

# Reconsidering the Selection Concept of Genetic Algorithms from a Population Genetics Inspired Point of View

Michael Affenzeller, Stefan Wagner  
Institute of Systems Theory and Simulation  
Johannes Kepler University  
Altenbergerstrasse 69  
A-4040 Linz, Austria  
email: {ma,sw}@cast.uni-linz.ac.at

## Abstract

In this paper we propose some generic extensions to the general selection concept of a Genetic Algorithm (GA). These bionically inspired interrelated further developments aim to make the algorithm more open for scalability on the one hand, and to stabilize the performance of weaker crossover operators on the other hand without necessitating the development of new coding standards and operators for certain problems. Furthermore, the underlying Genetic Algorithm is unrestrictedly included in all of the newly proposed hybrid variants under especial settings.

## 1 Introduction

Optimization techniques derived from nature include Simulated Annealing (SA) which draws an analogy between the annealing of material to its lowest energetic state and an optimization problem or Evolutionary Algorithms (EAs) which are basically inspired by biological evolution. Further recent approaches like Ant-Colony Optimization (ACO) or Particle Swarm Optimization (PSE) are also mentionable in the context of bionically inspired optimization techniques and also agent theory is on the verge of achieving greater importance in the field of heuristic optimization.

Genetic Algorithms (GAs), possibly the most prevalent representative of Evolutionary Computation, were first presented by Holland [Holland, 1975]. Since that time GAs have been successfully applied to a wide range of problems including multimodal function optimization, machine learning, and the evolution of complex structures such as neural networks. An overview of GAs and their implementation in various fields is given by Goldberg [Goldberg, 1989] or Michalewicz [Michalewicz, 1996].

A very essential question about the general performance of a GA is, whether or not good parents are able to produce children of comparable or even better fitness (the building block hypothesis implicitly relies on this). In natural evolution, this is almost always true. For Genetic Algorithms this property is not so easy to guarantee. The disillusioning fact is that the

user has to take care of an appropriate coding in order to make this fundamental property hold.

In order to somehow overcome this strong requirement we try to get to the bottom of the success factors of the new selection concepts from a technical as well as from a population genetics inspired point of view and draw some essential interconnections.

The basic idea of the new selection model is to consider not only the fitness of the parents in order to produce a child for the ongoing evolutionary process. Additionally, the fitness value of the evenly produced child is compared with the fitness values of its own parents. The child is accepted as a candidate for the further evolutionary process if and only if the reproduction operator was able to produce a child that could outperform the fitness of its own parents. This strategy guarantees that evolution is presumed mainly with crossover results that were able to mix the properties of their parents in an advantageous way.

The experimental part of this section conventionally analyzes the characteristics of the new principles: As commonly done when evaluating the capability of heuristic techniques, the characteristic features are analyzed separately on the one hand. On the other hand the potential of the new methods is analyzed by comparing the achievable results with the results of other techniques on the basis of hard benchmark problems.

## 2 The Selection Concept of Population Genetics vs. the Selection Concept of Genetic Algorithms

Population genetics aims to describe the topology and temporal dynamics of genetic variation in natural populations with the goal to understand the evolutionary forces that act on populations.

Like Evolutionary Computation also population genetics has an empirical as well as a theoretical component and especially for scientists in the field of Evolutionary Computation it should be a very fruitful approach to consider the latest developments of population genetics which should be kept in mind as the bionic role-model for further developments. Therefore, in the following we summarize some up

to date considerations of population genetics which are relevant for the advanced GA selection models as stated in this paper.

### Selection in Population Genetics

Since Darwin [Darwin, 1859] selection is considered to be the most important (and only) evolutionary mechanism for adaptation to the environment. Population genetics considers the basic model of selection for hermaphroditic organisms in the following way:

random mating → selection → random mating → selection → .....

