“Royal-Road” Landscapes for a Dual Genetic Algorithm

Philippe Collard and **Alessio Gaspar** [1]

Abstract. Genetic Algorithms may be viewed as an optimization method based on an *Artificial Evolution* which is at the center of most works in *Artificial Life*. Since they were introduced by J. Holland during the sixties, their weaknesses and effectiveness have been intensively studied in two ways; by designing “improved GA” such as the Dual Genetic Algorithms described later, or by studying features of functions particularly easy or hard for a GA to optimize.

This paper will first introduce main concepts concerning both *Simple Genetic Algorithms* (SGA), and *Dual Genetic Algorithms* (DGA). After briefly presenting the *Royal Road Functions* originally designed by S. Forrest, M. Mitchell and J. Holland, we extend them in order to build a set of new functions so-called *Dual Royal Road Functions* (DRR). While the former were based on traditional Building Blocks, the latter use what we call *Relational-Positional Schemata*. Both of them will be used in the second part of this paper to establish an empirical comparison between SGA and DGA.

Finally, we discuss on the relevance of the Dual approach and its implications over the Genetic Algorithms perspectives of evolution.

1 Introduction

This introduction features a brief overview of both traditional Genetic Algorithms (referred in the following as *SGA*) and Dual ones (*DGA*). Refer to [12] or [24] for more precisions on this field.

1.1 Simple Genetic Algorithms

Genetic Algorithms use a kind of artificial Neo-Darwinian evolution applied through successive populations (i.e. Generations) of individuals each coding in their(s) chromosome(s) a potential solution. The application of Genetic Operators such as Mutation or Crossover to the population and the use of a Selection mechanism based on the survival of the fittest (inspired from natural selection) leads to a successful evolution of more and more adapted solutions to the problem considered. This leads to an homogeneous population only composed of the best Individuals found so far. We repeat until we met a convergence criterion the process described by figure 1; starting with a random set of Chromosomes (bit strings), we apply three genetic operators in order to obtain the next population. While next population isn’t fulfilled, we first choose by a *selection* mechanism two parents in the current population. According to a specified *crossover* rate we recombinate their chromosomes and then *Mutate* sons according to a mutation rate. Resulting strings are incorporated in the new population.

As expected by J. Holland ([15]), selective pressure and recombination of genetic material among individuals both provided a robust

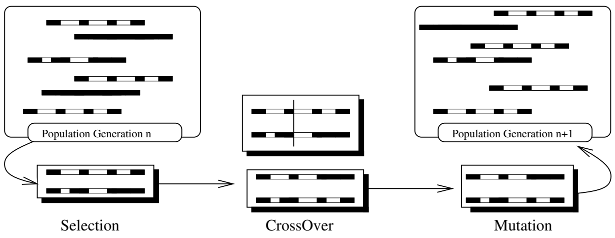

Figure 1. Standard Genetic Algorithm

and adaptive evolutionary process. GAs are able to handle function optimization tasks as well as computer programs design ([16]), or even Neural Network Morphogenesis ([14], etc),. . .

One of the first explanation of GA’s effectiveness is the so-called *Schemata Theorem* which has become a fundamental statement. Willing to understand GA has lead people to study features of difficult functions, that way *Deceptivity* has been early found to be one of the fundamental GA’s problem.[2]

Thus, analyzing properties of fitness Landscapes has become a promising gap; this paper focuses on the design of such particular fitness functions, on which we’ll compare both SGA and DGA.

1.2 Dual Genetic Algorithms

The DGA offers an improved robustness against most intrinsic weaknesses of GA ([5]). Levenick ([17]) has underlined the opportunity of adding *introns* (i.e. non expressives genes) into a chromosome; the DGA goes further and use a *Meta-gene* inspired from the transcription phase in Natural Genetic : “Gene Editing”. This single Meta-Gene alters the phenotype of the overall chromosome; if Meta bit is active (’1’) we translate normal genes by application of the *not* function, otherwise we keep original value. Consequently, populations may contain complementary[3] individuals called *Sosies* coding for the same expressed phenotype while having fundamentally different Genotypes. According to the fact that fully deceptive functions (the hardest ones) are known (cf Whitley) to have a local optima which Genotype is the complement of the global optima’s one, the DGA directly defers the competition between the local and global optima to the end of the exploration process simply by assuming that complementary strings are phenotypically equivalent in a symmetric doubled (and so dualised) search space. This brings, for example, an improved robustness against deceptivity which has lead to successful applications of DGAs to a wide area of problems such as traditional

[1] University of Nice Sophia Antipolis, Laboratory I3S-CNRS, bat. 4, 250, Ave. Albert Einstein 06560, Valbonne, France
e-mail:{pc,gaspar}@unice.fr
WWW: http://wwwi3s.unice.fr/mage.html

[2] Deceptive functions feature a local optima, which systematically (for fully deceptives) misleads the GA.

