Towards Analyzing Recombination Operators in Evolutionary Search

Yang Yu Chao Qian Zhi-Hua Zhou

National Key Laboratory for Novel Software Technology,
Nanjing University, Nanjing 210093, China,
{yuy,qianc,zhouzh}@lamda.nju.edu.cn

Abstract. Recombination (also called *crossover*) operators are widely used in EAs to generate offspring solutions. Although the usefulness of recombination has been well recognized, theoretical analysis on recombination operators remains a hard problem due to the irregularity of the operators and their complicated interactions to mutation operators. In this paper, as a step towards analyzing recombination operators theoretically, we present a general approach which allows to compare the runtime of an EA turning the recombination on and off, and thus helps to understand when a recombination operator works. The key of our approach is the *Markov Chain Switching Theorem* which compares two Markov chains for the first hit of the target. As an illustration, we analyze some recombination operators in evolutionary search on the LeadingOnes problem using the proposed approach. The analysis identifies some insight on the choice of recombination operators, which is then verified in experiments.

1 Introduction

Evolutionary algorithms (EAs) run a circle of reproducing offspring solutions from the current population, and then selecting to weed out bad solutions [1]. The most popular reproduction operators are *mutation* and *recombination* (also called *crossover*). Contrary to mutation operators that are defined on individual solutions, recombination operators are defined on a population of solutions, that is, they generate offspring solutions by mixing up a set of (usually two) solutions. Recombination operators were born together with the first genetic algorithm. They are a special characteristic of EAs, which makes EAs significantly differ from classical optimization techniques such as branch-and-bound strategy [17], as well as other heuristic search methods such as simulated annealing [17].

When EAs are used to tackle optimization problems, the runtime is among the central concerns. Due to recent advances in theoretical analysis of EAs, e.g. [2, 9, 10, 20, 3], a landscape of computational complexity of EAs is emerging. However, most of the previous studies focus on EAs using mutation only.

There are some recent studies on recombination operators [14, 18, 11–13, 15, 4, 5]. Lin and Yao [14] assumed that a 'step size' is critical for recombination operators. They compared the step sizes of four recombination operators, and empirically studied an EA with these recombination operators yet without mutation operator on several problems. Spears [18] studied the construction and destruction probabilities and equilibrium of recombination using the schema theory, and concluded that the recombination operators are beneficial when there are few local optima. Several recent studies proved some effects of recombination operators. On the positive side, Jansen and Wegener [11, 12] proved that, on the *Real Royal Road* problem, a recombination operator reduces the runtime of an EA from exponential to polynomial; Lehre and Yao [13] proved that, on the *TwoPath* problem of computing unique input-output sequences, a recombination operator reduces the runtime exponentially; Doerr et al. [4, 5] proved that, on the *all pairs shortest path problem*, a recombination operator reduces the runtime. On the negative side, Richter et al. [15] constructed a problem called *Ignoble Trails* on which the EA using mutation only requires polynomial runtime, yet the EA using mutation with recombination requires exponential runtime. These studies disclosed some properties of recombination operators from different aspects, however, there is no general approach for theoretically analyzing recombination operators.

Usually, a recombination operator is used together with a mutation operator in EA, and a nontrivial population (more than one solution) is maintained with a selection operator. Thus, the analysis of recombination operators needs to consider the interactions between mutation and recombination operators, the effect of population size, and the effect of the selection pressure. So, it is not surprising that theoretical analysis on recombination operators is more difficult than that on mutation operators.

In this paper, as a step towards theoretical analysis of recombination operators, we try to tackle the interaction between mutation and recombination operators. We present the *Markov Chain Switching* Theorem for the comparison of two Markov chains on their first hit of the target. This theorem allows to compare the one-step behaviors of two EAs, thus can be used to compare the average runtime of an EA turning on and off the recombination operator. This theorem provides a general tool towards the understanding of the behaviors of recombination operators. As an illustration, we theoretically analyze four strategies using two different recombination operators in an EA on the LeadingOnes problem using the tool. The LeadingOnes problem is a well-studied problem for EAs using mutation only [16], yet the effect of recombination operators on the problem remains untouched. Our analysis identifies helpful and unhelpful recombination strategies from the candidates, which are then verified in experiments. It is interesting to find from the analysis that similar recombination operators can have opposite effects.

The rest of this paper is organized as follows. Section 2 introduces preliminaries, Section 3 presents the main theorem, Section 4 shows the case study on the LeadingOnes problem, and Section 5 concludes.

