UNIVERSITY OF DORTMUND

REIHE COMPUTATIONAL INTELLIGENCE

COLLABORATIVE RESEARCH CENTER 531

Design and Management of Complex Technical Processes and Systems by means of Computational Intelligence Methods

The Ising Model on the Ring: Mutation versus Recombination

Simon Fischer and Ingo Wegener

No. CI-164/04

Technical Report ISSN 1433-3325 February 2004

Secretary of the SFB 531 · University of Dortmund · Dept. of Computer Science/XI
44221 Dortmund · Germany

This work is a product of the Collaborative Research Center 531, "Computational Intelligence," at the University of Dortmund and was printed with financial support of the Deutsche Forschungsgemeinschaft.

The Ising Model on the Ring: Mutation versus Recombination

Simon Fischer and Ingo Wegener*
FB Informatik, LS2, Univ. Dortmund, 44221 Dortmund, Germany
simon.fischer@uni-dortmund.de, ingo.wegener@uni-dortmund.de

Abstract

The investigation of genetic and evolutionary algorithms on Ising model problems gives much insight how these algorithms work as adaptation schemes. The Ising model on the ring has been considered as a typical example with a clear building block structure suited well for two-point crossover. It has been claimed that GAs based on recombination and appropriate diversity-preserving methods outperform by far EAs based only on mutation. Here, a rigorous analysis of the expected optimization time proves that mutation-based EAs are surprisingly effective. The $(1+\lambda)$ EA with an appropriate λ-value is almost as efficient as usual GAs. Moreover, it is proved that specialized GAs do even better and this holds for two-point crossover as well as for one-point crossover.

1 Introduction

Nowadays, genetic algorithms (GAs) and evolutionary algorithms (EAs) are mainly applied as optimization algorithms. Holland (1975) has designed GAs as adaptation systems. The building block hypothesis (see Goldberg (1989)) claims that GAs work by combining different building blocks in different individuals by crossover (or recombination). There is a long debate on the role of mutations in this context.

Naudts and Naudts (1998) have presented the Ising model as an interesting subject for the investigation of GAs and EAs. Ising (1925) has described the model now called Ising model to study the theory of ferromagnetism. In its most general form, the model consists of an undirected graph $G = (V, E)$ and a weight function $w\colon E \to \mathbb{R}$. Each vertex $i \in V$

*Supported in part by the Deutsche Forschungsgemeinschaft (DFG) as part of the Collaborative Research Center "Computational Intelligence" (SFB 531) and by the German Israeli Foundation (GIF) in the project "Robustness Aspects of Algorithms."

has a positive or negative spin $s_i \in \{-1, +1\}$. The contribution of the edge $e = \{i, j\}$ equals $f_s(e) := s_i \cdot s_j \cdot w(e)$. The fitness $f(s)$ of the state s equals the sum of all $f_s(e)$, $e \in E$, and has to be maximized. A GA or EA can be considered as a process adapting to better fitness or as an algorithm maximizing f. In this paper, we have chosen the language from optimization.

The Ising problem in its general form is NP-hard. Nevertheless, there are quite efficient algorithms for this problem (Pelikan and Goldberg (2003)). For the investigation of the adaptation capabilities of simple GAs and EAs, one is interested in the case where $w(e) = 1$ for all $e \in E$. By an affine transformation, we consider the state space $\{0, 1\}^n$ instead of $\{-1, +1\}^n$. The fitness $f(s)$ equals the number of monochromatic edges. As an optimization problem, the problem is trivial. The states 0^n and 1^n are optimal and they are the only optimal states for connected graphs. Connected monochromatic subgraphs are schemata of high fitness and, therefore, building blocks. However, the fitness function has the property of spin-flip symmetry, i. e., $f(s) = f(\overline{s})$ for all states s and their bitwise complement $\overline{s}$. Therefore, 0-colored building blocks compete with 1-colored building blocks. This property of fitness functions has also been discussed for other functions, e. g., the "hierarchically if and only if" function H-IFF (Watson (2001), Dietzfelbinger, van Hoyweghen, Naudts, and Wegener (2003)).

The Ising model on the ring is of particular interest. The ring is a graph on $V = \{1, \ldots, n\}$ with edges $\{i, i + 1\}$, $1 \le i \le n - 1$, between neighbored vertices and the turn-around edge $\{n, 1\}$. Building blocks are also blocks in the string (if the positions 1 and n are considered as neighbored) and two-point crossover can cut out a building block. Extensive experiments on GAs for this problem have been reported by van Hoyweghen (2002), van Hoyweghen, Goldberg, and Naudts (2002), and van Hoyweghen, Naudts, and Goldberg (2002). These papers contain also discussions how the algorithms work and some theoretical results but no run time analysis. In recent years, the rigorous run time analysis of EAs has led to interesting results. Most of this research is devoted to mutation-based algorithms (a good example is Droste, Jansen, and Wegener (2002)) but there are also results on steady-state GAs (Jansen and Wegener (2001, 2002)). Here, this approach is applied to the Ising model on the ring.

Sections 2, 3, 4, and 5 analyze mutation-based algorithms. Experiments have led to the conjecture that these algorithms are quite inefficient for the Ising model. The authors of the papers mentioned above do not explicitly state such a conjecture but they and many others have argued in discussions that mutation-based EAs will need exponential optimization time. In Section 2, we analyze randomized local search (RLS) flipping one bit per step and applying a plus-strategy for selection. This simple algorithm finds the optimum in an expected number of $O(n^3)$ steps and the constants in the O-term are surprisingly small. Based on this analysis, a similar bound is obtained in Section 3 for the (1+1) EA. There, the usual mutation operator is applied to create new search points. Hence, mutation-based EAs are much more efficient than conjectured.

Both, RLS and (1+1) EA, are sequential and produce one offspring per generation. In Sections 4 and 5, we analyze parallel variants of these algorithms, parallel RLS (PRLS) and the $(1+\lambda)$ EA, respectively. They produce λ offspring per generation and select a best individual. For $\lambda = n/\log n$, the expected optimization time consists of $O(n^2 \log n)$ generations and $O(n^3)$ fitness evaluations. This analysis follows the line of research started by Jansen and De Jong (2002) and Jansen, De Jong, and Wegener (2003). In Section 6, we compare our results with the experiments on GAs.