[3] Each of those strings is the complement of the other bit per bit, under Meta-Gene influence, they obviously express identical phenotypes.

ECAI 96. *12th European Conference on Artificial Intelligence*
Edited by W. Wahlster
Published in 1996 by John Wiley & Sons, Ltd.

optimization ([5]), Genetic Programming ([7]), Classifiers Systems ([6], [8]),. . .

We now consider the Royal Road Functions in order to compare SGAs and DGAs over a common adapted Testbed.

2 Royal Road Functions

After defining what are the original Royal Road Functions, we present several well-founded experimental results to establish a comparison between both previously defined Genetic Algorithms. We'll measure their effectiveness to find one occurrence of the solution string within a minimal amount of generations as done in original papers from Forrest and al.

2.1 An historical overview

Royal Road Functions have been originally designed by M. Mitchell, J. Holland and S. Forrest in order to become a set of easy problems for Genetic Algorithms; they revealed in fact to be very hard and interesting problems.

These Functions reward in each chromosome presence of user-specified building blocks. GA implicitly use Building Blocks during optimization, by specifying them, authors attempted to offer a "royal road" for the GA's convergence. Despite of these efforts GAs were still confronted with difficulties. As we'll see hereafter, several versions have been designed, but results became even more astonishing as the GA kept encountering problems despite of the increasing number of hand-defined building blocks included.

As previously exposed, DGAs are particularly well suited to solve GA-Hard problems, in order to establish a quantification of their advantages or disadvantages versus SGAs, we then define a new Testbed based on classical RR and introduce Duality in it.

2.2 Structural Definitions of RRs

We summarize the RR's definitions, extracted from [19], [20] and [9], used in the following experiments.

- Royal Road Function 1:
  The first royal road function is composed, as described in figure 1 of a set of eight basic buildings blocks rewarding presence

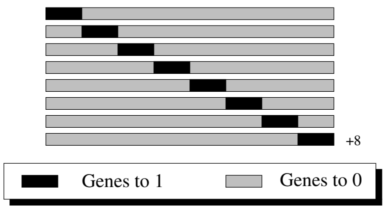

Figure 2. Royal Road 1 Decomposition

  of '1' on 8 consecutive genes among the 64 of the chromosome. Thus, optimal Chromosome is only composed of '1' genes, and more generally, for all described functions, the optimal string(s) contains all presented schemata and has a fitness summing all rewards (here 64 points).
- Royal Road Function 2:
  This second function introduces new building blocks rewarding presence of consecutive basic RR1's building blocks. Figure 2 presents this combination of RR1's blocks with new RR2 specific ones. The optimal String is the same as previous but is rewarded

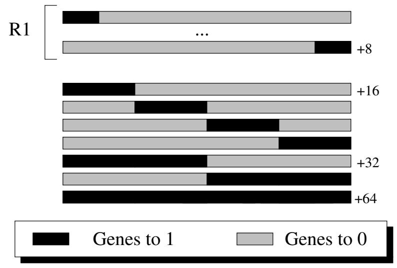

Figure 3. Royal Road 2 Decomposition.

  by 256 fitness points.

Now we'll describe our own model of Royal Road aimed at taking into account the extended expressiveness of schemata processed by Dual Genetic Algorithms.

3 Dual Royal Road Functions

This section is devoted to a description of our own definitions of extended Royal Road. In order to justify our choices, we first recall the structure of the new Schemata handled by DGA (named *R-Schemata*).

3.1 R-Schemata and RP-Schemata within GAs

Our paper will not present a complete analysis of a DGA's properties, such a work has already been exposed in previous papers, we only focus on a particular consequence of the meta approach : improvement of the Phenotype's expressiveness.