2 Preliminaries

We model EAs as Markov chains [9, 20]. A population of an EA can be mapped to a state of a Markov chain. Formally, let $\mathcal{X}$ denote the population space containing all possible populations of an EA. A Markov chain $\{\xi_t\}_{t=0}^{+\infty}$ modeling the EA can be constructed by taking $\mathcal{X}$ as the state space, i.e., $\xi_t \in \mathcal{X}$. Let $\mathcal{X}^* \subseteq \mathcal{X}$ denote the set of all optimal populations containing at least one optimal solution. The process of an EA seeking $\mathcal{X}^*$ can be analyzed by studying the corresponding Markov chain.

Given a Markov chain $\{\xi_t\}_{t=0}^{+\infty}(\xi_t \in \mathcal{X})$ and a target subspace $\mathcal{X}^* \subset \mathcal{X}$, $\{\xi_t\}_{t=0}^{+\infty}$ is said to be an *absorbing Markov chain* if $\forall t \geq 0 : P(\xi_{t+1} \notin \mathcal{X}^* \mid \xi_t \in \mathcal{X}^*) = 0$, and $\{\xi_t\}_{t=0}^{+\infty}$ is said to be a *homogeneous Markov chain* if $\forall t \geq 0 \forall x, y \in \mathcal{X} : P(\xi_{t+1} = x \mid \xi_t = y) = P(\xi_1 = x \mid \xi_0 = y)$. All practical EAs track the best-so-far solutions during the evolution process. These EAs can be modeled as absorbing Markov chains. EAs without time-variant operators can be modeled as homogeneous Markov chains.

Starting from time step $\tilde{t}$ when $\xi_{\tilde{t}} = x$, let $\tau_{\tilde{t}}$ be a random variable denoting the hitting events:

$$\begin{aligned}&\tau_{\tilde{t}} = 0 : \xi_{\tilde{t}} \in \mathcal{X}^*,\\ &\tau_{\tilde{t}} = 1 : \xi_{\tilde{t}+1} \in \mathcal{X}^* \wedge \xi_i \notin \mathcal{X}^* \ (i = \tilde{t}),\\ &\tau_{\tilde{t}} = 2 : \xi_{\tilde{t}+2} \in \mathcal{X}^* \wedge \xi_i \notin \mathcal{X}^* \ (i = \tilde{t}, \tilde{t}+1),\\ &\dots .\end{aligned}$$

The mathematical expectation of $\tau_{\tilde{t}}$, $\mathbb{E}[\![\tau_{\tilde{t}} \mid \xi_{\tilde{t}} = x]\!] = \sum_{i=0}^{+\infty} i \cdot P(\tau_{\tilde{t}} = i)$, is called the *conditional first hitting time* (CFHT) of the Markov chain from $\tilde{t}$ and $\xi_{\tilde{t}} = x$. If $\xi_{\tilde{t}}$ is drawn from a distribution π of states, the expectation of the CFHT over $\xi_{\tilde{t}}$,

$$\mathbb{E}[\![\tau_{\tilde{t}} \mid \xi_{\tilde{t}} \sim \pi]\!] = \mathbb{E}_{x\sim\pi}[\![\tau_{\tilde{t}} \mid \xi_{\tilde{t}} = x]\!] = \sum\nolimits_{x\in\mathcal{X}} \pi(x)\mathbb{E}[\![\tau_{\tilde{t}} \mid \xi_{\tilde{t}} = x]\!],$$

is called the *distribution-conditional first hitting time* (DCFHT) of the Markov chain from $\tilde{t}$ and $\xi_{\tilde{t}} \sim \pi$. The DCFHT of the chain from $t = 0$ and uniform distribution π_u,

$$\mathbb{E}[\![\tau]\!] = \mathbb{E}[\![\tau_0 \mid \xi_0 \sim \pi_u]\!] = \mathbb{E}_{x\sim\pi_u}[\![\tau_0 \mid \xi_0 = x]\!] = \sum\nolimits_{x\in\mathcal{X}} \frac{1}{|\mathcal{X}|}\mathbb{E}[\![\tau_0 \mid \xi_0 = x]\!],$$

is called the *expected first hitting time* (EFHT) of the Markov chain [9, 10, 20], which implies the average runtime of EAs.

About the CFHT, we will use the following two lemmas [8].

Lemma 1. *Given an absorbing Markov chain $\{\xi_t\}_{t=0}^{+\infty}(\xi_t \in \mathcal{X})$ and a target subspace $\mathcal{X}^* \subset \mathcal{X}$, we have, for CFHT,*

$$\forall x \notin \mathcal{X}^* : \mathbb{E}[\![\tau_t \mid \xi_t = x]\!] = 1 + \sum_{y\in\mathcal{X}} P(\xi_{t+1} = y \mid \xi_t = x)\mathbb{E}[\![\tau_{t+1} \mid \xi_{t+1} = y]\!],$$

and for DCFHT,

$$\mathbb{E}[\![\tau_t \mid \xi_t \sim \pi_t]\!] = \mathbb{E}_{x\sim\pi_t}[\![\tau_t \mid \xi_t = x]\!] = 1 - \pi_t(\mathcal{X}^*) + \mathbb{E}[\![\tau_{t+1} \mid \xi_{t+1} \sim \pi_{t+1}]\!],$$

where $\pi_{t+1}(x) = \sum_{y\in\mathcal{X}} \pi_t(y)P(\xi_{t+1} = x \mid \xi_t = y)$.

