Problem Hardness in Evolutionary Computation

Edgar Galván López

Trinity College Dublin

Distributed Systems Group
School of Computer Science & Statistics
Dublin, Ireland

November, 2012

Outline

EC Methods (1/2)

- Evolutionary Computation (EC) techniques allow computer systems to learn.
- EC methods (e.g., Genetic Algorithms, Genetic Programming) are inspired by biological mechanisms of evolution.

EC Methods (2/2)

- Despite the effectiveness of EC systems (e.g., see Hummies), they also have limitations.

Neutral Theory

- Neutral Theory was defined by Kimura 1960s. The theory suggests that different forms of the same gene are indistinguishable in their effects.

Controversial Claims on Neutrality!

- 'Finding Needles in Haystacks is not Hard with Neutrality' by Yu and Miller, EvoStar 2002.
- 'Finding Needles in Haystacks is Harder with Neutrality' by Collins, GECCO 2005.
- **Both papers were nominated as best papers in their conference tracks!**

Confusion regarding neutrality has several reasons (1/2):

- there is a lack of mathematical frameworks that explain how and why neutrality affects evolution;
- many studies have based their conclusions on performance statistics (i.e., on whether or not a system with neutrality could solve a particular problem faster or better than a system without neutrality), rather than a more in-depth analysis based on problem hardness measures and search characteristics;
- studies have often considered problems, representations and search algorithms that are relatively complex; as a consequence, results represent the compositions of multiple effects (e.g., bloat or spurious attractors in genetic programming);

Confusion regarding neutrality has several reasons (2/2):

- there is not a single definition of neutrality, and different studies have added neutrality to systems in radically different ways;
- very often studies focused their attention on particular 'properties' of neutrality without properly defining them; and
- the features of a problem's landscape and the behaviour of the search operators change when neutrality is artificially added, but rarely effort has an been made to understand in exactly what ways.

Fitness Distance Correlation (*fdc*)

- Fitness distance correlation (*fdc*) measures the hardness of a landscape according to the correlation between the distance from the optimum and the fitness of the solution.
- Given a set given a set $F = \{f_1, f_2, ..., f_n\}$ of fitness values of n individuals and the corresponding set $D = \{d_1, d_2, ..., d_n\}$ of distances to the nearest optimum, we compute the correlation coefficient r,

$$r = \frac{C_{FD}}{\sigma_F \sigma_D},$$

where:

$$C_{FD} = \frac{1}{n} \sum_{i=1}^{n} (f_i - \overline{f})(d_i - \overline{d})$$

Classification of hardness in *fdc*

- According to Jones, a problem can be classified in one of three classes:
  1. *misleading* ($r \geq 0.15$), in which fitness tends to increase with the distance from the global optimum,
  2. *difficult* ($-0.15 < r < 0.15$), for which there is no correlation between fitness and distance, and
  3. *easy* ($r \leq -0.15$), in which fitness increases as the global optimum approaches.

Samples of fitness distance scatter plots

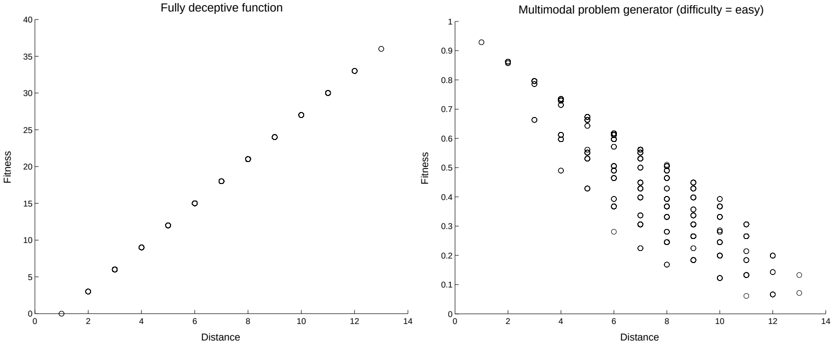


Figure : 1. A scatter plot for a fully deceptive function ($fdc = 1.0$) and a scatter plot for a multimodal problem of difficulty easy ($fdc = -0.8553$), shown in the left and right panels, respectively.

