Re-inspiring the genetic algorithm with multi-level selection theory: Multi-Level Selection Genetic Algorithm (MLSGA)

A.J. Sobey[a,*], P.A. Grudniewski[a]

[a]*Fluid Structure Interactions, University of Southampton, University Rd, Southampton SO17 1BJ, England, UK*

Abstract

Genetic algorithms are integral to a range of applications. They utilise Darwin's theory of evolution to find optimal solutions in large complex spaces such as engineering, to visualise the design space, Artificial Intelligence, for pattern classification, and financial modelling, improving predictions. Since the original Genetic Algorithm was developed, new theories have been proposed which are believed to be integral to the evolution of biological systems. However, genetic algorithm development has focused on mathematical or computational methods as the basis for improvements to the mechanisms, moving it away from its original evolutionary inspiration. There is a possibility that the new evolutionary mechanisms are vital to explain how biological systems developed but they are not being incorporated into the genetic algorithm; it is proposed that their inclusion may provide improved performance or interesting feedback to evolutionary theory. Multi-level selection is one example of an evolutionary theory that has not been successfully implemented into the genetic algorithm and these mechanisms are explored in this paper. The resulting MLSGA is unique in that it has different reproduction mechanisms at each level and splits the fitness function between these mechanisms. There are

[*]Corresponding author
Email address: ajs502@soton.ac.uk (A.J. Sobey)

two variants of this theory and these are compared with each other alongside a unified approach. This paper documents the behaviour of the two variants, which show a difference in behaviour especially in terms of the diversity of the population found between each generation. The multi-level selection 1 variant moves rapidly towards the optimal front but with a low diversity amongst its children. The multi-level selection 2 variant shows a slightly slower evolution speed but with a greater diversity of children. The unified selection exhibits a mixed behaviour between the original variants. The different performance of these variants can be utilised to provide specific solvers for different problem types when using the MLSGA methodology.

Keywords:
Evolutionary theory, algorithm development, multi-level selection,
multi-objective optimisation, single objective optimisation.

1. Applications using Genetic Algorithms

There is a large literature pertaining to applications and developments of evolutionary algorithms. Coello Coello [1] has collected 10714 references relating to genetic algorithms and a google scholar search of the term “genetic algorithm” for 2017 supplied 37600 examples using or developing these algorithms. These references cover a wide range of areas related to finding optimal solutions to complex problems across a wide range of different types of search space. A recent review by Zhou et al. [2] lists 50 major applications for evolutionary algorithms. This is augmented by recent high profile examples that utilise these algorithms from a number of different fields of study, including architecture, bioinformatics, computational science, evolutionary theory, environmental science and materials engineering ([3]; [4]; [5]; [6]; [7]; [8]; [9]; [10]; [11]; [12]; [13] and [14]). The problems being solved are becoming more complex and therefore promising methods for the improvement of genetic algorithms should be further explored and documented. This research investigates the adoption of multi-level selection, a popular explanation for the mechanisms of evolution, as a potential inspiration to improve evolutionary computation.

1.1. Evolutionary Inspiration

Evolutionary algorithms were originally proposed by Turing [15] and were first developed by Holland [[16] and [17]] to study evolution through simulations. The usages of genetic algorithms quickly expanded due to their ability to search large and complex spaces, with a corresponding increase in the number of available algorithms. Since these initial attempts, there has been limited success in improving genetic algorithm performance using evolutionary inspired methods. A recent benchmarking of top algorithms in the CEC'09, [18], show no biologically or evolutionary inspired mechanics and a recent review of the state-of-the-art in biomimetics doesn't list evolution in the top 100 topics[19]. However, other algorithms have benefited from developments that more closely mimic the original concepts that inspired them, such as the ant colony optimisation developed by Zhang [20].

In parallel with the improved performance in genetic algorithms, there have been developments in evolution theory, which have led to a larger range of mechanisms available to explain how organisms evolve; many researchers now believe that some of these mechanisms are necessary for biological evolution. These proposed mechanisms are missing from the genetic algorithm and the authors believe they should be used to re-inspire it; if they are critical to evolution then they should provide benefits to algorithm development. Amongst the newer evolutionary theories is the multi-level selection theory, originally proposed by Sober and Wilson [21], with the idea that evolution does not occur at only one level but is actually occurring at different levels of a hierarchical structure. An example of a hierarchical structure is shown in Figure

1 where evolution at multiple levels is shown using collectives, larger units, which contain a number of individuals, smaller units; darker shading of the individuals or the collectives is used to illustrate higher fitness. As the generations progress the collectives may continue, and possibly reproduce, or be eliminated, depending on their ability to survive. Within the surviving collectives, the individuals also reproduce or are eliminated, based on their fitness. The strongest individuals generally have more offspring and this leads to an increase in average fitness. Whilst the authors believe the process makes intuitive sense, and there is a growing body of evidence to support the idea that it increases evolutionary speed, the theory of multi-level selection is not without controversy. There are many arguments for other mechanisms through which evolution might occur, such as Dawkin's popular selfish gene theory [22] which emphasises a gene-centric level of evolution, or Sterelny and Kitcher [23] which questions whether there must be a uniquely correct identification of the level at which selection is occurring. The authors believe multi-level selection provides the most promising inspiration for improvements to the genetic algorithm, allowing a collective mechanism to be put in parallel with current individual mechanisms and allowing more flexibility in the manner in which the fitness function can be interpreted. Initial research by the authors investigating one variant of multi-level selection shows promise [24] and can be used to solve practical problems [25] but further investigation into the inspiration behind these mechanisms needs exploring. This paper therefore documents the behaviour of different variants of this algorithm to provide specific solvers for different problem types when using the MLSGA methodology. These will exploit the unique reproduction mechanisms at each level of MLSGA and the split in the fitness function.