Lemma 2. *Given an absorbing homogeneous Markov chain $\{\xi_t\}_{t=0}^{+\infty}(\xi_t \in \mathcal{X})$ and a target subspace $\mathcal{X}^* \subset \mathcal{X}$, it holds $\forall t_1, t_2 : \mathbb{E}[\![\tau_{t_1} \mid \xi_{t_1} = x]\!] = \mathbb{E}[\![\tau_{t_2} \mid \xi_{t_2} = x]\!]$.*

3 Markov Chain Switching Theorem

We focus on tackling the interaction between mutation and recombination operators. Considering that there are many studies devoted to the estimate of EFHT of EAs using mutation only [2, 7, 9, 20], we propose to analyze recombination operators by studying an EA that turns the recombination operator on and off. For this purpose, we present the following Markov Chain Switching Theorem.

Theorem 1 (Markov Chain Switching Theorem). *Given two absorbing homogeneous Markov chains* $\{\xi_t\}_{t=0}^{+\infty}$ *and* $\{\xi'_t\}_{t=0}^{+\infty}$. *Let* τ *and* τ' *denote the hitting events of* ξ_t *and* ξ'_t, *respectively. Let* π_t *denote the distribution of* ξ_t *on states. If it satisfies*

$$\forall t : \sum\nolimits_{x,y\in\mathcal{X}} \pi_t(x)P(\xi_{t+1} = y \mid \xi_t = x)\mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!] \le (\ge) \sum\nolimits_{x,y\in\mathcal{X}} \pi_t(x)P(\xi'_{t+1} = y \mid \xi'_t = x)\mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!] \,, \tag{1}$$

and both $\mathbb{E}[\![\tau]\!]$ *and* $\mathbb{E}[\![\tau']\!]$ *are finite, it holds that* $\mathbb{E}[\![\tau]\!] \le (\ge)\mathbb{E}[\![\tau']\!]$.

Before the proof, it is helpful to briefly explain the significance of the theorem. Let ξ_t corresponds to an EA that turns its recombination operator on, and ξ'_t be the opposite. In Eq. 1, only the CFHT of the EA turning recombination off, i.e., $\mathbb{E}[\![\tau'_{t+1}|\xi'_{t+1}]\!]$, is required to be calculated, whilst there is no such term as $\mathbb{E}[\![\tau_{t+1}|\xi_{t+1}]\!]$. Therefore, Theorem 1 enables us to analyze the effect of recombination operator by comparing the one-step transition probabilities, i.e., $P(\xi_{t+1} = y \mid \xi_t = x)$ and $P(\xi'_{t+1} = y \mid \xi'_t = x)$, avoiding complicated calculations on the CFHT of the EA turning recombination on. Note that, besides recombination, Theorem 1 can also be applied to analyze many other kinds of operators.

Proof. Here, we prove the "$\le$" case, while the "$\ge$" case can be proved similarly. Denote the operator used in the EA modeled by ξ as *op*, and that used in the EA modeled by ξ' as *op*$'$, respectively. Let Markov chain $\{\xi^k_t\}_{t=0}^{+\infty}$ corresponds to the EA using *op* at time steps $\{0, 1, \cdots, k-1\}$ and using *op*$'$ otherwise. Thus, for any k and any time step $t \ge k$, we have

$$\forall x \in \mathcal{X} : \mathbb{E}[\![\tau^k_t \mid \xi^k_t = x]\!] = \mathbb{E}[\![\tau'_t \mid \xi'_t = x]\!], \tag{2}$$

since the two chains use the same operator after the time step $k-1$. To prove the inequality $\mathbb{E}[\![\tau]\!] \le \mathbb{E}[\![\tau']\!]$, we prove its DCFHT version $\mathbb{E}[\![\tau_0 \mid \xi_0 \sim \pi_0]\!] \le \mathbb{E}[\![\tau'_0 \mid \xi'_0 \sim \pi'_0]\!]$ by induction on the number of time steps, where *op* is applied instead of *op*$'$. Note that $\pi_0 = \pi'_0$ for the random generation of solutions.