It would be even more interesting to obtain also bounds on the expected optimization time of GAs. We are not able to do this for the GAs used in experiments which apply an island model to preserve diversity. We analyze in Section 7 the GA introduced by Culberson (1992) and known as GIGA (Gene Invariant GA) and in Section 8 an idealized GA with fitness sharing. Both algorithms are tailored to cope with the given problem and perform better than RLS and the (1+1) EA. Their expected number of fitness evaluations is $O(n^2)$. We finish with some conclusions.

2 The Expected Run Time of Randomized Local Search

Randomized Local Search (RLS) chooses the first search point $x \in \{0,1\}^n$ uniformly at random. Afterwards, it chooses a position $i \in \{1,\dots,n\}$ uniformly at random, computes x' by flipping bit i of x, and replaces x by x' iff $f(x') \geq f(x)$. We are interested in the expected number of f-evaluations until $x \in \{0^n, 1^n\}$.

Instead of maximizing f, we investigate the equivalent problem of minimizing the number i of monochromatic blocks on the ring. This number is even for non-optimal points and has to be decreased from at most n to 1. For $2 \leq i \leq n$ and i even, let $t_i(n)$ be the expected time until i is decreased if we start with a worst search point with i blocks. We estimate the expected run time by the sum of all $t_i(n)$ and the term 1 for the initialization step.

By the pigeon-hole principle, there is one block whose length is bounded above by $N := \lfloor n/i \rfloor$. We investigate a shortest block B of the first search point x. A flip of a bit in the middle of a block is not accepted. If i is not decreased, the length of B can change at most by 1 per step. We distinguish relevant steps (either decreasing i or changing the length of B) from the other steps called non-relevant. First, we only investigate the relevant steps. It is possible that some block $B' \neq B$ gets shorter than B and vanishes earlier. Pessimistically, we ignore this. Only if B grows to length $N+1$ we switch our interest to another block whose length is at most N. Pessimistically, we assume that this length equals N. Then we obtain the following Markoff chain on $\{0,1,\dots,N\}$ where the state j describes the length of the considered block. Pessimistically, we start at N. If $j \in \{2,\dots,N-1\}$, by symmetry, $p(j,j-1) = p(j,j+1) = 1/2$. By the discussion above, "state $N+1$" is replaced by N and $p(N,N-1) = p(N,N) = 1/2$. State 1 is untypical, since there are two bits whose flip increases the block length but only one decreasing it. Hence, $p(1,0) = 1/3$ and $p(1,2) = 2/3$. We stop when reaching state 0. Let $T_N(j)$ be the expected time until reaching state 0 when starting in state j.

Lemma 2.1: *If $j \geq 1$, $T_N(j) = 4N - 1 + N(N-1) - (N-j+1)(N-j)$.*

Proof: We fix N and omit the index N. Obviously, $T(0) = 0$. We prove by induction that

$$T(j) = 2 \cdot (N - j + 1) + T(j-1),$$

if $j \geq 2$. By the law of total probability,

$$T(N) = 1 + (1/2) \cdot T(N) + (1/2) \cdot T(N-1)$$

implying that $T(N) = 2 + T(N-1)$. If $j < N$, by induction hypothesis,

$$\begin{aligned} T(j) &= 1 + (1/2) \cdot T(j+1) + (1/2) \cdot T(j-1) \\ &= 1 + (N-j) + (1/2) \cdot T(j) + (1/2) \cdot T(j-1). \end{aligned}$$

Solving for $T(j)$, this proves the claim. Finally,

$$\begin{aligned} T(1) &= 1 + (1/3) \cdot T(0) + (2/3) \cdot T(2) \\ &= 1 + (2/3) \cdot (2 \cdot (N-1) + T(1)) \end{aligned}$$

implying that $T(1) = 4N - 1$. This proves the lemma for $j = 1$ and, if $j \geq 2$,

$$T(j) = 2 \cdot (N-j+1) + 2 \cdot (N-j+2) + \cdots + 2 \cdot (N-1) + 4N - 1 \qquad (*)$$

which implies the lemma. □

For later purposes, we state that T_N is monotone increasing and concave, i. e.,

$$T_N(j+1) - T_N(j) \leq T_N(j) - T_N(j-1).$$

The last statement follows directly from equation $(*)$.

In order to estimate the expected number of relevant steps, it is sufficient to sum up all $T_{\lfloor n/i \rfloor}(\lfloor n/i \rfloor), i \in I := \{j \mid 2 \leq j \leq n, j \text{ even}\}$. Since $T_N(N) = N^2 + 3N - 1$, we obtain

$$\begin{aligned} \sum_{i \in I} T(\lfloor n/i \rfloor) &\leq n^2 \sum_{i \in I} (1/i^2) + 3n \sum_{i \in I} (1/i) - \lfloor n/2 \rfloor \\ &\leq 0.411 \cdot n^2 + 1.5 \cdot n \ln n + n. \end{aligned}$$

In the last step, we have used the following arguments. The sum of all $1/i$, $1 \leq i \leq n$ is bounded by $\ln n + 1$, the sum for the even i is less than half of this. The bound for the sum of all $1/i^2$, i even, has been obtained analytically.

In order to estimate the number of all steps, we are interested in the probability that a step is relevant. This is easy since we consider only one block B. There are 4 positions such that the length of B changes if one of the corresponding bits flips and the length of B is at least 2. If B has length 1, there are only 3 such positions. The expected waiting time until one of k bits flips is exactly n/k. In order to get good bounds, we estimate the expected number of relevant steps where the block length equals 1. Since the probability of reaching state 0 and finishing a phase equals $1/3$, the expected number of steps in state 1 equals 3 independent of i. Hence, $(3/2)n$ of the relevant steps have to be multiplied by $n/3$ and the other ones by $n/4$ to obtain an upper bound on the expected run time. Because of our rough estimate of the sum of all $1/i, i \in I$, we can omit the term 1 for the initialization and obtain the following result.

Theorem 2.2: *The expected number of steps until RLS finds an optimum for the Ising model on the ring is bounded above by*

$$T_{\mathrm{RLS}}(n) = 0.103 \cdot n^3 + 0.375 \cdot n^2 \cdot (\ln n + 1).$$

This bound is pessimistic in the following aspects:

- the first search point can have less than the maximal number of blocks,
- the first search point with i blocks can contain a block which is shorter than $\lfloor n/i \rfloor$,
- larger blocks can get shorter than the considered block.

