Understanding the Co-evolutionary Solution Exchange of the MOEA/D: Already the Smallest Communication Radius Gives the Best-Possible Speed-Up

Benjamin Doerr[1], **Moaad Elmoutassim**[2], **Martin S. Krejca**[1], **Guy Mushiata Kibulu**[2]
[1] Laboratoire d'Informatique (LIX), CNRS, École Polytechnique,
Institut Polytechnique de Paris, Palaiseau, France
[2] École Polytechnique, Institut Polytechnique de Paris, Palaiseau, France
firstname.all-lastnames@polytechnique.edu

Abstract

Besides domination-based algorithms such as the NSGA-II, the decomposition-based MOEA/D is one of the most popular and successful optimization heuristic for multi-objective problems. It decomposes the problem into several, single-objective subproblems. By solving these in a co-evolutionary manner and archiving non-dominated solutions, it approximates or computes the Pareto front. Interestingly, so far no advantages from the co-evolutionary communication of subproblem solutions were proven. In this first mathematical runtime analysis addressing this research question, we investigate how the mutation-based MOEA/D solves the classic OMM benchmark when solutions are exchanged among subproblems in different distances r. We show, for the first time, that the MOEA/D profits from an exchange of solutions, and that already very small communication radii r are sufficient to maximally profit from this feature of the algorithm. More precisely, for all values $r \geq 1$ we prove a tight runtime guarantee of $\Theta(n^2)$, where n is the problem size. This is asymptotically faster than the known bound of $O(n^2 \log n)$ guarantee for $r = 0$, which we complement with a matching bound of $\Omega(n^2 \log n)$.

1 Introduction

Many real-world problems require the optimization of multiple competing objectives. This naturally leads to various, mutually incomparable, optimal trade-offs, known as *Pareto optima*, where improving one objective necessarily reduces the quality of another objective. Thus, the overall optimization goal is to quickly find as many Pareto optima as possible. To this end, multi-objective evolutionary algorithms (MOEAs) are widely and very successfully applied [ZQL$^+$11, CLvV07]. Their population-based approach of maintaining a set of currently best-so-far solutions lends itself well to the multi-objective setting, and their general-purpose nature allows for great flexibility.

State-of-the-art MOEAs can be classified into two main approaches. Using Pareto dominance as main selection criterion and then a secondary criterion ensuring a balanced coverage of the Pareto front is the strategy of the NSGA-II [DPAM02]—the most popular MOEA, with over $60{,}000$ citations on Google Scholar. Other popular MOEAs using this general strategy include the NSGA-III [DJ14], SMS-EMOA [BNE07], and SPEA2 [ZLT01]. All these algorithms are intensively used in practice. Since more recently, they are also subject to intensive theoretical research, see [ZD23, WD23, BZLQ23, RBLQ24] and follow-up works.

An alternative and more recent strategy is to consider a decomposition of the objective function f into several less complex subproblems, and obtain a good set of solutions for f by optimizing these. The most popular MOEA making use of this approach is the *decomposition-based MOEA*

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

(MOEA/D) [ZL07], which decomposes f into a user-specified number of single-objective subproblems. The MOEA/D proceeds by optimizing these subproblems instead of f directly. However, the best-so-far solutions of each subproblem are stored in a shared archive that stores all non-dominated solutions according to the original objective function f, thus ensuring that high-quality solutions for f are obtained. The subproblems themselves are optimized in a co-evolutionary fashion, where each subproblem exchanges its best-so-far solution with all subproblems of a user-specified *communication radius* r. The intention of r is to benefit from potential positive correlations among the different subproblems.

The MOEA/D has been subject to rigorous mathematical investigation for considerably longer than dominance-based MOEAs [LZZZ16], however, the theoretical understanding of this algorithm seems less developed (see Section 2 for a more detailed discussion). In particular, we have to this day no theoretical results on the necessity of or benefits from the co-evolutionary solution exchange in the MOEA/D. All theoretical results so far conduct their analysis as if there was no exchange of solutions. Hence this central aspect of the MOEA/D, distinguishing it from a simple parallel optimization of the subproblems, is not at all understood via theoretical means by now.

Our Contribution. We prove tight bounds for how the expected runtime of the MOEA/D depends on the choice of its communication radius r. As our main result, we show that already a radius $r = 1$ leads to asymptotic speed-ups when compared to a trivial communication radius of $r = 0$. In particular, as we discuss at the end of Section 2, the guarantees that we obtain with non-trivial communication are better than that of various state-of-the-art MOEAs based on non-dominated sorting, when using a comparable parametrization. Interestingly, we obtain the same runtime guarantees for all $r \geq 1$, that is, increasing r further does not lead to further speed-ups when waiting for the entire Pareto front to be witnessed. However, for achieving partial progress, we show that the runtime scales linearly in $\frac{1}{\sqrt{r}}$, the details of which we discuss at the beginning of Section 4.

In more detail, we study how the mutation-only MOEA/D optimizes the bi-objective OneMinMax (OMM) benchmark of size n, when using the Chebyshev decomposition into $n + 1$ subproblems with uniform weights. Hence, each Pareto optimum is represented by the optimum of exactly one subproblem. We further assume that the reference point is initially set to the ideal value of (n, n). This is a standard setting for runtime results for the MOEA/D.

In our main result (Theorem 1), we show that if the MOEA/D uses a communication radius of $r \geq 1$, it exhibits an expected runtime of $\Theta(n^2)$ function evaluations until its archive witnesses the entire Pareto front of OMM for the first time. This result improves the previously best known bound of $O(n^2 \log n)$ [LZZZ16] by a factor of $O(\log n)$, and our bound is asymptotically tight. Our result follows from a careful analysis of the exchange of solutions among subproblems. We show that an extremal subproblem propagates its solution quickly to sufficiently many other subproblems. In turn, this increases the probability to pick a suitable subproblem for finding an improving solution. The amount of how many subproblems can actually benefit from such an exchange is limited by the structure of OMM, which is why the expected runtime is the same for all communication radii greater or equal to 1. However, the benefit of larger search radii is observable to some extent, before this capping occurs.

We prove Theorem 1 for both a uniform and cyclical way of selecting which subproblem to choose next in an iteration. This shows that the exact choice in which communication occurs is not relevant. Moreover, we note that we prove the upper and lower quadratic bound separately (in Theorems 2 and 6), for more general types of mutation, of which standard bitwise mutation—the most widely applied type of mutation—is a special case (Lemma 7).

We complement our quadratic bound in Corollary 8 with a bound of $\Theta(n^2 \log n)$ for a trivial communication radius of $r = 0$. This result follows directly by combining known results in the literature for similar processes. It shows that the speed-up for $r \geq 1$ is, indeed, a result of the communication radius, since all other parameters remain the same as in the setting for $r \geq 1$.

Outline. We discuss related work in Section 2. In Section 3, we introduce our mathematical notation and terminology, and we present the MOEA/D. In Section 4, we discuss and prove our tight $\Theta(n^2)$ runtime bound for non-trivial communication radii. We complement this result in Section 5 with a $\Theta(n^2 \log n)$ runtime bound for trivial radii. Last, we conclude our paper in Section 6.

2 Related Work

The MOEA/D has been studied mathematically from several angles so far. However, none of these results studied the impact of the communication radius on the runtime. In fact, none of the papers take the communication among subproblems into account when conducting a runtime analysis.

The first runtime analysis of the MOEA/D was conducted by Li et al. [LZZZ16]. Besides other results, the authors prove an $O(n^2 \log n)$ bound for the expected runtime on the OMM benchmark[1] in the same setting as in this article.

Huang and Zhou [HZ20] study the MOEA/D with contiguous hypermutation, which is a non-standard mutation operator from artificial immune systems that performs larger changes more frequently than standard operators. The authors study a general decomposition of the objective functions into N subproblems, where N is a user-specified value. Important for the comparison to our work, Huang and Zhou [HZ20] prove that the MOEA/D with contiguous hypermutation optimizes the OMM benchmark in $O(Nn^2 \log n)$ expected time. Hence, even for a minimal value of $N = 2$, their upper bound only matches the upper bound of Li et al. [LZZZ16] for the classic MOEA/D.