(a) Initialization. We prove by Eq. 1 that $\mathbb{E}[\![\tau^1_0 \mid \xi^1_0 \sim \pi_0]\!] \le \mathbb{E}[\![\tau'_0 \mid \xi'_0 \sim \pi'_0]\!]$ as follows. Since *op* is applied only at $t = 0$, we have

$$\begin{aligned}
&\mathbb{E}[\![\tau^1_0 \mid \xi^1_0 \sim \pi_0]\!] = \sum\nolimits_{x\in\mathcal{X}} \pi_0(x)\mathbb{E}[\![\tau^1_0 \mid \xi^1_0 = x]\!] \\
&= 1 - \pi_0(\mathcal{X}^*) + \sum\nolimits_{x,y\in\mathcal{X}} \pi_0(x)P(\xi^1_1 = y \mid \xi^1_0 = x)\mathbb{E}[\![\tau^1_1 \mid \xi^1_1 = y]\!] \\
&= 1 - \pi_0(\mathcal{X}^*) + \sum\nolimits_{x,y\in\mathcal{X}} \pi_0(x)P(\xi^1_1 = y \mid \xi^1_0 = x)\mathbb{E}[\![\tau'_1 \mid \xi'_1 = y]\!] \\
&\le 1 - \pi'_0(\mathcal{X}^*) + \sum\nolimits_{x,y\in\mathcal{X}} \pi'_0(x)P(\xi'_1 = y \mid \xi'_0 = x)\mathbb{E}[\![\tau'_1 \mid \xi'_1 = y]\!] \\
&= \mathbb{E}[\![\tau'_0 \mid \xi'_0 \sim \pi'_0]\!] \,,
\end{aligned}$$

where the first, second and last equations are obtained by Lemma 1, the third equation is obtained Eq. 2, and the followed inequality is obtained by Eq. 1 and by $\pi_0 = \pi'_0$.

(b) Inductive Hypothesis. Assume that at the induction step $K > 0$, $\forall k \le K-1 :$ $\mathbb{E}[\![\tau^k_0 \mid \xi^k_0 \sim \pi_0]\!] \le \mathbb{E}[\![\tau'_0 \mid \xi'_0 \sim \pi'_0]\!]$. Then, at the time step $t = K$, we have

$$\mathbb{E}[\![\tau^K_{K-1} \mid \xi^K_{K-1} \sim \pi_{K-1}]\!]$$

$$
\begin{aligned}
&= 1 - \pi_{K-1}(\mathcal{X}^*) + \sum_{x,y\in\mathcal{X}} \pi_{K-1}(x) P(\xi_K^K = y | \xi_{K-1}^K = x) \mathbb{E}[\![\tau_K^K | \xi_K^K = y]\!] \\
&= 1 - \pi_{K-1}(\mathcal{X}^*) + \sum_{x,y\in\mathcal{X}} \pi_{K-1}(x) P(\xi_K^K = y | \xi_{K-1}^K = x) \mathbb{E}[\![\tau'_K | \xi'_K = y]\!] \\
&\le 1 - \pi_{K-1}(\mathcal{X}^*) + \sum_{x,y\in\mathcal{X}} \pi_{K-1}(x) P(\xi'_K = y | \xi'_{K-1} = x) \mathbb{E}[\![\tau'_K | \xi'_K = y]\!] \\
&= \sum\nolimits_{x\in\mathcal{X}} \pi_{K-1}(x) \mathbb{E}[\![\tau'_{K-1} \mid \xi'_{K-1} = x]\!] \\
&= \sum\nolimits_{x\in\mathcal{X}} \pi_{K-1}(x) \mathbb{E}[\![\tau_{K-1}^{K-1} \mid \xi_{K-1}^{K-1} = x]\!] \\
&= \mathbb{E}[\![\tau_{K-1}^{K-1} \mid \xi_{K-1}^{K-1} \sim \pi_{K-1}]\!] ,
\end{aligned}
$$

where the first and the third equations are obtained by Lemma 1, the second and the fourth equations are obtained by Eq. 2, and the inequality is obtained by Eq. 1. Thus, by the induction hypothesis, we get

$$
\begin{aligned}
&\mathbb{E}[\![\tau_0^K \mid \xi_0^K \sim \pi_0]\!] = K - 1 - \sum\nolimits_{t=0}^{K-2} \pi_t(\mathcal{X}^*) + \mathbb{E}[\![\tau_{K-1}^K \mid \xi_{K-1}^K \sim \pi_{K-1}]\!] \\
&\le K - 1 - \sum\nolimits_{t=0}^{K-2} \pi_t(\mathcal{X}^*) + \mathbb{E}[\![\tau_{K-1}^{K-1} \mid \xi_{K-1}^{K-1} \sim \pi_{K-1}]\!] \\
&= \mathbb{E}[\![\tau_0^{K-1} \mid \xi_0^{K-1} \sim \pi_0]\!] \le \mathbb{E}[\![\tau'_0 \mid \xi'_0 \sim \pi'_0]\!] .
\end{aligned}
$$