In any case, the bound of Theorem 2.2 is surprisingly small when considering the discussions about this problem. Experiments have shown that, in the case $i = 2$, the shorter block has an average block length of $0.28n$ when reaching this phase. We can prove a lower bound on the expected run time only under a reasonable assumption.

Theorem 2.3: *Starting with two blocks of length εn and $(1-\varepsilon)n$, $0 < \varepsilon \leq 1/2$ a constant, the expected number of steps until RLS finds an optimum for the Ising model on the ring is $\Theta(n^3)$.*

Proof: Since there are only two blocks, our analysis for the upper bound considers always the shorter one. The expected number of relevant steps equals $T_{\lfloor n/2 \rfloor}(\varepsilon n) = \Theta(n^2)$ and has to be multiplied by $n/4$ to obtain the expected number of steps (for 3 steps we have to multiply by $n/3$). This leads to the expected number of $\Theta(n^3)$ steps. □

3 The Expected Run Time of the (1+1) EA

The (1+1) EA can be considered as the simplest evolutionary algorithm. It works like RLS with the exception of the search operator. The mutant x' is obtained from x by flipping each bit of x independently of the others with probability $1/n$. Steps without flipping bits do not count since they do not lead to a fitness evaluation. Let $e = 2.718\ldots$ be the Eulerian constant.

Theorem 3.1: *The expected number of steps until the (1+1) EA finds an optimum for the Ising model on the ring is bounded above by $T_{(1+1)}(n) = (e-1) \cdot (1+o(1)) \cdot T_{\mathrm{RLS}}(n) \leq 0.177 \cdot n^3 + o(n^3)$.*

Proof: We use the same ideas as in the proof of Theorem 2.2. In particular, we concentrate our analysis on the length of one block and we consider first only relevant steps, i. e., steps changing the length of the chosen block. We investigate another block if the chosen block has a length larger than $\lfloor n/i \rfloor$. The main idea is that we do not estimate the number of steps directly but we compare the (1+1) EA with RLS. For this purpose, we investigate some stochastic processes "between" RLS and the (1+1) EA.

We start with RLS* which applies the search operator of the (1+1) EA but only mutants x' where exactly one bit has flipped are considered for selection. Then the expected run time increases by the expected waiting time for a step flipping exactly one bit. Let X_n be the random variable counting the number of flipping bits. Then X_n is asymptotically Poisson distributed with parameter $\lambda = 1$. Since we do not consider steps without flipping bits, let Y be Poisson distributed with $\lambda = 1$ and let Y^* have the distribution of Y under the condition $Y \geq 1$. Then

$$\begin{aligned}\text{Prob}(Y^* = 1) &= \text{Prob}(Y = 1 \mid Y \geq 1) = \text{Prob}(Y = 1)/\,\text{Prob}(Y \geq 1) \\ &= e^{-1}/(1 - e^{-1}) = 1/(e-1).\end{aligned}$$

Hence, the expected waiting time until $Y^* = 1$ equals $e - 1$. The corresponding waiting time for X_n is bounded by $(e-1) \cdot (1 + o(1))$. This indeed is the essential factor why the (1+1) EA is slower than RLS. If the number of blocks is not too large, the probability that a step flipping more than one bit is relevant is much less than the corresponding probability for steps flipping one bit. The reason is that the other flipping bits typically increase the number of blocks.

Nevertheless, there are relevant steps flipping more than one bit and there are relevant steps changing the length of the considered block by more than 1. For each search point x let $p_k^+(x)$ be the probability that the next step is accepted and produces a search point where the length of the considered block B has been increased by k and let $p_k^-(x)$ be the corresponding probability for decreasing the length of B. We know from Section 2 that $p_k^+(x)$ may be larger than $p_k^-(x)$. To simplify the analysis, we investigate two further stochastic processes called (1+1) EA$_{\text{sym}}$ and RLS$_{\text{sym}}$. They are based on the algorithms (1+1) EA and RLS, respectively, but, if $p_k^+(x) > p_k^-(x)$, the probability of increasing the length of B is decreased to $p_k^-(x)$. As before, we consider another block if the length of B is larger than $\lfloor n/i \rfloor$. First, RLS$^*_{\text{sym}}$ is obviously faster than RLS*. We show that the expected run time of the (1+1) EA$_{\text{sym}}$ is bounded by the upper bound proven for RLS* and, therefore, also for RLS$^*_{\text{sym}}$ and, later, we compare the (1+1) EA and the (1+1) EA$_{\text{sym}}$.

Let A_t be the algorithm working t steps like the (1+1) EA$_{\text{sym}}$ and afterwards like RLS$^*_{\text{sym}}$. We prove by induction on t that the expected run time of A_t is not larger than the upper bound obtained for RLS*. This is true for $t = 0$, since $A_0 = \text{RLS}^*_{\text{sym}}$. For the induction step, we compare A_t and A_{t+1}. They are identical for the first t steps and we consider the (random) search point x after t steps. The probability of a relevant step is for the (1+1) EA$_{\text{sym}}$ not smaller than for RLS$^*_{\text{sym}}$. We compare the algorithms conditioned to some events and prove the claim for each of the cases. If the next step is neither relevant for A_t nor for A_{t+1}, the claim is obvious since the upper bound for RLS*

only depends on the length of the considered block. The perhaps larger probability of a relevant step of A_{t+1} is only in favor for A_{t+1}. Finally, we have to compare the effect of relevant steps. Instead of having steps changing the length of B by $+1$ and -1 (with the same probability), we now may change the length of B by $+k$ and $-k$ (with the same probability). Afterwards, we apply $\text{RLS}^*_{\text{sym}}$ in both cases. The upper bound for RLS^* (and also $\text{RLS}^*_{\text{sym}}$), namely the function T_N from Section 2 is increasing and concave (see Figure 1). Therefore, a $\pm k$-step instead of a ± 1-step reduces the expected run time, i. e., $(T(j+k)+T(j-k))/2 < (T(j+1)+T(j-1))/2$, if $k \geq 2$. For $t \to \infty$, we obtain the claim.