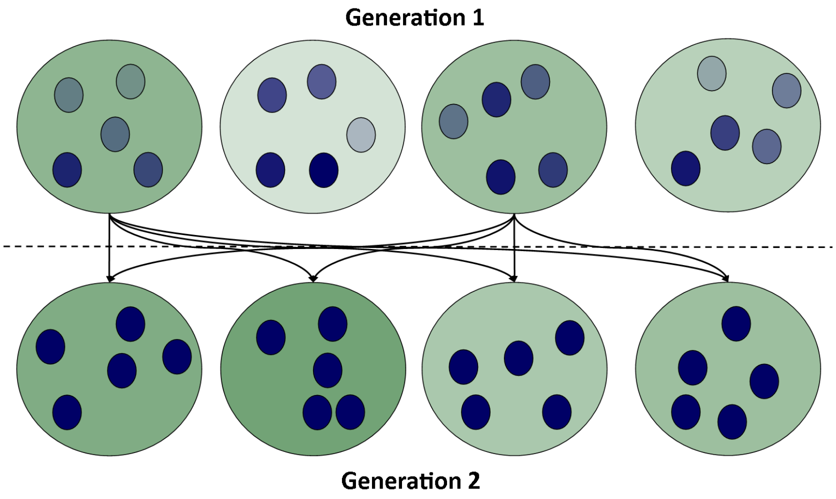

Figure 1: Nested hierarchical collective reproduction

1.2. *Multi-level selection theory*

Natural selection encourages individuals to adapt to the environment developing traits that increase their chance of survival; it is therefore common to consider adaptions at the individual level, for example, giraffes with longer necks can reach leaves on trees that others cannot. Multi-level selection proposes that it is rare for a population to compete as individuals without some kind of beneficial interaction; often a number of individuals will group together with the aim of helping each other. This can be in the form of a number of organisms grouping into a tribe or pack but it can also be at a lower level where a number of organelles create a hybrid form. An argument is often made to support the statement that there are characteristics that have developed which cannot be explained satisfactorily at the individual level, for instance altruism; which is a trait that does not benefit the individual and, by definition, actually results in a cost to itself. Sober and Wilson [21] suggested that evolution of altruists can be observed at a level higher than the individual by watching the development of colonies.

Multi-level selection is an outcome of three determinants based around those proposed by Lewontin [26], which holds that evolution occurs when:

1. Different individuals in a population have physical variations.
2. Different individuals have different rates of survival due to different fitness.
3. There is a correlation between parents and offspring so that fitness is heritable.

Natural selection occurs if there are variations in characteristics of individuals and some offspring are considered to be fitter than others. Individuals with these different characteristics can be defined as a 'unit of selection'. Individuals are encouraged to adapt over time to the environment developing traits that increase their chance of survival. This causes evolution as weaker members of the population are lost and the stronger members, on average, survive. Gradually the make-up of the population will change over successive generations to favour the stronger characteristics. A species can be described based on a hierarchical organisation, which is normally viewed as a nested hierarchy, with one level being enclosed within another. A nested hierarchical organisation is presented in Figure 1 which forms the basis of the Multi-Level Selection Genetic Algorithm (MLSGA). McShea [[27], [28]] proposes an interactionist approach indicating that selection at a higher level will occur when individuals at a lower level perform fitness-affecting interactions with each other, a physical connection is not required, and this is supported by Sober and Wilson [21] in their theory of multi-level selection. The new algorithm is developed by creating a hierarchical organisation, including collective level reproduction. However, since classical genetic algorithms already utilise selection of individuals based on their adaption to their environment it is therefore possible to utilise a standard Genetic Algorithms as the reproduction method at the individual level.

In addition to the collective level reproduction mechanism the new algorithm relies on a change to the way fitness is defined. Amongst the theories for multi-level selection, there are two main strands, multi-level selection 1 and 2 (MLS1 and MLS2) which are dependent on how fitness is defined on the individual and collective levels. For multi-level selection 1 the fitness is defined the same way at each level, the aggregate of the individuals. For multi-level selection 2 the fitness is defined differently at each level. For multi-level selection to occur there are a number of important concepts defined in Table 1 including the unit evolution is working on at each hierarchical level, the character or abilities of a unit used to determine the fitness, the fitness used to judge survival for each unit and finally the implications of these factors on the heritability of the unit. Both of these theories have proponents and a large background of literature with which to support these views. For a larger and more in-depth review the authors recommend the excellent text by Okasha [29] outlining the history, theory and gaps in multi-level selection research.

1.3. Comparison to the state-of-the-art

In reviewing algorithms specifically pertaining to multi-level selection there are already a few attempts to use the process as inspiration to improve evolutionary algorithms, but these fail to replicate the key aspects of this theory and the resulting performance is poor. Lenaerts el al. [30] were the first to use multi-level selection and they study a biological model based on multi-level selection. It shows the importance of variation between groups as selection at group level occurs only when there is enough variation. However, the mechanics to generate selection at different levels are prescribed based on interactions between individuals, as opposed to fitness. Akbari et al. [[31] and [32]] looked at multilevel selection including selection at different levels. However, they do not present group, collective, mechanisms such as characters, fitness and heritability, necessary for multilevel selection. The algorithm relies on a