(c) Conclusion. From (a) and (b), it holds that $\mathbb{E}[\![\tau_0^\infty \mid \xi_0^\infty \sim \pi_0]\!] \le \mathbb{E}[\![\tau'_0 \mid \xi'_0 \sim \pi'_0]\!]$, i.e., $\mathbb{E}[\![\tau^\infty]\!] \le \mathbb{E}[\![\tau']\!]$. Finally, since $\mathbb{E}[\![\tau]\!]$ is finite, we have $\mathbb{E}[\![\tau]\!] = \mathbb{E}[\![\tau^\infty]\!]$, and thus $\mathbb{E}[\![\tau]\!] \le \mathbb{E}[\![\tau']\!]$. □

4 Analysis of LeadingOnes Problem

4.1 The Evolutionary Algorithm

We study the (2:2)-EA as in Definition 1, which is generalized with the minimum difference from the (1+1)-EA [6] for enabling recombination. The (2:2)-EA uses the population size two and a *direct offspring selection* (i.e., selecting between a parent and its direct offspring, so denoted as '2:2' instead of '2+2'). This helps to isolate the influences of population size and selection pressure that are beyond the scope of this paper. Note that as used in [19, 15], though the population size of two is small, it is sufficient for showing the effect of recombination operators.

Definition 1 ((2:2)-EA). *Encode each solution by a string with n binary bits, and let every population, denoted by variable ξ, contain 2 solutions. The (2:2)-EA consists of the following steps:*

1. (Initialization) Let $t \leftarrow 0$.
$\xi_0 :=$ *randomly generated population.*
2. Let $(s_1, s_2) := \xi_t$.
3. (Reproduction) If $UseRecombination$ *is true,*
$(s_1^R, s_2^R) :=$ *M&R*(s_1, s_2)
4. Else, $(s_1^R, s_2^R) :=$ *Mutation*(s_1, s_2)
5. (Selection) $\xi_{t+1} := \big(\arg\max_{s \in \{s_1, s_1^R\}} f(s), \arg\max_{s' \in \{s_2, s_2^R\}} f(s') \big)$.
6. (Stop Criterion) Terminates if the optima is reached.
7. (Loop) Let $t \leftarrow t + 1$*, goto step 2.*
Mutation : $\mathcal{X} \to \mathcal{X}$ *is a mutation operator, M&R* : $\mathcal{X} \times \mathcal{X} \to \mathcal{X} \times \mathcal{X}$ *is the strategy of combining mutation and recombination operators, and* $UseRecombination$ *is a switching parameter. Note that, when a parent has the same fitness as its offspring in selection, the parent is selected.*

We describe several operators below. Note that, when using recombination operators, two parents generate two offsprings by exchanging some bits. To apply the *direct offspring selection*, we need identify which of the two generated solutions is the *direct offspring* of a parent. This is realized by considering which of the two generated solutions is closer to the parent, i.e., with smaller Hamming distance (ties are broken randomly).

- **One-bit mutation.** For each solution, randomly choose one of the n bits and flip (0 to 1 or inverse) the chosen bit.
- **First-bit recombination.** For the two current solutions, scan the solutions from left to right, and exchange the first different bits.
- **One-bit recombination.** For the two current solutions, randomly choose one of the n positions and exchange the bits on that position.

4.2 Analysis

The LeadingOnes problem is given in Definition 2. Denote $s(j)$ be the j-th bit of solution s counting from left to right (thus $s(1)$ is the left-most bit). We define $LO(s) = \{\max i; s.t. \forall j \leq i, s(j) = 1\}$ and $\delta(s_1, s_2) = \|s_1\| - \|s_2\|$, where $\|\cdot\|$ denotes the 1-norm, i.e., the number of one bits.

Definition 2 (LeadingOnes Problem). *LeadingOnes Problem of size* n *is to find an* n *bits binary string* s^* *such that* $s^* = \arg\max_{s \in \{0,1\}^n} LO(s)$.

It is easy to find out that the optimal solution of the LeadingOnes problem is $s^* = (1, 1, \cdots, 1)$, which has n leading one bits, i.e., $LO(s^*) = n$; also note that $LO((0, 1, \cdots, 1)) = 0$ since the leading bit is zero. This problem is one of the most widely studied problems for EAs using mutation operators [16], however, previous analysis on this problem did not touch recombination operators.

In the following, we analyze (2:2)-EA on LeadingOnes problem with Mutation implemented by one-bit mutation and M&R implemented by four strategies described below. The function LO is used as the fitness function in (2:2)-EA. For the two solutions (s_1, s_2) in the current population such that $\delta(s_1, s_2) \geq 0$, we simply denote $LO_1 = LO(s_1)$, $LO_2 = LO(s_2)$, and $\delta = \delta(s_1, s_2)$.