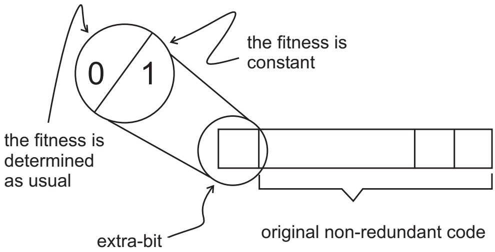


Figure : 1. Creation of a typical GA individual using Constant Neutrality.

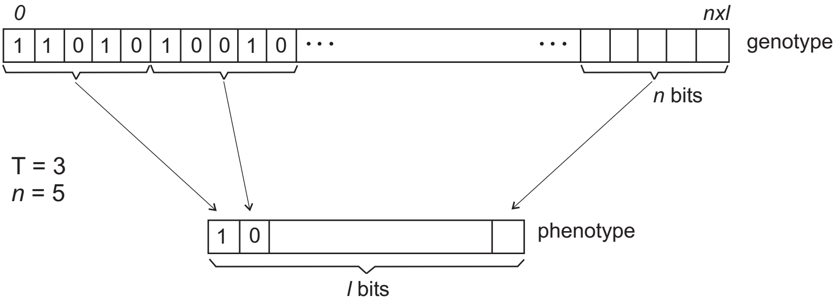
We try to shed some light on neutrality. That is:

- We consider a mutation based, binary GA without crossover.
- We analise performance figures and population flows from and to the neutral network and the basins of attraction of the optima.
- We use two problems: a unimodal landscape where we expect neutrality to always be detrimental and a multimodal deceptive landscape, where there are conditions where neutrality is more helful than others.

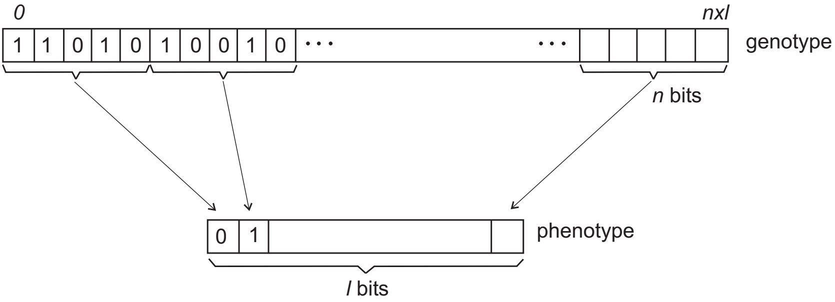
For analysis purposes:

- In the presence of the form of neutrality discussed above, the landscape is therefore divided into two areas of identical size, which we call *neutral layer* and *normal layer*.
- For bit strings of length l there are 2^l points in each layer. However, we still only have one global optimum.
- Neutrality is often reported to help in multimodal landscapes, so, in the case of our multimodal deceptive problem, should we expect a uniform neutral network to increase performance? And what sort of population dynamics should we expect?
- For analysis purposes we further divide the layers depending on which of the two basins of attraction a string belongs to. That is: "global neutral", "local neutral", "global normal" and "local normal".

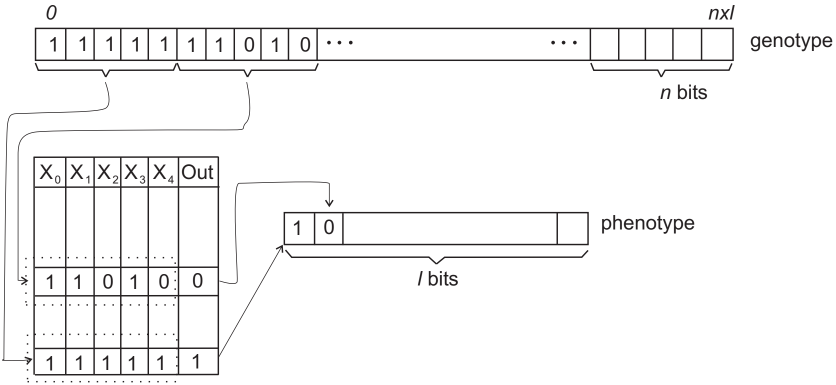
Majority Encoding



Parity Encoding



Truth Table encoding



Search Space VS. Solution Space

- Because each bit is encoded using n bits, the same phenotype can be obtained from different genotypes, and so, neutrality is artificially added.
- The search of the search space is $2^{\ell n}$, where ℓ is the length of a bit string and n is the number of bits required to encode each bit.
- With the types of encodings explained earlier, we have increased not only the size of the search space but also the size of the solution space.
- Neutrality is often reported to help in multimodal landscapes, in that it can prevent a searcher from getting stuck in local optima.

