

A Non-Linearity Measure of a Problem's Crossover Suitability

Andrew Mason

Department of Engineering Science

School of Engineering, University of Auckland

Private Bag 92019, New Zealand

ph: +64 (9) 3737599 x7909, fax: +64 (9) 3737468

a.mason@auckland.ac.nz

ABSTRACT

For any problem with an unknown fitness function, justification of an Evolutionary Algorithm as a search method necessarily relies upon conjectures about that fitness function. This paper formulates apparent crossover partition coefficients (a generalisation of Walsh transforms) and uses these to develop a new model of crossover non-linearity ratios. Experimental runs demonstrate that this theory can offer insights into the apparent tractability of problems under crossover.

1. Introduction

With the increasing use of Evolutionary Algorithms (EAs), it is important that practitioners have some means by which the effectiveness of the EA's trademark operator, the crossover, can be measured. Traditional schema-processing based approaches such as those expounded by Holland [9] have proven of limited value [15, 13, 17], while the recent advances made by Vose and colleagues [11, 22] in theoretical characterisations of Genetic Algorithms (GAs) are currently unable to offer guidance of a practical value.

2. Focus on Exploration

If progress is to be made in our understanding of the GA it is essential that the focus of research move away from issues such as compliance with the Schema Theorem and the degree of intrinsic parallelism and instead consider the GA's exploration of new points in the solution space. If any non-enumerative search algorithm is to outperform random search it must make assumptions about the search space. Only by gaining a greater understanding of the implicit assumptions underlying the GA's search will we succeed in characterising those problems best suited to optimisation using GAs.

The hallmark of the GA is its use of the crossover operator to combine properties of two good solutions. If the GA is to distinguish itself then it must do so via this operator. Therefore, the understanding of crossover will serve as a focus for this discussion.

We consider the problem of finding a solution

(individual) that maximises some associated fitness function $f(\cdot)$. We will assume the GA is using a traditional 'cut and swap' crossover in which the children represent a strict redistribution (without loss) of the parental gene values. That is, we shall consider crossover to be any symmetric reversible operator that maps one pair of parental fitnesses to a second child pair. Clearly this symmetry requirement excludes crossover operators in which only a single child is produced. Furthermore, it also requires that the crossover be 'fitness blind'; we must be equally likely to travel 'forward' from parents to children as we are to travel 'backwards' from children to parents, even if one direction is considered an improvement while the other a degradation in quality. Fitness aware crossover operators that can effectively hill-climb differ fundamentally from the blind crossover considered here.

The GA is likely to outperform other algorithms on some problem if its crossover operator is able to exploit structure in the problem space. To achieve this, the crossover operator must prove successful in generating new improved solutions from the existing solution populations. That is, crossover must be able to generate improved solutions by rearranging building blocks present in the existing good solutions. Because the crossover operator is reversible, no claims can be made as to its ability per se. Instead, we require that crossover of good individuals produce at least one better than average (and hopefully improved) individual.

Previous characterisation of problems as suitable or otherwise for solution using crossover has focussed on two approaches – the construction of easy and hard problems, and the analysis of ex-

isting problems. Both these approaches attempt in some way to consider the interaction of building blocks. A formal analysis of such interactions was first made by Bethke [2] who introduced *partition coefficients* for analysing binary-encoded individuals. Partition coefficients have recently been generalised to non-binary encodings by Mason [14], and it is this notation that we use here. Each schema H has an associated partition coefficient $\epsilon(H)$ that quantifies the non-linearities in the fitness function associated with H . Using Mason's notation, the $\epsilon(\cdot)$ values can be uniquely defined by:

$$f(H) = \sum_{H' \supseteq H} \epsilon(H') \quad (1)$$

where $H' \supseteq H$ denotes that H is a subset of H' , e.g. $1*1*3* \supseteq 1*123*$. The partition coefficients are defined over the set of *all* n possible individuals, and consequently equation (1) allows the fitness of any schema (or individual) to be written as a sum of appropriate partition coefficients. Note, however, that dependency relationships between the partition coefficients reduce the number of independent coefficients to the n required for the system to be uniquely determined. For a full appreciation of partition coefficients, the reader is referred to the above papers or one of [3, 5, 6, 8, 7, 10].

