Abstract: The metaheuristics have to explore the search space successfully i.e., their search process has to explore distinctive areas of the search space and to move to unexplored parts of the search space. There are numerous factors that affect this important feature of the metaheuristics that lead to getting high quality solutions. Recently, it has been shown that the performance of the coarse-grained algorithms outperforms many other parallel algorithms. This work experimentally investigates the impact of the key parameters of the coarse-grained algorithms such as migration on the diversification (creating various solutions in order to investigate the search space on the global level) of these algorithms and subsequently their ability to find high quality solutions. The observed results show comparable recommendations.

Key words: Metaheuristics, coarse-grained algorithms, diversification, high dimensional optimization problems.

1. Introduction

Metaheuristics are general approximate algorithms that are not particular for a specific optimization problem. They are sets of algorithmic ideas for characterizing heuristic strategies to handle various optimization problems with few adjustments to adjust them to a specific optimization problem [1]. They have been effectively dealt with a variety of optimization problems. A good instance of metaheuristics is evolutionary algorithms.

Evolutionary algorithms mimic the survival of the fittest in nature which became equivalent to reusing good solutions in these algorithms. This is by generating a population of candidate solutions, evaluating their quality, producing a new population by applying genetic operators, and repeating these phases for a specific number. Prior to this process is representing the underlying solutions to the problem being tackled.

Genetic algorithms (GAs) are the main example of evolutionary algorithms that were presented as an abstract type of the natural evolution to make PCs perform what nature does. A simple GA can be seen as a sequence of steps for moving from one population of potential solutions to a new one containing high quality potential solutions [2], [3].

Generally speaking, genetic algorithms and more generally the other metaheuristics require setting the values of their parameters that have great effect on their performance. This also applies for parallel metaheuristics as there are additional particular parameters related to paralyzing these algorithms such as migration interval and size, and topology in coarse-grained approaches. Hence, the significance of studying the parameters impact on the performance of these algorithms [4].

Mainly, the ideal values for these parameters rely upon:

- the given problem,
the problem instance, and
- the spent computational time in handling this optimization problem.

There is a tradeoff between the solution fineness and the search time when setting the parameters [4]-[7].

There are several approaches for setting metaheuristics’ parameters. These can be parameters tuning (preceding utilizing the metaheuristics), or self-adaptation to the problem being handled. The latter is suitable when good average performances across diverse problems is required but the computation overhead leads to less efficiency on specific problems compared to parameter tuning. Hence, parameter tuning is utilized in this paper. Common techniques for this method are sequential parameter optimization, racing, and meta-parameter setting (also known as meta-algorithm) [4]-[7].

Metaheuristics should be able to explore the search space intelligently. One of the essential factors for achieving this is the diversification. The necessity of keeping this important feature in coarse-grained metaheuristics is to perform enough information exchange between islands during combining partial solutions and to create novel ones.

There are various factors that affect achieving the diversification which leads to achieving successful global search of the whole search space. Some of these are particular to coarse-grained approaches (such as migration and its parameters) while the others keep the diversification in both parallel and traditional ones such as mutation in evolutionary algorithms. In addition, changing a component of an algorithm can improve its diversification capabilities such as tuning a parameter’s value.

In this article, an investigation of the impact of the key factors on the solutions diversification and hence of the performance of islands approaches for high dimensional optimization problems is performed.

The rest of this work is structured as follows. Section 2 addresses the parallel genetic algorithms. Section 3 exhibits the population sizing. Section 4 clarifies the diversification in coarse-grained approaches. Section 5 details the experiments and the acquired results. These results are analyzed in Section 6. Conclusion remarks and possible future work are presented in Section 7.

2. Parallel Genetic Algorithms

It has been demonstrated that the performance of the coarse-grained approaches is superior to that of the conventional single population ones [8] as shown in the following subsections.

2.1. The Fine-Grained Approach

As a rule, it is efficient when the fitness evaluation is time consuming i.e., the required time for evaluating the fitness is more costly than the overhead resulted in the communication between the cores [9].