OneMax problem

- The first problem used is the OneMax problem which consist in maximizing the number of ones of a bitstring. Seen as a function of unitation the problem is represented by $f(u) = u$ or $f(x) = u(x)$.

Multimodal problem generator

- For the second problem, we used the multimodal problem generator. The idea is to create problem instances with a certain degree of multimodality

$$Peak_n(x) = \arg\min_i H(Peak_i, x)$$

$$f(x) = \frac{\ell - H(x, Peak_n(x))}{\ell} \times Height(Peak_n(x))$$

Trap function

- The third problem is a Trap function, which is a deceptive function of unitation. For this example, we have used the function:

$$f(x) = \begin{cases} \frac{a}{u_{min}}(u_{min} - u(x)) & \text{if } u(x) \leq u_{min}, \\ \frac{b}{\ell - u_{min}}(u(x) - u_{min}) & \text{otherwise} \end{cases}$$

where a is the deceptive optimum, b is the global optimum, and u_{min} is the slope-change location.

Example of a Trap Function

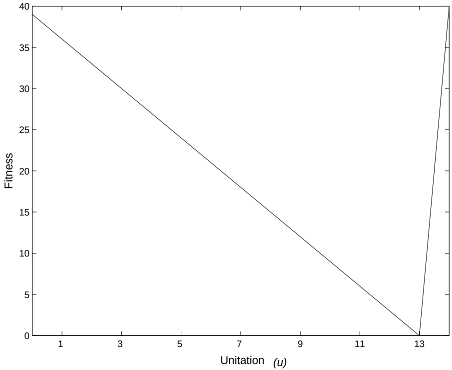


Figure : . Trap function used in our experiments ($u_{min} = 13$, a=39, $b = 40$).

MAX-SAT Problem Class

- The target in Boolean satisfiability problem (SAT) is to determine whether it is possible to set the variables of a given Boolean expression in such a way to make the expression true. The expression is said to be satisfiable if such an assignment exists.
- A related problem, known as the Maximum Satisfiability problem, or MAX-SAT, consists in determining the maximum number of clauses of a given Boolean formula that can be satisfied by some assignment. MAX-k-SAT is the maximum satisfiability problem for k-SAT instances.

MAX-SAT Problem Class

- We treat MAX-3-SAT as an optimisation problem with the following objective function:

$$f(x) = \sum_{i=1}^{c} S_i(x),$$

where $S_i(x)$ is 1 if clause i is satisfied by assignment x and 0 otherwise. A clause is satisfied if at least one of the literals it contains is true. Since our random MAX-3-SAT instances are all satisfiable, we declared a MAX-3-SAT problem as solved as soon as a string x such that $f(x) = c$ was generated by the EA.

Calculations

$$fdc_n = \frac{\frac{3}{8}(f_n - \bar{f}_p) + \frac{1}{2}C_{FD}}{\sqrt{\frac{(f_n - \bar{f}_p)^2}{4} + \frac{1}{2}\sigma_F^2} \times \sqrt{\frac{1}{2} + \sigma_D^2}}. \quad (1)$$

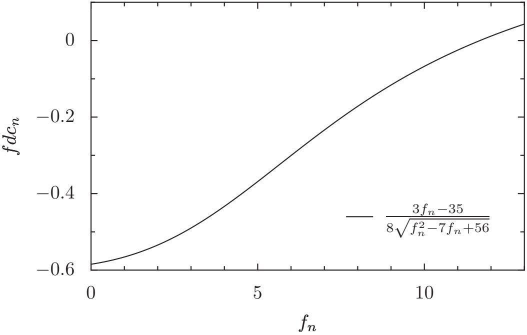


Figure : Fitness distance correlation in OneMax in the presence of constant neutrality as a function of f_n for $\ell = 14$.

fdc for OneMax Under Parity Bitwise Neutrality.

$$fdc = \frac{-\frac{1}{4}}{\sqrt{\frac{1}{4}}\sqrt{\frac{1}{4}}} = -1.$$

fdc for OneMax Under Truth Table Bitwise Neutrality.