In the simplest case, a function can be written as the sum of the first-order partition coefficients associated with individual genes. Such a problem would be classed as *veridical*, in that the optimal solution is contained within the best of each set of competing schemata, and in fact can be found by independently choosing values for each gene. A number of authors have also created *deceptive* problems in which the apparently best schemata contain some false optimum to which the GA invariably converges. Such problems can be created through careful choice of the associated partition coefficients.

Partition coefficients and linearity measures can also be used to characterise an existing function. This is the approach taken by Davidor [4] who introduced *epistasis variance* as a measure of the difference between a function and its gene-wise linear approximation. Motivated by Davidor's work, Reeves and Wright [21, 20] have recently taken a statistical approach to estimating apparent partition coefficients. In their work they use analysis of variance (ANOVA) tables to determine the extent to which low order partition coefficients can explain the fitness function. They also discuss how such an analysis extends to the analysis of subsets of individuals as seen during actual GA runs.

Manderick et al. [12] have considered a statistically based *correlation coefficient* r for the crossover operator, where the range of values from $r = 0$

to $r = 1$ indicates the degree of linear separability of the fitness function. In similar work, Mühlenbein and Schlierkamp-Voosen [18] examine the 'heritability' of individuals across each successive generation by forming a linear regression model relating the fitness of a crossover-produced child to the average fitness of the two parents, and then calculating correlation coefficients that measure the degree to which the linear regression model explains the observed data.

Menczer and Parisi [16] compared the performance of GAs with and without crossover in the optimisation of weights in a neural net, and found that crossover did make a significant contribution to the GA's performance. They conducted experiments measuring the correlation coefficients for both the mutation and crossover operators as these operations were observed during the actual GA run. They found that the fitness landscape appeared more correlated (and thus smoother) under the crossover operator than under mutation. (Note that under crossover Menczer and Parisi considered a child as belonging to the one parent (of the two possible) with which it matched in the greatest number of gene positions; the correlation coefficient was based on each such parent/child pair, and not the average parent/child fitness.)

In general, the use of some algorithm to solve a particular problem can only be supported by appeal to specific properties of the fitness function under consideration. Given the inability to fully enumerate the fitness function, the strongest support for such claims must derive from sample observations of the function. While several of the previously discussed correlation measures go some way in serving this need, the next section presents a crossover-specific measure that can successfully quantify those properties of the crossover operator that we believe to be of greatest importance.

3. Crossover Non-Linearity Ratios

Work conducted by the author [15] has shown that the GA makes rapid progress on both veridical and deceptive problems, suggesting that both problems present fitness paths that can be easily and reliably ascended by the GA to either the optimum or a near-optimal point. One explanation for the rapid progress of the GA on these veridical and deceptive problems is that they are both linearly separable to a high degree — their fitness functions, at least over the range seen by the GA's exploration, can be written as a linear sum of coefficients associated with each gene position. It is this effective separability of both problems that permits the GA's rapid progress.

While linearly separable functions are GA-easy, such problems are also solved more efficiently using

simple bit-flipping hill climbers. If we are to characterise those problems for which GAs excel as a solution method then we need to isolate that property of linearly separable functions that makes such functions crossover-easy but also generalise this beyond a simple classification of trivially separable problems.

Let us assume that individuals I_{P1} and I_{P2} form children I_{C1} and I_{C2} after crossover. Now, letting a, b, p and q denote the appropriate gene subsequences, we can write $I_{P1} = ab, I_{P2} = pq, I_{C1} = aq$ and $I_{C2} = pb$. If we treat the representation as comprising two binary genes, the first with alleles $\{a, p\}$, and the second with alleles $\{b, q\}$ then we can treat the four individuals associated with the given crossover as if they formed a complete problem domain. In particular, we can calculate the 4 unique partition coefficients defined by the fitnesses $f(ab), f(pq), f(aq), f(pb)$ over the domain $\{a, p\} \times \{b, q\}$; these are termed the crossover's pseudo partition coefficients. Using the partition coefficient notation presented earlier, the four independent coefficients $\epsilon(**), \epsilon(a*) = -\epsilon(p*), \epsilon(*b) = -\epsilon(q*),$ and $\epsilon(ab) = -\epsilon(aq) = -\epsilon(pb) = \epsilon(pq)$ are defined by equation (1), giving