– **M&R1a.** Use the first-bit recombination if either $LO_1 < LO_2$ holds or both $\delta = 0$ and $LO_1 \neq LO_2$ hold; otherwise use the one-bit mutation.
– **M&R1b.** Use the first-bit recombination if both $LO_1 > LO_2$ and $0 < \delta \leq 2$ hold; otherwise use the one-bit mutation.
– **M&R1.** Use the first-bit recombination if either M&R1a or M&R1b holds; otherwise use the one-bit mutation.
– **M&R2.** Use the one-bit recombination if both $LO_1 < LO_2$ and $s_1(LO_2 + 1) = 0$ hold, or both $LO_1 > LO_2$ and $s_2(LO_1 + 1) = 0$ hold; otherwise use the one-bit mutation.

Our analysis below shows that M&R1a and M&R1b are both superior to one-bit mutation (Propositions 2 and 3), and the combination of the two strategies, M&R1, is also superior to one-bit mutation (Proposition 4). However, M&R2 is inferior to one-bit mutation (Proposition 5).

In the following, unless stated, we let $\{\xi_t\}_{t=0}^{+\infty}$ correspond to the (2:2)-EA using M&R, and let $\{\xi'_t\}_{t=0}^{+\infty}$ correspond to the (2:2)-EA using Mutation. Thus, $\mathbb{E}[\![\tau]\!]$ and $\mathbb{E}[\![\tau']\!]$ denote the EFHT of (2:2)-EA using M&R and Mutation, respectively. We denote $\mathbb{E}(i,j)$ as the CFHT $\mathbb{E}[\![\tau'_t | \xi'_t = \{s_1, s_2\}]\!]$ of EA using one-bit mutation, given that $\|s_1\| = n - i$ and $\|s_2\| = n - j$. We present proof sketches below due to space limit, and full proof will be provided in a longer version.

Proposition 1. *The CFHT of* $\{\xi'_t\}_{t=0}^{+\infty}$ *satisfies that*
$\forall i \geq 1, \delta \geq 0 : \frac{n}{2} < \mathbb{E}(i, i+\delta) - \mathbb{E}(i-1, i+\delta) \leq n - \frac{3n-1}{2^{\delta+3}}$,
and $\forall i \geq 1, \delta \geq 1 : \frac{n}{2^{\delta+2}} \leq \mathbb{E}(i, i+\delta) - \mathbb{E}(i, i+\delta-1) < \frac{n}{2}$.

Proof. It is easy to prove the proposition by induction on $i + i + \delta$. □

Proposition 2. *Given M&R in (2:2)-EA being implemented by M&R1a, when* $n \geq 2$, *we have* $\mathbb{E}[\![\tau]\!] \leq \mathbb{E}[\![\tau']\!]$.

Proof. For any population $x = \{s_1, s_2\}$ such that $\|s_1\| = n - i$ and $\|s_2\| = n - i - \delta(\delta \geq 0)$:
a) in the case $\delta = 0 \wedge LO_1 \neq LO_2$ or $LO_1 < LO_2$, the first-bit recombination and the selection reproduce two solutions $\{s'_1, s'_2\}$ such that $\|s'_1\| = n - i + 1$ and $\|s'_2\| = n - i - \delta$. We have, using Proposition 1,

$$\sum\nolimits_{y \in \mathcal{X}} P(\xi_{t+1} = y \mid \xi_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!]$$
$$= \mathbb{E}(i-1, i+\delta) < \mathbb{E}(i, i+\delta) - \frac{n}{2} \leq \mathbb{E}(i, i+\delta) - 1.$$

b) Otherwise, it uses the one-bit mutation, which yields

$$\sum\nolimits_{y \in \mathcal{X}} P(\xi_{t+1} = y \mid \xi_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!]$$
$$= \sum\nolimits_{y \in \mathcal{X}} P(\xi'_{t+1} = y \mid \xi'_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!] = \mathbb{E}(i, i+\delta) - 1.$$

Thus, for any population x, it holds that

$$\sum\nolimits_{y \in \mathcal{X}} P(\xi_{t+1} = y \mid \xi_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!]$$

$$\leq \mathbb{E}(i, i+\delta) - 1 = \sum_{y \in \mathcal{X}} P(\xi'_{t+1} = y \mid \xi'_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!].$$

By Theorem 1, we immediately have $\mathbb{E}[\![\tau]\!] \leq \mathbb{E}[\![\tau']\!]$. □