$$fdc \approx -\frac{\frac{1}{4} + 2^{-n-2}}{\sqrt{\frac{1}{4} + 2^{-n}}\sqrt{\frac{1}{4}}} = -\frac{2^{(-n-1)} + \frac{1}{2}}{\sqrt{2^{-n} + \frac{1}{4}}}.$$

fdc for OneMax Under Majority Bitwise Neutrality.

$$fdc \approx -\frac{0.315 + 0.185\sqrt{n}}{\sqrt{0.133\,n - 0.117\,\sqrt{n} + 0.263}}.$$

Constant Neutrality

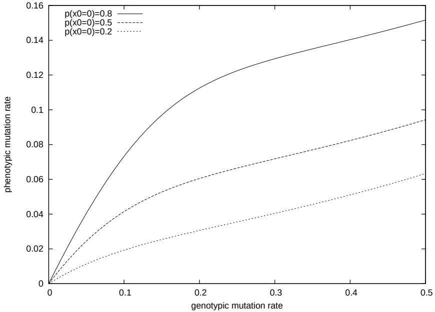


Figure : Effective phenotypic mutation rate for invertible fitness functions as a function of the genotypic mutation rate for strings of length $\ell = 14$ and different values of the selection probability for strings outside the neutral network induced by constant neutrality.

Bitwise Neutrality

Table : Phenotypic mutation rates corresponding to different genotypic mutation rates for different forms of bitwise neutrality.

Type of redundancy	$p_m = 0.01$	$p_m = 0.06$	$p_m = 0.1$
Parity (n bits $= 5$)	0.0480	0.2361	0.3362
Parity (n bits $= 6$)	0.0571	0.2678	0.3689
Parity (n bits $= 7$)	0.0659	**0.2957**	0.3951
Parity (n bits $= 8$)	0.0746	0.3202	0.4161
Truth Table (n bits $= 5$)	0.0245	0.1331	0.2048
Truth Table (n bits $= 6$)	0.0293	0.1551	0.2343
Truth Table (n bits $= 7$)	0.0340	0.1758	0.2609
Truth Table (n bits $= 8$)	0.0386	0.1952	**0.2848**
Majority ($n = 5$, $T = 2.5$)	0.0168	0.0916	0.1530
Majority ($n = 7$, $T = 3.5$)	0.0204	0.1072	0.1725

Parameters

Table : Parameters used for the experiments using constant and bitwise neutrality for the OneMax, Trap and multimodal problems.

Parameter	Value
Length of the genome	14
Population Size	80
Generations	100
Mutation Rate (per bit)	0.01, 0.06, 0.1
Generation gap	1
Independent Runs	1,000

Performance of a mutation-based EA on the OneMax problem using Constant Neutrality

	fdc	$p_m = 0.01$		$p_m = 0.06$		$p_m = 0.1$	
		Avr. Gen	% Suc.	Avr. Gen	% Suc.	Avr. Gen	% Suc.
No neutrality	-1.0	21.11	100.0	14.39	100.0	16.47	100.0
$f_n = 11$	-0.1645	38.12	63.0	29.60	99.0	31.81	99.1
$f_n = 12$	-0.0914	38.79	19.8	46.15	68.9	44.77	82.1
$f_n = 13$	-0.0396	27.47	2.0	48.63	12.3	43.14	13.1

Performance of a mutation-based EA on the Trap function with and without constant neutrality.

	fdc	$p_m = 0.01$		$p_m = 0.06$		$p_m = 0.1$	
		Avr. Gen	% Suc.	Avr. Gen	% Suc.	Avr. Gen	% Suc.
No neutrality	1.0	1	0.6	1	0.6	1	0.6
$f_n = 30$	0.5909	-	0.0	38.37	1.6	39.91	3.6
$f_n = 38$	0.4908	28.66	0.3	44.34	2.9	51.33	5.3

Fitness distance correlation in MAX-3-SAT problems of increasing complexity.