In this approach (depicted in Fig. 1), one population is stored on the master. Evaluating the fitness is divided among a few processors i.e., assessing the solutions is carried out in parallel by them. This is on the grounds that evaluating a chromosome does not rely upon the others [10] and is accomplished by assigning a number of chromosomes to each available processor. Communication happens only when every slave gets its individuals and when it returns the fitness values. The master is in charge of the allocation of the fitness evaluation tasks among the slaves, and for applying genetic operators [9], [10]. This is the reason it is additionally called master-slaves.

![Fig. 1. The fine-grained GA [10].](image)
2.2. The Coarse-Grained Approach

The key difference between the traditional evolutionary algorithms and the islands ones is dividing the whole population (in the latter) into several islands that communicate with each other via migration. They have numerous populations interconnected according to a particular topology (such as a ring topology [8]) where meager migrations of chromosomes between its islands are executed [11] as shown in Fig. 2. In other words, the entire population is divided into a few subsets whose members having higher fitness will have higher chances of relocation [8] that allows a small amount of genetic material to cross-fertilize the population of each island which may enhance the diversity via introducing new chromosomes i.e., re-injecting diversity periodically into islands that disallows converging prematurely. Keeping high diversity is crucial for getting novel chromosomes.

![Fig. 2. The coarse-grained GAs [12].](image)

Utilizing these algorithms usually leads to superior performance as each island can search in different portions of the search space and hence enhancing their exploratory feature and providing different solutions for the given optimization problem [9]-[11]. They are highly advanced and can converge rapidly since the sub-populations' size is not as much as the used population by the conventional GAs [10]. They suit complex problems that are not easily tackled by conventional GAs [13]. However, they are distinguished by a large computation/communication rate [4]. They have additional synonyms such as distributed, islands, or multi-population GAs [10], [11].

3. Population Sizing

The population size (the number of individuals) is viewed as one of the key GAs parameters and modifying it can highly affect the performance of a GA. It is significant because it not only influences whether the GA can discover good solutions (the quality of the found solutions differs depending on it) but also the taken time to get them. If it is overlarge, the GA will squander computational resources dealing with unneeded chromosomes. If it is too small, there might not be a sufficient supply of good traits that act as a unit to influence the wellness of chromosomes, and it will be difficult to distinguish good solutions. A well choice of it can improve both solution quality and the computation time as well. This balance between the quality of the solution and the time that a basic GA requires to get it exists for parallel GAs as well [10].

Because of its significant influence on the solution quality and the search time, it has been considered one of the important research topics in the evolutionary computation community. Different researches about the proper population size are found in literature. It has been claimed that a small population size can guide the algorithm to poor solutions and that a large one can lead to expending more computation time [4].

According to many authors, the ideal value for this parameter is around 100 chromosomes. This means when we have several islands, the whole population should be an integer that can be divided into several values each is around 100 [4]. In this paper, we exclude this parameter from studying its effect as it was...
investigated previously.

4. Diversification in Coarse-Grained Approaches

The metaheuristics ought to be developed with the point of viably and proficiently investigating the search space. The search procedure ought to be keen to investigate unexplored parts of the search space.

The investigation of the search space distinguishes parts with high quality solutions in a problem specific close ideal way. This helps avoid spending too much time in the portions that are already well explored or that comprise of only low fitness solutions.

Diversification (one of the crucial components in recent metaheuristics) is the exploration of the regions that have not been investigated well i.e., enabling the search procedure to investigate unexplored areas which prompts accomplishing good quality solutions by better examining the search space. It makes metaheuristics able to avoid any local optimum so as to investigate the search space globally through decreasing the convergence rate. This is because their search procedure prefers chromosomes with larger fitness that may prompt premature convergence (failing to get the global optimum due to converging prematurely on suboptimal chromosomes. This is because of excluding speedily unfit chromosomes in the early generations while these chromosomes may be unique) [1], [14]-[17].