Remarks. We can observe from the proof of Proposition 2 that how the analysis has be simplified by Theorem 1. By Theorem 1, we need to bound

$$\sum_{y \in \mathcal{X}} P(\xi_{t+1} = y \mid \xi_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!] ,$$

where $P(\xi_{t+1} = y \mid \xi_t = x)$ is the one-step behavior of the EA using M&R, and $\mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!]$ is the CFHT of the EA using Mutation from y. In the proposition, the expression is calculated to be $\mathbb{E}(i-1, i+\delta)$. By the analysis of the EA using Mutation in Proposition 1, we then bound that $\mathbb{E}(i-1, i+\delta) \leq \mathbb{E}(i, i+\delta) - 1$, while

$$\mathbb{E}(i, i+\delta) - 1 = \sum_{y \in \mathcal{X}} P(\xi'_{t+1} = y \mid \xi'_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!] .$$

Thus the condition of Theorem 1 is satisfied.

Proposition 3. *Given M&R in (2:2)-EA being implemented by M&R1b, when* $n \geq 16$*, we have* $\mathbb{E}[\![\tau]\!] \leq \mathbb{E}[\![\tau']\!]$.

Proof. For any population $x = \{s_1, s_2\}$ such that $\|s_1\| = n - i$ and $\|s_2\| = n - i - \delta (\delta \geq 0)$. Note that in the case $0 < \delta \leq 2 \wedge LO_1 > LO_2$ it uses the first-bit recombination. We have

$$\begin{aligned}
&\sum_{y \in \mathcal{X}} P(\xi_{t+1} = y \mid \xi_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!] \\
&= \mathbb{E}(i, i+\delta-1) \leq \mathbb{E}(i, i+\delta) - \frac{n}{2^{\delta+2}} \leq \mathbb{E}(i, i+\delta) - 1 .
\end{aligned}$$

The remaining of the proof is the same as in Proposition 2. □

Since we only need to deal with the one-step transition probability, we get Proposition 4 directly.

Proposition 4. *Given M&R in (2:2)-EA being implemented by M&R1, when* $n \geq 16$*, we have* $\mathbb{E}[\![\tau]\!] \leq \mathbb{E}[\![\tau']\!]$.

Proposition 5. *Given M&R in (2:2)-EA being implemented by M&R2, when* $n \geq 2$*, we have* $\mathbb{E}[\![\tau]\!] \geq \mathbb{E}[\![\tau']\!]$.

Proof. For any population $x = \{s_1, s_2\}$ such that $\|s_1\| = n - i$ and $\|s_2\| = n - i - \delta (\delta \geq 0)$. Note that in the case $LO_1 < LO_2 \wedge s_1(LO_2 + 1) = 0$, it uses the one-bit recombination. So we have

$$\begin{aligned}
&\sum_{y \in \mathcal{X}} P(\xi_{t+1} = y \mid \xi_t = x) \mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1} = y]\!] \\
&= \big(\mathbb{E}(i-1, i+\delta) + (n-1)\mathbb{E}(i, i+\delta)\big)/n \\
&\geq \mathbb{E}(i, i+\delta) - \frac{1}{n}(n - \frac{3n-1}{2^{\delta+3}}) \\
&= \mathbb{E}(i, i+\delta) - 1 + \frac{3 - 1/n}{2^{\delta+3}} \geq \mathbb{E}(i, i+\delta) - 1 .
\end{aligned}$$

While in the case $LO_1 > LO_2 \wedge s_2(LO_1+1)=0$, it also uses the one-bit recombination. We have

$$\begin{aligned}
&\sum\nolimits_{y\in\mathcal{X}} P(\xi_{t+1}=y \mid \xi_t=x)\mathbb{E}[\![\tau'_{t+1} \mid \xi'_{t+1}=y]\!]\\
&= \big(\mathbb{E}(i,i+\delta-1)+(n-1)\mathbb{E}(i,i+\delta)\big)/n\\
&\geq \mathbb{E}(i,i+\delta) - \frac{5+1/n}{8} > \mathbb{E}(i,i+\delta)-1\ .
\end{aligned}$$

The remaining of the proof is similar to that in Proposition 2. □

To verify the theoretical results, we run the (2:2)-EA on the LeadingOnes problem with problem size ranging up to 100. On each size, we repeat independent runs of each implementation of the EA for 1,000 times, and then average the runtimes as an estimate of the EFHT. The results are plotted in Figure 1. It can be observed that the runtime of both M&R1a and M&R1b is smaller than that using one-bit mutation, and the two strategies have similar runtime such that their curves overlap largely. It is also observable that M&R1, which combines M&R1a and M&R1b, is more efficient than one-bit mutation, as well as M&R1a and M&R1b. Meanwhile, M&R2 is worse than one-bit mutation. These observations verify our analysis results in Propositions 2-5.