I.e. selection mainly depends on the probability of survival of newborn individuals until they reach pubescence which is called viability in the terminology of population genetics. Sexual selection which considers selection as selection for reproduction (as it is the approach in Genetic Algorithm theory) is rather considered as a side issue in population genetics. Our advanced selection scheme allowing self-adaptive steering of selection pressure as described in section 3.1 is therefore much more similar to the population genetics understanding of selection. In our new selection model the survival probability is determined by a comparison of the fitness of the newly generated individual with the fitness values of its parents. Indeed, as demonstrated in the experimental part, it appears that the first sexual selection step (roulette wheel, linear-rank, tournament) as in case of a standard GA does not drastically effect the qualitative or quantitative performance of the algorithm. Even with random sexual selection (corresponding to the basic model of the population genetic's selection model) the results were about the same or even better than with roulette wheel or linear-rank as the first selection step.

A very important consequence of selection in population genetics as well as in evolutionary computation is its influence on certain alleles. As a matter of principle there are four possibilities for each allele in the population :

- The allele may be fixed in the population.
- The allele may disappear in the population.
- The allele may converge to an equilibrium state.
- No change in the allele frequency.

For the purpose of technical optimization it is desirable that alleles which are part of a global optimal solution are fixed in the entire population and on the other hand it is also desirable that alleles representing genetic information that is definitely not part of a global optimal solution is sieved out in a rather early stage of evolutionary optimization.

### 2.1 Selection in Genetic Algorithms

In Genetic Algorithms a fitness function assigns a score to each individual in a population that indicates the 'quality' of the solution the individual represents.

The fitness function is often given as part of the problem description or based upon the objective function. Developing an appropriate fitness function can involve the use of simulation, heuristic techniques or the experience of an expert. Evaluation of the fitness function for each individual must be relatively fast because of the number of times it will be invoked. If the evaluation is likely to be slow then an approximate function evaluation technique or a technique that only considers elements that have changed, may be employed.

Once that a population has been generated and its fitness has been measured, the set of solutions that are to be 'mated' in a given generation is produced. In the Standard Genetic Algorithm (SGA) the probability that a chromosome in the current population is selected for reproduction is proportional to its fitness. Actually there are many ways of accomplishing this selection. These include roulette-wheel selection, linear-rank selection or tournament selection ([Michaliwicz, 1996], [Schoneburg *et al.*, 1994]).

However, all evenly mentioned GA-selection principles have one thing in common:

They all just consider the aspect of sexual selection, i.e. mechanisms of selection only come into play for the selection of parents for reproduction. The enhanced selection model which will be described in the following section defies this limitation by considering selection in a more general sense.

## 3 Enhanced Selection Models for Genetic Algorithms

The handling of selection pressure in the context of Genetic Algorithms mainly depends on the choice of the selection operator and the replacement scheme [Michaliwicz, 1996]. 'Generational replacement', for example, replaces the entire population by the next one, whereas 'elitism replacement' keeps the best individuals of the last generation and only replaces the rest and therefore usually performs faster. On the other hand, elitism likely causes too homogeneous populations, i.e. little population diversity, and therefore might cause unwanted premature convergence. Anyway, there exists no manageable model for controllable selection pressure handling within the theory of Genetic Algorithms [Schoneburg *et al.*, 1994].

Therefore, for the first extended selection model, we have introduced some kind of intermediate step (a 'virtual population') into selection which provides a handling of selection pressure very similar to that of Evolution Strategies (ES) [Affenzeller, 2001c], [Affenzeller, 2002]. As pointed out in the above mentioned publications, the most common replacement mechanisms can easily be implemented in this intermediate selection step. Furthermore, this Evolution Strategy like variable selective pressure is well-suited to steer the degree of population diversity on the one hand and, on the other hand, it acts as a basic model for the SEGA algorithm as being proposed in [Affenzeller, 2001a] or [Affenzeller, 2001b].

However, even if systematic steering of selection pressure becomes possible within this model there

is absolutely no self-adaptivity and it is a time consuming job even for a well experienced user to adapt the parameter settings according to a particular application, the problem instance and also to the characteristics of the available operators in order to achieve high quality results.