It has not been investigated enough in coarse-grained approaches as there are a few papers dealing with this research point. One of the main merits of the coarse-grained approaches is that they cause the slowness of the premature convergence of the overall population because every island’s population independently evolves from each other.

The first and principle factor influences diversification in this type of parallel evolutionary algorithms is migration. It does not simply mean a number of exchanged members. Rather, one should also take into his account that it can affect the algorithm’s capabilities of diversification.

Migration disallows the search from getting stuck in local optima as it causes diversity in the subpopulations because migrants can help change a stagnated target island [9], [10], [13]. This makes migration one of the key parameters of any islands model and subsequently its main parameters will have significant impact on the whole system performance.

It is commonly implemented by sending some chromosomes (migration size - how many members will migrate) every certain number of generations (migration interval- how many iterations at which substitution of chromosomes will take place). Migration can be executed when particular criterion is satisfied but fixed schedules are easier to analyze. The migration size and interval (how frequently migrations take place) are considered the most significant migration parameters [9], [10], [13].

Another migration parameter characterizes what is being sent (migration policy). For example, the best-worst migration approach selects the best chromosomes from the source population and replaces the worst chromosomes in the destination one. Nevertheless, this results in expanded selection pressure. Another option is the random-random migration strategy which selects random chromosomes and subrogates random chromosomes in another island. This does not extend the selection intensity, and thus it is reasonable for investigating other migration parameters [9], [10], [13].

One more migration parameter is the migration topology that depicts which islands will send selected chromosomes to which ones. The most widely recognized topologies are: an array, a ring, a grid, a star or a fully connected topology. Sparser connected topologies cause slower spread of information [9], [10], [13].

Besides migration, mutation plays an important role as it helps avoid the waste of genetic diversity. The repeating of selection and crossover operations can stagnate at indistinguishable chromosomes. Thus, the population’s fitness won’t be improved. In this manner, genetic algorithms guarantee the continuous improvements of the population’s average fitness and after sufficient iterations, the population advances to
a close ideal solution [2], [3].

5. Computational Experiments

Ten experiments were executed as follows.

5.1. Method

To explore thoroughly the role of a single factor, we fixed the other ones. We keep the same setting for the other parameters (that are not investigated in this study) in the experiments and we set their values prior to the experiments in view of various previous researches. These are:

- The entire chromosomes’ number was set to 200 since we used 2 islands. This is to keep the population of any island equals to 100 members.
- Crossover probability was set to 0.8.
- The mutation probability was set to 0.1. This is because it should be kept to a small value.
- Maximum 500 generations was utilized.
- In view of the stochastic nature of these algorithms, we did 10 repetitions.

We utilized one machine (benefiting from its multiple cores) rather than distributing computations over numerous machines to ensure that the experiments are not influenced by any external factors and to avoid the potential differences in the behavior of the utilized islands. The experiments were executed utilizing 2.

Two sorts of investigations are executed:

- With migration frequency equals to 50 and a variety of migrants’ number, and
- With migration frequency equals to 40 and a variety of migrants’ number.

5.2. The Utilized Optimization Problem

0-1 knapsack problem [18] was utilized. It is a problem of packing a subset of the total elements into the knapsack with the aim of maximizing the value of this knapsack without exceeding its maximum capacity as shown in the following figure.

![Fig. 3. A knapsack and 6 items.](image)

The details of the used instance is appeared in the following table. We utilized the high dimensional knapsack problem (KP) since it is an NP-hard optimization problem and frequently emerges in a variety of resource allocation applications and has numerous applications in real-world problems [7].

<table>
<thead>
<tr>
<th>No. of items</th>
<th>Knapsack Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>2732</td>
</tr>
</tbody>
</table>

5.3. Results

Tables 2 and 3 demonstrate the experiments’ results which are the average of ten independent runs (a
considerable number of autonomous runs has to be performed so as to obtain meaningful outcomes like any stochastic method).