$$\begin{aligned} f(I_{P1}) &= f(ab) \\ &= \epsilon(**) + \epsilon(a*) + \epsilon(*b) + \epsilon(ab) \quad (2) \end{aligned}$$

$$\begin{aligned} f(I_{P2}) &= f(pq) \\ &= \epsilon(**) - \epsilon(a*) - \epsilon(*b) + \epsilon(ab) \quad (3) \end{aligned}$$

$$\begin{aligned} f(I_{C1}) &= f(aq) \\ &= \epsilon(**) + \epsilon(a*) - \epsilon(*b) - \epsilon(ab) \quad (4) \end{aligned}$$

$$\begin{aligned} f(I_{C2}) &= f(pb) \\ &= \epsilon(**) - \epsilon(a*) + \epsilon(*b) - \epsilon(ab). \quad (5) \end{aligned}$$

The relationships between the fitnesses and these pseudo partition coefficients are shown in Figure 1. It is important to note that these $\epsilon(\cdot)$ quantities are apparent partition coefficients associated with just the single crossover operation under consideration; they are not the values that would be generated by a full partition coefficient analysis over the entire problem domain.

This formulation suggests that we think of the crossover operation as a rearranging of the building blocks $\pm\epsilon(a*)$ and $\pm\epsilon(*b)$, the parents and children representing all four possible combinations of these building blocks. The $\epsilon(ab)$ is a nonlinear contribution that is then added to the parents or subtracted from the children. We can quantify the linear and non-linear effects by considering the ratio of these two contributions:

$$\psi'(I_{P1}, I_{P2}, I_{C1}, I_{C2}) \stackrel{\text{def}}{=} \frac{|\epsilon(ab)|}{|\epsilon(a*)| + |\epsilon(*b)|}.$$

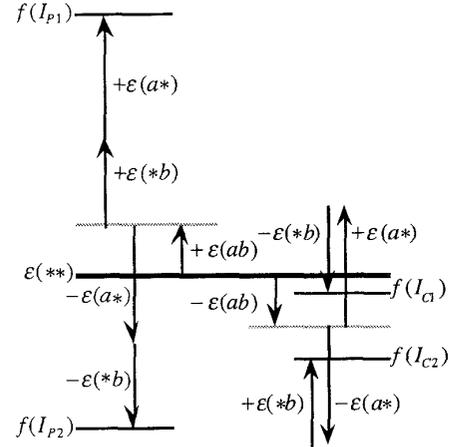


Fig. 1: An example showing the apparent partition coefficients associated with a crossover. For clarity this example assumes that $\epsilon(a*), \epsilon(*b)$ and $\epsilon(ab)$ are all positive, and $f(I_{C1}) > f(I_{C2})$. This figure is best understood by tracing the successive fitness changes involved in progressing from the mean family fitness $\epsilon(**)$ through to each individual's fitness.

