatorial optimization problem defined over binary strings which is commonly studied due to its simplicity. The objective function for ONEMAX is defined as:

$$f(x) = |x| = \sum_{i=1}^n x_i. \quad (37)$$

Using properties of Walsh function given in (12) we obtain:

$$f(x) = \sum_{i=1}^n x_i = \sum_{i=1}^n \frac{1 - \psi_i(x)}{2} = \frac{n}{2} - \frac{1}{2} \sum_{i=1}^n \psi_i(x), \quad (38)$$

and we deduce that the Walsh coefficients for ONEMAX are:

$$a_w = \begin{cases} n/2 & \text{if } |w| = 0, \\ -1/2 & \text{if } |w| = 1, \\ 0 & \text{if } |w| > 1. \end{cases} \quad (39)$$

Since all the nonzero Walsh coefficients have order 0 or 1, only the coefficients $A_{x,y}^{(0)}$ and $A_{x,y}^{(1)}$ can be nonzero, yielding a linear polynomial in ρ for the expected value $\mathbb{E}\{f(U_\rho(x, y))\}$.

LEMMA 2. *Let $x, y \in \mathbb{B}^n$ be two binary strings, the polynomial coefficients $A_{x,y}^{(0)}$ and $A_{x,y}^{(1)}$ for the ONEMAX problem are:*

$$A_{x,y}^{(0)} = \frac{1}{2}|x \oplus y| + |x \wedge y|, \quad A_{x,y}^{(1)} = -\frac{1}{2}|x \oplus y| + |\bar{x} \wedge y|. \quad (40)$$

PROOF. The development of the $A_{x,y}^{(0)}$ coefficient is:

$$\begin{aligned} A_{x,y}^{(0)} &= \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = 0}} a_w \psi_w(y) = \frac{n}{2} - \frac{1}{2} \sum_{\substack{i=1 \\ x_i = y_i}}^n (1 - 2y_i) \\ &= \frac{n}{2} - \frac{1}{2}(n - |x \oplus y|) + \sum_{\substack{i=1 \\ x_i = y_i}}^n y_i \\ &= \frac{1}{2}|x \oplus y| + \sum_{\substack{i=1 \\ x_i = y_i}}^n y_i \\ &= \frac{1}{2}|x \oplus y| + |\overline{(x \oplus y)} \wedge y|, \end{aligned}$$

where $\overline{(x \oplus y)}$ denotes the complement of $x \oplus y$ (bitwise XNOR). The binary string $\overline{(x \oplus y)}$ has 1 in the positions in which $x_i = y_i$. The development of the $A_{x,y}^{(1)}$ coefficient is:

$$\begin{aligned} A_{x,y}^{(1)} &= \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = 1}} a_w \psi_w(y) = -\frac{1}{2} \sum_{\substack{i=1 \\ x_i \neq y_i}}^n (1 - 2y_i) \\ &= -\frac{1}{2}|x \oplus y| + \sum_{\substack{i=1 \\ x_i \neq y_i}}^n y_i \\ &= -\frac{1}{2}|x \oplus y| + |(x \oplus y) \wedge y|, \end{aligned}$$

which gives the expressions in (40) taking into account that $(x \oplus y) \wedge y = x \wedge y$ and $(x \oplus y) \wedge y = \bar{x} \wedge y$. \square

THEOREM 2. *Let $x, y \in \mathbb{B}^n$ be two binary strings and $\rho \in [0, 1]$. In the ONEMAX problem an expression for $\mathbb{E}\{f(U_\rho(x, y))\}$ is:*

$$\mathbb{E}\{f(U_\rho(x, y))\} = |x \wedge y| + \rho|x \wedge \bar{y}| + (1 - \rho)|\bar{x} \wedge y|, \quad (41)$$

which allows one to efficiently evaluate $\mathbb{E}\{f(U_\rho(x, y))\}$ using bitwise operations and simple arithmetic.