As a logical consequence in the further development of the first extended selection principles our main aim was to find concepts for self-adaptive selection pressure steering. The basic idea to create and evaluate a certain amount (at least population size) of offspring to be considered for becoming members of the next generation is adopted from our 'first' selection pressure steering model. Self-adaption comes into play when considering the question which amount of offspring has to be created at each round and which of these candidates are to be selected as members of the next generation, i.e. for the ongoing evolutionary process. In order to keep the concepts generic no problem specific information about the solution space is allowed to be used for stating the self-adaptive model. Thus, it is desirable to systematically utilize just the fitness information of the individuals of the actual generation for building up the next generation of individuals in order to keep the new concepts and methods generic and also to maintain the Markov property.

After having already borrowed some basic ideas from the theory of Evolution Strategies in the design phase of the first selection pressure steering model, we discovered some further useful hints in the theory of ES for integrating self-adaptation into the selection procedure of Genetic Algorithms. Consequently the basic concepts and ideas of our new self-adaptive selection pressure model, are built upon Rechenberg's success rule in a broader interpretation. Rechenberg's  $\frac{1}{5}$  success rule provides some heuristics (derived not mathematically, but experimentally) for controlling the amount of successful offspring:

*The ratio of successful mutations to all mutations should be  $\frac{1}{5}$ . If it is greater than  $\frac{1}{5}$ , increase the mutation variance; if it is less, decrease the variance (if search successful increase search in larger steps, otherwise in smaller) [Rechenberg, 1973].*

For GAs this means to adopt and further develop the self-adaptive concepts which are inherently included in this rule to the sexual recombination process of a GA (in case of ES the generation of new individuals is asexual as there is no crossover and new individuals are just generated by mutation) [Schoneburg *et al.*, 1994]. As being described and reasoned later on in this section, this strategy leads to a number of benefits especially for GAs even if the first consideration in this direction have been undertaken for ES.

Actually all modifications that are taken into account, use exactly the same operators for crossover and mutation as the corresponding Genetic Algorithm. As no further problem specific information is used, the new concepts are applicable to all problems Genetic Algorithms can be applied to.

### 3.1 An Enhanced Selection Model Introducing Self-Adaptive Steering of Selection Pressure

The first selection step chooses the parents for crossover either randomly or in the well-known way of Genetic Algorithms by roulette-wheel, linear-rank, or some kind of tournament selection strategy. After having performed crossover and mutation with the selected parents we introduce a further selection mechanism that considers the success of the apparently applied reproduction in order to assure the proceeding of genetic search mainly with successful offspring in that way that the used crossover and mutation operators were able to create a child that surpasses its parents' fitness. Therefore, a new parameter, called success ratio ( $SuccRatio \in [0, 1]$ ), is introduced. The success ratio gives the quotient of the next population members that have to be generated by successful mating in relation to the total population size. Our adaptation of Rechenberg's success rule for Genetic Algorithms says that a child is successful if its fitness is better than the fitness of its parents, whereby the meaning of 'better' has to be explained in more detail: is a child better than its parents, if it surpasses the fitness of the weaker, the better, or is it in fact some kind of mean value of both?

For this problem we have decided to introduce a cooling strategy similar to Simulated Annealing. Following the basic principle of Simulated Annealing we claim that an offspring only has to surpass the fitness value of the worse parent in order to be considered as 'successful' at the beginning and while evolution proceeds the child has to be better than a fitness value continuously increasing between the fitness of the weaker and the better parent. Like in the case of Simulated Annealing, this strategy effects a broader search at the beginning whereas at the end of the search process this operator acts in a more and more directed way. Having filled up the claimed ratio ( $SuccRatio$ ) of the next generation with successful individuals in the above meaning, the rest of the next generation  $((1 - SuccRatio) \cdot |POP|)$  is simply filled up with individuals randomly chosen from the pool of individuals that were also created by crossover but did not reach the success criterion. The actual selection pressure  $ActSelPress$  at the end of a single generation is defined by the quotient of individuals that had to be considered until the success ratio was reached and the number of individuals in the population in the following way:

$$ActSelPress = \frac{|POP_{i+1}| + |POOL|}{|POP|}$$

Figure 1 shows the operating sequence of the above described concepts. With an upper limit of selection pressure ( $MaxSelPress$ ) defining the maximum number of children considered for the next generation (as a multiple of the actual population size) that may be produced in order to fulfill the success ratio, this new model also functions as a precise detector of premature convergence:

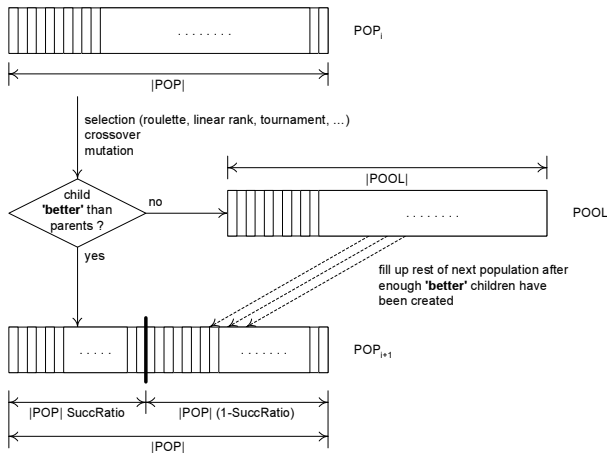


Figure 1: Flowchart for embedding the new selection principle into a Genetic Algorithm.

If it is no longer possible to find a sufficient number of  $(SuccRatio \cdot |POP|)$  offspring outperforming their own parents even if  $(MaxSelPress \cdot |POP|)$  candidates have been generated, premature convergence has occurred.

As a basic principle of this selection model a higher success ratio causes higher selection pressure. Nevertheless, higher settings of success ratio and therefore of selection pressure do not necessarily cause premature convergence. The reason for this is mainly, that the new selection step (after crossover) per definition does not accept clones that emanate from two identical parents. In conventional GAs such clones represent a major reason for premature convergence of the whole population around a suboptimal value (cf. take-over time), whereas the new selection step specifically counteracts against this phenomenon.

## 4 Experimental Results

The analysis of producible results considering various algorithms and benchmark test cases still denotes the most commonly used and possibly also the most objective way to analyze the potential of heuristic optimization techniques.

In our experiments, all computations are performed on a Pentium 4 PC with 1 GB RAM under Windows XP. The programs are written in the C# programming language using the Microsoft .NET framework 1.1. For the tests we have selected the Travelling Salesman Problem (TSP) as a well documented instance of a typical multimodal combinatorial optimization problem. We have tested the new concepts on a selection of symmetric TSP benchmark problem instances taken from the TSPLIB [Reinelt, 1991] using updated results<sup>1</sup> for the best or at least the best known solutions. In all experiments, the results are represented as the relative difference to the best known solution defined as  $relativeDifference = \left(\frac{Result}{Optimal} - 1\right) \cdot 100\%$ .

<sup>1</sup>Updates for the best (known) solutions can for example be found on <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>

We have selected the Travelling Salesman Problem (TSP) as our main test instance because of several reasons:

Firstly, there are a plenty of standardized benchmark test cases (of various degree of difficulty) available for this possibly most established instance of an NP-complete combinatorial optimization problem which allows to draw fair comparisons between different optimization techniques. Secondly, as the TSP is discussed for GA applications in great detail, there are a lot of GA representations and GA operators available for this problem instance which is of great interest especially for our considerations and analyses.

However, it is to be pointed out once again that the newly introduced methods are by no means restricted or somehow optimized to routing problems like the TSP. Further experimental tests on other combinatorial optimization problems like the JSSP (Job Shop Scheduling Problem), the MPSP (Multi Processor Scheduling Problem), or the TTP (Timetabling Problem) have already been performed and show the same effects and improvements as being described here for the TSP.

In order to do not dilute the effects of the different aspects too much, some selected aspects are pointed out separately in this section.