Huang et al. [HZC$^+$21] consider the MOEA/D with standard bitwise mutation and with the slightly problem-specific one-point crossover, also in the setting of a general decomposition into N subproblems. Relevant to our article, the authors prove an $O(Nn \log n)$ bound for the OMM benchmark. For the minimal choice of $N = 2$, this results in an $O(n \log n)$ bound, which is faster than the traditional $O(n^2 \log n)$ by Li et al. [LZZZ16] by a factor of $O(n)$. The reason for this speed-up is that the crossover operator allows to easily find all Pareto optima once the extremal solutions of the Pareto front have been found. While impressive, this work highlights the benefit of crossover, and it does not focus on communication among the solutions.

Similar to the result by Huang et al. [HZC$^+$21], Doerr, Krejca, and Weeks [DKW24] also study the MOEA/D with a general decomposition into $N + 1$ subproblems, studying only the OMM benchmark. The authors analyze the impact of mutation, studying both standard bitwise mutation and fast mutation, the latter of which performs changes following a power law with exponent $\beta > 1$. They prove for the version with bitwise mutation an expected $O(nN \log n + n^{n/(2N)} N \log n)$ runtime[2] and for the version with fast mutation an $O(nN \log n + n^\beta \log n)$ expected runtime. As with the previous results, this result highlights how different operators can lead to a far higher tolerance with respect to the number of problem decompositions, in particular making the setting $N = 1$ feasible. Nonetheless, the communication radius is not discussed.

Huang et al. [HZLL21] study the mutation-only MOEA/D for problems with $m \geq 2$ objectives. They are interested in the first time that the algorithm finds any optimal solution, which is very different from the standard setting of witnessing the entire Pareto front at once. The authors consider three different decompositions, into m subproblems each, studying the performance of the MOEA/D on the mLOTZ and mCOCZ benchmarks. The main result of the paper is that the weighted-sum decomposition performs drastically outperforms the other two decomposition methods. This paper explicitly sets the communication radius to the lowest-possible value.

Zhang et al. [ZZZD26] also consider the mutation-only MOEA/D for a flexible number N of subproblems. Similar to Huang et al. [HZLL21], the authors study the impact of the decomposition type, but they consider the standard runtime definition, which is the time until the archive witnesses the entire Pareto front. Most important to our work, Zhang et al. [ZZZD26] show that with a logarithmic number of subproblems, the MOEA/D with the weighted-sum decomposition has an expected runtime of $O(n \log^2 n)$ on the OMM benchmark, which is considerably faster than the typical $O(n^2 \log n)$ bound. Again, this highlights the importance of the decomposition choice, but the communication radius is not studied. The latest work studying the MOEA/D on classic benchmarks [DKW26] proves that a self-adjusting way to adapt the decomposition can lead to drastic runtime gains for unevenly scaled Pareto fronts.

In the only work on a classic optimization problem, Do et al. [DNNS23] study how the MOEA/D solves multi-objective minimum-weight-base problems. The authors introduce a special decomposition for their problem and prove that it finds the entire Pareto front in an expected time that is fixed-parameter tractable.

[1] The authors erroneously refer to this benchmark as COCZ in their paper.

[2] The authors complement a specific part of this runtime with a loose lower bound that is also exponential in n/N.

Besides the MOEA/D, other MOEAs have been theoretically studied. Most relevant to our work, Doerr and Qu [DQ23] showed that the NSGA-II has a lower bound of $\Omega(Nn \log n)$ on OMM, where $N \geq 4(n+1)$ denotes the population size of the algorithm. Thus, for an optimal choice of N, the algorithm has an $\Omega(n^2 \log n)$ lower bound, which is tight [ZLD22]. The same bound of $\Theta(n^2 \log n)$ also holds for the global SEMO, another MOEA that is commonly studied theoretically [BS24]. We note that while for other algorithms based on non-dominated sorting, when relying solely on mutation, only a comparable $O(n^2 \log n)$ for best-possible parameter values is known [WD24a, WD24b, ZLD22, RBLQ24, WD23, ZD24], even for variants that are more robust in their parameter settings [DIK25, DKS26], there is little reason to believe that these algorithms can optimize OMM faster than $\Omega(n^2 \log n)$ when only using mutation.

Outside the theory domain, some suggestions on the optimal size of communication radius exists, but due to their experimental nature, they are less clear than a typical theoretical result. In the work [ZL07] proposing the MOEA/D, generally the MOEA/D was perceived as not very sensitive to the size of the neighborhood used for the exchange of solutions. However, the authors observed that for a knapsack instance, a large radius was detrimental, whereas for the ZDT1 benchmark, a large radius was preferable. A more detailed study on the influence of the communication radius [IAN13] finds that it depends on the number of objectives, but that generally a small constant fraction of the population size such as 10 to 20 percent gives good results. We are not aware of more detailed results, in particular, we have found none that discuss whether the coevolutionary solution exchange is advantageous at all.

3 Preliminaries

Let $\mathbb{N}$ denote the set of natural numbers, including 0, and let $\mathbb{R}$ be the set of reals. For all $a, b \in \mathbb{R}$, let $[a..b] = [a, b] \cap \mathbb{N}$. For all $m \in \mathbb{N}_{\geq 1}$, $v \in \mathbb{R}^m$, and $i \in [1..m]$, we denote the value of v at component i by v_i.

We investigate *pseudo-Boolean* bi-objective problems, that is, for a given $n \in \mathbb{N}_{\geq 1}$, we consider the *maximization* of *objective functions* $f \colon \{0,1\}^n \to \mathbb{R}^2$. We call each $x \in \{0,1\}^n$ an *individual* and $f(x)$ its *objective value*. We refer to the value of x in the first and second objective, respectively, by $f_1(x)$ and $f_2(x)$. Last, we let $|x|_0$ and $|x|_1$ denote, respectively, the number of 0s and 1s of x.

When we mention objective functions, we assume that the dimension $n \in \mathbb{N}_{\geq 1}$ is implicitly given, and when we use big-O notation, we consider asymptotics in this n.

Multi-Objective Terminology

We compare objective values via the *weak dominance* preorder $\succeq$. For all $u, v \in \mathbb{R}^2$, we say that v *weakly dominates* u, written $v \succeq u$ if and only if $v_1 \geq u_1$ and $v_2 \geq u_2$. In all other cases, we say that u *and* v *are incomparable*.

Given an objective function f of dimension $n \in \mathbb{N}_{\geq 1}$, we use the dominance terminology also when comparing individuals, referring to their objective values. We call each $x \in \{0,1\}^n$ *Pareto-optimal* if and only if for all $y \in \{0,1\}^n$, we have that $x \succeq y$ or that x and y are incomparable. If x is Pareto-optimal, we call $f(x)$ a *Pareto optimum*. The set of all Pareto optima of f is its *Pareto front*.

The OneMinMax Benchmark

The OneMinMax (OMM) benchmark [GL10] is the most widely studied objective function for multi-objective evolutionary algorithms, and it is typically the first function that is considered when conducting a new type of analysis, as we are doing.

As the name suggests, OMM aims at both minimizing and maximizing the number of 1s. Formally, $\mathrm{OMM} \colon \{0,1\}^n \to [0..n]^2$ is defined as $x \mapsto (|x|_0, |x|_1)$. Note that each individual is Pareto-optimal, and the Pareto front is $\{(k, n-k) \mid k \in [0..n]\}$, that is, it has $n+1$ objective values.

The MOEA/D Algorithm

The *decomposition-based multi-objective evolutionary algorithm* (MOEA/D, Algorithm 1) [ZL07] decomposes a given objective function f via a user-defined number $N \in \mathbb{N}_{\geq 1}$ into $N+1$ single-objective optimization problems $(g_k)_{k \in [0..N]}$. Each subproblem is subject to *minimization*. Moreover,

Algorithm 1: The MOEA/D [ZL07] maximizing an objective function $f\colon \{0,1\}^n \to \mathbb{R}^2$, given decomposition number $N \in \mathbb{N}_{\geq 1}$, subproblems $\{g_k\}_{k\in[0..N]}$, reference point $z \in \mathbb{R}^2$, and communication radius $r \in \mathbb{N}_{\geq 1}$.

```
P ← ∅;
for k ∈ [0..N] do
    choose x^(k) uniformly at random from {0,1}^n;
    P ← P ∪ {x^(k)};
remove all elements from P dominated by other elements in P;
while stopping criterion is not met do
    choose a subproblem k ∈ [0..N];
    y ← create offspring from x^(k) via bitwise mutation;
    for i ∈ [k − r..k + r] ∩ [0..N] do
        if g_i(y, z) ≤ g_i(x^(i), z) then x^(i) ← y;
    remove all elements weakly dominated by y from P and add y to P if it is not weakly
      dominated by an element of P;
return P;
```

for each subproblem $k \in [0..N]$, a single best-so-far individual $x^{(k)}$ is maintained, each of which is initialized uniformly at random. Last, the MOEA/D maintains an archive P of best-so-far individuals.