The above definition has not considered the *direction* of any non-linearity. The linear analysis will generally suggest a certain combination of building blocks as being good. As an example, consider the case where $f(a*) > f(p*) \Leftrightarrow \epsilon(a*) > 0$, and $f(*b) > f(*q) \Leftrightarrow \epsilon(*b) > 0$. Now, if the non-linear interaction term $\epsilon(ab)$ is zero, then individual ab will be the most fit with fitness $f(ab) = \epsilon(**) + \epsilon(a*) + \epsilon(*b)$ (see equations (2)-(5)). However, in most cases $\epsilon(ab)$ will not be zero, and so by equation (2), $f(ab)$ will be increased by $\epsilon(ab)$. Now, a positive $\epsilon(ab)$ will be aligned with, and thus reinforce the fitness contributions of $\epsilon(a*)$ and $\epsilon(*b)$ within individual ab , while a negative $\epsilon(ab)$ will oppose the linear components, and perhaps reduce ab 's fitness below that of one of the other three family members. If $f(a*) < f(p*) \Leftrightarrow \epsilon(a*) < 0$, and $f(*b) < f(*q) \Leftrightarrow \epsilon(*b) < 0$, then individual pq will be the most fit if $\epsilon(ab) = 0$. If $\epsilon(ab) \neq 0$, then as before, a positive $\epsilon(ab)$ will improve pq 's fitness (i.e. reinforce the fitness contributions of $\epsilon(a*)$ and $\epsilon(*b)$), while a negative $\epsilon(ab)$ will oppose the linear components and thus worsen pq 's fitness. The converse will be true when $\epsilon(a*)$ and $\epsilon(*b)$ are of opposite sign; i.e. a negative $\epsilon(ab)$ will improve the best individual's fitness, while a positive $\epsilon(ab)$ will worsen that individual's fitness. We can take these effects into account to construct an improved non-linearity measure ψ .

Let $\text{sign}(x)$ denote the sign of x (being +1 if $x \geq 0$ and -1 otherwise). We now form what we shall term the *crossover non-linearity ratio*:

$$\psi(I_{P1}, I_{P2}, I_{C1}, I_{C2}) \stackrel{\text{def}}{=} \frac{\epsilon(ab)}{\text{sign}[\epsilon(a*)\epsilon(*b)] \cdot [|\epsilon(a*)| + |\epsilon(*b)|]}$$

If ψ is near zero then the fitness function can be modelled locally as a linear function of building blocks. If this is the case for many crossover operations then crossover of fit individuals will recombine these blocks in a linear fashion, thus generating improved individuals. We would expect the GA to be successful in generating improved solutions for such problems. This scenario of low ψ values includes the special case of $\psi \equiv 0$ for fully linearly separable functions.

A large ψ (either positive or negative) indicates that crossover has failed to locate linear building blocks. If many crossovers give such ψ values then we would suspect a lack of useful building blocks and thus a problem unsuited to GAs. In such cases the focus on good solutions provided by selection cannot aid the search process because the fitnesses of good individuals are associated with features of the individuals that crossover is unable to manipulate. Indeed, the large non-linearity of each crossover will dominate the crossover operation, giving offspring whose fitnesses differ little from those of randomly generated individuals — a degenerate random search is likely to result.

If ψ is both large and negative then the non-linear component of the crossover is not only large but also working in opposition to the linear building blocks; not only is the function non-linear, but also locally *crossover-deceptive*. Analysis of the sign of the ψ values can offer insights into any deceptive structures within a problem.

We note that the symmetry of crossover will result in the same ψ value being generated if either the parents are crossed to produce the children, or the children are crossed to produce the parents. The value of ψ is a measure of the underlying fitness structure and as such goes beyond any characterisations of the fitness function based simply upon the observable improvements generated by crossover.

Given the insensitivity to direction in ψ , let us assume, without loss of generality, that $|f_{P1} - f_{P2}| \geq |f_{C1} - f_{C2}|$, i.e. the ‘parents’ have the greater fitness range. The following result is proven in Appendix A.

$$f_{C1} + f_{C2} = f_{P1} + f_{P2} + 2\psi(I_{P1}, I_{P2}, I_{C1}, I_{C2})|f_{P1} - f_{P2}|$$

That is, ψ is the correlation coefficient relating the non-linear change in mean fitness to the linear component given by the fitness range in the parents.

It is easy to see from the above expression that our measure $\psi(I_{P1}, I_{P2}, I_{C1}, I_{C2})$ is invariant under an affine fitness scaling $f'(I) = af(I) + b$. We would expect this to be so given that the underlying difficulty of a problem is in no way changed by such scaling.

4. Experimental Results

In our discussion we have argued that crossover of fit individuals will successfully generate improved solutions whenever the fitness function, at least over the range of good individuals focused on by the GA, is such that the crossover operations produce near-zero ψ values. In this section we illustrate our work with brief summaries of a number of initial experimental results. These experiments are discussed more fully in Mason [15].