While RR functions are based on traditional Schemata, call them *P-Schemata*[4], DRR will use R-Schemata and RP-Schemata as presented bellow. First, let us consider the figure 4. We may study through

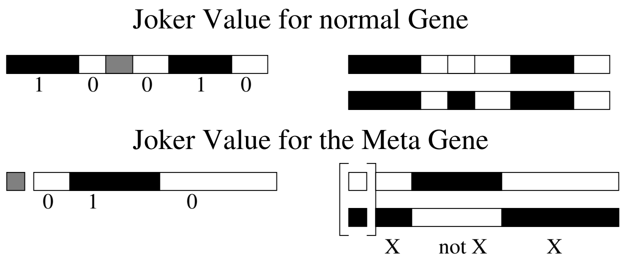

Figure 4. Introduction of variables in R-Schemata

this simplified representation the consequences of the instantiation of meta-genes in traditional P-Schemata. A GA explicitly manipulates, during its search, *n* individuals but implicitly handles a cubic amount of Schemata. This property, known as *Implicit Parallelism* of Genetic Algorithms, allows them to handle *Buildings Blocks* that correspond to particular features (Loci's values) in individuals. Wild spread ones are those appearing in most of best individuals, spreading Building

[4] Underlining the fact that they handle only positional instead of relational information about loci's values.

Blocks results in a competition that is a good interpretation of a Genetic Exploration of search space.

As shown in the figure, schemata including instanciated meta-genes don't offer interesting properties, but non-instanciated meta-genes introduce *relational* dependencies between loci; each gene value can be decoded normally or as its complement. So far, [1 0] leads to [1 0] or [0 1]. In this particular case we can obtain two different expressed phenotypes from the same P-Schemata [# 1 0]. The underlying information may be expressed by a R-Schemata [$X\ \overline{X}$] defined over an extended alphabet including a boolean variable X and its complement $\overline{X}$.

We naturally define a third kind of Schema, aimed to combine within a unique structure, expressed over the alphabet $\{0, 1, \#, X, \overline{X}\}$, both P-Schemata and R-Schemata. We'll refer to them in the following as *RP-Schemata* and use them in the definition of the Dual Royal Road Functions.

We can notice that building blocks represented as R-Schemata behave the same as standard ones. Thus, with a simple redefinition of the concepts of *order* and *definition length* we can directly apply the schema theorem on those new schemata. At first, the schema's definition length remains unchanged when passing from P-Schemata to R-Schemata or RP-Schemata. In the latter we simply consider variables (X and $\overline{X}$) as instancied genes and thus measure the definition length as the longest chromosome segment delimited by non-joker characters. The order of a R-Schema becomes the number of variables it contains decremented by one. This new definition remains consistent with the standard one as a $\mathcal{O}$ order schema can be relaxed into $2^{\lambda-\mathcal{O}}$ instances. more generally, concerning RP-Schemata, the order is thus expressed by the number of variables and instancied genes decremented by one if variables are used. According to these modifications, low order R-Schemata having a small definition length and over average fitted will be exponentially reproduced through successive generations conformly with the building block hypothesis.

3.2 DRR and Relational Building Blocks.

In a Machine Learning context ([6]), R-Schemata and RP-Schemata has featured interesting results, we tried naturally to apply them to optimization tasks by adding to Royal Road Functions the concept of *Relational Building-Blocks* thus allowing us to reward relationnal interactions between loci. In the following, we'll refer to classical Royal Roads just as RR functions or even P-RR functions to emphasize the fact that they are only based on P-Schemata for describing their building blocks.

All functions next defined will be particular *Royal Road Functions* based on dualism. While the former will be called from now RR or P-RR, we'll name the *Dual Royal Roads* according to the type of their building blocks : P-Schemata, R-Schemata or RP-Schemata.

More formally we get :

- Relational Royal Road :
  Our first Dual Royal Road Function called *R-RR* emphasizes the reward for the following R-Schemata: This way it rewards the alternation of alternated 8-bits sequence of '1' and '0. We have two global optima starting respectively by '1' or '0'. In fact the alternation constraint is propagated all over the chromosome, and adding more building blocks will result in an over specification similar to the one of P-RR2 but that has not been relevant here. There are two optimal strings each featuring alterning 8 genes long sequences with '1' or '0' consecutive values and both having a 64 points fitness.

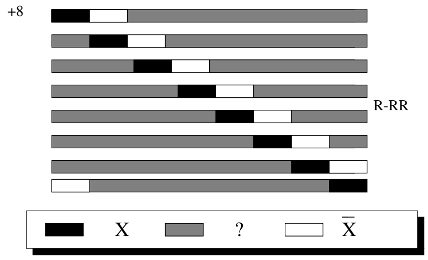

Figure 5. Relational Dual Royal Road Decomposition.