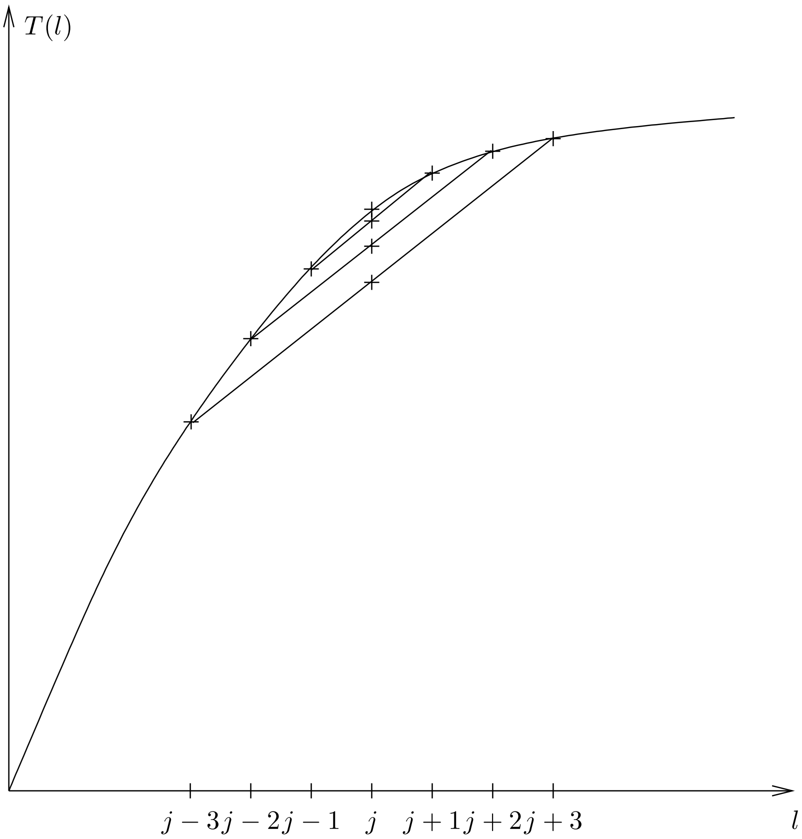

Figure 1: The function T.

Finally, we have to compare the (1+1) EA and the (1+1) EA_{sym}. We investigate a phase of length $n^{7/2}$. By Markoff's inequality, the probability that the (1+1) EA_{sym} needs more than $n^{7/2}$ steps is $O(n^{-1/2}) = o(1)$. In this case, we repeat the arguments for the next phase leading to an additional $1 + o(1)$ factor. In the following, we investigate a phase of length $n^{7/2}$. Events which altogether have a probability of $o(1)$ can be ignored since then the phase can be considered as unsuccessful also leading to a $1 + o(1)$ factor.

Let k be the length of the considered shortest block B, w. l. o. g. a block of ones. If $k \geq 4$, the string contains $0^4111^{k-4}110^4$. We consider the substrings 0^411 and 110^4. The probability that a phase contains a step with at least four flipping bits at these positions is $o(1)$ and this event can be ignored. Steps with at most three flipping bits at these positions do not eliminate one of the blocks. The situation is symmetric with respect to lengthenings and shortenings of B.

We are left with the situation $k \leq 3$. Recalling the analysis of RLS in Section 2, it is easy to obtain the result that the (1+1)*EA$_{\text{sym}}$ has an expected number of $O(n)$ steps where $k \leq 3$. By Markoff's inequality, we can ignore runs where this number is larger than $n^{3/2}$. The probability that a phase contains a step with at least two flipping bits in the substring $0^k1^k0^k$ is $o(1)$.

Finally, decreasing the length of B from k to 0 does not imply that we decrease the number of blocks. A new block may be created somewhere else. The probability of no bit flipping elsewhere is at least e^{-1}. Hence, with a probability of $1 - e^{-1}$ we are still in the same situation as before, i. e. , we have the same value of $k \in \{1, 2\}$. This happens on average $e/(e-1)$ times, each time increasing the expected time by $O(n^2)$. Hence, we have proved the theorem. □

It is worth noticing that we were not able to prove such a small bound by analyzing the (1+1) EA directly. It was helpful to analyze the simpler algorithm RLS and to compare RLS and the (1+1) EA.

Finally, we prove a lower bound similarly to the lower bound of Theorem 2.3.

Theorem 3.2: *Starting with two blocks of length εn and $(1-\varepsilon)n$, $0 < \varepsilon \leq 1/2$ a constant, the expected number of steps until the (1+1) EA finds an optimum for the Ising model on the ring is $\Theta(n^3)$.*

Proof: The upper bound is contained in Theorem 3.1. For the lower bound, we know that the probability of a relevant step equals $\Theta(1/n)$ since we have exactly two blocks. Hence, it is sufficient to prove a bound of $\Omega(n^2)$ on the number of relevant steps. Such a bound holds for RLS. We have seen in the proof of Theorem 3.1 that the (1+1) EA may gain from steps changing the block length by more than 1.

Again we compare the stochastic processes (1+1) EA$_{\text{sym}}$ and RLS$_{\text{sym}}$ with each other. They are faster than (1+1) EA and RLS, respectively. Hence, it is enough to prove the lower bound for the (1+1) EA$_{\text{sym}}$. With the same arguments as in the proof of Theorem 3.1 we obtain the result that the expected number of relevant steps of RLS$_{\text{sym}}$ is only by an additive term of $O(n)$ smaller than the corresponding number for RLS. Let $T_0^*(j)$ be the expected number of relevant steps of RLS$_{\text{sym}}$ starting with two blocks where the shorter one has length j. Then, by Lemma 2.1 and the considerations above,

$$T_0^*(j) \geq n^2/4 - cn - (n/2 - j + 1)(n/2 - j)$$

for some constant c. Let $T_t^*(j)$ be the expected number of relevant steps of RLS$^t_{\text{sym}}$ which works for t relevant steps like the (1+1) EA$_{\text{sym}}$ and then like RLS$_{\text{sym}}$. We prove by induction that