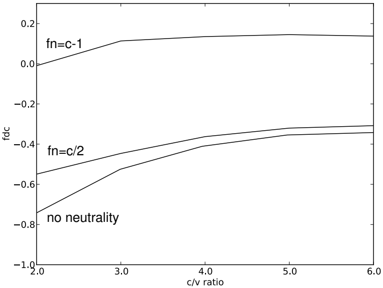


Figure : fdc with two different values of f_n: $f_n = c/2$ where c is the number of clauses and $f_n = c - 1$ (c is also the fitness of the global optimum in our SAT problems). The data are averages over 100 random satisfiable 3-SAT instances.

Plots of success rate of a mutation-based EA on MAX-3-SAT problems.

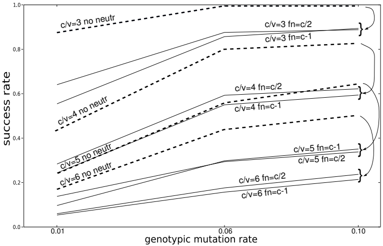


Figure : MAX-3-SAT problems with 14 variables as a function of the problem difficulty, the genotypic mutation rate and the fitness of the neutral network induced by constant neutrality (solid lines). The correspondence between these and success rates in the absence of neutrality (dashed lines) is indicated by the curved arrows on the right.

fdc estimated for the OneMax problem, the Multimodal Problem generator and the Trap function

Type of redundancy	OneMax Problem	Multimodal Problem	Trap Function
No neutrality	-1	0.5114	1
Parity ($n = 5$)	-1	0.5190	0.9925
Parity ($n = 6$)	-1	0.5190	0.9999
Parity ($n = 7$)	-1	0.5144	0.9999
Parity ($n = 8$)	-1	0.5086	0.9999
Truth Table ($n = 5$)	-0.9999	0.5102	0.9999
Truth Table ($n = 6$)	-1	0.5374	0.9925
Truth Table ($n = 7$)	-1	0.5264	0.9999
Truth Table ($n = 8$)	-0.9999	0.5233	0.9925
Majority ($n = 5, T = 2.5$)	-0.8488	0.4444	0.8434
Majority ($n = 7, T = 3.5$)	-0.8308	0.4471	0.8308

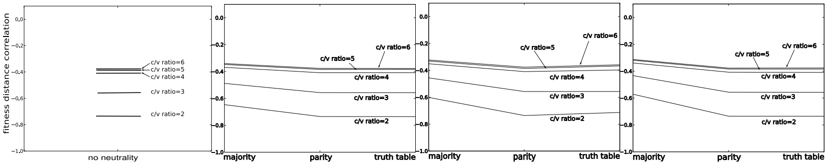
Performance of a mutation-based EA on the OneMax problem

Table : Pairs of numbers in **boldface**, underlined, doubly underlined and underlined with a wavy line represent situations with almost identical phenotypic mutation rates.

Type of redundancy	$p_m = 0.01$ Avr. Gen	$p_m = 0.01$ % Suc.	$p_m = 0.06$ Avr. Gen	$p_m = 0.06$ % Suc.	$p_m = 0.1$ Avr. Gen	$p_m = 0.1$ % Suc.
No neutrality	21.35	100	14.39	100	16.58	100
Parity ($n = 5$)	14.55	100	36.06	$\underline{\underline{90.1}}$	44.02	62.7
Parity ($n = 6$)	14.46	100	38.38	82.6	45.14	54.4
Parity ($n = 7$)	14.49	100	40.09	**73.3**	42.12	49.7
Parity ($n = 8$)	15.06	100	43.26	68.2	44.56	47.6
Truth Table ($n = 5$)	16.63	99.9	20.02	99.5	29.21	$\underline{95.0}$
Truth Table ($n = 6$)	16.89	100	22.87	99.4	33.14	$\underline{\underline{90.5}}$
Truth Table ($n = 7$)	15.89	100	24.41	97.5	35.49	84.5
Truth Table ($n = 8$)	15.01	100	28.16	$\underline{97.4}$	38.89	**78.8**
Majority (n=5, T=2.5)	23.39	99.8	17.26	99.7	22.08	99.3
Majority (n=7, T=3.5)	23.51	99.8	17.93	100	22.50	98.6

Fitness distance correlation estimated for MAX-SAT problems of various degrees of difficulty

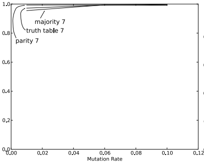
Encoding			
no neutrality	3 bits	5 bits	7 bits