PROOF.

$$\begin{aligned} \mathbb{E}\{f(U_\rho(x, y))\} &= A_{x,y}^{(0)} + A_{x,y}^{(1)}(1 - 2\rho) \\ &= A_{x,y}^{(0)} + A_{x,y}^{(1)} - 2\rho A_{x,y}^{(1)} \\ &= (|\bar{x} \wedge y| + |x \wedge y|) - 2\rho \left(-\frac{1}{2}|x \oplus y| + |\bar{x} \wedge y| \right). \end{aligned}$$

If we take into account that $|x \oplus y| = |\bar{x} \wedge y| + |x \wedge \bar{y}|$ we obtain (41) after some manipulation. \square

The formula (41) can be also explained as follows. The term $|x \wedge y|$ counts the bits which are 1 in both x and y , and these bits keep their value in any child. The term $|x \wedge \bar{y}|$ are the bits which are 1 in x and 0 in y and each one of these bits will be in the child with probability ρ . For this reason, the expected number of these bits in the child is $\rho|x \wedge \bar{y}|$. Finally, the term $|\bar{x} \wedge y|$ counts the bits which are 1 in y and 0 in x . These bits will be in the child with probability $1 - \rho$, which explains the contribution of $(1 - \rho)|\bar{x} \wedge y|$ to the expected value.

In the case of ONEMAX the formalism presented in this paper is not required to find the expectation formula. The argument in the last paragraph is enough to find an expression. For the MAX-kSAT problem the formalism is helpful, since it is difficult to reach a formula of the expectation using arguments similar to the ones in the previous paragraph.

4.2 MAX-kSAT

MAX-kSAT is an NP-hard combinatorial optimization problem with the objective of maximizing the number of satisfied clauses of a Boolean formula in conjunctive normal form. It is related with the SAT decision problem, since finding the optimum (maximum) in MAX-kSAT solves the related SAT decision problem.

Let us assume that n Boolean decision variables exist in the Boolean formula and let C be a set of clauses. In the MAX-kSAT problem each clause $c \in C$ is composed of k literals, each one being a decision variable x_i or a negated decision variable \bar{x}_i . For each clause $c \in C$ we define the vectors $v(c) \in \mathbb{B}^n$ and $u(c) \in \mathbb{B}^n$ as follows (see [5]):

$$v_i(c) = \begin{cases} 1 & \text{if } x_i \text{ appears (negated or not) in } c, \\ 0 & \text{otherwise,} \end{cases} \quad (42)$$

$$u_i(c) = \begin{cases} 1 & \text{if } x_i \text{ appears negated in } c, \\ 0 & \text{otherwise.} \end{cases} \quad (43)$$

According to this definition $u \wedge v = u$. The objective function of this problem is defined as

$$\begin{aligned} f(x) &= \sum_{c \in C} f_c(x); \quad \text{where} \\ f_c(x) &= \begin{cases} 1 & \text{if } c \text{ is satisfied with assignment } x, \\ 0 & \text{otherwise.} \end{cases} \quad (44) \end{aligned}$$

A clause c is satisfied with x if at least one of the literals is 1. Using the vectors $v(c)$ and $u(c)$ we can say that c is satisfied by x if $\bar{x} \wedge u \vee x \wedge v \wedge \bar{u} \neq 0$.

Sutton et al. [5] provide the Walsh decomposition for the MAX-kSAT problem. Let the function f_c evaluate one clause $c \in C$. The Walsh coefficients for f_c are:

$$a_w = \begin{cases} 0 & \text{if } w \wedge \bar{v} \neq 0, \\ 1 - \frac{1}{2^k} & \text{if } w = 0, \\ -\frac{1}{2^k} \psi_w(u) & \text{otherwise.} \end{cases} \quad (45)$$

The following provides the polynomial coefficients $A_{x,y}^{(r)}(c)$ for the function f_c , where we include the clause in the coefficient to distinguish the value of one clause from another.