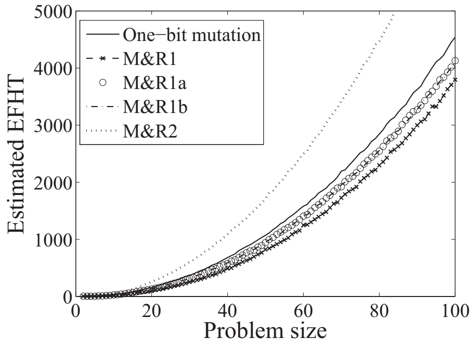

Fig. 1. Estimated EFHT with problem size $[2, 100]$.

5 Conclusion

It is difficult to analyze recombination operators theoretically in terms of runtime, since they operate on population and interact with mutation operators. In this paper, we present a general approach which allows to compare the runtime of an EA turning the recombination on and off. The key is the *Markov Chain Switching Theorem* which compares two Markov chains for the first hit of the target.

For the simplicity of analysis, in this paper we only present a case study of a simple EA on the LeadingOnes problem to show how the proposed approach can be helpful. We will present the analysis on the OneMax problem in a longer version. For more realistic EAs that use uniform mutation and uniform recombination, since one-step of operation could generate many different solutions, compact analysis using our approach is an interesting future work.

Acknowledgements

We want to thank the anonymous reviewers for their helpful comments and suggestions. This research was supported by the Jiangsu Science Foundation (BK2008018) and the Jiangsu 333 High-Level Talent Cultivation Program.

References

1. Bäck, T.: Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford University Press, Oxford, UK (1996)
2. Beyer, H.G., Schwefel, H.P., Wegener, I.: How to analyse evolutionary algorithms. Theoretical Computer Science 287(1), 101–130 (2002)
3. Chen, T., Tang, K., Chen, G., Yao, X.: On the analysis of average time complexity of estimation of distribution algorithms. In: Proceedings of CEC'07. pp. 25–28. Singapore (2007)
4. Doerr, B., Happ, E., Klein, C.: Crossover can provably be useful in evolutionary computation. In: Proceedings of GECCO'08. pp. 539–546. Atlanta, GA (2008)
5. Doerr, B., Theile, M.: Improved analysis methods for crossover-based algorithms. In: Proceedings of GECCO'09. pp. 247–254. Montreal, Canada (2009)
6. Droste, S., Jansen, T., Wegener, I.: A rigorous complexity analysis of the (1 + 1) evolutionary algorithm for linear functions with boolean inputs. Evolutionary Computation 6(2), 185–196 (1998)
7. Droste, S., Jansen, T., Wegener, I.: On the analysis of the (1+1) evolutionary algorithm. Theoretical Computer Science 276(1-2), 51–81 (2002)
8. Freĭdlin, M.I.: Markov Processes and Differential Equations: Asymptotic Problems. Birkhäuser Verlag, Basel, Switzerland (1996)
9. He, J., Yao, X.: Drift analysis and average time complexity of evolutionary algorithms. Artifical Intelligence 127(1), 57–85 (2001)
10. He, J., Yao, X.: Towards an analytic framework for analysing the computation time of evolutionary algorithms. Artificial Intelligence 145(1-2), 59–97 (2003)
11. Jansen, T., Wegener, I.: The analysis of evolutionary algorithms - a proof that crossover really can help. Algorithmica 34(1), 47–66 (2002)
12. Jansen, T., Wegener, I.: Real royal road functions - where crossover provably is essential. Discrete Applied Mathematics 149, 111–125 (2005)
13. Lehre, P.K., Yao, X.: Crossover can be constructive when computing unique input output sequences. In: Proceedings of SEAL'08. pp. 595–604. Melbourne, Australia (2008)
14. Lin, G., Yao, X.: Analysing crossover operators by search step size. In: Proceedings of CEC'97. pp. 107–110. Indianapolis, IN (1997)
15. Richter, J.N., Wright, A., Paxton, J.: Ignoble trails - where crossover is provably harmful. In: Proceedings of PPSN X. pp. 92–101. Dortmund, Germany (2008)
16. Rudolph, G.: Convergence Properties of Evolutionary Algorithms. Verlag Dr. Kovač, Hamburg, Germany (1997)
17. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd ed. Pearson Education, Englewood Cliffs, NJ. (2003)
18. Spears, W.: Evolutionary Algorithms: The Role of Mutation and Recombination. Springer, Berlin (2000)
19. Storch, T., Wegener, I.: Real royal road functions for constant population size. Theoretical Computer Science 320(1), 123–134 (2004)
20. Yu, Y., Zhou, Z.H.: A new approach to estimating the expected first hitting time of evolutionary algorithms. Artificial Intelligence 172(15), 1809–1832 (2008)