$$T_t^*(j) \geq T_0^*(j) - c't/n$$

for some constant c'. Having proved this claim it is easy to obtain the lower bound. By Markoff's inequality, there is a constant d such that the considered algorithms have a success probability of at least $1/2$ after dn^2 relevant steps. Then we have saved an expected number of $O(n)$ relevant steps. If the (1+1) $\mathrm{EA}_{\mathrm{sym}}$ was not successful in such a phase, it starts again with some value of j. We can repeat the arguments and the expected number of phases is at most 2 leading to an expected saving of only $O(n)$ steps. Hence, it is sufficient to prove the claim above. For a relevant step, the probability to change the block length by a constant k equals $\Theta(1/n^{k-1})$. Since the success probability after $n^{5/2}$ relevant steps is $1-2^{-\Omega(n^{1/2})}$, we can ignore relevant steps changing the block length by at least 4. They have a probability of $O(n^{-1/2})$ within $n^{5/2}$ steps and in the case of such a step we use a lower bound of 0. Comparing $T^*_{t+1}(j)$ with $T^*_t(j)$, we apply the fact that the lower bounds for $T^*_0(j)$ and $T^*_0(j-1)$ differ by $n-2j+2$. Let p_i, $1 \le i \le 3$, be the probability that a relevant step changes the block length by i. By our remark above, $p_1+p_2+p_3=1$. The following inequalities hold if all $T^*_0(k^*)$ are defined, i. e. , $0 \le k^* \le n/2$. Then, by induction hypothesis and the law of total probability,

$$\begin{aligned} T^*_{t+1}(j) &\ge -c't/n + (p_1/2)\cdot(T^*_0(j-1)+T^*_0(j+1)) \\ &\quad +(p_2/2)\cdot(T^*_0(j-2)+T^*_0(j+2)) + (p_3/2)\cdot(T^*_0(j-3)+T^*_0(j+3)). \end{aligned}$$

Now

$$\begin{aligned} T^*_0(j-2) &= T^*_0(j-1)-(n-2(j-1)), \\ T^*_0(j+2) &= T^*_0(j+1)+(n-2(j+2)), \text{ and} \\ T^*_0(j-2)+T^*_0(j+2) &= T^*_0(j-1)+T^*_0(j+1)-6. \end{aligned}$$

Similarly,

$$T^*_0(j-3)+T^*_0(j+3) = T^*_0(j-1)+T^*_0(j+1)-16.$$

Altogether, since $T^*_0(j) = (1/2)\cdot(T^*_0(j-1)+T^*_0(j+1))$,

$$\begin{aligned} T^*_{t+1}(j) &\ge -c't/n + (1/2)\cdot(T^*_0(j-1)+T^*_0(j+1)) - c''/n \\ &\ge c'(t+1)/n + T^*_0(j), \end{aligned}$$

if c' is chosen in an appropriate way. This proves the claim and the theorem. □

4 The Expected Number of Generations of Parallel RLS

A GA works with a population of $s(n)$ individuals and, in most cases, run time is defined as the number of generations. The number of fitness evaluations is larger by a factor of $s(n)$. For RLS and the (1+1) EA, the number of generations equals the number of fitness evaluations. In order to have a fair comparison with GAs, we consider population-based RLS and (1+1) EA. Parallel RLS (PRLS) or (1+λ) RLS creates λ children from the parent x using the search operator of RLS. The children are created independently. Selection chooses x if all children are worse and chooses one of the fittest children uniformly at random otherwise.

Let x be a search point with i blocks. The probability of a step changing the length of the considered block B is at least $3/n$ and the probability of a step creating a child with at least the same fitness as x is at most $2i/n$.

For $\lambda = n$, there is, on average, a constant fraction of steps creating at least one child changing the length of B and not creating more than $4i$ children with i blocks. Therefore, the probability of a relevant step is $\Omega(1/i)$. The expected number of relevant steps with i blocks is bounded above by $O(n^2/i^2)$ (see Lemma 2.1). Hence, (1+n) RLS needs an expected number of $O(n^2/i)$ generations with strings with i blocks. Considering all even $i \leq n$, this leads to $O(n^2 \log n)$ generations and $O(n^3 \log n)$ fitness evaluations. In the following we improve this result.

Theorem 4.1: *The expected number of generations until (1+$\lfloor n/\log n \rfloor$) RLS finds the optimum for the Ising model on the ring is bounded above by $O(n^2 \log n)$ and the expected number of fitness evaluations by $O(n^3)$.*

Proof: It is sufficient to investigate the number of generations since each generation consists of $\lfloor n/\log n \rfloor$ fitness evaluations. Let B be the considered block (as in the sections before). The probability that no child shortens or lengthens B equals $(1 - c/n)^{\lfloor n/\log n \rfloor} = 1 - \Theta(1/\log n)$, where $c \in \{3, 4\}$ depends on the length of B. In any case, the expected waiting time for a generation with a child changing B equals $\Theta(\log n)$. If x contains i blocks, the expected number of children with the same number of blocks as x is $\Theta(i/\log n)$ and the probability that this number is bounded by $O(i/\log n)$ is at least 1/2 (Markoff's inequality).

If $i \geq \log n$, the probability of choosing a child where B is changed, if such a child is created, is $\Omega(\log n/i)$. The waiting time for such a step is $O(i/\log n)$. Hence, each step has a probability of $\Omega(1/i)$ of being relevant. We can continue as in the case of $\lambda = n$ to obtain a bound of $O(n^2 \log n)$ on the expected number of generations.

If $i < \log n$ and one child changes B, the probability that all other children have more blocks equals $(1 - \Theta(i/n))^{\lfloor n/\log n \rfloor - 1}$ which is bounded below by a positive constant. Then the generation is relevant. Hence, the expected waiting time for a relevant generation equals $\Theta(\log n)$ and the expected number of generations is bounded by $O((n^2 \log n)/i^2)$. Considering all $i < \log n$ and even, this gives an additional term of $O(n^2 \log n)$. □

5 The Expected Number of Generations of the (1+λ) EA

The (1+λ) EA applies the search operator of the (1+1) EA and produces independently λ children from the parent which is the only individual of the current population. We have to be careful with the selection operator. It is likely that many children are a replica of the parent. In order to guarantee exploration of the search space, we select the parent x only if all children $y \neq x$ have a worse fitness than x. Otherwise, we randomly select an individual among the fittest children $y \neq x$.

Our analysis of the (1+1) EA in Section 3 was based on a comparison with RLS. The analysis of (1+λ) RLS in Section 4 essentially was an analysis of the waiting time for a relevant generation. We obtain asymptotically the same results for the (1+λ) EA. In a relevant step, the probability distribution describing how the length of B changes is the same for the (1+1) EA and the (1+λ) EA and it is the same for RLS and (1+λ) RLS. Hence, the arguments of Section 3 can be applied and lead to the following result.