- Relational Positional Royal Road :
  In order to define our second test function, we use RP-Schemata to describe its building blocks. This definition is similar to the one

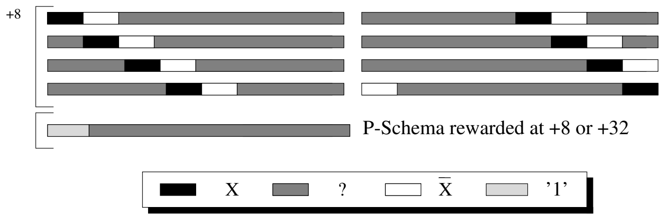

Figure 6. Relational Positional Royal Road Decomposition.

  of R-RR but it introduces an additional building block described by a P-Schema that breaks the function's symmetry by rewarding the presence of a '1' sequence at the beginning of the optimal string. By combining both kind of schemata we obtained a RP-RR function and deleted one of the prior global optima, there only remains the one beginning by 8 genes at '1' value for a fitness of 72.

4 Experiments

At first, this section provides data on experimental parameters settings and then details results obtained by both Simple and Dual Genetic Algorithms. Results presented in the following will allow us to establish a referential for further experiments in the field with respect to Holland's original definitions of P-RR.

4.1 Experimental Global conditions

All Experiments have been achieved using G.I.G.A.S., a Genetic Algorithm Experimentation Software, in respect of specifications published in the previously cited papers and summarized below.

- **Population Size** : 128
  This size has been changed to investigate opportunity of resizing when using a DGA in a few experiments.
- **String Length** : 64
  Though RR can be defined over variable string lengths we restrict to the original size.
- **Maximal number of iterations** : 2000
  Fixes a kind of time limit for optima discovery by GA.

- **Genetic Operators** : $\mu = 0.001$, $\tau = 0.05$ and $X_c = 0.7$
  Those settings are rather traditional and conform to the original paper on Royal Roads Functions except for the *Mirroring* operator's[5] application rate only settled for the use within DGA's experiments. Concerning crossover, standard 1-point crossover method is used.
- **Selection Scheme** : Stochastic sampling
  Our selection method replicates an intermediate population *P'* where individuals are cloned *x* times according to their fitness value (with *x* varying from 0 to 2) before to apply Genetic Operators on *P'* using random picking of parents in the population.
- **Fitness treatments** : Σ-Troncature, Linear Scaling
  We used Sigma troncature and linear scaling in all experiments.

All other non-specified parameters are assumed to be standard or uninteresting ones, each result presented in the following subsection is an average value computed over 500 experiments at least. We have, once again, taken these parameters from Forrest and al original RR's papers.

4.2 Positional Royal Road Results

Here are obtained results under the previously described conditions. Table 1 clearly features a SGA's superiority within the Classical Royal Roads Testbed, but a complete analysis of these results may take care of the following facts; first of all, we have not adjusted population size to the real size of the processed search space; DGAs double their search space and so may naturally need more individuals than SGAs to "cover" their search space. Meanwhile, presented results only give a light advantage for SGAs, this advantage can be discussed according to the previous argument and seems to remain constant when passing from RR1 to RR2. This advocates the fact that the difference doesn't come from a DGA's intrinsic weakness but rather from an insufficient population size. The supplementary amount of search space to sample remains constant in both case (RR1 and RR2) and may explain this difference in experimental results.

Table 1. Results on Royal Roads Functions

Functions	SGA	DGA
P-RR1	699	748
P-RR2	1579	1624

To establish this hypothesis we have performed some experiments with a population sized to 200 individuals for both GA. To ease comparisons we have grouped all obtained results in a single table where we have put number of individual evaluations rather than number of generations elapsed during genetic optimization[6].

The results, presented in table 2, clearly show that DGAs were lacking in a well-sized population in previous experiments. With 200 individuals their effectiveness, in comparison to SGAs using populations sized to 128 individuals is unchallenged even over P-RR which have not been especially designed to fit their particular capacities.

For adjusted population sizes, SGAs and DGAs become equivalent, and DGAs even gain a light superiority when applied to P-RR2 function although P-RR functions has been especially designed for SGAs.

Table 2. Results on Royal Roads Functions (Population 200).

Functions	Pop Size	SGA	DGA
P-RR1	128	89 472	95 744
P-RR1	200	78 400	84 000
P-RR2	128	202 112	207 872
P-RR2	200	251 400	248 600

This leads us to consider the fact that, in our search of an universal, adaptive, and robust optimization method, we must advocate that a DGA offers multiple almost interesting features despite of the need of increasing the population's search space to obtain comparable results. In order to measure the improvement induced by the particular feature of DGAs we now consider similar experiments concerning DRR.