Iteratively, the MOEA/D selects in a user-defined manner a subproblem $k \in [0..N]$ and attempts to improve $x^{(k)}$ by creating a new individual y, called *offspring*. If $g_k(y)$ is at least as good as $g_k(x^{(k)})$, then $x^{(k)}$ is replaced by y. Moreover, y is also communicated to all subproblems whose index is within a certain user-defined radius $r \in \mathbb{N}_{\geq 1}$ of k, and their best-so-far individual is also potentially updated. Finally, the archive is updated with respect to y.

Subproblem definition. We consider the *Chebyshev* decomposition of problem f, which is one of the most common MOEA/D decompositions. Given a weight vector $\lambda \in [0,1]^{N+1}$ in increasing order as well as a reference point $z^* \in \mathbb{R}^2$, the Chebyshev decomposition aims at minimizing the weighted L_∞-distance of a given objective value to the reference point, where each subproblem receives a different weight. Formally, for each $k \in [0..N]$, subproblem k is defined for all $x \in \{0,1\}^n$ as

$$g_k(x,z) = \min\{\lambda_k \cdot |f_1(x) - z_1^*|, (1-\lambda_k) \cdot |f_2(x) - z_2^*|\}.$$

In our analysis of OMM, we choose $z^* = (n,n)$ to reflect the best-possible value in each objective. Moreover, we use a uniform weight vector, that is, for all $k \in [0..N]$, we choose $\lambda_k = \frac{k}{N}$. We note that, together with the assumption $N = n$, we thus obtain for all $k \in [0..N]$ and all $x \in \{0,1\}^n$ that

$$g_k(x) = \min\left\{\frac{k}{n}(n - |x|_0), \frac{n-k}{n}(n-|x|_1)\right\} = \begin{cases} \frac{k(n-|x|_0)}{n} & \text{if } |x|_0 < k, \text{ and} \\ \frac{|x|_0(n-k)}{n} & \text{if } |x|_0 \geq k. \end{cases} \tag{1}$$

In particular, this implies for each $k \in [0..n]$ that all individuals with exactly k 0s are minimizers for g_k, as both terms in the minimum are equal. Hence, each subproblem corresponds to exactly one Pareto optimum of OMM.

Subproblem selection. We consider two types of selecting a subproblem: uniformly at random and cyclically, the former of which is self-explanatory.

When choosing a subproblem cyclically, the algorithm chooses consecutively a subproblem from $[0..N]$, starting in the first iteration with subproblem 0, and going after subproblem N back to 0.

Bitwise mutation. Given an individual x as input (the *parent*), bitwise mutation creates an offspring y by first copying x and then deciding independently for each position $i \in [1..n]$ of y whether to flip y_i; each flip occurs with probability $\frac{1}{n}$.

Runtime on OMM

We define the *runtime* of the MOEA/D with decomposition number $N = n \in \mathbb{N}_{\geq 1}$ on OMM of problem dimension n to be the random number of *function evaluations* (of OMM) until the entire

Pareto front of OMM is witnessed by the archive P of the MOEA/D, that is, $\{(k, n-k) \mid k \in [0..n]\} \subseteq \{\text{OMM}(x) \mid x \in P\}$.

We assume that each individual created by the MOEA/D is evaluated once upon its creation and then never again. Hence, since the MOEA/D creates $n+1$ individuals initially and then exactly one offspring in each iteration, its runtime is $n+1$ plus the number of iterations until termination.

4 Tight Runtime Guarantees for Non-Trivial Communication Radii

Our main result is Theorem 1 below, which proves a tight expected runtime of $\Theta(n^2)$ function evaluations for the MOEA/D on OMM when using a communication radius of at least 1.

Theorem 1. *Let $n \in \mathbb{N}_{\geq 1}$. Consider the MOEA/D (Algorithm 1) with decomposition number n, Chebyshev decomposition, reference point (n, n), communication radius $r \in [1..n]$, and either uniform or cyclical subproblem selection, optimizing* OMM. *Then the expected runtime is $\Theta(n^2) = O\left(\frac{n^2}{\sqrt{r}} + n^2\right) \cap \Omega(n^2)$ function evaluations.*

Theorem 1 shows a speed-up of order $\ln n$ compared to the same setting but with a trivial communication radius of 0. The reason for this speed-up is that individuals with a currently best-so-far value in one objective are successively communicated to neighboring subproblems, for which these individuals are also best-possible at that time. This accumulation of such extremal individuals improves the rate at which an individual is chosen that has a decent chance to expand the Pareto front. In contrast, for communication radius 0, each subproblem needs to find such an extremal individual by itself.

More formally[3], we consider subproblem 0 and assume that its number of 0s is $k \in [1..\frac{n}{2}]$. Except for at most once during the optimization, if subproblem 0 has a value of at most k, so do subproblems 1 to $\min\{k, r\} =: M$, where the cap at k is due to the selection preference stated by Lemma 3. We call an individual with fewer than k 0s an *improving* individual, for this discussion. When choosing subproblems 0 to M for offspring creation and not creating an improving individual but instead a copy, that is, an individual with exactly k 0s, this individual is the new best-so-far individual for at least subproblem $M+1$. Analogously, subproblems 1 to $M+1$, if chosen, communicate an individual with exactly k 0s with constant probability to at least subproblem $M+2$, if it exists, and so on. Since each subproblem is chosen with probability $\frac{1}{n}$, the communication spreads, in expectation, to $\ell \in [1..k]$ subproblems in $\ell\frac{n}{r}$ iterations. Since creating an improving offspring from an individual with exactly k 0s has a probability of order $\frac{k}{n}$, if $\ell \in [1..k]$ subproblems in the population have an individual with exactly k 0s, the probability to create an improving individual is of order $k\ell/n^2$ and thus takes $n^2/(k\ell)$ iterations in expectation. Hence, if $\ell \approx \sqrt{nr/k}$, then these two expected times coincide, resulting in finding an improving solution in order $n^{3/2}/\sqrt{rk}$, provided $k \geq (nr)^{1/3}$. For smaller k, it takes order n^2/k^2 iterations in expectation, as there are not enough subproblems available for which a solution with exactly k 0s is the best-so-far solution while no improvement is found. Due to this limit, larger communication radii do not provide any further asymptotic speed-up when aiming to witness the entire Pareto front. In other words, while there are enough solutions to reach, the upper runtime bound is $O(n^2/\sqrt{r})$, and the algorithm profits from larger search radii. However, at some point in time, this effect is capped, and the remaining runtime is $O(n^2)$, which is independent of r, save for $r \geq 1$.

The lower bound follows a different line of thought. Assume that subproblem 0 does not have an optimal individual yet, but subproblems 1 and 2 do, which we prove to occur very likely. Then all remaining subproblems have a best-so-far solution of at least two 0s and, in particular, discard an individual with exactly one 1. Consequently, only subproblems 0 and 1 have, each, a probability of order $\frac{1}{n}$ to create the optimum for subproblem 0. This results in at least order n^2 iterations in expectation in order to solve subproblem 0. All other subproblems have at most a chance of order $\frac{1}{n^2}$, also resulting in a quadratic lower bound.

In the following, we prove the upper and lower bound of Theorem 1 separately, following the outline above. For each of these bounds, rather than only considering bitwise mutation, we consider a general class of mutation operators, which is always suited to our proof method. Bitwise mutation is just one operator that lies in the intersection of these larger classes.

[3] We assume uniform subproblem selection in this discussion, noting that cyclical selection, when viewed over n consecutive iterations, behaves similarly to uniform selection.

4.1 A Quadratic Upper Bound for Non-Trivial Communication Radii

For the upper bound of Theorem 1, we rely on communicating copies of individuals with currently extremal single-objective values among neighboring subproblems, as explained in Section 4. Our analysis generalizes to a certain type of mutation operators that we call *type-A operators*.