<table>
<thead>
<tr>
<th>10 and 50</th>
<th>20 and 50</th>
<th>40 and 50</th>
<th>60 and 50</th>
<th>80 and 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of selected items</td>
<td>Total weight</td>
<td>Total profit</td>
<td>No. of selected items</td>
<td>Total weight</td>
</tr>
<tr>
<td>61.7</td>
<td>2726.6</td>
<td>4144.4</td>
<td>61.7</td>
<td>2726.65</td>
</tr>
</tbody>
</table>

Table 3. The Obtained Results (2)

<table>
<thead>
<tr>
<th>10 and 40</th>
<th>20 and 40</th>
<th>40 and 40</th>
<th>60 and 40</th>
<th>80 and 40</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of selected items</td>
<td>Total weight</td>
<td>Total profit</td>
<td>No. of selected items</td>
<td>Total weight</td>
</tr>
<tr>
<td>61.8</td>
<td>2726.1</td>
<td>4167.1</td>
<td>61.8</td>
<td>2728</td>
</tr>
</tbody>
</table>

These experiments were implemented utilizing R language [19]. The binary representation was utilized since it is the most fitting representation dealing with the utilized optimization problem.

The third, sixth, ninth, twelfth, and fifteenth column in table two represent the average of the total profits using migration frequency equals to 50 with a variety of migrants number. Similarly, the third, sixth, ninth, twelfth, and fifteenth column in table three are the average of the total profits using migration frequency equals to 40 and a variety of migrants number.

The second, fifth, eighth, eleventh, and fourteenth column in Table 2 represent the average of the total weights profits using migration frequency equals to 50 with a variety of migrants number. Similarly, The second, fifth, eighth, eleventh, and fourteenth column in Table 3 represent the average of the total weights profits using migration frequency equals to 40 and a variety of migrants’ number.

The first, fourth, seventh, tenth, and thirteenth column in table two and three represent the average of the selected items.

With migration frequency equals to 50, the best results were obtained using the number of migrants equals to 10. With migration frequency equals to 40, the best results were obtained using number of migrants equals to 20.

6. Discussion

In this paper, we focus on handling real-world optimization problems. Henceforth, high dimensional optimization problems were utilized in the experiments.

This work aims at clarifying the role of the key parameters on the diversification of the coarse-grained approaches for high dimensional optimization problems:

- The population size of each island.
- The migrants’ number which was set to 10, 20, 40, 60, and 80. These numbers were tried using migration frequency equals to 40 and 50.
- The migration frequency (interval).

7. Conclusions and Future Work

The aim of this paper was to examining the main diversification parameters' roles on the coarse-grained approaches. This leads to getting a thorough understanding of these factors' roles which is needed due to the large number of the parallel approaches’ parameters arising from those of the fundamental algorithm besides the parameters of including the parallelism feature of such algorithm.
Too rare migrations should be avoided as this prevents benefiting from exchanging chromosomes which affects the solutions quality. However, rare migrations with medium sizes lead to diverse solutions and hence getting better solutions. We should avoid the situation where the migrants will substitute individuals before a successful exchange and recombination of chromosomes could be done.

For the future work, other optimization problems should be utilized. In addition, investigating the importance of particular parameters should be considered with the utilization of hybrid algorithms.

As these results were gotten using a ring topology (a strongly connected one), other setting for the utilized parameters can affect the performance of the whole system utilizing other topologies. For instance, the performance is decreased with decreasing the migration interval. For that, other topologies should be considered in future studies.

References


Nadia Abd-Alsabour (BSc, MSc, and PhD) is an assistant professor at Cairo University, Cairo, Egypt. Her teaching experiences are intelligent systems, data mining, operating systems, programming languages (C++, advanced C++, C#), advanced topics in systems development, management information systems, information systems developments, web development (HTML, ASP.net, SQL server, PHP, MYSQL), E-commerce, software engineering, and systems analysis. Her research interests are swarm intelligence, optimization, data mining, machine learning, artificial intelligence and software engineering. Dr. Abd-Alsabour is a PC member, session chair, publicity chair, guest editor, and a reviewer in many international conferences and journals.