4.3 Dual Royal Road Results.

Let us now consider the second part of our experimental work, table 3 presents results concerning the DGA-SGA comparison on the Dual Royal Road functions.

Table 3. Results on Dual Royal Roads Function.

Functions	SGA	DGA
R-RR	1641	1116
RP-RR	1659	979

As expected, DGA revealed an unchallenged superiority over SGA when applied to the DRR functions. The difference is even more significative than in experiments concerning classical Royal Roads, and as we have not increased population size during DGA's experiments, we should view those results as a lower bound for DGA capacities. A look at Table 3 advocates the hypothesis of DGA ability to deal with higher level building blocks that we may refer to as *R-Building Blocks* or *RP-Building Blocks* according to the nature of their representation as schemata.

For RP functions we have used two different rewards for the additional P-Schema; all results presented in this paper stand for a +8 reward since values 8 and 32 don't induce a relevant difference on results.

Thus, while DGAs can do as well as SGAs on a set of functions specially designed for traditional GAs, they show a significative superiority on Dual Functions. This clearly involve that Dual Functions represent a new perspective in function optimization; after deceptiveness, multimodality, royal road structure, our Dual Royal Road and more generally Dual Functions study appears as a promising gap in the area. Furthermore, they bring us an example of the intrinsic limitations of Positional Schemata implicitly processed by GAs.

4.4 Conclusions

Dual Genetic Algorithms are new, effective improved Genetic Algorithm that keep close enough to traditional GA to be formally studied while exhibiting new powerful features. This paper is devoted to an experimental analysis of their theoretical capacities, we based our work on RP-Schemata and defined an appropriate set of testbed functions. Our functions were built by analogy with the Royal Road ones

[5] Mirroring is a DGA's specific Genetic Operator that mute all genes in a Chromosome (including the heading Meta-Gene) according to a rate denoted by τ.

[6] This is simply accomplished by multiplying the average value of the halting generation by the population size.

to enable DGA's analysis in the same way that the latter did for traditional GA.

Results are encouraging; they confirmed DGA's superiority on *DRR* and even on *P-RR* Functions if we adjust their population size parameter. The third part of our experimentations even lead us to establish their superiority while combining traditional Schemata and specific R-Schemata within RP-Schemata.

This DGA's particular ability to handle DRR functions is fully based on the extension of the chromosome's expressiveness, such an extension has already lead to interesting results in Learning Classifiers Systems and these combined results suggest an intrinsic limitation of the concept of *Positional Schemata* on which SGAs rely. DGA's effectiveness hides a more powerful concept; *Relational* and *Relational Positional Schemata*. Both represent an improvement of chromosome's expressiveness as previously underlined. They allow the GA to be successfully applied to a new kind of functions (deceptive, temporal, and Dual Royal Road ones), and they even implies a new dynamics to the GA.

Such an article is aimed to begin an intensive study on the reason of their effectiveness. A few years ago understanding GA lead to DGA, we hope that understanding Dual GA will help us in defining a new kind of GA even more powerful.

The reason to introduce DGAs as representatives of a new generation of GAs can be found in comparing them with other device seeking for similar goals. A Brief overview of related work shows that many attempts to increase chromosome's expressiveness (which appear to be the key of all exhibited effectiveness) are based on an extension of its allelic alphabet. Unlike this, DGA keeps using a low level approach; it remains based upon easy-to-study (cf. [22], [23]) Binary chromosomes. GA using larger alphabets ([13]), Evolutionary Algorithms, or even Genetic Programming feature more expressive chromosomes, but none of these paradigms provides enough formal, theoretical foundations comparing to GA's Schema Theorem. All theoretical results on SGA are thus re-used in the Dual approach with only light adaptations, providing a strong theoretical background for their analysis.

For this reason, the Meta approach appears very promising. DGAs are not just an other "good device" but contain strong theoretical foundations. This paper tried to provide an experimental way to understand and quantify their behavior. Just as Royal Roads did and with similar methods, we hope to have provided sufficient matter to establish the basis for designing experiments that may lead to a better understanding of DGA's effectiveness.

5 Discussion and Future Works

A natural extension to both DGA and DRR should consist in adapting meta gene influence; rather than applying the meta bit to all genes it may be relevant to restrict it to a few selected loci only by using a kind of mask. This approach should be particularly promising when used with RP-Schemata, it may allow a GA to decide whenever it should use DGA mode to sample its search space and whenever relational variables become irrelevant.