We say that a mutation operator $\pi\colon \{0,1\}^n \to \{0,1\}^n$ is a *type-A operator* if and only if there exist constants $C_A, \alpha \in \mathbb{R}_{>0}$ such that for all $x \in \{0,1\}^n$, we have

- $\Pr(|\pi(x)|_0 = |x|_0 - 1) \geq C_A \frac{|x|_0}{n}$,
- $\Pr(|\pi(x)|_0 = |x|_0 + 1) \geq C_A \frac{n-|x|_0}{n}$, and
- $\Pr(|\pi(x)|_0 = |x|_0) \geq \alpha$.

Our main result is the following.

Theorem 2. *Let $n \in \mathbb{N}_{\geq 1}$. Consider the MOEA/D (Algorithm 1) with decomposition number n, Chebyshev decomposition, reference point (n,n), communication radius $r \in [1..n]$, and either uniform or cyclical subproblem selection, optimizing* OMM. *Assume that the MOEA/D makes use of a type-A mutation operator in line 8. Then the expected runtime is $O\left(\frac{n^2}{\sqrt{r}} + n^2\right)$ function evaluations.*

In order to prove Theorem 2, we define more formally for each subproblem $k \in [0..n]$ when an individual replaces the best-so-far individual of subproblem k.

Let $k \in [0..n]$. For all $x, x' \in \{0,1\}^n$, we say that x is *at least as attractive as x' for subproblem k*—denoted by $x \preceq_k x'$—if and only if $g_k(x) \leq g_k(x')$. Furthermore, if and only if $g_k(x) < g_k(x')$, we say that x is *strictly more attractive* than x' for subproblem k.

The following lemma shows for all subproblems $k \in [0..n]$ how the attractiveness of an individual is related to its distance of its numbers of 0s to k.

Lemma 3. *Let $k \in [0..n]$ and $x, x' \in \{0,1\}^n$. If $|x|_0, |x|'_0 \geq k$, then $x \preceq_k x'$ if and only if $|x|_0 \leq |x'|_0$. Likewise, if $|x|_0, |x|'_0 \leq k$, then $x \preceq_k x'$ if and only if $|x|_0 \geq |x|'_0$.*

Moreover, if $k \leq \frac{\sqrt{n}}{2}$, $|x|_0 \leq k$, and $|x'|_0 > k$, then $x \prec_k x'$.

Proof. First, note that eq. (1) is only depends on the number of 0s of its argument. Hence, for all $y \in \{0,1\}^n$, let $g'_k(|y|_0) \coloneqq g_k(y)$. The first two statements follow then directly from the fact that g'_k is decreasing on $[0..k]$ and increasing on $[k..n]$.

We now consider the case $k \leq \frac{\sqrt{n}}{2}$, $|x|_0 \leq k$, and $|x'|_0 > k$, relying again on the definition of g' as introduced above. Let $j = |x|_0$ and $j' = |x'|_0$, and let $\ell \in \mathbb{N}$ and $r \in \mathbb{N}_{\geq 1}$ such that $j = k - \ell$ and $j' = k + r$. Using the piecewise form of g'_k, we obtain

$$g'_k(j) \leq g'_k(j') \iff \frac{k(n-(k-\ell))}{n} \leq \frac{(k+r)(n-k)}{n} \iff \ell \leq r\,\frac{n-k}{k}.$$

Since $k \leq \frac{\sqrt{n}}{2}$, we have $\frac{n-k}{k} \geq \frac{2n-\sqrt{n}}{\sqrt{n}} > \frac{\sqrt{n}}{2} \geq k \geq \ell$. Hence, since $r \geq 1$, we have $\ell < r\,\frac{n-k}{k}$ and thus $g'_k(k-\ell) < g'_k(k+r)$, concluding the proof. □

Lemma 4. *Consider the same setting as in Theorem 2. Then the expected runtime for solving subproblems 0 and n is $O\left(\frac{n^2}{\sqrt{r}} + n^2\right)$ function evaluations.*

The proof of Lemma 4 follows the same outline as mentioned at the beginning of Section 4. Details are deferred to the appendix. Letting $k \in [1..n]$ denote number of 0s of the best-so-far individual of subproblem 0, we bound the expected time until (1) subproblems 1 to $\ell_k = \min\{\lceil\sqrt{nr/k}\rceil, k\}$ also have a best-so-far solution with exactly k 0s or (2) the minimum number of 0s among these subproblems decreases. By Lemma 3, for each subproblem with index in $[0..\ell_k]$, a solution with exactly k 0s is at least as attractive as a larger value of the subproblem while we did not find an improvement. Since copies are created with a constant probability, by the definition of a type-A operator, and since we inform either all relevant subproblems immediately or at least r of them every

$O(\frac{n}{r})$ iterations on average, it takes $O(\frac{n}{r}\ell_k)$ in expectation in order to propagate a solution with exactly k 0s to ℓ_k subproblems (or to find an improvement earlier).

Given ℓ_k subproblems as above, the expected time for finding an improvement is $O(\frac{n^2}{k\ell_k})$, as we have a probability of at least $\frac{\ell_k}{n}$ of choosing a suitable subproblem, each of which has a probability of at least order $\frac{k}{n}$ to flip exactly one of the k 0s. Substituting ℓ_k and summing over all values for k concludes the proof.

We show that once the extremal subproblems are solved, the remaining subproblems are also solved in at most quadratic time.

Lemma 5. *Consider the same setting as in Theorem 2. Then once the subproblems* 0 *and* n *are solved, the expected runtime for witnessing the entire Pareto front of* OMM *is* $O(n^2)$ *function evaluations.*

Proof. We show that, starting from subproblem 0, we solve all subproblems in $[0..\lceil\frac{n}{2}\rceil]$ in expected time $O(n^2)$. For the other subproblems, an analogous argument applies by swapping all 0s and 1s in the argument, as explained also initially in the proof of Lemma 4. Hence, the overall expected time is at most twice the time that we estimate for the lower half.

We show for all $k \in [0..\lceil\frac{n}{2}\rceil]$ that once subproblem k is solved, we solve subproblem $k+1$ in expected time $O(n)$, leading to an overall expected time of $O(n^2)$ when summing over all values of k. To this end, we use the same terminology as in the proof of Lemma 4, calling the number of 0s of the best-so-far individual of a subproblem its *value*.

Consider an iteration such that $k \in [0..\lceil\frac{n}{2}\rceil - 1]$ is maximal such that for all $i \in [0..k]$, we have that subproblem i is solved. When selecting subproblem k, then by the definition of a type-A operator, mutation produces value $k+1$ with probability at least $C\frac{n-k}{n} \geq \frac{C}{2}$.

For uniform selection, subproblem k is selected, in expectation, in $O(n)$ iterations. For cyclical selection, it is selected for sure in n iterations. Hence, for either subproblem selection, we solve subproblem $k+1$ in expected time $O(n)$, concluding the proof. □

Theorem 2 follows by first applying Lemma 4 and then Lemma 5.

4.2 A Quadratic Lower Bound for Non-Trivial Communication Radii

Similar to Section 4.1, we introduce a general mutation operator, called *type-B operator*, before we state our quadratic lower bound. In contrast to a type-A operator, which requires a decent probability for copies and for exploring the 1-bit neighborhood of an individual, a type-B operator guarantees that the all-1s individual is not created too likely when its parent has at least one 1.

We say that a mutation operator $\pi\colon \{0,1\}^n \to \{0,1\}^n$ is a *type-B operator* if and only if for all $x \in \{0,1\}^n$, we have

- if $|x|_0 = 1$, then $\Pr(|\pi(x)|_0 \neq 0) \geq 1 - O(\frac{1}{n})$,
- if $|x|_0 = 1$, then there is a constant $C_B \in (0,1)$ such that for all $i \in [2..n]$, we have $\Pr(|\pi(x)|_0 \in [2,i] \mid |\pi(x)|_0 \neq 1) \geq 1 - C_B(\frac{1}{i^2} + \frac{1}{n}) > 0$, and
- if $|x|_0 \geq 2$, then $\Pr(|\pi(x)|_0 \geq 1) \geq 1 - O(\frac{1}{n^2})$.

Our main result in this section is the following lower bound.