The first problems we consider are the functions developed by Tanese and subsequently studied by Mitchell and Forrest [5]. Tanese found that these functions were difficult to solve using the GA, a result which Mitchell and Forrest confirmed. We conducted our own experiments on 4 of these functions, defined by chromosome lengths $k = 32$ and $k = 128$ and the length of the contributing non-zero partition coefficients ∂ being either unrestricted or $\partial = 8$. For each problem, the GA was run for exactly 22 generations and the best fitness and mean ψ and $|\psi|$ values across all crossovers recorded. After performing 250 such runs the means ($\% \bar{f}$) and standard errors ($s_{\% \bar{f}}$) of these values were calculated, along with the percentage of runs finding the optimum ($\% \text{opt}$); this data is shown in Table 1. The crossover and mutation rates used for the table entries were chosen as follows. The first entry for each problem corresponds to that value of p_c which maximised $\% \bar{f}$ using a low mutation rate $p_m = 0.001$, while the second is for runs with no crossover and a mutation rate p_m chosen to again maximise $\% \bar{f}$. Thus each pair of entries for a problem indicates the best performance achievable using either crossover (with a low rate of mutation) or just mutation.

For the first two problems in the table we see that mutation generates a better search than crossover – crossover’s performance is no better than a poor mutation operator. These problems should not be solved using a GA. For the two problems with $\partial = 8$, and in particular the second of these, we see that crossover has successfully exploited properties inherent in the problems in a way that cannot be mimicked by mutation. When we consider the $|\psi|$ values, we observe that the lower values are indeed associated with those problems for which crossover’s contribution is most beneficial. Indeed, the smallest $|\psi|$ value corresponds to a problem in which crossover found the optimum 10 times more frequently than runs using just mutation. This table shows that crossover can contribute positively to solving these problems if the defining length ∂ is kept sufficiently low.

The second example we consider is a 2 class minimal flow time job scheduling problem with changeover times incurred whenever processing switches between classes. Using a suite of test

	p_c	p_m	$\% \bar{f}$	$s_{\% \bar{f}}$	$\% \text{opt}$	$\bar{\psi}$	$s_{\bar{\psi}}$	$ \bar{\psi} $	$s_{ \bar{\psi} }$
$k = 32$	1.0	0.001	81.62	0.24	0.0	-1.717	0.037	5.281	0.031
	0.0	0.050	85.69	0.18	0.0				
$k = 128$	0.9	0.001	89.66	0.22	0.0	-1.352	0.031	4.475	0.027
	0.0	0.010	94.70	0.14	0.8				
$k = 32$ $\partial = 8$	1.0	0.001	98.14	0.13	37.6	-1.717	0.030	3.218	0.028
	0.0	0.110	96.83	0.16	20.0				
$k = 128$ $\partial = 8$	1.0	0.001	98.47	0.10	29.6	-0.249	0.007	0.700	0.006
	0.0	0.025	94.01	0.16	2.4				

Table 1: Analysis of various Tanese Functions solved by the GA firstly using an experimentally optimised crossover rate p_c (with a low mutation rate $p_m = 0.001$), and secondly using an experimentally optimised mutation rate p_m with $p_c = 0$. Each problem was solved 250 times using exactly 22 generations. The average quality of the GA's solutions ($\% \bar{f}$, where 100% is the known maximum), the standard error in this value, and the percentage of problems solved optimally ($\% \text{opt}$) by the GA are given.

problems we tested our GA using a method similar to that discussed above and obtained the results shown in Table 2. Note that 250 runs were made; $\% \text{err}$ gives the deviation from optimality, and $\% \text{opt}$ gives the percentage of runs finding the true optimum. As before we observe that the ability of crossover to outperform mutation appears to be associated with low $|\bar{\psi}|$ values.

p_c	p_m	$\% \text{err}$	$s_{\% \text{err}}$	$\% \text{opt}$
0	0.15	0.045	0.004	76.0
1	0.05	0.036	0.004	83.6
		$\bar{\psi} = 0.2437, s_{\bar{\psi}} = 0.0049$		
		$ \bar{\psi} = 0.6511, s_{ \bar{\psi} } = 0.0048$		

Table 2: The percentage error in the GA's flow times above the known minima ($\% \text{err}$), the standard error in this value, and the percentage of problems solved optimally ($\% \text{opt}$) by the GA for a sequencing test suite using an experimentally optimised crossover rate p_c (with a low mutation rate $p_m = 0.05$), and secondly using an experimentally optimised mutation rate p_m with $p_c = 0$. The ψ values for the $p_c = 1$ run are also given.