Theorem 5.1: *The expected number of generations until the* (*1+*$\lfloor n/\log n \rfloor$) *EA finds the optimum for the Ising model on the ring is bounded above by* $O(n^2 \log n)$ *and the expected number of fitness evaluations by* $O(n^3)$.

6 A Comparison with GA Experiments

We have no doubt that crossover can play an essential role for the Ising model on the ring. A theoretical fundament for this argument will be presented in Sections 7 and 8. Here, we want to argue that mutation-based EAs are better than expected in many papers. Van Hoyweghen (2002) claims that “the presence of spin-flip symmetry in the one-dimensional Ising model prevents an unspecialized GA to find an optimum in a reasonable amount of time.” Van Hoyweghen, Goldberg, and Naudts (2001) indicate in this context that “the Ising model shows that for a certain class of optimization problems niching becomes a necessity for a GA to solve these problems.” Our results have shown that unspecialized EAs solve this problem in reasonable time. The upper bounds on the expected run times of RLS ($0.103n^3 + 0.375n^2(\ln n + 1)$ and even $117,957$ for $n = 100$) and of the (1+1) EA (by a factor of 1.72 slower than RLS) show this even for populations of size 1. The time bounds are much better, namely $O(n^2 \log n)$, if $n/\log n$ children are generated in parallel. Hence, the optimization is finished in a reasonable amount of time without any niching. Van Hoyweghen (2001) has considered the case of GAs for $n = 100$ and a population size of 100. The best parameters for tournament selection and two-point crossover lead to an average number of $35,857$ generations. This can be decreased to $10,881$ using SAWing (Stepwise Adaptation of Weights). With an Island model and a distributed GA there is a good chance that $10,000$ generations suffice. In all these cases a population of size $s(n) \geq 100$ is used. In general, it is claimed that a population size of $10.9n^{0.57}$ suffices. These algorithms need less generations than the mutation-based algorithms examined in this paper but they do not beat RLS with respect to the expected number of fitness evaluations (at least for $n = 100$).

7 The Expected Run Time of GIGA

Although mutation-based algorithms are surprisingly efficient for the Ising model on a ring, it is believed that GAs can be faster. It is difficult to analyze the effect of crossover if one is interested in the expected optimization time. We are not able to analyze distributed GAs. Therefore, we analyze GAs which are specialized to work on the Ising model on the ring. We cannot expect to obtain the same good time bounds for unspecialized GAs.

In this section, we analyze a simple variant of GIGA (Gene Invariant Genetic Algorithm) introduced by Culberson (1992) and also called (1+1) GA by Dietzfelbinger, van Hoyweghen, Naudts, and Wegener (2003). The population has size 2 and consists of a search point $x \in \{0,1\}^n$ and its bitwise complement $\overline{x}$. In the initialization step, x is chosen uniformly at random. Later, a new pair of search points $(y, \overline{y})$ is produced from $(x, \overline{x})$ by crossover. Since $f(x) = f(\overline{x})$, the new pair $(y, \overline{y})$ replaces $(x, \overline{x})$ if $f(y) \geq f(x)$ (or equivalently, if y has not more monochromatic blocks than x). Since we want to cut out a block in x and to replace it by its bitwise complement, two-point crossover seems to be the appropriate recombination operator. Let us consider the effect of crossover at the positions j and k, $0 \leq j < k < n$. A position p is called border of x, if $x_p \neq x_{p+1}$ or $x_n \neq x_1$ if $p = 0$. Let i be the number of blocks of x.

Case 1: The positions j and k are not borders. Then y has $i+2$ blocks and $(y, \overline{y})$ is not accepted.

Case 2: Exactly one of the positions j and k is a border. Then y also has i blocks and $(y, \overline{y})$ is accepted but the fitness is not changed.

Case 3: The positions j and k are borders. If $i > 2$, y has $i - 2$ blocks. If $i = 2$, y has one block. In any case, $(y, \overline{y})$ is accepted and the fitness is improved.

As long as x is not optimal, $i \geq 2$ and there are $\binom{i}{2}$ among $\binom{n}{2}$ pairs of positions which lead to an improved fitness. Hence, the expected optimization time can be bounded above by (remember that $I = \{i \mid 2 \leq i \leq n, i \text{ even}\}$)

$$
\begin{aligned}
\sum_{i \in I} \binom{n}{2} \Big/ \binom{i}{2} &= n(n-1) \sum_{i \in I} \frac{1}{i(i-1)} \\
&= n(n-1) \sum_{i \in I} \left(\frac{1}{i-1} - \frac{1}{i} \right) \leq 0.70 \cdot n \cdot (n-1).
\end{aligned}
$$

Moreover, the expected i-value of the initial search point is $n/2$ and, by Chernoff bounds, the probability that it is larger than $n/3$, is $1 - 2^{-\Omega(n)}$. Hence, we get a lower bound on the expected optimization time if we compute the same sum for all $i \in I' := \{i \mid i \leq n/3, i$ even$\}$. Altogether, we have obtained the following result.

Theorem 7.1: *The expected number of steps until GIGA with two-point crossover finds an optimum for the Ising model on the ring is bounded above by* $0.70 \cdot n \cdot (n-1)$ *and bounded below by* $0.69 \cdot n^2 - o(n^2)$.

We can generalize GIGA to (1+λ) GIGA where λ offspring pairs are produced independently and a best one is chosen if it is not worse than the parent. We analyze the (1+n) GIGA. The probability of producing a better offspring is bounded below by a positive constant, if $i > n^{1/2}$, and by $\Omega(i^2/n)$, otherwise. Hence, the expected number of generations equals $\Theta(n)$.