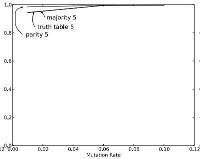


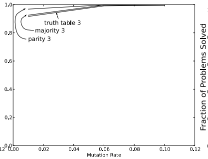
Success rates for MAX-SAT problems of varying difficulty

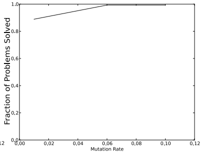
	Encoding			
c/v	no neutrality	3 bits	5 bits	7 bits

3

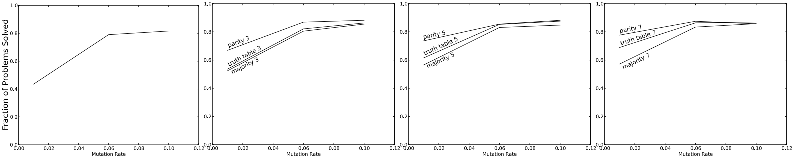




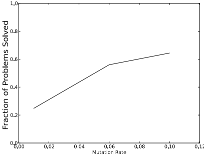
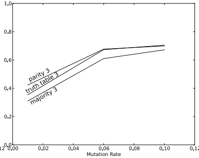
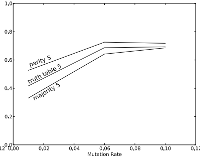
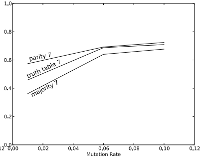




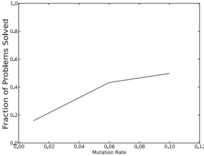
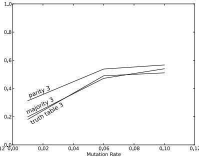
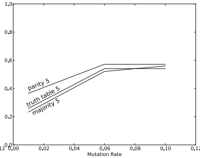
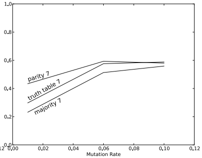
Success rates for MAX-SAT problems of varying difficulty

	Encoding			
c/v	no neutrality	3 bits	5 bits	7 bits
4				

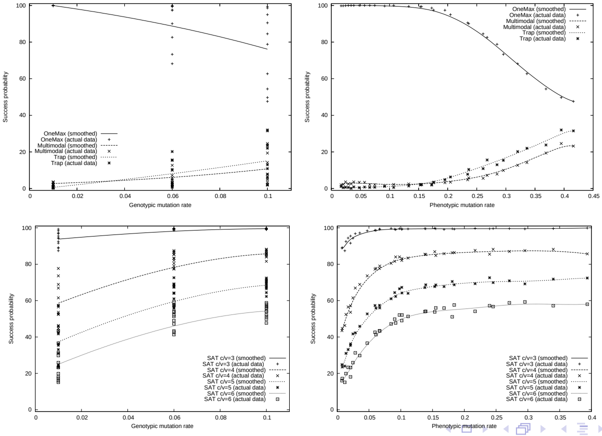
Success rates for MAX-SAT problems of varying difficulty

	Encoding			
c/v	no neutrality	3 bits	5 bits	7 bits
5				

Success rates for MAX-SAT problems of varying difficulty

	Encoding			
c/v	no neutrality	3 bits	5 bits	7 bits
6				

Plots of the success probability as a function of the genotypic mutation rate and the phenotypic mutation rate



.

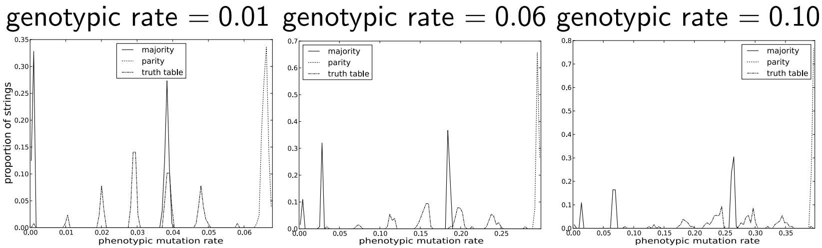


Figure : Distribution of phenotypic mutation rates with 7-bit bitwise neutrality for different encodings and genotypic mutation rates.

Performance of an EA on the Trap function

Table : Pairs of numbers in **boldface**, underlined, doubly underlined and underlined with a wavy line represent situations with almost identical phenotypic mutation rates.