Table 1: Important concepts of multi-level selection

	Individual level	Collective level
Hierarchical Unit	Number of individuals in each collective	Number of collectives composed of individuals
Character	The qualities shown by the individual which can affect its fitness e.g. speed from body type	Collective characters are the qualities of the collective that can affect its fitness and can be the average or total character of individuals (aggregate, MLS1), e.g. the aggregate speed of each individual, or different from individual's character (emergent, MLS2) with independent qualities, e.g. the ability to communicate abstract concepts
Fitness	Individual fitness measured directly on individuals	Collective fitness measured the same as the individual (aggregate, MLS1) or differently to individual's (emergent, MLS2).
Heritability	Individuals with higher fitness produce more offspring with characters correlated to the parent individuals	Collectives with higher fitness leave more collectives with characters correlated to the parent collectives.

number of complex mechanisms to replicate the sub-problem generation and individual replacement. Finally, Wu and Banzhaf [33] focused on selections using any hierarchical model. However, in multilevel selection, selection at one level influences the selection at the adjacent levels and the concepts of multilevel selection, such as units of selection at all levels and the products induced by selections between levels, are not obviously applied. The authors feel that these efforts miss the key aspects of multi-level selection in that they are orientated around complex prescriptive mechanisms, forcing the selection to occur rather than letting it emerge as part of the process. As these mechanisms do not replicate the key aspects of multi-level selection the authors provide a novel evolutionary algorithm, supported by the fact that the results provide an improved performance compared to those previously documented.

Furthermore, multiple different methodologies in the current state of the art show similarities to the idea of a collective and fitness split, such as subpopulation approach and problem decomposition. In Niching algorithms, such as NSGA-II [34] and U-NSGA-III [35], Co-evolutionary algorithms, for example BCE [36] and HEIA [37] and Island model algorithms proposed by [38] the population is divided into groups with often different operations performed on each sub-population. However, between these groups, only one level of selection is used with no competition between sub-groups and split in the fitness function, unlike MLSGA. The decomposition-based methods as MOEA/D [39], MOEA/D-M2M [40], CS-NSGA-II [41] DMOEA-DD [42], LiuLi [43], RVEA [44] and K-RVEA [45] operate by partitioning the entire objective space into subspaces, dividing the problem into a number of separate subproblems each with its own subpopulation. This is done by using a set of uniformly spread reference points, or weight vectors, and scalarization functions. However, the effectiveness of these methods decreases for discontinuous problems as subpopulations can be assigned to regions where feasible solutions do not exist. A priori knowledge about the objective and search spaces are therefore required to adjust the mechanisms. In MLSGA, the sub-regional search is created using different fitness definitions, instead of forced decomposition, and therefore all individuals operate on the same region.

Introduction of selection at multiple levels provides the inspiration for the development of a new algorithm for optimisation, Multi-Level Selection Genetic Algorithm (MLSGA), adding multi-level concepts to the classic genetic algorithm. Current genetic algorithms consider the evolution of its individuals at a single level similar to the manner in which Holland [16] first developed the algorithm. It is proposed here that if multi-level selection speeds the process of evolution then its addition will elicit an increase in performance in the genetic algorithm, speeding up the rate at which higher fitness solutions will be found. This evolutionary inspiration has been used by the authors to develop an algorithm, Multi-Level Selection Genetic Algorithm but there are a number of different variants of multi-level selection theory and these are explored on single- and multi-objective problems to categorize any differences in behaviour.

2. Methodology

Outlined are the mechanisms that are used, inspired by the concept of multi-level selection, and the methodology used, showing the adaptations from the classic genetic algorithm. This is documented for the two variants of multi-level selection theory to show the differences in performance, the unified theory and also to the commonly used NSGA-II algorithm and also to the commonly used NSGA-II algorithm and MOEA/D developed by Zhang and Li [39]. NSGA-II was created by Deb et al. [34] and is the most commonly used Genetic Algorithm but is also selected to represent a niching algorithm. The specific method selected is the 2011 updated version which shows similar results on 2 objective functions to the more mature U-NSGA-III. MOEA/D developed by Zhang and Li [39], represents sub-region search algorithms, and is the highest performing genetic algorithm for unconstrained test functions, for multi-objective problems. In these cases the hyperparameters are taken from the CECŠ09 benchmarking with no additional tuning [18].

2.1. Multi-Level Selection Genetic Algorithm (MLSGA)

The Genetic Algorithm starts with a randomly generated population of individuals each of which represents a set of variables representing the search space for an optimisation problem. This population is then evaluated against a fitness function. Once the variables for each individual have been assessed, and the fitness determined, a new generation can be created from the current parent generation. Whatever the specific selection methodology chosen the process involves finding the fittest individuals within a population to mate and produce a child generation. These children are produced by the process of crossover where the chromosome of the parents is split. The chromosomes are then combined forming a new offspring different to the parent generation. Further diversity is found through mutation of the chromosome, where random changes are made normally based on a probability value, which is chosen to be low. Once this process has been repeated for the whole population a new population is then ready for the process to be repeated. Over successive generations, the average fitness of the population will become lower until either the optimal value or the limit of the number of generations is reached. This replicates a simplified process of evolution first developed to simulate the process but since then many adaptions and improvements have been made.

The Multi-Level Selection Genetic Algorithm is similar to previous genetic algorithms except that it utilises the key concepts from Table 1, defining fitness and heritability, through reproduction mechanisms, at different hierarchical units, individuals and collectives. The main addition is therefore the collective, which contains a number of individuals and has a fitness dependent on the character of those individuals.