5. Conclusion

We believe that the concept of generating new solutions from the best elements of existing solutions is a powerful one, and this contribution should not be obscured by all-encompassing claims for the GA's utility based on uncertain evolutionary analogies. We hope that our discussion of the issues associated with crossover and our contribution of crossover non-linearity ratios and initial supporting experimental results will encourage further research on improved characterisations of crossover and consequently, a better understanding of those problems for which GAs are an appropriate solution method.

References

- [1] R. Belew and L. Booker, eds., *Proceedings of the Fourth International Conference on Genetic Algorithms*. San Mateo, CA: Morgan Kaufmann, 1991.
- [2] A. Bethke, "Genetic algorithms as function optimizers," *Dissertation Abstracts International*, vol. 41, no. 9, 1980. (Doctoral dissertation, University of Michigan, 3503B, University Microfilms No. 8106101).
- [3] C. Bridges and D. Goldberg, "A note on the non-uniform Walsh-schema theorem," TCGA Report No. 89004, The Clearinghouse for Genetic Algorithms, Tuscaloosa: University of Alabama, 1989.
- [4] Y. Davidor, "Epistasis variance: A viewpoint on GA-hardness," in [19], 1991.
- [5] S. Forrest and M. Mitchell, "What makes a problem hard for a genetic algorithm? Some anomalous results and their explanation," *Machine Learning*, vol. 13, pp. 285–319, 1993.
- [6] D. Goldberg, "Genetic algorithms and Walsh functions: Part I, a gentle introduction," *Complex Systems*, vol. 3, pp. 129–152, 1989.
- [7] D. Goldberg, "Genetic algorithms and Walsh functions: Part II, deception and its analysis," *Complex Systems*, vol. 3, pp. 153–171, 1989.
- [8] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Massachusetts: Addison-Wesley, 1989.
- [9] J. Holland, *Adaptation in natural and artificial systems*. Ann Arbor: The University of Michigan Press, 1975.
- [10] J. Holland, "Searching nonlinear functions for high values," *Applied Mathematics and Computation*, vol. 32, pp. 255–274, 1989.
- [11] J. Juliany and M. Vose, "The genetic algorithm fractal," *Evolutionary Computation*, vol. 2, no. 2, 1994.
- [12] B. Manderick, M. de Weger, and P. Spiessens,

- “The genetic algorithm and the structure of the fitness landscape,” in [1], 1991.
- [13] A. Mason, “Crossover non-linearity ratios and the genetic algorithm: Escaping the blinkers of schema processing and intrinsic parallelism,” Technical Report No. 535b, Department of Engineering Science, University of Auckland, Private Bag 92019, Auckland, New Zealand, 1993.
- [14] A. Mason, “Partition coefficients, static deception and deceptive problems for non-binary alphabets,” in [1], 1991.
- [15] A. Mason, “Genetic algorithms and scheduling problems,” Ph.D. Dissertation, University of Cambridge, Cambridge, U.K., 1992.
- [16] F. Menczer and D. Parisi, “Evidence of hyperplanes in the genetic learning of neural networks,” *Biological Cybernetics*, vol. 66, pp. 283–289, 1992.
- [17] H. Mühlenbein, “Evolution in time and space - the parallel genetic algorithm,” in [19], 1991.
- [18] H. Mühlenbein and D. Schlierkamp-Voosen, “The science of breeding and its application to the Breeder Genetic Algorithm (BGA),” *Evolutionary Computation*, vol. 1(4), 1993.
- [19] G. Rawlins, ed., *Foundations of Genetic Algorithms*. Los Altos, CA: Morgan Kaufman, 1991.
- [20] C. Reeves and C. Wright, “Epistasis in genetic algorithms: An experimental design perspective,” Technical Report, School of Mathematical and Information Sciences, Coventry University, U.K., 1995.
- [21] C. Reeves and C. Wright, “An experimental design perspective on genetic algorithms,” Technical Report, School of Mathematical and Information Sciences, Coventry University, U.K., 1995.
- [22] M. Vose, “Generalizing the notion of schema in genetic algorithms,” *Artificial Intelligence*, vol. 50, pp. 385–396, 1989.