LEMMA 3. *Let $x, y \in \mathbb{B}^n$ be two binary strings and $r \geq 0$. Then, the following identity holds for the polynomial coefficients $A_{x,y}^{(r)}(c)$ in the case of the function f_c :*

$$A_{x,y}^{(r)}(c) = \delta_0^r - \frac{\delta_0^\gamma}{2^\beta} \mathcal{K}_{r,\alpha}^{(\beta)}, \quad (46)$$

where $\alpha = |v(c) \wedge (x \oplus y) \wedge (u(c) \oplus y)|$, $\beta = |v(c) \wedge (x \oplus y)|$ and $\gamma = |v(c) \wedge \overline{(x \oplus y)} \wedge (u(c) \oplus y)|$.

PROOF. In the following we will remove the argument c in the vectors $v(c)$ and $u(c)$ to alleviate the notation. Let us assume that $r > 0$. The nonzero Walsh coefficients a_w are the ones for which $w \wedge \bar{v} = 0$, which are exactly $w \in \mathbb{B}^n \wedge v$, then we can restrict the sum of (34) to these binary strings. We can also assume that the strings w in the sum are $w \neq 0$, since $r > 0$. Then we can write:

$$\begin{aligned} A_{x,y}^{(r)} &= \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = r}} a_w \psi_w(y) \\ &= \sum_{\substack{w \in \mathbb{B}^n \wedge v \\ |(x \oplus y) \wedge w| = r}} \frac{-1}{2^k} \psi_w(u) \psi_w(y) \quad \text{by (45)} \\ &= \frac{-1}{2^k} \sum_{\substack{w \in \mathbb{B}^n \wedge v \\ |(x \oplus y) \wedge w| = r}} \psi_w(u \oplus y) \quad \text{by (8)}. \end{aligned} \quad (47)$$

We can now write each w as the sum of two strings w' and w'' where $w' \in \mathbb{B}^n \wedge (v \wedge (x \oplus y))$ and $w'' \in \mathbb{B}^n \wedge (v \wedge \overline{(x \oplus y)})$.

$$\begin{aligned} A_{x,y}^{(r)} &= \frac{-1}{2^k} \sum_{\substack{w' \in \mathbb{B}^n \wedge (v \wedge (x \oplus y)) \\ |w'| = r}} \sum_{w'' \in \mathbb{B}^n \wedge (v \wedge \overline{(x \oplus y)})} \psi_{w'+w''}(u \oplus y) \\ &= \frac{-1}{2^k} \left(\sum_{\substack{w' \in \mathbb{B}^n \wedge (v \wedge (x \oplus y)) \\ |w'| = r}} \psi_{w'}(u \oplus y) \right) \\ &\quad \cdot \left(\sum_{w'' \in \mathbb{B}^n \wedge (v \wedge \overline{(x \oplus y)})} \psi_{w''}(u \oplus y) \right). \end{aligned}$$

Let us now define $\alpha = |v(c) \wedge (x \oplus y) \wedge (u(c) \oplus y)|$, $\beta = |v(c) \wedge (x \oplus y)|$ and $\gamma = |v(c) \wedge \overline{(x \oplus y)} \wedge (u(c) \oplus y)|$. Then, using the results of Proposition 3 we can write:

$$\begin{aligned} A_{x,y}^{(r)} &= \frac{-1}{2^k} \mathcal{K}_{r,\alpha}^{(\beta)} \left(2^{|v \wedge (x \oplus y)|} \delta_0^\gamma \right) \\ &= -\frac{\delta_0^\gamma}{2^\beta} \mathcal{K}_{r,\alpha}^{(\beta)}, \end{aligned} \quad (48)$$

where we used the fact that $2^k = 2^\beta \cdot 2^{|v \wedge \overline{(x \oplus y)}|}$. When $r = 0$ we have to take into account that $w = 0$ is one possible string in the sum and $a_0 = 1 - 1/2^k$. Then we have:

$$\begin{aligned} A_{x,y}^{(0)} &= \sum_{\substack{w \in \mathbb{B}^n \\ |(x \oplus y) \wedge w| = 0}} a_w \psi_w(y) \\ &= a_0 - \frac{1}{2^k} \sum_{\substack{w \in \mathbb{B}^n \wedge v \\ |(x \oplus y) \wedge w| = 0, w \neq 0}} \psi_w(u \oplus y) \\ &= a_0 - \frac{1}{2^k} \sum_{\substack{w \in \mathbb{B}^n \wedge (v \wedge \overline{(x \oplus y)}) \\ w \neq 0}} \psi_w(u \oplus y) \\ &= a_0 - \frac{1}{2^k} \sum_{w \in \mathbb{B}^n \wedge (v \wedge \overline{(x \oplus y)})} \psi_w(u \oplus y) + \frac{1}{2^k} \\ &= 1 - \frac{1}{2^k} \sum_{w \in \mathbb{B}^n \wedge (v \wedge \overline{(x \oplus y)})} \psi_w(u \oplus y) \\ &= 1 - \frac{1}{2^k} \delta_0^\gamma 2^{|v \wedge \overline{(x \oplus y)}|} \\ &= 1 - \frac{\delta_0^\gamma}{2^\beta}. \end{aligned} \quad (49)$$

According to the definition of Krawtchouk matrices we have $\mathcal{K}_{0,\alpha}^{(\beta)} = 1$, which allows us to combine (48) and (49) to yield (46). \square

All the $A_{x,y}^{(r)}(c)$ coefficients for each particular clause can be efficiently computed in $O(k)$ time. The bitwise operations required to compute α , β and γ only need to explore the bits set to one in $v(c)$ (that is, k bits). With the values of α , β and γ each of the $k+1$ coefficients can be computed in $O(1)$. The coefficients for the MAX-kSAT objective function f are given in the following theorem.

THEOREM 3. *Let $x, y \in \mathbb{B}^n$ be two binary strings and $r \geq 0$. Then, the following identity holds for the polynomial coefficients $A_{x,y}^{(r)}$ of the MAX-kSAT problem:*

$$A_{x,y}^{(r)} = \sum_{c \in C} A_{x,y}^{(r)}(c) \quad (50)$$

where $A_{x,y}^{(r)}(c)$ is given by (46). These coefficient can be computed in $O(km)$ where m is the number of clauses.

PROOF. It is a direct consequence of Lemma 3. \square

COROLLARY 2. *Let $x, y \in \mathbb{B}^n$ be two binary strings and f the objective function for the MAX-kSAT problem, defined in (44). The expected value of the fitness function after applying UX to solutions x and y with bias $\rho = 1/2$ is:*

$$\mathbb{E}\{f(U_{1/2}(x, y))\} = m - \sum_{\substack{c \in C \\ \gamma(c) = 0}} \frac{1}{2^{\beta(c)}} \quad (51)$$

where $\beta(c) = |v(c) \wedge (x \oplus y)|$ and $\gamma(c) = |v(c) \wedge \overline{(x \oplus y)} \wedge (u(c) \oplus y)|$.

PROOF. Combining the results of Corollary 1 and Theorem 3 we obtain the desired result:

$$\mathbb{E}\{f(U_{1/2}(x, y))\} = \sum_{c \in C} \left(1 - \frac{\delta_0^{\gamma(c)}}{2^{\beta(c)}} \right)$$

\square

From these results, we conclude that the expectation curve of the fitness value after applying UX to two solutions in the MAX-kSAT problem is a polynomial in ρ with degree at most k .

5. FURTHER ANALYSIS

We next analyze some of the consequences of these results. In particular, we study what is the value of the expected fitness if the second child of the UX is selected instead of the first one. We also investigate the optimal value for the crossover bias ρ .