1. Initialisation - a population of individuals is created at random.
2. Classification - sorted into collectives of the most similar individuals, in this case using a Support Vector Machine (SVM). The quality of the result is dependent on a good spread of results between the collectives where SVM is used over clustering algorithms to allow the user control over the size of the collectives.
3. Individual reproduction - As the algorithm progresses the individuals inside each of the collectives are treated in the same manner as for the standard genetic algorithm so that the process of selection and mating continues as normal.
4. Collective reproduction - the difference is that the collectives themselves have a fitness and elitism process where the least fit are eliminated and the fittest reproduce. Hyperparameter tuning is performed which shows that the results are relatively insensitive to the new parameters. Large or small numbers of collectives are shown to provide poor performance with 6 giving the best performance for these problems and 4 or 8 collectives giving similar but worse results. In this case 2 collectives are eliminated and 2 are created from the old surviving collectives with the individuals being populated from the fittest 4 collectives, where the number of new individuals in the offspring collective is equal to the number of individuals in the eliminated one. Individuals are generated by replicating the best individuals, according to the collective fitness function, equally from the remaining collectives. As an example in the case where there are 6 collectives and 2 are eliminated then if there are 40 individuals in the first eliminated collective and 60 in the second then the best 10 individuals from the remaining 4 collectives are replicated and added to the first new collective and the next 15 best are then taken from the remaining 4 collectives and are added to the second new collective.
5. Termination - The algorithm terminates, in these cases after a given num-

ber of generations.

This process, with the specific details used in this paper for each stage, are documented in Table 2.

2.1.1. MLS1

The two types of multi-level selection differ in the manner in which the fitness function is represented. In MLS1 the fitness of the collectives is the aggregate of the sum of the individuals and is illustrated using the ZDT1 function where $f_1(x)$ is,

$$f_1(x) = x_1, \tag{1}$$

and $f_2(x)$, is,

$$f_2(x) = 1 + \frac{9}{n-1}\sum_{i=1}^{n} x_i \left(1 - \sqrt{\frac{x_1}{1 + \frac{9}{n-1\sum_{i=1}^{n} x_i}}}\right), \tag{2}$$

where variables are in the range $0 \leq x_i \leq 1$, n is the number of variables. The first objective function is arbitrarily taken as $f_1(x)$ and the second, remaining, objective function is $f_2(x)$. This means that the same fitness function can be used to determine the selection in the collectives as well as the individuals. The fitness function for the MLS1 individuals and collectives is,

$$\text{Fitness} = \frac{f_1(x) + f_2(x)}{2}. \tag{3}$$

2.1.2. MLS2

The MLS2 theory states that the fitness for the collectives is different to the fitness for the individuals. Therefore, this variant only allows the use of multi-objective optimisation problems or the generation of an abstract objective for single objective problems. For the cases chosen here, the fitness function has been split into two with one of the objectives forming the fitness for the individuals,

Table 2: Multi-level selection algorithm methodologies

Step	Parameter	Value
1. Initialisation	Type	Random
	Encoding	Real values
	Pop. Size	600
2. Classification	Method	SVM
	Number of collectives	6
3. Individual Reproduction		
Fitness Evaluation	Type	MLS1 or MLS2 individual fitness
Selection	Type	Roulette wheel
Mating	Crossover type	Real variable SBX
	Crossover rate	0.7
	Mutation type	Polynomial mutation
	Mutation Rate	0.08
Elitism	Rate	0.1
4. Collective Reproduction		
Fitness Evaluation	Type	MLS1 or MLS2 collective fitness
Elimination	Number of eliminated collectives	2
Replacement	Number of new collectives	2
5. Termination	Criterion	Reaching 500 generations

$$\text{Individual Fitness} = f_1(x), \tag{4}$$

and the other forming the fitness function for the collective,

$$\text{Collective Fitness} = f_2(x). \tag{5}$$

The effect of reversing this order is also investigated to determine whether it affects the results of the optimisation and is designated MLS2R, i.e in equations 4 and 5 individual fitness is assigned to be f_2 and collective fitness f_1.

2.1.3. MLS-U

Contradicting the need to define MLS1 or MLS2, Sterelny and Kitcher [23] indicate that a single mechanism for selection does not need to be defined at each level. United Multi-Level Selection (MLS-U) is introduced, inspired by this definition, exhibiting characteristics of both MLS types. In MLS-U all MLS variants are used in parallel where some collectives only utilise MLS1, MLS2 or MLS2R, maintaining an equal number of collectives of each type.

2.1.4. Computational complexity of one generation

The computational cost of the MLSGA is determined by three operations: the PF selection, individual reproduction and collective reproduction. The PF selection identifies nondominated individuals from 1.5N members at most, aggregate of external population of maximum size equal to the overall population size N and current collective of maximum size of 0.5 N, requires *O(m(1.5N)^2)* comparisons using fast nondominated sorting approach, where m is the number of objectives [34]. In individual reproduction the full replacement of old generation with elitims take place which require *O(N^2)* at most, as only one of the objectives is considered. For collective reproduction, the fitness evaluation step requires *O(N)* computations, as is calculated as the average of single fitnesses of the individuals and collective replacement of *O((0.5N)^2)* complexity at most, due to maximum size of the collective.

To summarize, the overall computational compexlity of one generation of MLSGA is bounded by *O(mN^2)*.

3. Characterisation of multi-level selection variants

The results of genetic algorithms developed using the two strands of multilevel selection, MLS1 and MLS2, are compared to each other for the different test functions from Ziztler et al. [46], shown in Table 3, on 30 separate runs. These are popular in testing genetic algorithms due to the wide range of near optimal points which are difficult to find, and which the classical Genetic Algorithms cannot solve. Whilst there are more complex functions available, these are not selected at this stage, as it is difficult to identify differences between variants on these problems. Furthermore, these problems have clear differences in complexity for both fitness functions, where f_2 is harder, making it easier to investigate the differences between MLS2, MLS2R and MLS1. For each of these functions the number of variables used is 30, except in the case of ZDT4 where 10 are used. The variable bounds are $[0, 1]$ for all cases except in ZDT4 where these are: $x_1 = [0, 1], x_{2..10} = [-5, 5]$. The stopping criterion is when a value of 300,000 function evaluations, generations multiplied by population size, is reached. If the optimal result is not found at this stage then the best fitness to this point is used as the minimum fitness value. To demonstrate the accuracy of each algorithm results show the minimum, the maximum and the average optimal results found over the all the runs. The robustness of the algorithm is investigated to determine the percentage of runs over which the algorithm found the optimal result.