### Operator Improvement and Independence of Sexual Selection

The first aspect to be considered is the effect of the enhanced selection model to the quality improvement of different crossover operators. To visualize the positive effects of the new methods in a more obvious way we also present results that were generated by a classical GA with generational replacement and 1-elitism.

Remarkable in this context is the effect that also crossover operators that are considered as rather unsuitable for the TSP [Larranaga *et al.*, 1999] achieve quite good results in combination with the new selection model. The reason for this behavior is given by the fact that in our selection principle only children that have emerged as a good combination of their parents' attributes are considered for the further evolutionary process, if the success ratio is set to a higher range. In combination with a higher upper value for the maximum selection pressure genetic search can therefore be guided advantageously also for poor crossover operators as the larger amount of handicapped offspring is simply not considered for the further evolutionary process.

Additionally to the already mentioned aspect that the enhanced selection mechanism is able to improve the performance of the certain crossover operators it is furthermore observable that the new self-adaptive selection model makes the performance of the GA almost independent of sexual selection in terms of qualitative performance.

In Tab. 2, 3, and 4 the results achieved with the conventional GA using either roulette-wheel (Tab. 2), linear-rank (Tab. 3) respectively random, i.e. no, (Tab. 4) selection are listed. On the other hand Tab. 5,

Table 1: Parameter values used in the test runs of the several algorithms

Classical GA with several selection mechanisms (Tab. 2, Tab. 3, Tab. 4)		
Generations		100'000
Population Size		120
Elitism Rate		1
Mutation Rate		0.05
Selection Operator	Roulette, Linear Rank resp. Random	
Mutation Operator	Inversion & Translocation	
GA with self-adaptive selection in combination with several sexual selection mechanisms (Tab. 5, Tab. 6, Tab. 7)		
Population Size		500
Elitism Rate		1
Mutation Rate		0.1
Selection Operator	Roulette, Linear Rank resp. Random	
Mutation Operator	Inversion & Translocation	
Success Ratio		0.7
Maximum Selection Pressure		250

6, and 7 show the results achieved with the enhanced self-adaptive selection concept using either roulette-wheel (Tab. 5), linear-rank (Tab. 6), or no (Tab. 7) sexual selection.

The fixed parameter values for all algorithms that were used in the different test runs of the present subsection are given in Tab. 1. All values presented in the following tables are the best resp. average values of five independent test runs executed for each test case.

Table 2: Experimental results achieved with the classical GA using roulette-wheel selection

		Results with roulette-wheel selection		
Problem	Crossover	Best	Av:	Evaluated Solutions
berlin52	OX	0.00	3.76	12'000'000
berlin52	ERX	5.32	7.73	12'000'000
berlin52	MPX	21.74	26.52	12'000'000
ch130	OX	3.90	5.41	12'000'000
ch130	ERX	142.57	142.62	12'000'000
ch130	MPX	83.57	85.07	12'000'000
kroa200	OX	3.14	4.69	12'000'000
kroa200	ERX	325.92	336.19	12'000'000
kroa200	MPX	146.94	148.08	12'000'000

Similar improvements of solution quality are also observable when comparing the GA using linear-rank selection with the enhanced GA using linear-ranking as the first selection step. So far the results underpin the crossover improvement not only for roulette-wheel selection but also for linear-ranking.

What is really notable is the comparison of Tab. 4 and Tab. 7. Firstly, it is barely remarkable that a GA with no (i.e. random) sexual selection is unable of producing high-quality results. So the results of Tab. 4 are in the region of random search which is caused by the 1-elitism (the only goal-oriented force under these settings). What is really remarkable is that the results of the new GA with enhanced selection and no sexual selection (Tab. 7) are about the same than the results obtained with roulette-wheel respectively with linear-rank as the first selection step. This observation supports the theory of population genetics that sexual selection really plays a rather inferior role in the natural selection process.