That way, natural Biology, for which the *Genome Expression* remains an unresolved question, brings us new theories on meta genes ([3], [11], [4]) that may even allow us to apply *Dualism* to Genetic Algorithm Self Adaptation which is a particularly active area using meta bits to adapt crossover's parameters ([18], [21]), mutation ([1], [2]) or even in a more general Self Adaptive Genetic Algorithm perspective ([10], [18]).

Among all these perspectives, Dual Royal Roads are presented as a first experimental and quantitative look over DGA, they may be a first step toward a general Meta GA theory and open us new ways for Genetic optimization understanding.

REFERENCES

[1] T. Back, 'Self adaptation in genetic algorithms', *Toward a Practice of Autonomous Systems : ECAL'91*, (1991).

[2] T. Back, 'The interaction of mutation rate, selection, and self adaptation within a genetic algorithm', *Parallel Problem Solving From Nature*, 85–99, (1992).

[3] R. Benne and H. Van der Spek, 'L'editing des messages genetiques', *La Recherche*, **23**(245), 846–854, (July-August 1992).

[4] T. Chouard and M. Yaniv, 'Le controle de l'expression des genes', *La Recherche*, **25**, 626–635, (June 1994).

[5] P. Collard and J.P. Aurand, 'Dga: An efficient genetic algorithm', *ECAI'94: European Conference on Artificial Intelligence*, (1994).

[6] P. Collard and C. Escazut, 'Relational schemata: A way to improve expressiveness of classifiers.', *ICGA'95: Proceedings of the sixth International Conference on Genetic Algorihtms*, 397–404, (1995).

[7] P. Collard and J.L. Segapeli, 'Using a double based genetic algorithm on a population of computers program', *TAI'94: 6th IEEE International Conference on Tools for Artificial Intelligence*, (1994).

[8] C. Escazut, *Contribution aux systemes declassifieurs: etude des mecanismes de specialisation et extention du langage de representation*, Phd thesis, University of Nice Sophia Antipolis, May 1995.

[9] S. Forrest and M. Mitchell, 'Relative building-block fitness and the building-block hypothesis', *Foundations of Genetic Algorithms 2*, (1993).

[10] B. Freisleben and M. Hartfelder, 'Optimization of genetic algorithm by genetic algorithm', *Proceedings of the 1993 International Conference on Artificial Neural Nets and Genetic Algorithms*, 392–399, (1993).

[11] W.L. Gerlacj and C. Robaglia, 'Les ribosymes', *La Recherche*, **23**(247), 1144–1151, (October 1992).

[12] D. E. Goldberg, *Algorithmes Genetiques -Exploration, optimisation et apprentissage Automatique-*, Vie Artificielle, Addisson Wesley, 1994.

[13] D.E. Goldberg, 'Real-coded genetic algorithms, virtual alphabets, and blocking', *Complex Systems*, (5), 139–167, (1991).

[14] F. Gruau, *Synthese de Reseaux de Neurones par codage cellulaire et Algorithmes Genetiques.*, Phd thesis, ENS Lyon, 1994.

[15] J. H. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, 2nd edn., 1992.

[16] J. Koza, *Genetic Programming: On the Programming of Computers by the mean of Natural Selection*, MIT Press, 1992.

[17] J.R. Levenick, 'Inserting introns improves genetic algorithms success rate: Taking a cue from biologie', *Genetic Algorithms and their applications: Proceedings of the Fourth International Conference on Genetic Algorithms*, (1991).

[18] J.R. Levenick, 'Metabits: Generic endogenous crossover control.', *Proceedings of the sixth international conference on genetic algorithms*, 88–95, (1995).

[19] S. Forrest M. Mitchell and J. Holland, 'The royal road for genetic algorithms: Fitness landscapes and ga performance.', *Toward a Practice of Autonomous Systems : ECAL'91*, (1991).

[20] M. Mitchell and J. Holland, 'When will a genetic algorithm outperform hill-climbing', *Toward a Practice of Autonomous Systems : ECAL'91*, (1991).

[21] W.M. Spears, 'Adapting crossover in evolutionary algorithms'.

[22] M.D. Vose, 'Modeling simple genetic algorithms', *Fundations of Genetic Algorithms*, (1991).

[23] D. Whitley, 'An executable model of a simple genetic algorithm', *Foundations of Genetic Algorithms*, (1991).

[24] D. Whitley, 'A genetic algorithm tutorial', Technical Report CS-93-103, University of Colorado, (1993).