3.1. Weighted average optimisation - evolutionary speed

A weighted average optimization is used to demonstrate the algorithm working for single objective problems. MLS2 and MLS1 become the same for a

Table 3: ZDT Test Functions

Problem	Objective Functions	Comments
ZDT1 [47]	$f_1(x) = x_1;\, f_2(x) = g(x)\left[1 - \sqrt{\frac{x_1}{g(x)}}\right]$ $g(x) = 1 + 9\frac{\sum_{i=2}^{n} x_i}{n-1}$	Convex, continuous
ZDT2 [47]	$f_1(x) = x_1;\, f_2(x) = g(x)\left[1 - \left[\frac{x_1}{g(x)}\right]^2\right]$ $g(x) = 1 + 9\frac{\sum_{i=2}^{n} x_i}{n-1}$	Nonconvex, continuous
ZDT3 [47]	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1 - \sqrt{\frac{x_1}{g(x)}} - \frac{x_1}{g(x)} sin(10x_1)\right]$ $g(x) = 1 + 9\frac{\sum_{i=2}^{n} x_i}{n-1}$	Convex Disconnected
ZDT4 [47]	$f_1(x) = x_1,\, f_2(x) = g(x)\left[1 - \sqrt{\frac{x_1}{g(x)}}\right]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^{n}\left[x_i^2 - 10cos(4x_i)\right]$	Convex, large search space
ZDT6 [47]	$f_1(x) = x_1 - exp(-4x_1)sin^6(6x_1)$ $f_2(x) = g(x)\left[1 - \left[\frac{x_1}{g(x)}\right]^2\right]$ $g(x) = 1 + 9\left[\frac{\sum_{i=2}^{n} x_i}{n-1}\right]^{0.25}$	Nonconvex Non-uniformly spaced

true single objective problem, $f_1 = f_2$, however, in the weighted average case two objectives still exist and so MLS2 can still be examined, even though it still splits the function into two components but is trying to find the solution illustrated in eq. 3, conversely MLS1 considers all problems as a weighted average. The results are shown in Table 4 with the best performing algorithm shaded for each category other than the standard deviation.

The results show that the MLS1 function rapidly finds the solution to all of these problems, except for ZDT4; in the case of the ZDT4 function a close to best value is found. A similar performance is seen for the MLS2R function but with a slightly greater variation in the solutions for the more complex functions, ZDT3-6, finding the best solution 43% of the time for the ZDT3 function and 77% for the ZDT6 function. The MLS2 variant performs extremely poorly for most of the functions only finding the best solution for the ZDT2 function and in this case only 30% of the time. The unified variant, MLS-U, exhibits a similar performance to the MLS1 and MLS2R variants and outperforms MLS2, in terms of the final accuracy of the Pareto front. However, the results take considerably more generations to form than for the other variants, especially MLS1. The results demonstrate the manner in which the MLS1 algorithm dives towards the Pareto front reaching it quickly, in single objective optimisation or scenarios where the optimal value is not of so much interest and the near optimal is required the MLS1, with a low generation cut off, would provide an excellent solution in few generations. For the MLS2 the choice of function for the individual and collective is important in determining the performance with one variant finding the Pareto front rapidly and the other struggling to find any optimal solutions. Computationally the new algorithm is rapid and the only extra computational effort over a standard genetic algorithm is the generation and elimination of the collectives.

Table 4: Comparison of multi-level selection algorithms for weighted average optimisation

Test Function	Fitness				Convergence (Generations)
	Max.	Min.	Mean	Standard Deviation	
MLS1					
ZDT1	0.375	0.375	0.375	0	49
ZDT2	0.500	0.500	0.500	0	34
ZDT3	0.039	0.039	0.039	0	100
ZDT4	0.375	0.375	0.375	1.90E-06	500
ZDT6	0.500	0.500	0.500	0	145
MLS2					
ZDT1	0.500	0.584	0.508	0.017	500
ZDT2	0.500	0.558	0.505	0.013	412
ZDT3	0.500	0.583	0.506	0.016	500
ZDT4	3.38	32.8	16.7	7.58	500
ZDT6	0.516	0.601	0.567	0.025	500
MLS2R					
ZDT1	0.375	0.375	0.375	0	169
ZDT2	0.500	0.500	0.500	0	57
ZDT3	0.039	0.039	0.039	1.00E-06	368
ZDT4	0.375	0.375	0.375	1.19E-04	500
ZDT6	0.500	0.501	0.500	4.11E-04	377
MLS-U					
ZDT1	0.378	0.393	0.388	2.84E-03	500
ZDT2	0.500	0.500	0.500	0	299
ZDT3	0.039	0.039	0.039	0	465
ZDT4	0.375	0.375	0.375	4.70E-05	500
ZDT6	0.500	0.501	0.500	4.33E-04	479

3.2. Multi-objective optimisation - evolutionary diversity

The speed and accuracy with which the optimal point is found is interesting and useful for a subset of problems. However, the benefits of evolutionary algorithms lie in their ability to assess multi-objective cases where a spread of optimal points is required. Whilst these fronts are useful they can be difficult to generate, as the algorithm must be accurate while ensuring an even spread of results to ensure that sharp variations in the front are found. The algorithm is adapted from the weighted-average results by storing the 400 best values in the Pareto front the objective functions are evaluated in the same way and only the objectives against which they are judged are changed, i.e. the spread of results is now evaluated unlike in the previous section. In this case, the 3 algorithms, including the 4 variants of MLSGA, are used to generate a Pareto front for the previously defined functions. The results for ZDT1-4 and 6 are found and are illustrated for the MLSGA variants at 50 and 100 generations for the ZDT1 case at 50 generations, Figure 2, and 100 generations, Figure 3, to demonstrate the difference in performance. This is shown again for the more complex ZDT6 function in Figure 4, for 50 generations, Figure 5, 100 generations, and Figure 6, 250 generations. Blue diamonds represent MLS1, red squares represent MLS2, green triangles represent MLS2R and purple circles represent MLS-U.