Table 3: Experimental results achieved with the classical GA using linear-rank selection

		Results with linear-rank selection		
Problem	Crossover	Best	Av:	Evaluated Solutions
berlin52	OX	0.00	5.40	12'000'000
berlin52	ERX	2.52	4.58	12'000'000
berlin52	MPX	20.90	27.31	12'000'000
ch130	OX	5.60	8.88	12'000'000
ch130	ERX	99.18	128.47	12'000'000
ch130	MPX	85.78	97.46	12'000'000
kroa200	OX	8.58	12.24	12'000'000
kroa200	ERX	351.41	365.80	12'000'000
kroa200	MPX	144.25	150.34	12'000'000

Table 4: Experimental results achieved with the classical GA using no (random) sexual selection

		Results with random selection		
Problem	Crossover	Best	Av:	Evaluated Solutions
berlin52	OX	25.07	31.85	12'000'000
berlin52	ERX	80.54	89.96	12'000'000
berlin52	MPX	52.24	78.52	12'000'000
ch130	OX	148.54	161.77	12'000'000
ch130	ERX	397.46	406.94	12'000'000
ch130	MPX	252.59	286.18	12'000'000
kroa200	OX	296.22	309.71	12'000'000
kroa200	ERX	667.71	692.22	12'000'000
kroa200	MPX	420.76	464.49	12'000'000

Table 5: Experimental results achieved with the new selection principle using using roulette-wheel sexual selection

		Results with roulette-wheel selection		
Problem	Crossover	Best	Av:	Evaluated Solutions
berlin52	OX	0.00	3.88	15'964'680
berlin52	ERX	0.00	3.10	16'337'700
berlin52	MPX	0.00	1.45	11'775'071
berlin52	OX,ERX,MPX	0.00	0.72	7'204'601
ch130	OX	3.88	5.40	15'602'824
ch130	ERX	4.02	5.30	16'920'451
ch130	MPX	1.83	3.53	13'994'686
ch130	OX,ERX,MPX	2.50	4.40	7'702'818
kroa200	OX	2.25	5.72	10'814'980
kroa200	ERX	5.10	5.99	18'268'888
kroa200	MPX	5.21	7.65	12'296'581
kroa200	OX,ERX,MPX	2.78	4.61	6'647'256

Table 6: Experimental results achieved with the new selection principle using using linear-rank sexual selection

		Results with linear-rank selection		
Problem	Crossover	Best	Av:	Evaluated Solutions
berlin52	OX	2.29	4.94	7'448'762
berlin52	ERX	0.00	1.92	399'296
berlin52	MPX	0.00	3.92	8'199'592
berlin52	OX,ERX,MPX	0.00	1.59	60'920'006
ch130	OX	3.04	7.90	2'515'637
ch130	ERX	4.36	5.36	1'245'727
ch130	MPX	2.22	3.61	9'029'807
ch130	OX,ERX,MPX	2.16	2.80	61'759'481
kroa200	OX	8.14	9.30	2'011'929
kroa200	ERX	6.28	8.12	4'822'588
kroa200	MPX	5.63	6.37	8'527'427
kroa200	OX,ERX,MPX	1.75	2.79	57'493'081

## 5 Conclusion and Future Perspectives

Genetic Algorithms do not use any auxiliary information about the objective function such as derivatives. Therefore, they can be applied to any kind of discrete

Table 7: Experimental results achieved with the new selection principle using using no (random) sexual selection

Problem	Crossover	Results with random selection		
		Best	Av:	Evaluated Solutions
berlin52	OX	3.09	5.62	16'045'200
berlin52	ERX	0.00	1.35	16'938'904
berlin52	MPX	0.00	3.78	19'307'034
berlin52	OX,ERX,MPX	0.00	1.45	7'233'215
ch130	OX	2.24	4.59	15'281'043
ch130	ERX	2.27	5.20	18'840'038
ch130	MPX	3.60	4.77	23'164'733
ch130	OX,ERX,MPX	1.00	3.05	6'797'867
kroa200	OX	3.77	6.49	13'188'469
kroa200	ERX	118.61	121.78	28'406'603
kroa200	MPX	3.13	4.04	22'728'010
kroa200	OX,ERX,MPX	3.21	5.72	6'171'308