5.1 Expectation of the Sibling

Until now we only have considered one of the two possible children that UX generates. In particular, for two solutions x and y we always considered the one built taking the bits of x with probability ρ , which we call z . In this section we consider the other child, which have inverted bits in the positions in which x and y differ. The following result provides the expected fitness of this second child given by $z \oplus (x \oplus y)$.

THEOREM 4. *Let f be a pseudo-Boolean function defined over \mathbb{B}^n , $x, y \in \mathbb{B}^n$ two binary strings. The expected fitness of the second child in the UX when it is applied to x and y (in that order) with bias ρ is $\mathbb{E}\{f(U_\rho(y, x))\} = \mathbb{E}\{f(U_{1-\rho}(x, y))\}$.*

PROOF. The first child takes the bits of x with probability ρ and bits of y with probability $1 - \rho$. The second child takes the value of y with probability ρ and the value of x with probability $1 - \rho$. As a consequence to compute the expected fitness value for the second child we have to commute the order of x and y in the expectation formula without changing ρ or we have to replace ρ by $1 - \rho$ without changing the order of x and y in the expectation formula. \square

In Figure 1 we graphically interpret the result of the previous theorem. We can observe that the expectation curve (expectation as a function of ρ) of the second child is just a reflection of the one of the first child using as axis the line $\rho = 1/2$.

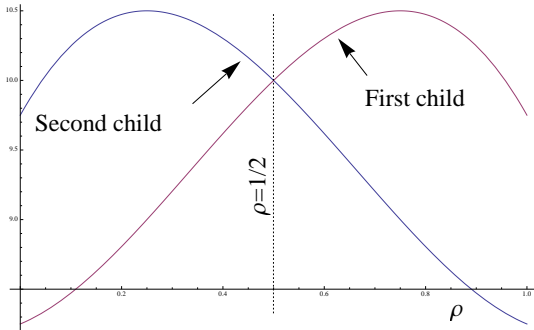


Figure 1: The expectation curve of the second child in the UX is a reflection of the expectation curve of the first child with respect to the line $\rho = 1/2$.

5.2 Optimal Crossover Bias

Given a particular problem and two solutions x and y , we can compute the coefficients $A_{x,y}^{(r)}$ of the polynomial and find the optimal value of ρ to maximize (or minimize) the

expected fitness of the child. This can always be done using numerical analysis, but we consider here some cases in which a closed-form formula can be derived, namely, when $\mathbb{E}\{f(U_\rho(x, y))\}$ is a polynomial of degree less than or equal to 3^1 .

In the order 1 case, the expectation is $\mathbb{E}\{f(U_\rho(x, y))\} = A_{x,y}^{(0)} + (1 - 2\rho)A_{x,y}^{(1)}$. In this case the optimum is one of the extremes: $\rho = 0$ or $\rho = 1$. If we consider maximization and $f(x) \geq f(y)$, then an optimal bias is $\rho^* = 1$, which can be interpreted as “select x as the child”. One problem having always a degree-1 polynomial is ONEMAX. In fact, if for a particular objective function f the expectation curves for all the possible solution pairs x, y is linear in ρ , then the function $f(x)$ has to be a weighed sum of the variables x_i . This kind of functions can always be solved in $O(n)$.

Consider the case in which the expectation is quadratic in ρ . Then the polynomial has a maximum if $A_{x,y}^{(2)} < 0$. We can find the derivative and solve the corresponding linear equation to obtain a tentative optimal value for ρ . The value for this optimum is

$$\rho = \frac{1}{2} + \frac{A_{x,y}^{(1)}}{4A_{x,y}^{(2)}}. \quad (52)$$

If this value is inside the interval $[0, 1]$ then it is an optimal value for the bias, otherwise, one optimal value is $\rho^* = 1$, since we assume $f(x) \geq f(y)$. One interesting observation here is that if (52) gives the value for the optimal bias, and $f(x) > f(y)$ then $\rho > 1/2$. That is, the optimal bias would suggest to increase the probability of selecting the components of the best solution (x). This scenario is graphically represented in Figure 2. This is common sense, since one expects the best individual to have the best solution components, those that increase the fitness value of the solution. Problems having a quadratic polynomial are those with at most two-variable interactions in their fitness function. For example, the subset sum problem [3] or the 0-1 Unconstrained Quadratic Optimization [4], both NP-hard problems.