The Pareto front evolution shows a large difference between the algorithms at the early generations. However, by the time 100 generations have been completed then the algorithms have all found, or are close to finding the Pareto front. Figure 3 demonstrates the speed with which the Pareto front is found. In this case, the MLS2R algorithm finds the front rapidly with a good spread of results. The MLS2 algorithm finds a few points on the top left hand part of the front, but is unable to provide a greater diversity of points. The MLS1 algorithm finds some points along the front rapidly and is the first to reach it but struggles to diversify these with extra points along this front. The MLS-U shows the worst speed for reaching the front, and never reaches it in over the

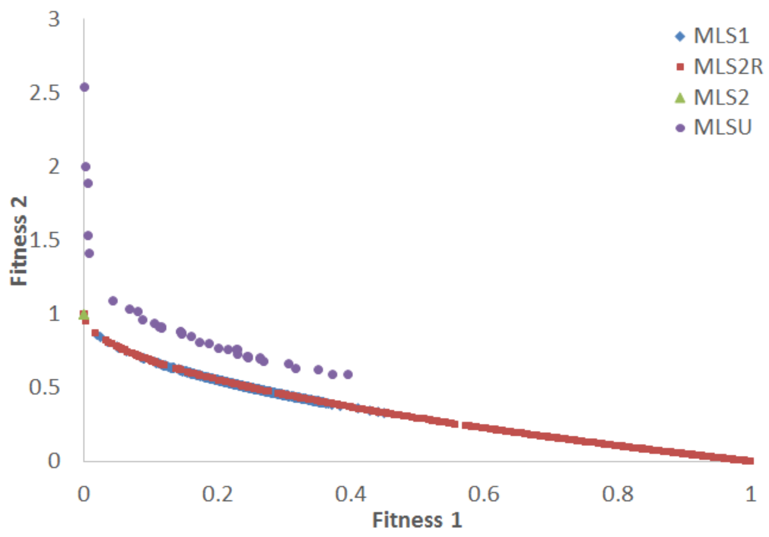

Figure 2: Pareto front comparison at 50 generations ZDT1 function

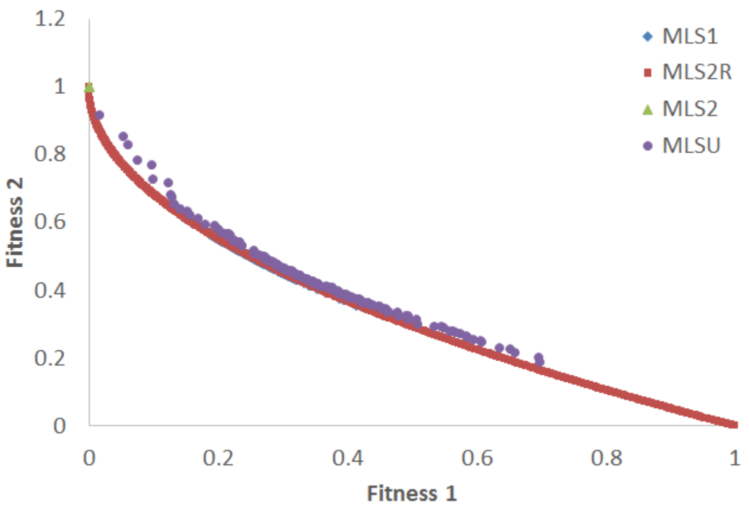

Figure 3: Pareto front comparison at 100 generations ZDT1 function

30 runs. This is because the MLS-U variant uses 3 different methods simultaneously where each has approximately only a third of the overall population. In this case, the evolutionary pressure is not as concentrated as when only a single variant is used and results in slower progression.

The process is repeated for the more complex ZDT6 functions where all of the algorithms are slower to find the front, shown in Figures 4, 5 and 6. The MLSGA variants find the front with the MLS2 and MLS2R finding a good spread of results. The MLS2 in this case is able to find the front more rapidly than the MLS2R. The MLS1 algorithm finds some on the front but many of the points are a considerable distance from the front. The MLS-U results show a slower progression to form the Pareto front to the extent that the results for Figure 5 are not visible as they are out of range, resulting from a change in axes from Figure 4 to Figure 5. Figure 6 shows the results after 250 generations where MLS-U has not found the front due to the lower evolutionary pressure exhibited by this variant.