Type of redundancy	$p_m = 0.01$ Avr. Gen	$p_m = 0.01$ % Suc.	$p_m = 0.06$ Avr. Gen	$p_m = 0.06$ % Suc.	$p_m = 0.1$ Avr. Gen	$p_m = 0.1$ % Suc.
No neutrality	0.6	0.3	7.2	0.7	4.55	0.7
Parity ($n = 5$)	1	0.5	47.77	10.4	44.85	22.0
Parity ($n = 6$)	1	0.8	45.96	15.6	44.73	23.8
Parity ($n = 7$)	1	0.6	48.62	**15.4**	46.82	32.0
Parity ($n = 8$)	13.57	0.7	46.27	20.2	46.69	31.5
Truth Table ($n = 5$)	1	0.7	13.05	1.4	41.49	6.3
Truth Table ($n = 6$)	1.25	0.6	35.16	2.1	47.19	7.8
Truth Table ($n = 7$)	1	0.1	32.36	3.5	47.32	10.9
Truth Table ($n = 8$)	1	0.9	34.44	4.8	58.54	**13.0**
Majority (n=5, T=2.5)	1	1.1	4.4	1.2	19.91	2.3
Majority (n=7, T=3.5)	1	0.5	1.16	0.6	28.15	1.9

Performance of an EA on the Multimodal function

Table : Pairs of numbers in **boldface**, underlined, doubly underlined and underlined with a wavy line represent situations with almost identical phenotypic mutation rates.

Type of redundancy	$p_m = 0.01$ Avr. Gen	% Suc.	$p_m = 0.06$ Avr. Gen	% Suc.	$p_m = 0.1$ Avr. Gen	% Suc.
No neutrality	8.56	3.2	5.22	2.7	11.54	1.9
Parity ($n = 5$)	5.61	3.4	41.2	5.8	44.07	14.2
Parity ($n = 6$)	4.76	3.4	45.27	7.2	50.41	19.4
Parity ($n = 7$)	2.80	2.1	44.41	**9.9**	46.31	24.6
Parity ($n = 8$)	4.85	2.1	42.14	12.7	46.94	23.2
Truth Table ($n = 5$)	6.41	3.6	15.86	2.5	34.11	3.5
Truth Table ($n = 6$)	8.18	2.5	20.27	2.2	34.32	4.8
Truth Table ($n = 7$)	6.59	2.6	24.07	3.1	44.44	5.6
Truth Table ($n = 8$)	4.95	3.6	19.10	3.2	33.03	**7.9**
Majority (n=5, T=2.5)	11.41	2.0	23.6	1.4	15.62	1.9
Majority (n=7, T=3.5)	9.76	2.3	9.44	2.2	25.42	2.4

Simplest definition

- We have used two problems to analyse neutrality. The first one is the OneMax problem. Naturally this problem has only one global optima in 111...1, and the landscape is unimodal.
- The second problem has two optima: a global optimum at position 111...1 and a local optimum at position 000...0. The global optimum is given a fitness n while the local optimum has fitness n-1. The remaining points in the landscape are assigned values that decrease with the distance from one of the optima.
- In our experiments we use the chromosomes of length $l = 8$ and $n = 40$.

Table : Parameters used for the experiments using constant and bitwise neutrality.

Parameter	Value
Length of the genome	14
Population Size	80
Generations	100
Mutation Rate (per bit)	0.01, 0.06, 0.1
Number of $extra - bits$ (Constant Neutrality)	1
Independent Runs	1,000

The OneMax problem

Table : 2. Average number of generations required to reach the optimal solution for the OneMax problem.

Population size	Without neutral layer	Value on neutral 7	Value on neutral 5
20	9.6	111.3	17
40	6.5	101.2	11.6
60	5.4	82.3	8.2
80	4.5	64.6	7.5
100	3.5	50.5	6.7

The Deceptive problem

Table : 3. Percentage of runs that reached the optimal solution for the Deceptive problem. Random initialisation.

Population Size	Without neutral layer	Value on neutral 38	Value on neutral 23
20	61%	42%	51%
40	78%	56%	60%
60	81%	67%	72%
80	85%	81%	75%
100	93%	94%	84%

Table : 4. Percentage of runs that were able to reach the optimal solution for the Deceptive problem. Fixed initialisation.