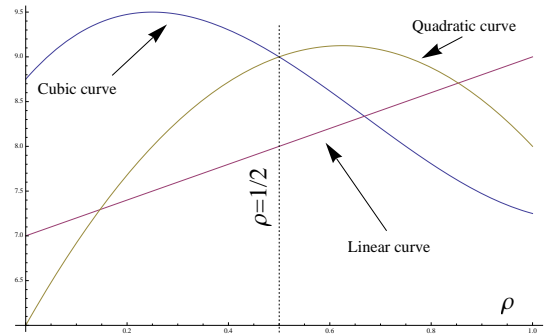


Figure 2: Shape of the expectation curves depending on the degree of the polynomial.

Finally, let us assume that $\mathbb{E}\{f(U_\rho(x, y))\}$ is a cubic polynomial. In this case the derivative polynomial can have zero,

¹Closed-form formulas of the expectation-optimal bias ρ can also be derived for polynomials up to degree 5, but we don't consider them here.

one or two roots in the interval $[0, 1]$. Their values are:

$$\rho = \frac{-A_{x,y}^{(2)} \pm \sqrt{(A_{x,y}^{(2)})^2 - 3A_{x,y}^{(1)}A_{x,y}^{(3)}}}{3A_{x,y}^{(3)}}. \quad (53)$$

It can happen that one of these values is the optimal bias for UX (it is not possible that both values are optimal) or it could be that the optimal value is in one of the extremes (or both). The interesting observation here is that if (53) computes the optimal bias, then it can happen that $\rho < 1/2$ (in Figure 2 we plot a cubic polynomial having this behaviour). In general, if the degree of the polynomial in ρ is higher than 2, then we can find situations in which increasing the probability of selecting components of the worst solution (y) the expected fitness value of the child is higher.

Table 1: Instance of the MAX-3SAT problem with $n = 7$ for which the optimal bias ρ is less than 0.5 when $x = 1111000$ and $y = 0000000$.

Clauses		
$x_1 \vee \bar{x}_2 \vee \bar{x}_3$	$x_5 \vee x_6 \vee \bar{x}_1$	$x_1 \vee x_2 \vee x_5$
$x_3 \vee \bar{x}_1 \vee \bar{x}_2$	$x_5 \vee x_6 \vee \bar{x}_2$	$x_1 \vee x_3 \vee x_5$
$x_2 \vee \bar{x}_1 \vee \bar{x}_3$	$x_5 \vee x_6 \vee \bar{x}_3$	$x_2 \vee x_3 \vee x_5$
$x_1 \vee \bar{x}_2 \vee \bar{x}_4$	$x_5 \vee x_6 \vee \bar{x}_4$	$x_1 \vee x_4 \vee x_5$
$x_4 \vee \bar{x}_1 \vee \bar{x}_2$	$x_5 \vee x_7 \vee \bar{x}_1$	$x_2 \vee x_4 \vee x_5$
	$x_6 \vee x_7 \vee \bar{x}_2$	$x_3 \vee x_4 \vee x_5$
		$x_1 \vee x_2 \vee x_6$

We designed an instance of MAX-3SAT for which we find the optimal bias ρ is less than 0.5. The instance has $n = 7$ variables and 18 clauses which are shown in Table 1. Let $x = 1111000$ and $y = 0000000$, then $f(x) = 12$ and $f(y) = 11$. For these two solutions the expectation curve for the uniform crossover is the one plotted in Figure 3, with expression

$$\mathbb{E}\{f(U_\rho(x, y))\} = \frac{101}{8} + \frac{1}{8}(1-2\rho) - \frac{9}{8}(1-2\rho)^2 - \frac{5}{8}(1-2\rho)^3.$$

The optimal expected fitness value is 12.6284 and can be found at $\rho = 0.473401 < 0.5$.