or also continuous optimization problem. The only thing to be done is to specify a meaningful representation and the corresponding operators. In order to do not restrict this essential property, the advanced GA concepts are designed in an absolutely generic non-problem specific way that allows to reuse all representations and the corresponding operators that have already been developed for certain GA-applications without any adaptation. The extended concepts are motivated by reconsidering recent results of population genetics that may be relevant for improving the performance of a GA in a non problem specific way. The proposed techniques couple aspects from Evolution Strategies (selection pressure and success rule) and Simulated Annealing (growing selection pressure). Furthermore, established crossover and mutation operators for certain problems may be used analogously to the corresponding Genetic Algorithm.

Anyway, under special settings the corresponding Genetic Algorithm is fully included within the introduced concepts achieving a performance only marginally worse than the performance of the equivalent Genetic Algorithm. In other words, the introduced models can be interpreted as a superstructure of the GA model or as a technique downwards compatible to Genetic Algorithms. Therefore, an implementation of the new algorithm(s) for a certain problem is quite easy to do, presumed that the corresponding Genetic Algorithm (coding, operators) is known.

Because of better comparability, no additional problem specific methods like commonly used hill-climbing or certain other pre- or post-optimization techniques have been considered in the examples presented in the experimental results section. However, there exists absolutely no objection against doing so in order to improve the solution quality for a certain problem.

As we have already shown exemplarily a main area of applications for the new Evolutionary Algorithm is given by problems that are already treated by Genetic Algorithms: Independent from the actually employed variant of a Genetic Algorithm it should be possible in almost all cases to implement the additional concepts of self-adaptive selection pressure quite easily.

## References

- [Affenzeller, 2001a] M. Affenzeller. A new approach to evolutionary computation: Segregative genetic algorithms (SEGA). *Connectionist Models of Neurons, Learning Processes, and Artificial Intelligence, Lecture Notes of Computer Science*, 2084:594–601, 2001.
- [Affenzeller, 2001b] M. Affenzeller. Segregative genetic algorithms (sega): A hybrid superstructure upwards compatible to genetic algorithms for retarding premature convergence. *International Journal of Computers, Systems and Signals (IJCSS)*, 2(1):18–32, 2001.
- [Affenzeller, 2001c] M. Affenzeller. Transferring the concept of selective pressure from evolutionary strategies to genetic algorithms. *Proceedings of the 14th International Conference on Systems Science*, 2:346–353, 2001.
- [Affenzeller, 2002] M. Affenzeller. New variants of genetic algorithms applied to problems of combinatorial optimization. *Proceedings of the EMCSR 2002*, 1:75–80, 2002.
- [Darwin, 1859] C. Darwin. *The Origin of Species By Means of Natural Selection or the Preservation of Favoured Races in the Struggle for Life*. Penguin Books London, 1859.
- [Goldberg, 1989] D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley Longman, 1989.
- [Holland, 1975] J.H. Holland. *Adaption in Natural and Artificial Systems*. University of Michigan Press, 1975.
- [Larranaga et al., 1999] P. Larranaga, C.M.H. Kuijpers, R.H. Murga, and S. Dizdarevic. Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artificial Intelligence Review*, 13(2):129–170, 1999.
- [Michaliwicz, 1996] Z. Michaliwicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag Berlin Heidelberg New York, 3. edition, 1996.
- [Rechenberg, 1973] I. Rechenberg. *Evolutionstrategie*. Friedrich Frommann Verlag, 1973.
- [Reinelt, 1991] G. Reinelt. TSPLIB - A traveling salesman problem library. *ORSA Journal on Computing* 3, 3:376–384, 1991.
- [Schoneburg et al., 1994] E. Schoneburg, F. Heinzmann, and S. Feddersen. *Genetische Algorithmen und Evolutionstrategien*. Addison-Wesley, 1994.