Theorem 6. *Let* $n \in \mathbb{N}_{\geq 1}$. *Consider the MOEA/D (Algorithm 1) with decomposition number* n*, Chebyshev decomposition, reference point* (n,n)*, communication radius* $r \in [1..n]$*, and either uniform or cyclical subproblem selection, optimizing* OMM. *Assume that the MOEA/D makes use of a type-B mutation operator in line 8. Then the expected runtime is* $\Omega(n^2)$ *function evaluations.*

We prove Theorem 6 by showing that with constant probability, during the first n^2 iterations, the all-1s individual is not created. To this end, we show that with high probability, none of the $n+1$ initial individuals is the all-1s individual. Afterward, if a subproblem $i \in [2..n]$ is selected for the first time and its best-so-far individual has exactly one 1, its offspring has, with constant probability, at least two 1s, which are at least as attractive and thus retained, by Lemma 3. Moreover, we show

that with constant probability, subproblems 0 and 1 are chosen at most $3n$ times during the first n^2 iterations, and each time that they are chosen, their offspring has at least one 0. Last, we show that with constant probability, whenever during the first n^2 iterations a subproblem is selected whose currently best-so-far solution has at least two 0s, its offspring has at least one 0. This shows the claim. Details are deferred to the appendix.

4.3 Proof of the Tight Quadratic Runtime Guarantee

Theorem 1 is a consequence of Theorems 2 and 6 when knowing that bitwise mutation is both a type-A and a type-B operator. We show this with the following statement.

Lemma 7. *For $n \in \mathbb{N}_{\geq 16}$, bitwise mutation is a type-A and a type-B operator.*

Proof. Let $x \in \{0,1\}^n$ be the parent, and let $\pi\colon \{0,1\}^n \to \{0,1\}^n$ denote bitwise mutation. We show the properties of the two types separately.

Type-A operator. In order for the offspring to have one fewer 0 than x, exactly one of the 0s of x needs to flip, and all other bits must not flip. Consequently, we obtain $\Pr(|\pi(x)|_0 = |x|_0 - 1) = \frac{|x|_0}{n}(1-\frac{1}{n})^{n-1} \geq \frac{1}{e}\frac{|x|_0}{n}$. For increasing the number of 0s by one, an analogous reasoning applies, requiring to flip exactly one of the $n - |x|_0$ 1s of x, satisfying both bounds with $C_A = \frac{1}{e} \in \mathbb{R}_{>0}$.

Last, in order to create a copy, no bit of x must flip. Thus, we have $\Pr(|\pi(x)|_0 = |x|_0) = (1-\frac{1}{n})^n \geq \frac{1}{e}(1-\frac{1}{n}) \geq \frac{1}{2e}$, also satisfying the last property.

Type-B operator. If $|x|_0 = 1$, then in order for the offspring not to be the all-1s individual, it is sufficient that the single 0 in x does not flip. Thus, we obtain $\Pr(|\pi(x)|_0 \neq 0) \geq 1 - \frac{1}{n}$, as required.

For the next property, let $i \in [2..n]$, and assume that $|x|_0 = 1$. We estimate the probability that the offspring has a number of 0s in $[2..i]$, conditional on the offspring not being a copy. To this end, let $p_c \coloneqq (1-\frac{1}{n})^n$ denote the probability of creating a copy. Let Z denote the unconditional event that the offspring is the all-1s individual, and let B denote the unconditional event that the offspring has more than i 0s. Then we have $\Pr(|\pi(x)|_0 \in [2,i] \mid |\pi(x)|_0 \neq 1) = \big(1-p_c-\Pr(Z)-\Pr(B)\big)/(1-p_c) = 1 - (\Pr(Z)+\Pr(B))/(1-p_c)$. We continue with bounding the probabilities of Z and B.

For Z, the single 0 in x must flip and none of the other bits. Hence, $\Pr(Z) = \frac{1}{n}(1-\frac{1}{n})^{n-1} = \frac{1}{n(1-1/n)}(1-\frac{1}{n})^n \leq \frac{1}{e}\frac{1}{n-1}$. For B, it is necessary that at least i 1s in x flip. Hence, since $i \geq 2$, we have $\Pr(B) \leq \frac{1}{n^i}\binom{n-1}{i} \leq \frac{(n-1)^i}{n^i}\frac{1}{i!} \leq \frac{1}{i(i-1)} \leq \frac{2}{i^2}$.

By $1-p_c \geq 1-\frac{1}{e}$ and combining the bounds above, we obtain $\Pr(|\pi(x)|_0 \in [2,i] \mid |\pi(x)|_0 \neq 1) \geq 1 - \frac{2}{1-1/e}(\frac{1}{i^2}+\frac{1}{n})$, which is strictly positive for $i \geq 2$ and $n \geq 16$, proving this property.

Last, assume $|x|_0 \geq 2$. In order for the offspring to be the all-1s individual, at least all 0s in x must flip. Thus, we have $\Pr(|\pi(x)|_0 \geq 1) \geq 1 - \frac{|x|_0}{n^{|x|_0}} \geq 1 - \frac{2}{n^2}$, concluding the proof. □

We note that while Lemma 7 requires $n \in \mathbb{N}_{\geq 16}$, the expected runtime of the MOEA/D on OMM for each value $n \in [1..15]$ is constant, thus also proving our claim for these cases.

5 A Super-Quadratic Lower Bound for Trivial Communication Radius

We complement Theorem 1 by showing that its runtime performance is worse by a factor of order $\ln n$ if no communication exchange among the subproblems is used.

Corollary 8. *Let $n \in \mathbb{N}_{\geq 1}$. Consider the MOEA/D (Algorithm 1) with decomposition number n, Chebyshev decomposition, reference point (n,n), communication radius $r = 0$, and either uniform or cyclical subproblem selection, optimizing* OMM. *Then the expected runtime is $\Theta(n^2 \log n)$ function evaluations.*

Corollary 8 follows directly from well-known results. The upper bound is given by Li et al. [LZZZ16, Proposition 4], following from an n-times application of the coupon collector problem. The lower

bound follows from noting that solving subproblem 0 takes $\Omega(n \log n)$ function evaluations if only considering function evaluations spent on this subproblem. This is a special case of the fact that the $(1+1)$ evolutionary algorithm (EA) takes $\Omega(n \log n)$ function evaluations on each objective function with a unique optimum [DJW98, Theorem 1]. Since the MOEA/D with communication radius 0 runs an independent $(1+1)$ EA for each subproblem, this result translated directly to subproblem 0. The overall lower bound follows by noticing that, on average, we require $n+1$ function evaluations in order to perform 1 function evaluation for subproblem 0.

6 Conclusion

We derived tight runtime guarantees for the MOEA/D (Algorithm 1) when optimizing the classic OMM benchmark of size n. Theorem 1 shows that for any communication radius $r \geq 1$, the expected runtime is $\Theta(n^2)$ function evaluations, which improves the previously best bound of $O(n^2 \log n)$ [LZZZ16]. Our improvement is a result of studying the process more carefully, showing that currently best-so-far individuals are propagated to sufficiently many subproblems quickly. This results in a higher chance of selecting a useful subproblem for finding new Pareto-optima. However, as our lower bound shows, the exact size of r, once at least 1, is irrelevant, as the number of subproblems that share the same sub-optimal individuals is limited by the problem structure of OMM.

We contrast Theorem 1 with Corollary 8, showing that if $r = 0$, the expected runtime is $\Theta(n^2 \log n)$. Hence, the speed-up for $r \geq 1$ is, indeed, due to the exchange of individuals among subproblems.

As usual in the mathematical runtime analysis of randomized search heuristics [NW10, AD11, Jan13, ZYQ19, DN20], we were able to analyze rigorously only one particular but commonly studied situation, and for arbitrary value of the parameter r and for a broader class of mutation operators. From the working principles of our proofs, we are optimistic that the co-evolutionary solution exchange is advantageous also in other settings. This is clear when crossover is used with some constant probability $p_c < 1$, simply because our proof is monotonic in the sense that the creation of other individuals at most slows down the proven progress by the time spent on these other individuals. Similarly, when using the PBI decomposition with weight $\theta > 1$, we obtain the same set of subproblems, and hence the same process. The weighted-sum decomposition leads to a particular behavior on the OMM benchmark that is most likely not representative for a typical use of the MOEA/D [ZZZD26]. For the number of subproblems N different from the size $n+1$ of the Pareto front, it is somewhat clear that $N > n+1$ simply implies that some points of the Pareto front are targeted by more than one subproblem. This should give a similar behavior as shown in our work except for that all numbers of individuals are scaled up appropriately. When $N < n+1$, for any value of the communication radius, there is the risk that points of the Pareto front are skipped and then are hard to find. Apart from this, the co-evolutionary solution exchange should work as in our setting, that is, the outermost solution is communicated to a large number of subproblems, which then in parallel search for more outermost solutions.