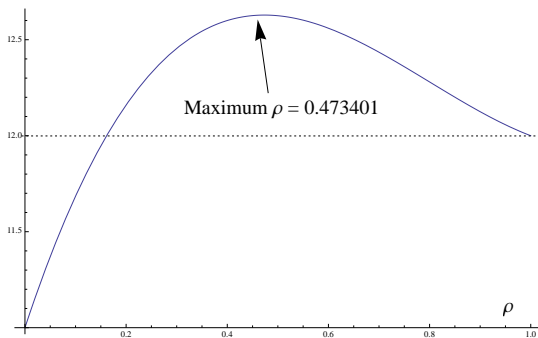


Figure 3: Expectation curve of the MAX-3SAT instance in Table 1 when $x = 1111000$ and $y = 0000000$. The cubic polynomial is $\mathbb{E}\{f(U_\rho(x, y))\} = \frac{101}{8} + \frac{1}{8}(1-2\rho) - \frac{9}{8}(1-2\rho)^2 - \frac{5}{8}(1-2\rho)^3$. The optimal expected value is 12.6284 at $\rho = 0.473401$.

6. CONCLUSIONS AND FUTURE WORK

We have derived an expression for computing the expected fitness value of a solution which is the result of applying the uniform crossover to two solutions x and y . Since UX has only one parameter, the bias ρ , this expression is obviously a function of ρ when x and y are fixed. We prove that this function is a polynomial in ρ and the degree of the polynomial is bounded by the number of bits in which x and y differ and the maximum order of the nonzero coefficients in the Walsh decomposition of the objective function.

We have developed the expression as a closed-form formula for two optimization problems: ONEMAX and MAX-kSAT. The complexity of computing the expectation for these two problems is similar to the complexity of evaluating the objective function. With the help of these polynomials it is possible to compute the bias for which the expected fitness is optimal, which could be used to create new crossover operators exploiting this information. We found that it is not always the case that the optimal value for ρ is above 0.5 and we have provided an instance of MAX-3SAT for which the optimal value is less than 0.5.

As future work we plan to extend the results in this paper and we plan to analyze other variation operators using the same tools. We can also propose new variation operators or search algorithms based on the expected fitness.

7. ACKNOWLEDGMENTS

This research has been partially funded by the Spanish Ministry of Science and Innovation and FEDER under contract TIN2011-28194 (the roadME project) and the Andalusian Government under contract P07-TIC-03044 (DIRICOM project).

8. REFERENCES

- [1] T. Baeck, D. B. Fogel, and Z. Michalewicz, editors. *Evolutionary Computation 1: Basic Algorithms and Operators*. Taylor & Francis, 2000.
- [2] P. Feinsilver and J. Kocik. Krawtchouk polynomials and krawtchouk matrices. In R. Baeza-Yates, J. Glaz, H. Gzyl, J. Hüslér, and J. Palacios, editors, *Recent Advances in Applied Probability*, pages 115–141. Springer US, 2005.
- [3] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman, 1979.
- [4] F. Glover, B. Alidaee, C. Rego, and G. Kochenberger. One-pass heuristics for large-scale unconstrained binary quadratic problems. *European Journal of Operational Research*, 137(2):272–287, Mar 2002.
- [5] A. M. Sutton, L. D. Whitley, and A. E. Howe. A polynomial time computation of the exact correlation structure of k-satisfiability landscapes. In *Proceedings of GECCO*, pages 365–372, New York, NY, USA, 2009. ACM.
- [6] A. Terras. *Fourier Analysis on Finite Groups and Applications*. Cambridge University Press, 1999.
- [7] J. L. Walsh. A closed set of normal orthogonal functions. *American Journal of Mathematics*, 45(1):5–24, 1923.