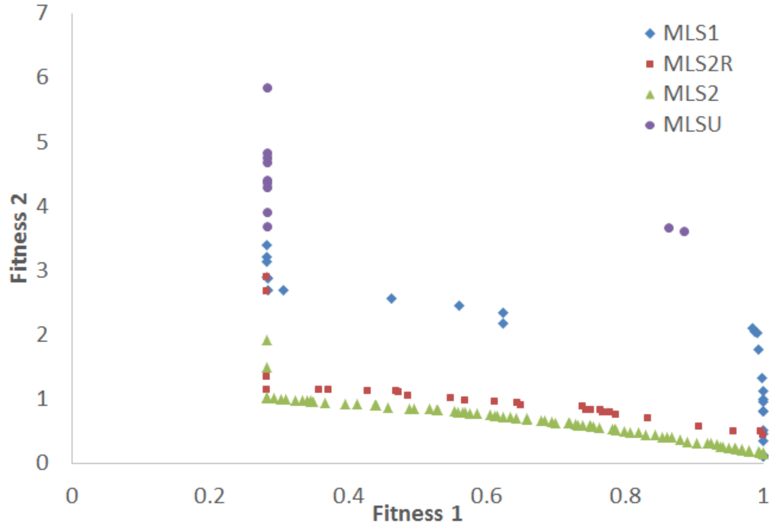

Figure 4: Pareto front comparison at 50 generations ZDT6 function

Table 5 illustrates the IGD values for the different algorithms with the best

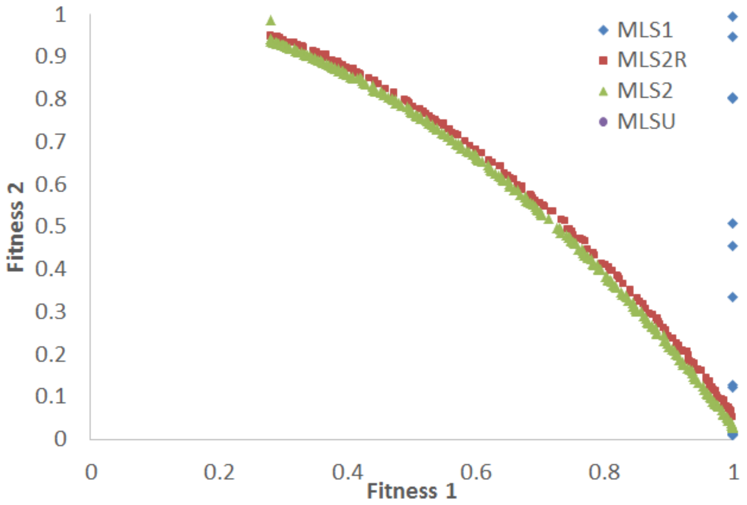

Figure 5: Pareto front comparison at 100 generations ZDT6 function, where the MLS-U results are not visible due to poor convergence

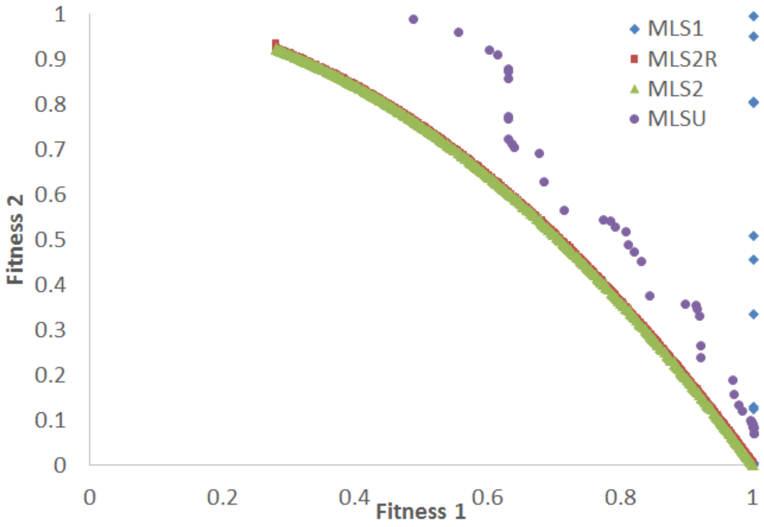

Figure 6: Pareto front comparison at 250 generations ZDT6 function

results shaded, MLS-U is ignored because of its poor performance. The MLS2R algorithm provides slightly better results in almost all cases except for the ZDT6 function, where MLS2 provides better results. This raises a question about how the collective and individual functions should be defined. It appears that the collective level mechanism is much weaker than that of the individual. Improvements to the collective level mechanism should result in a considerable increase in performance.

Table 6 presents the best results from MLS2, the best performing variant, using the best result from MLS2 or MLS2R, making the assumption that the correct algorithm is selected. These results show an increase in performance for MLSGA over the most popularly used GA, NSGA-II, and the top algorithm on unconstrained functions, MOEA/D. This is despite the fact that the algorithm uses a simple mechanism at the individual level and shows the potential for this method. In one case the MOEA/D algorithm has a better mean performance that MLSGA, but NSGA-II never shows better performance.

4. Evolutionary performance

Amongst the theories for multi-level selection, there are two main strands, multi-level selection 1 and 2 (MLS1 and MLS2) which are dependent on how fitness is defined on the individual and collective levels. For multi-level selection 1 the fitness is defined the same way at each level, the aggregate of the individuals, which results in deep specialisation of individuals, and greater evolutionary pressure thus increasing the rate of evolution. For multi-level selection 2 each level has different "goals", which results in competition between individuals and collectives, and leads to increased diversity of population as a compromise between the two goals is found. Importantly separate groups, across the whole environment, can have different fitness definitions at each level, similar to species where each becomes specialised, developing different traits. According to Okasha [29], multi-level selection is proposed to enhance

Table 5: Comparison of multi-level selection algorithms for multi-objective optimisation