What is less clear from our work, and an interesting next step is to study, is for which other benchmarks such a speed-up exists and whether the runtime differs for different non-trivial communication radii. A natural next benchmark is LOTZ, for which, different to OMM, only very few individuals are on the Pareto front. Deriving in such a setting a central lemma like Lemma 3, which succinctly summarizes when a subproblem updates its best-so-far individual, seems to be a logical next step.

Another natural question is to study an additional positive effect of crossover. In this setting, since r also determines which subproblems are chosen for the crossover, the exact value of r may play a more important role. In particular, it could be that a small value of r leads to a more suitable parent selection than a uniform parent selection as often used in domination-based MOEAs.

Last, we currently lack mathematical guarantees for the importance of the communication radius for problems with at least three objectives. So far, there is only a single work studying such a setting for the MOEA/D, and it does not consider Pareto front coverage [HZLL21].

Acknowledgments

Martin S. Krejca is supported by the French National Agency for Research (ANR) via the JCJC grant titled MultiEDA (ANR-25-CE23-2418-01). This research is also supported by the FMJH Program Gaspard Monge for optimization and operations research and their interactions with data science.

References

[AD11] Anne Auger and Benjamin Doerr, editors. *Theory of Randomized Search Heuristics*. World Scientific Publishing, 2011.

[BNE07] Nicola Beume, Boris Naujoks, and Michael Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181:1653–1669, 2007.

[BS24] Jakob Bossek and Dirk Sudholt. Runtime analysis of quality diversity algorithms. *Algorithmica*, 86:3252–3283, 2024.

[BZLQ23] Chao Bian, Yawen Zhou, Miqing Li, and Chao Qian. Stochastic population update can provably be helpful in multi-objective evolutionary algorithms. In *International Joint Conference on Artificial Intelligence, IJCAI 2023*, pages 5513–5521. ijcai.org, 2023.

[CLvV07] Carlos Artemio Coello Coello, Gary B. Lamont, and David A. van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, 2nd edition, 2007.

[DIK25] Benjamin Doerr, Tudor Ivan, and Martin S. Krejca. Speeding up the NSGA-II with a simple tie-breaking rule. In *Conference on Artificial Intelligence, AAAI 2025*, pages 26964–26972. AAAI Press, 2025.

[DJ14] Kalyanmoy Deb and Himanshu Jain. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, 18:577–601, 2014.

[DJW98] Stefan Droste, Thomas Jansen, and Ingo Wegener. A rigorous complexity analysis of the $(1 + 1)$ evolutionary algorithm for separable functions with boolean inputs. *Evolutionary Computation*, 6:185–196, 1998.

[DKS26] Benjamin Doerr, Martin S. Krejca, and Milan Stankovic. Improved runtime guarantees for the SPEA2 multi-objective optimizer. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *AAAI Conference on Artificial Intelligence, AAAI 2026*, pages 36855–36863. AAAI Press, 2026.

[DKW24] Benjamin Doerr, Martin S. Krejca, and Noé Weeks. Proven runtime guarantees for how the MOEA/D computes the Pareto front from the subproblem solutions. In *Parallel Problem Solving from Nature, PPSN 2024, Part III*, pages 197–212. Springer, 2024.

[DKW26] Benjamin Doerr, Martin S. Krejca, and Noé Weeks. A dynamic decomposition strategy for the MOEA/D with proven performance guarantees. In *Advances in Neural Information Processing Systems, NeurIPS 2026*, 2026. To appear.

[DN20] Benjamin Doerr and Frank Neumann, editors. *Theory of Evolutionary Computation—Recent Developments in Discrete Optimization*. Springer, 2020. Also available at `http://www.lix.polytechnique.fr/Labo/Benjamin.Doerr/doerr_neumann_book.html`.

[DNNS23] Anh Viet Do, Aneta Neumann, Frank Neumann, and Andrew M. Sutton. Rigorous runtime analysis of MOEA/D for solving multi-objective minimum weight base problems. In *Advances in Neural Information Processing Systems, NeurIPS 2023*, pages 36434–36448, 2023.

[DPAM02] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6:182–197, 2002.

[DQ23] Benjamin Doerr and Zhongdi Qu. From understanding the population dynamics of the NSGA-II to the first proven lower bounds. In *Conference on Artificial Intelligence, AAAI 2023*, pages 12408–12416. AAAI Press, 2023.

[GL10] Oliver Giel and Per Kristian Lehre. On the effect of populations in evolutionary multi-objective optimisation. *Evolutionary Computation*, 18:335–356, 2010.

[HZ20] Zhengxin Huang and Yuren Zhou. Runtime analysis of somatic contiguous hypermutation operators in MOEA/D framework. In *Conference on Artificial Intelligence, AAAI 2020*, pages 2359–2366. AAAI Press, 2020.

[HZC+21] Zhengxin Huang, Yuren Zhou, Zefeng Chen, Xiaoyu He, Xinsheng Lai, and Xiaoyun Xia. Running time analysis of MOEA/D on pseudo-Boolean functions. *IEEE Transactions on Cybernetics*, 51:5130–5141, 2021.

[HZLL21] Zhengxin Huang, Yuren Zhou, Chuan Luo, and Qingwei Lin. A runtime analysis of typical decomposition approaches in MOEA/D framework for many-objective optimization problems. In *International Joint Conference on Artificial Intelligence, IJCAI 2021*, pages 1682–1688, 2021.

[IAN13] Hisao Ishibuchi, Naoya Akedo, and Yusuke Nojima. Relation between neighborhood size and MOEA/D performance on many-objective problems. In *Evolutionary Multi-Criterion Optimization, EMO 2013*, pages 459–474. Springer, 2013.

[Jan13] Thomas Jansen. *Analyzing Evolutionary Algorithms – The Computer Science Perspective*. Springer, 2013.

[LZZZ16] Yuan-Long Li, Yu-Ren Zhou, Zhi-Hui Zhan, and Jun Zhang. A primary theoretical study on decomposition-based multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 20:563–576, 2016.

[NW10] Frank Neumann and Carsten Witt. *Bioinspired Computation in Combinatorial Optimization – Algorithms and Their Computational Complexity*. Springer, 2010.

[RBLQ24] Shengjie Ren, Chao Bian, Miqing Li, and Chao Qian. A first running time analysis of the Strength Pareto Evolutionary Algorithm 2 (SPEA2). In *Parallel Problem Solving from Nature, PPSN 2024, Part III*, pages 295–312. Springer, 2024.

[WD23] Simon Wietheger and Benjamin Doerr. A mathematical runtime analysis of the non-dominated sorting genetic algorithm III (NSGA-III). In *International Joint Conference on Artificial Intelligence, IJCAI 2023*, pages 5657–5665. ijcai.org, 2023.

[WD24a] Simon Wietheger and Benjamin Doerr. Near-tight runtime guarantees for many-objective evolutionary algorithms. *CoRR*, abs/2404.12746, 2024.

[WD24b] Simon Wietheger and Benjamin Doerr. Near-tight runtime guarantees for many-objective evolutionary algorithms. In *Parallel Problem Solving from Nature, PPSN 2024, Part IV*, pages 153–168. Springer, 2024.

[ZD23] Weijie Zheng and Benjamin Doerr. Mathematical runtime analysis for the non-dominated sorting genetic algorithm II (NSGA-II). *Artificial Intelligence*, 325:104016, 2023.

[ZD24] Weijie Zheng and Benjamin Doerr. Runtime analysis of the SMS-EMOA for many-objective optimization. In *Conference on Artificial Intelligence, AAAI 2024*, pages 20874–20882. AAAI Press, 2024.

[ZL07] Qingfu Zhang and Hui Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11:712–731, 2007.

[ZLD22] Weijie Zheng, Yufei Liu, and Benjamin Doerr. A first mathematical runtime analysis of the non-dominated sorting genetic algorithm II (NSGA-II). In *Conference on Artificial Intelligence, AAAI 2022*, pages 10408–10416. AAAI Press, 2022.

[ZLT01] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm. *TIK report*, 103, 2001.