A. Proof of Equation (3)

Let $P = f(I_{P1}) - f(I_{P2})$, and $C = f(I_{C1}) - f(I_{C2})$. Now, from equations (2)-(5), we note that

$$\epsilon_{ab} = \frac{f(I_{P1}) + f(I_{P2}) - f(I_{C1}) - f(I_{C2})}{4}$$

$$\begin{aligned} \epsilon_{a*} &= \frac{[f(I_{P1}) - f(I_{P2})] + [f(I_{C1}) - f(I_{C2})]}{4} \\ &= (P + C)/4 \\ \epsilon_{*b} &= \frac{[f(I_{P1}) - f(I_{P2})] - [f(I_{C1}) - f(I_{C2})]}{4} \\ &= (P - C)/4 \end{aligned}$$

Now, let $Q = |\epsilon_{a*}| + |\epsilon_{*b}|$, giving

$$\begin{aligned} 4Q &= |P + C| + |P - C| \\ &= (P + C) \cdot \text{sign}(P + C) \\ &\quad + (P - C) \cdot \text{sign}(P - C). \end{aligned}$$

We note that if $|P| \geq |C|$, then $\text{sign}(P + C) = \text{sign}(P - C) = \text{sign}(P)$, and so

$$\begin{aligned} 4Q &= P \cdot \text{sign}(P) + C \cdot \text{sign}(P) \\ &\quad + P \cdot \text{sign}(P) - C \cdot \text{sign}(P) \\ &= 2|P| \end{aligned}$$

Alternatively, if $|P| \leq |C|$ then $\text{sign}(P + C) = -\text{sign}(P - C) = \text{sign}(C)$, and so

$$\begin{aligned} 4Q &= P \cdot \text{sign}(C) + C \cdot \text{sign}(C) \\ &\quad - P \cdot \text{sign}(C) + C \cdot \text{sign}(C) \\ &= 2|C|. \end{aligned}$$

Thus we have $Q = \max(|P|, |C|)/2$. This proves that

$$\begin{aligned} |\epsilon_{a*}| + |\epsilon_{*b}| \\ &= \max[|f(I_{P1}) - f(I_{P2})|, |f(I_{C1}) - f(I_{C2})|]/2 \end{aligned}$$

Now, arbitrarily assign a parent/child direction to the crossover such that $|f(I_{P1}) - f(I_{P2})| \geq |f(I_{C1}) - f(I_{C2})|$, giving $|P| \geq |C|$, and hence

$$\begin{aligned} \text{sign}(\epsilon(a*)\epsilon(*b)) &= \text{sign}\left(\frac{P+C}{4}\right) \text{sign}\left(\frac{P-C}{4}\right) \\ &= \text{sign}(P)\text{sign}(P) \\ &= 1 \end{aligned}$$

Given $|f(I_{P1}) - f(I_{P2})| \geq |f(I_{C1}) - f(I_{C2})|$, we have $|\epsilon_{a*}| + |\epsilon_{*b}| = |f(I_{P1}) - f(I_{P2})|/2$, and thus

$$\psi = \frac{[f(I_{P1}) + f(I_{P2}) - f(I_{C1}) - f(I_{C2})]/4}{|f(I_{P1}) - f(I_{P2})|/2}$$

giving the result

$$\begin{aligned} f(I_{P1}) + f(I_{P2}) &= f(I_{C1}) + f(I_{C2}) \\ &\quad + 2\psi(I_{P1}, I_{P2}, I_{C1}, I_{C2})|f(I_{P1}) - f(I_{P2})|. \end{aligned}$$