Theorem 7.2: *The expected number of generations until the* $(1 + n)$ *GIGA with two-point crossover finds an optimum for the Ising model on the ring equals* $\Theta(n)$*, the expected number of fitness evaluations equals* $\Theta(n^2)$*.*

We have seen that we have to hit the borders in order to improve the fitness. This is more difficult if the number of borders is small. Using k-point crossover for $k \geq 3$ or uniform crossover is, therefore, worse than two-point crossover. What about one-point crossover? This can be interpreted as two-point crossover where the first border is fixed to $j = 0$. If $j = 0$ is not a border and the cut point k is a border, the new pair $(y, \overline{y})$ has the same fitness and is accepted. Moreover, $j = 0$ is a border of $(y, \overline{y})$. If $j = 0$ is a border, the next offspring is accepted in any case. If one-point crossover does not hit a border, the offspring has the same fitness but position 0 is no longer a border. If k hits a border, we have improved the fitness. Hence, if position 0 is not a border, we improve the fitness, if, within two subsequent steps, we hit a border. This probability is $i/(n-1)$ for the first step (since 0 is not a border) and $(i-1)/(n-1)$ for the second step (since 0 is a border). A phase consists of one step if 0 is not a border and we do not hit a border and it consists of two steps otherwise. The expected number of phases of length 2 equals $(n-1)/(i-1)$. Altogether, the expected number of steps until the fitness is increased equals

$$\frac{(n-1)^2}{i \cdot (i-1)} + \frac{n-1}{i} = \frac{n^2 - 3n + 2 + in - i}{i \cdot (i-1)}$$

which is close to the corresponding value for two-point crossover. Hence, the expected number of fitness evaluations equals $\Theta(n^2)$. The (1+n) GIGA with one-point crossover needs an expected number of $\Theta(n)$ generations. If 0 is not a border, only an offspring, where 0 is a border, is accepted. For n offsprings, the success probability is bounded below by a positive constant. Then, in the next step, the probability that at least one offspring has a better fitness is again bounded below by a positive constant. This leads to the surprising result that one-point crossover is almost as efficient as two-point crossover for the Ising model on the ring.

Theorem 7.3: *The expected number of fitness evaluations until GIGA or the* (1+n) *GIGA with one-point crossover finds an optimum for the Ising model on the ring equals* $\Theta(n^2)$*. For the* $(1 + n)$ *GIGA, the number of generations equals* $\Theta(n)$*.*

8 The Expected Run Time of a GA with Fitness Sharing

The variant of GIGA analyzed in Section 7 is highly specialized. Diversity in the population of size 2 is guaranteed by choosing always individuals with the maximal Hamming distance. Here, we consider a GA with the unusual small population size 2 where diversity is supported by fitness sharing. Populations are multisets. In fitness sharing, the closeness of x and y is measured by

$$S(x,y) := \max\{1 - d(x,y)/\sigma, 0\}$$

where d is an appropriate distance measure and σ is a critical value deciding when x and y are so far from each other that they do not share fitness. In our case, d is the Hamming distance and $\sigma := n$ since we like to produce individuals with large Hamming distance. Then, for population P,

$$S(x,P) := \sum_{y \in P} S(x,y).$$

The shared fitness of x in the population P is defined by

$$f(x,P) := f(x)/S(x,P)$$

if f is the real fitness. Finally, the fitness $f(P)$ is defined as the sum of all $f(x,P)$, $x \in P$.

The following GA applies two-point crossover to produce two children and mutations flipping each bit independently with probability $1/n$.

Algorithm 8.1: (Steady-state GA with population size 2 and fitness sharing)

1.) The initial population P consists of two individuals chosen independently and uniformly at random.

2.) Selection for reproduction. Both individuals x and y are chosen.

3.) Offspring creation. One of the Steps 3a and 3b is chosen uniformly at random.

3a.) $x' := \text{mutate}(x),\ y' := \text{mutate}(y),\ P' := P \cup \{x', y'\}$.

3b.) $(\tilde{x}, \tilde{y}) := \text{two-point-crossover}(x,y),\ x' := \text{mutate}(\tilde{x}),\ y' := \text{mutate}(\tilde{y}), P' := P \cup \{x', y'\}$.

4.) Selection of the next generation. Choose a population $P \subseteq P'$ of size 2 with the maximal $f(P)$-value.

Since we work with populations of very small size, it is not too time-consuming to choose in Step 4 a population with the largest f-value. This reflects the real idea behind fitness sharing. The shared fitness of the population should be large. For large populations of size $n^{1/2}$ or n, there are exponentially many possible successor populations if we have produced the same number of offspring. Therefore, many algorithms only compare the $f(x, P')$-values.

Let the population P consist of the individuals x and y with a Hamming distance of $d = d(P)$. Let $i(z)$ be the number of borders within the individual z and let $i = i(P) := i(x) + i(y)$. Then $f(z) = n - i(z)$ and

$$f(x, P) = \frac{n - i(x)}{1 - H(x, x)/n + 1 - H(x, y)/n} = \frac{n - i(x)}{2 - d/n}$$

and

$$f(P) = \frac{2n - i}{2 - d/n}.$$

Hence, we can increase P by decreasing i and/or by increasing d. As long as we do not decrease i, we hope to increase d. If $d = n$, we have two complementary individuals and two-point crossover is a good operator to decrease i (see Section 7). Since $0 \leq f(P) \leq 2n$ and f cannot decrease because of the plus-strategy for selection, we try to analyze the expected time until f has been increased at least by a constant additive term c. For this purpose, we classify the possible populations P:

- type OPT contains all P where $i \leq 1$, i. e., at least one individual is optimal,
- type $A(i)$, $i \geq 2$, contains all P where $i = i(P)$ and $d = n$,
- type B contains all P where $2 \leq i \leq n$ and $d < n$, and
- type C contains all P where $i > n$ and $d < n$.

Theorem 8.2: *The expected number of fitness evaluations until the steady-state GA with population size 2 and fitness sharing finds an optimum for the Ising model on the ring is bounded above by $O(n^2)$.*

Proof: All populations of type $A(i)$ have the same fitness $2n - i$. After having increased the fitness, we will never accept a population of type $A(i)$. Moreover, if $P = \{x, y\}$ is of type $A(i)$, then $y = \overline{x}$. The expected waiting time until two-point crossover creates a population P' of type $A(i-4)$ is bounded by $O(n^2/i^2)$, see Section 7. Then $f(P') - f(P) = 4$. The probability of performing Step 3b and flipping no bit by mutation is bounded below by a positive constant. Hence, the expected time with populations of type $A(i)$ is bounded above by $O(n^2/i^2)$ and the expected time with populations of type A is bounded above by $O(n^2)$.

For populations of type B or C, we prove that the probability of increasing the fitness by at least $1/4$ is bounded below by $\Omega(1/n)$. We have to wait for at most $8n$ of such steps which proves the theorem.