[ZQL+11] Aimin Zhou, Bo-Yang Qu, Hui Li, Shi-Zheng Zhao, Ponnuthurai Nagaratnam Suganthan, and Qingfu Zhang. Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation*, 1:32–49, 2011.

[ZYQ19] Zhi-Hua Zhou, Yang Yu, and Chao Qian. *Evolutionary Learning: Advances in Theories and Algorithms*. Springer, 2019.

[ZZZD26] Danyang Zhang, Zerong Zhong, Weijie Zheng, and Benjamin Doerr. Superior runtime guarantees for the MOEA/D multi-objective optimizer via weighted-sum decomposition. In *AAAI Conference on Artificial Intelligence, AAAI 2026*, pages 37187–37194. AAAI Press, 2026.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our contribution, which is clearly highlighted in the introduction, explicitly references our main results and mentions their scope. Moreover, we refer in the introduction explicitly to Section 2, where we detail how our paper relates to existing results, thus providing a good overview of the scope.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our results in the introduction and in the conclusion. Moreover, the details in Section 2 highlight differences to existing results, thus informing about limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We provide a proof or a detailed proof sketch for each mathematical claim in the paper, with the full proofs of Lemma 4 and Theorem 6 given in the appendix. Theorem 1 and Corollary 8 follow directly from other results, which we state in sufficient detail, respectively at the end of Section 4 (just before Section 4.1) and of Section 5.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [N/A]

Justification: This paper does not contain any experiments.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [N/A]

   Justification: This paper does not contain any experiments.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [N/A]

   Justification: This paper does not contain any experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: This paper does not contain any experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [N/A]

   Justification: This paper does not contain any experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We carefully reviewed the NeurIPS Code of Ethics and concluded that none of our results or assumptions violate the code.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Our results provide theoretical guarantees for an existing algorithm. Hence, the algorithm itself, and thus its potential impact, is as potent as before, only our understanding of the algorithm improved.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This paper does not deal with data or models.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [N/A]

    Justification: This paper does not use existing assets.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This paper does release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper neither involves crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper neither involves crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The research conducted in this paper does not involve LLMs.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.

A Appendix for "Understanding the Co-evolutionary Solution Exchange of the MOEA/D: Already the Smallest Communication Radius Gives the Best-Possible Speed-Up"

This appendix contains the proofs of Lemma 4 and Theorem 6 and is meant to be read at the reviewer's discretion only. For the sake of convenience, we repeat the lemmas before we state their proof.

Lemma 4. *Consider the same setting as in Theorem 2. Then the expected runtime for solving subproblems* 0 *and* n *is* $O\left(\frac{n^2}{\sqrt{r}} + n^2\right)$ *function evaluations.*

Proof. Note that the discussion for subproblem n is the same as for subproblem 0 when swapping all 0s and 1s in the discussion, due to the symmetry of eq. (1) and of type-A operators in terms of these numbers. Thus, we consider in the following only the case for subproblem 0. The case for subproblem n follows immediately and increases the total expected number of function evaluations for solving both subproblems by at most a factor of 2.

Let $k \in [1..n]$. In the following, assuming that the number of 0s of the currently best-so-far individual for subproblem 0 is k, we derive an upper bound on the expected time until we find a solution with at most $k-1$ 0s. This value is then communicated to subproblems 1 to r, which are updated if they have a worse value. The expected number of fitness evaluations until solving subproblem 0 is then, by linearity of expectation, at most the sum of all of these expected waiting times.

For the sake of brevity, we refer to the number of 0s of the best-so-far solution for any subproblem as its *value*, and we consider the single-objective function $g'_0 \colon [0..n] \to \mathbb{R}$ such that, for all $x \in \{0,1\}^n$, we have $g'_k(|x|_0) = g_k(x)$.

We first estimate the time until sufficiently many individuals have the same value as subproblem k. Afterward, we estimate the time it takes in order to improve the value of subproblem 0. Last, we conclude.

Communicating copies. Let $\ell_k = \min\{\lceil\sqrt{nr/k}\rceil, k\}$ be the number of subproblems to which we communicate value k, and let $M = \min\{r, \ell_k\}$ be the index of the largest subproblem such that, except for at most one during the run, all subproblems 0 to M have at most value k, which we detail further below. Moreover, let $i \in [M..\ell_k]$ be the largest index such that for all $i' \in [0..i]$, subproblem i' has value at most k. We estimate the expected time until $i \geq \ell_k$ by estimating the time it takes for i to increase by one and then adding up these ℓ_k waiting times. Note that, except if subproblem 0 is the initial solution, subproblems 0 to M all have value at most k, because when subproblem 0 is updated, the following happens: Consider subproblem $j \in [1..M]$, noting that $j \leq k$. If subproblem j had a value greater than k before, then by Lemma 3, when updating subproblem 0, the solution to subproblem 0 is at least as attractive to subproblem i, and so subproblem i also has a value of k. Otherwise, if subproblem j already has a value less than k, then its value after the update is at most that of subproblem 0, which has value k. We discuss the case that not all subproblems 0 to M have value at most k at the end of the proof, as this case occurs at most once during the run of the algorithm.

Note that if $r \geq \ell_k$, then immediately subproblems 0 to ℓ_k have value at most k. Hence, the complexity in this case is bounded from above by any positive value. Thus, we assume in the following $r < \ell_k$. In this case, when any subproblem i to $i - M = i - r$ is selected, then by the definition of a type-A operator, with probability at least $\alpha > 0$, this subproblem mutates and then propagates to at least subproblem $i+1$ an individual of value $j \in [0..k]$ by creating a copy of its current best-so-far individual. Since i to $i-r$ is selected on average every $O(\frac{n}{r})$ iterations, this communication occurs after an expected $O(\frac{n}{r\alpha}) = O(\frac{n}{r})$ number of iterations. As discussed above, each such update leads to a value of at most k for subproblem $i+1$.

Since we need to reduce i at most $\ell_k - M$ times, index i is at least ℓ_k after an expected time of $O\big(\frac{n}{r}(\ell_k - M)\big) = O(\frac{n}{r}\ell_k)$.

Once at least ℓ_k subproblems have a value of at most k, we wait until at least one of these subproblems produces, via mutation, an individual with value at most $k-1$. To this end, we make a case distinction with respect to the subproblem selection mechanism. The communication radius is irrelevant for this step, since we only consider ℓ_k subproblems.

Uniform selection for reaching $k-1$. The probability to pick a subproblem with value at most k and that mutation then produces an individual with a strictly smaller amount of 0s is by definition of a type-A operator at least $C_A \cdot \frac{k}{n} \cdot \frac{\ell_k}{n}$. Consequently, the expected time until such an improvement occurs is $O(\frac{n^2}{k\,\ell_k})$.

Cyclical selection for reaching $k-1$. The probability that no subproblem with value at most k mutates to value at most $k-1$ after all subproblems are selected once is, by the definition of a type-A operator, at most $\left(1 - C_A \frac{k}{n}\right)^{\ell_k}$. Thus, with probability at least $p_k \coloneqq 1 - (1 - C_A \frac{k}{n})^{\ell_k}$, at least one subproblem generates an individual of value at most $k-1$. The expected waiting time for this event is $O(n/p_k)$ iterations, because p_k considers n consecutive iterations.

We bound $\frac{n}{p_k}$, noting that $\frac{k\ell_k}{n} \leq 1$. For $x \in (0,1)$, using for all $t \in \mathbb{R}_{\geq 0}$ that $e^{-t} \leq \frac{1}{1+t}$, we obtain for $t = xC_A$ that $\frac{1}{1-e^{-xC_A}} \leq \frac{1}{1-\frac{1}{1+xC_A}} = \frac{1+xC_A}{xC_A} \leq \frac{1+C_A}{xC_A}$. Thus,

$$\frac{n}{p_k} \leq n \cdot (1 + C_A) \frac{n}{C_A k \ell_k} = O\left(\frac{n^2}{k\ell_k}\right).$$

Reaching $k-1$. For either subproblem selection, we get in expected time $O(\frac{n^2}{k\ell_k})$ a solution with value at most $k-1$ for a subproblem in $[0..\ell_k]$. Then, using an analogous argument to the one for creating copies, we propagate this value to subproblem 0 in time $O(\frac{n}{M}\ell_k)$. Hence, in expected time $O\big(\frac{n}{M}\ell_k + \frac{n^2}{k\ell_k}\big)$, subproblem 0 has a value of at most $k-1$.