Population Size	Without neutral layer	Value on neutral 38	Value on neutral 23
20	7%	26%	2%
40	9%	48%	6%
60	17%	68%	12%
80	17%	74%	21%
100	31%	86%	23%

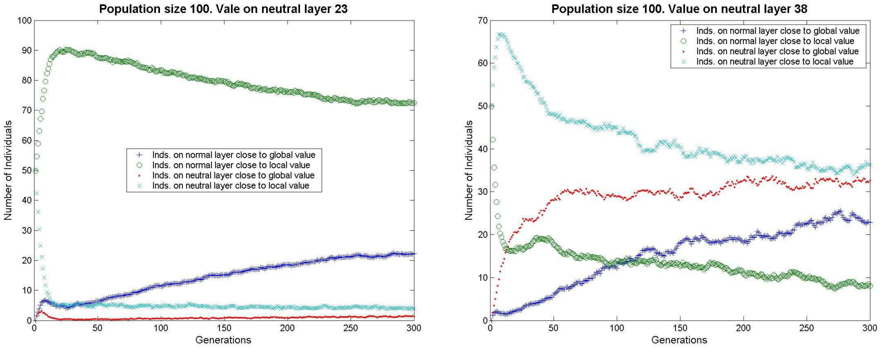


Figure : 1. Number of individuals situated in one of the four parts of the landscape. Fixed initialisation. Fitness of neutral layer 23 (left) and 38 (right).

More on Neutrality

Details of this presentation can be found in [2]. Other revelant readings include the following [1, 3, 4, 5, 6, 9, 10, 7, 8, 11].

Bibliography I

E. Galvan-Lopez.
Efficient graph-based genetic programming representation with multiple outputs.
International Journal of Automation and Computing, 5(1):81–89, 2008.

E. Galván-López.
An Analysis of the Effects of Neutrality on Problem Hardness for Evolutionary Algorithms.
PhD thesis, School of Computer Science and Electronic Engineering, University of Essex, United Kingdom, 2009.

E. Galván-López, S. Dignum, and R. Poli.
The effects of constant neutrality on performance and problem hardness in gp.
In M. O'Neill, L. Vanneschi, S. Gustafson, A. Esparcia-Alcázar, I. D. Falco, A. D. Cioppa, and E. Tarantino, editors, EuroGP, volume 4971 of Lecture Notes in Computer Science, pages 312–324. Springer, 2008.

E. Galván-López, J. McDermott, M. O'Neill, and A. Brabazon.
Defining locality in genetic programming to predict performance.
In Evolutionary Computation (CEC), 2010 IEEE Congress on, pages 1–8, 2010.

E. Galván-López and R. Poli.
An empirical investigation of how and why neutrality affects evolutionary search.
In M. Cattolico, editor, GECCO, pages 1149–1156. ACM, 2006.

Bibliography II

E. Galván-López and R. Poli.
How and why a bit-wise neutrality with and without locality affects evolutionary search.
In H. Lipson, editor, GECCO, page 1508. ACM, 2007.

E. Galván-López and R. Poli.
An empirical investigation of how degree neutrality affects gp search.
In A. H. Aguirre, R. M. Borja, and C. A. R. García, editors, MICAI, volume 5845 of Lecture Notes in Computer Science, pages 728–739. Springer, 2009.

E. Galván-López, R. Poli, A. Kattan, M. O'Neill, and A. Brabazon.
Neutrality in evolutionary algorithms... what do we know?
Evolving Systems, 2(3):145–163, 2011.

E. Galván-López and K. Rodríguez-Vázquez.
The importance of neutral mutations in gp.
In T. P. Runarsson, H.-G. Beyer, E. K. Burke, J. J. M. Guervós, L. D. Whitley, and X. Yao, editors, PPSN, pages 870–879, 2006.

E. Galván-López, J. Togelius, and S. M. Lucas.
Towards understanding the effects of neutrality on the sudoku problem.
In H. Lipson, editor, GECCO, page 1509. ACM, 2007.

Bibliography III

R. Poli and E. Galván-López.
On the effects of bit-wise neutrality on fitness distance correlation, phenotypic mutation rates and problem hardness.
In FOGA, pages 138–164, 2007.