Let $P = \{x, y\}$ be of type B. Since $d < n$, $x \neq \overline{y}$. Let j be the rightmost position where $x_j = y_j$. Then $x_{j+1} \neq y_{j+1}$ (where $n + 1$ is identified with 1 since we are on a ring). W.l.o.g. $x_j = x_{j+1}$ and $y_j \neq y_{j+1}$. With a probability of $\Omega(1/n)$, we choose Step 3a and only bit j is flipped when producing y'. Then $f(y') \geq f(y)$ and $H(x,y') = H(x,y)+1$. The population $P' = \{x, y'\}$ is a possible successor population and

$$\begin{aligned} f(P') - f(P) &\geq \frac{2n-i}{2-(d+1)/n} - \frac{2n-i}{2-d/n} \\ &= \frac{(2n-i)\cdot(2-d/n)-(2n-i)\cdot(2-d/n-1/n)}{(2-(d+1)/n)\cdot(2-d/n)} \geq \frac{1}{4}, \end{aligned}$$

since the numerator equals $2-i/n \geq 1$ ($i \leq n$ for type B populations) and the denominator is at most 4.

Let $P = \{x, y\}$ be of type C. Since $i > n$, w. l. o. g. $i(x) > n/2$. Then x has more than $n/2+1$ monochromatic blocks and, therefore, at least one block B of length 1. With a probability of $\Omega(1/n)$, we choose Step 3a and only the bit of B is flipped when producing x'. Then $f(x') = f(x)+2$ and $H(x', y) \geq H(x,y)-1$. The population $P' = \{x', y\}$ is a possible successor population and

$$\begin{aligned} f(P') - f(P) &\geq \frac{2n-(i-2)}{2-(d-1)/n} - \frac{2n-i}{2-d/n} \\ &= \frac{(2n-i+2)\cdot(2-d/n)-(2n-i)\cdot(2-d/n+1/n)}{(2-(d-1)/n)\cdot(2-d/n)} \geq \frac{1}{4}, \end{aligned}$$

since the numerator equals $(4-2d/n)-(2-i/n) \geq i/n \geq 1$ ($i > n$ for type C populations) and the denominator again is at most 4. This finishes the proof of the theorem. □

Finally, we can consider a GA with population size 2 and fitness sharing which produces $P'_1, \ldots, P'_n$ by performing Step 3 n times independently in parallel. Then it selects a population $P \subseteq P'_i$ for some i which has the largest $f(P)$-value. For populations of type B or type C, the probability of producing a population whose fitness is by an additive term of at least $1/4$ better is then $\Omega(1)$. The proof of Theorem 7.2 shows that the expected number of generations with populations of type A is bounded by $O(n)$. Hence, we obtain the following result.

Theorem 8.3: *The expected number of generations until the GA with population size 2, fitness sharing, and n pairs of offspring per generation finds an optimum for the Ising model on the ring is bounded above by $O(n)$.*

Conclusions

The Ising model is a good model to analyze the adaptation capabilities of EAs and GAs. In particular, the Ising model on the ring leads to surprising results. Mutation-based algorithms and even randomized local search are much more efficient than expected in the GA community. This is especially true if we consider the number of generations in the case of producing more than one offspring. Nevertheless, recombination can decrease the expected optimization time. This has been proved rigorously for two specialized GAs which work with very small populations. It is an open problem to analyze generic GAs with niching for the Ising model on the ring.

Acknowledgement. Thanks to Carsten Witt for discussions on the paper, in particular, those leading to Theorem 3.2.

References

[1] Culberson, J. (1992). Genetic invariance: A new paradigm for genetic algorithm design. TR 92–02, Univ. of Alberta.

[2] Dietzfelbinger, M., Naudts, B., van Hoyweghen, C., and Wegener, I. (2003). The analysis of a recombinative hill-climber on H-IFF. IEEE–Trans. on Evolutionary Computation 7, 417–423.

[3] Droste, S., Jansen, T., and Wegener, I. (2002). On the analysis of the (1+1) evolutionary algorithm. Theoretical Computer Science 276, 51–81.

[4] Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, Reading, MA.

[5] Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems.* Univ. of Michigan, MI.

[6] Ising, E. (1925). Beiträge zur Theorie des Ferromagnetismus. Z. Physik 31, 235–258.

[7] Jansen, T. and De Jong, K. (2002). An analysis of the role of offspring population size in EAs. Proc. of the Genetic and Evolutionary Computation Conference (GECCO 2002), 238–246. Morgan Kaufmann, San Mateo, CA.

[8] Jansen, T., De Jong, K., and Wegener, I. (2003). On the choice of the offspring population size in evolutionary algorithms. Submitted to Evolutionary Computation.

[9] Jansen, T. and Wegener, I. (2001). Real royal road functions – where crossover is provably essential. Proc. of the Genetic and Evolutionary Computation Conference (GECCO 2001), 375–382. Morgan Kaufmann, San Mateo, CA.

[10] Jansen, T. and Wegener, I. (2002). The analysis of evolutionary algorithms – a proof that crossover really can help. Algorithmica 34, 47–66.

[11] Naudts, B. and Naudts, J. (1998). The effect of spin-flip symmetry on the performance of the simple GA. Proc. of 5th Conf. on Parallel Problem Solving from Nature (PPSN - V), LNCS 1498, 67–76.

[12] Pelikan, M. and Goldberg, D. E. (2003). Hierarchical BOA solves Ising spin glasses and MAXSAT. Proc. of the Genetic and Evolutionary Computation Conference (GECCO 2003), LNCS 2724, 1271–1282.

[13] Van Hoyweghen, C. (2002). *Symmetry in the Representation of an Optimization Problem.* Ph. D. Thesis. Univ. Antwerpen.

[14] Van Hoyweghen, C., Goldberg, D. E., and Naudts, B. (2001). Building block superiority, multimodality and synchronization problems. IlliGAL Rep. No. 2001020, Illinois Genetic Algorithms Laboratory.

[15] Van Hoyweghen, C., Goldberg, D. E., and Naudts, B. (2002). From twomax to the Ising model: Easy and hard symmetrical problems. Proc. of the Genetic and Evolutionary Computation Conference (GECCO 2002), 626–633. Morgan Kaufmann, San Mateo, CA.

[16] Van Hoyweghen, C., Naudts, B., and Goldberg, D. E. (2002). Spin-flip symmetry and synchronization. Evolutionary Computation 10, 317–344.

[17] Watson, R. (2001). Analysis of recombinative algorithms on a non-separable building block problem. In *Foundations of Genetic Algorithms* 6 (Eds. Martin, W. and Spears, W.), 69–89, Morgan Kaufmann, San Mateo, CA.