Concluding. Except for the initial case, which we discuss last, the overall runtime is bounded by $O\Big(\frac{n}{r}\sum_{k=1}^{n}\ell_k + \sum_{k=1}^{n}\frac{n^2}{k\ell_k}\Big)$. We estimate this quantity by making a case distinction with respect to k. If $k \geq \lceil (nr)^{1/3} \rceil$, then $\ell_k = \lceil \sqrt{nr/k} \rceil$. The runtime contribution in this case amounts to $O\left(\frac{n}{r}\sum_{k=\lceil (nr)^{1/3}\rceil}^{n}\lceil\sqrt{nr/k}\rceil + n^2\sum_{k=\lceil n^{1/3}\rceil}^{n}\frac{1}{k\lceil\sqrt{nr/k}\rceil}\right) = O\Big(\frac{n^{3/2}}{\sqrt{r}}\sum_{k=\lceil (nr)^{1/3}\rceil}^{n}\frac{1}{\sqrt{k}}\Big) = O\big(\frac{n^2}{\sqrt{r}}\big)$.

If $k < \lceil (nr)^{1/3} \rceil$, then $\ell_k = k$, and we obtain a total runtime contribution of $O\Big(\frac{n}{r}\sum_{k=1}^{\lceil (nr)^{1/3}\rceil} k + n^2 \sum_{k=1}^{\lceil (nr)^{1/3}\rceil}\frac{1}{k^2}\Big) = O(n^2)$.

For the final case that subproblem 0 contains the initial solution, we wait at most until a copy of this solution is made, as this communicates it immediately to subproblems 1 to M, which then have a value of at most k, as discussed above. By the definition of a type-A operator, this happens after an expected $O(\frac{n}{\alpha})$ iterations, which is negligible for the overall runtime contribution. This concludes the proof. □

Theorem 6. *Let $n \in \mathbb{N}_{\geq 1}$. Consider the MOEA/D (Algorithm 1) with decomposition number n, Chebyshev decomposition, reference point (n, n), communication radius $r \in [1..n]$, and either uniform or cyclical subproblem selection, optimizing* OMM. *Assume that the MOEA/D makes use of a type-B mutation operator in line 8. Then the expected runtime is $\Omega(n^2)$ function evaluations.*

Proof. As in the proofs of Section 4.1, we use the terminology that the *value* of a subproblem is the number of 0s of its best-so-far individual. We prove that within $\Omega(n^2)$ iterations, if all subproblems have an initial value of at least 1, which occurs with overwhelming probability, then the all-1s individual is not produced with an at least constant probability. Hence, the expected runtime is at least quadratic.

As sketched at the beginning of Section 4, we show that it is likely that most subproblems have a value of at least 2 before mutation even creates the all-1s individual, due to the properties of the type-B operator. In this case, at most the subproblems 0 and 1 have value 1, which is too few in order to create the all-1s individual with a probability that is sub-constantly close to 1. More formally, we aim at proving that the probability of a set of favorable events $Q = A \cap B \cap C \cap D \cap E$, defined in the following, satisfies $\Pr(Q) = \Theta(1)$. We define:

- **Event A.** All initial values are at least 1.
- **Event B.** For all $i \in [2..n]$, the first time that subproblem i is selected and has value 1, its offspring has a number of 0s in $[2..i]$.
- **Event C.** In the first n^2 iterations, subproblem 0 or 1 is selected at most $3n$ times.
- **Event D.** In the first n^2 iterations, if subproblem 0 or 1 has value 1, its offspring has at least one 0.
- **Event E.** In the first n^2 iterations, every time a selected subproblem has a value greater than or equal to 2, its offspring has at least one 0.

Assuming that all of these events have positive probability, which we show in the following, we obtain $\Pr(Q) = \Pr(A) \cdot \Pr(C \mid A) \cdot \Pr(D \mid A, C) \cdot \Pr(B \mid A, C, D) \cdot \Pr(E \mid A, B, C, D)$. We analyze each of these terms separately.

During initialization, for each subproblem $k \in [0..n]$, the probability that subproblem k has the all-1s individual is $\frac{1}{2^n}$, independently of any other random choices. Hence, $\Pr(A) = \left(1 - \frac{1}{2^n}\right)^{n+1} = \Theta(1)$.

For uniform selection, let $N_{01} \in [0..n^2]$ be the random number of times that subproblem 0 or 1 are selected during the first n^2 iterations. Hence, N_{01} follows a binomial law with n^2 trials and a success probability of $2/(n+1)$, and it is independent of A. Hence, by a Chernoff bound, it follows that $\Pr(N_{01} > 3n \mid A) \leq \exp(-\Omega(n))$. Thus, $\Pr(C \mid A) \geq 1 - \exp(-\Omega(n)) = \Theta(1)$.

In cyclical selection, each element is selected every n iterations, so subproblem 0 or 1 are selected at most $2n$ times in n^2 iterations. Thus, $\Pr(C \mid A) = 1 = \Theta(1)$, too.

Conditional on C, subproblem 0 or 1 are selected at most $3n$ times during the first n^2 iterations. Each time they have a value of at least 1, by the definition of a type-B operator, the probability of the offspring being the all-1s individual is $O(1/n)$. Hence, $\Pr(D \mid A, C) \geq \left(1 - O\left(\frac{1}{n}\right)\right)^{3n} = \Theta(1)$.

Let S denote the random variable representing the set of indices from $[2..n]$ such that for all $i \in S$, subproblem i triggers the event B during execution, that is, the first time that subproblem i is selected, its offspring has a number of 0s in $[2..i]$. Let $i \in S$, and let p_i be the probability that the mutation that triggers event B of i produces an offspring with value in $[2..i]$. By the definition of a type-B operator, we have $p_i \geq 1 - C_B(1/i^2 + 1/n)$. Let $c^* = C_B(1/4 + 1/n)$. Noting that for all $\gamma, x \in (0,1)$, the inequality $1 - x \geq e^{-\gamma x}$ is equivalent to $\gamma \geq \frac{1}{x}\ln(\frac{1}{1-x})$, and noting that the function $x \mapsto \frac{1}{x}\ln(\frac{1}{1-x})$ is monotonically increasing over $[0,1]$, we see that for all constants $\gamma \in (0,1)$ and for all $x \in [0..\gamma]$, we have $1 - x \geq e^{-\frac{1}{\gamma}\ln(\frac{1}{1-\gamma})x}$, where $\frac{1}{\gamma}\ln(\frac{1}{1-\gamma})$ is also a constant. Thus, there is a constant $c' \in (0,1)$ such that for all $i \in S$, we have $1 - p_i \geq e^{-c' p_i}$. Using this inequality and the fact that mutations are independent of each other, we obtain

$$\begin{aligned}\Pr(B \mid A, C, D) &= E[\Pr(B \mid S, A, C, D)] = E\left[\textstyle\prod_{i \in S} p_i \mid A, C, D\right] \\ &\geq \prod_{i=2}^{n}\left(1 - C_B\left(\frac{1}{i^2} + \frac{1}{n}\right)\right) \geq \exp\left(-c' \cdot C_B \sum_{i=2}^{n}\left(\frac{1}{i^2} + \frac{1}{n}\right)\right) = \Theta(1).\end{aligned}$$

For the last term in our probability bound for Q, each time a subproblem with a value of at least 2 is selected, by the definition of a type-B operator, the offspring has at least one 0 with probability at least $1 - O(1/n^2)$. Hence, $\Pr(E \mid A, B, C, D) \geq \left(1 - O\left(\frac{1}{n^2}\right)\right)^{n^2} = \Theta(1)$.

Altogether, we obtain $\Pr(Q) = \Theta(1)$, as desired. Last, note that any run of the MOEA/D conditional on Q lasts at least n^2 iterations. By A, each subproblem has initially a value of at least 1. By D, if subproblem 0 or 1 is selected during the first n^2 iterations, their offspring is never the all-1s individual. By E, for each subproblem $i \in [2..n]$, if their value is at least 2 during the first n^2 iterations, their offspring has at least one 0. And by B, if subproblem i with value 1 creates an offspring, then its number of 0s is in $[2..i]$. By Lemma 3, such an offspring is at least as attractive as the current best-so-far individual of subproblem i. Hence, the offspring is the next best-so-far individual. This implies that subproblem i cannot have value 1 anymore. Consequently, E concludes the proof. □