Test Function	IGD			
	Max.	Min.	Mean	Std. deviation
MLS1				
ZDT1	0.11500	0.18000	0.16000	1.6E-02
ZDT2	0.11800	0.25100	0.15800	3.1E-02
ZDT3	0.02900	0.08100	0.05100	1.3E-02
ZDT4	0.02900	0.15300	0.09800	3.6E-02
ZDT6	0.13700	0.46800	0.29400	1.0E-01
MLS2				
ZDT1	0.84000	0.88100	0.84600	1.2E-02
ZDT2	0.60900	0.72500	0.61900	2.4E-02
ZDT3	0.82000	0.96200	0.83500	3.4E-02
ZDT4	9.8	62.3	34.4	13.4
ZDT6	0.00313	0.00334	0.00320	5.10E-05
MLS2R				
ZDT1	0.00390	0.00403	0.00395	3.80E-05
ZDT2	0.00389	0.00403	0.00396	3.00E-05
ZDT3	0.00477	0.00561	0.00499	1.40E-04
ZDT4	0.00388	0.00414	0.00401	6.70E-05
ZDT6	0.00316	0.00397	0.00340	2.60E-04
MLS-U				
ZDT1	0.00567	0.02770	0.01585	5.28E-03.
ZDT2	0.00405	0.39442	0.14610	1.08E-01
ZDT3	0.02903	0.17426	0.07854	4.92E-02
ZDT4	0.00443	0.06126	0.00776	1.00E-02
ZDT6	0.00397	0.10147	0.05535	3.98E-02

Table 6: Comparison of MLSGA with the state-of-the-art

Test Function	IGD			
	Max.	Min.	Mean	Std. deviation
MLSGA				
ZDT1	0.00390	0.00403	0.00395	3.80E-05
ZDT2	0.00389	0.00403	0.00396	3.00E-05
ZDT3	0.00477	0.00561	0.00499	1.40E-04
ZDT4	0.00388	0.00414	0.00401	6.70E-05
ZDT6	0.00313	0.00334	0.00320	5.10E-05
NSGA-II				
ZDT1	0.00407	0.00425	0.00416	5.30E-05
ZDT2	0.00401	0.00420	0.00409	4.80E-05
ZDT3	0.00488	0.00521	0.00510	8.60E-05
ZDT4	0.00409	0.25500	0.05730	7.70E-02
ZDT6	0.00340	0.00361	0.00351	5.00E-05
MOEA/D				
ZDT1	0.00407	0.00422	0.00414	3.20E-05
ZDT2	0.00406	0.00417	0.00410	2.90E-05
ZDT3	0.00485	0.00503	0.00500	4.10E-05
ZDT4	0.00409	0.00415	0.00413	1.50E-05
ZDT6	0.00335	0.00345	0.00337	2.20E-05

the specialization of different sub-groups, and this correlates with the results found here.

The results show that the introduction of MLS1 or MLS2 provides an increase in performance over the classic Genetic Algorithm with respect to both optimisation and generation of the Pareto front. This provides some promising behaviour on some simple test functions when judged against that of its contemporaries, as an example NSGA-II and MOEA/D. The MLS2R algorithm outperforms the other algorithms in the generation of Pareto fronts on the simple test functions proposed. The Pareto front itself is found rapidly but, more importantly, the diversity is high whereby the Pareto front has a good spread of results. This is in contrast to the MLS1 code, which searches one specific zone to a high degree but does not develop any front, it finds a close-to-optimal location rapidly but there is a lack of spread of results. MLS1 shows the best performance on single objective problems. Whilst on the multi-objective problems the MLS1 code can outperform the MLS2 in rapidly finding near optimal results the performance is not that much faster than the MLS2R and the absolute optimum is never found. The MLS2R performs strongly in both tasks with a wide spread of different optimal areas investigated showing a wide diversity. Interestingly the MLS2 algorithm outperforms the MLS2R algorithm for the ZDT6 function, in contrast to the weighted fitness results. The difference between MLS2 and MLS2R results are caused by the fact that individual selection is more efficient than collective selection, and the f_2 functions are more complex than the f_1 functions in most of the test problems, this is reversed in the ZDT6 function. The authors suggest that the individual level should have the more complex function assigned to it. MLS-U is outperformed only by the best variants, MLS2R for the ZDT1-4 functions and MLS2 for ZDT6, and shows better results than the remaining variants. This sensitivity can be removed by utilisation of a unified MLS-U approach. MLS-U, despite a decrease in performance in comparison to the best variants, does not require extensive a priori knowledge about the optimised problem and eliminates the necessity of performing the optimisation process repeatedly in order to find the best possible solutions, in the case when close-to-the-best result is sufficient. However, in many real world cases the complexity of the problem will be well understood, through domain knowledge or simple computational tests, meaning that the most powerful variant can be selected. Improvements to the novel collective level reproduction mechanism will further reduce this sensitivity and reduce the hyperparameters associated with MLSGA.

Whilst the authors propose the MLSGA method they are surprised at the increase in robustness, diversity and evolutionary speed generated from the simple process of considering the fitness at multiple levels. The optimal fitness is found rapidly and the average fitness of the populations is high, making it suitable for finding Pareto fronts. The authors hesitate to draw overarching conclusions into evolutionary theory based on a biologically inspired algorithm however, it is interesting to note the increase in speed with which the generations find higher fitness solutions due to the increased evolutionary pressure. Furthermore, there are considerable differences in behaviour between the MLS1, MLS2 and MLS2R algorithms and the resultant differences in fitness. The MLS1 seems to find the optimal value correctly but lacks the diversity of children seen in the MLS2 algorithm. In this case, the MLS1 has a concentrated search in which there are two layers to remove individuals that are unfit. In MLS2 this phenomenon is weaker but still provides a concentrated search towards the individual's objective function, many of the poor solutions are removed at each generation pushing the search quickly towards the Pareto front but using the collectives to retain some diversity; on finding the Pareto front the search then rapidly spreads out increasing the range of solutions. This also differs from other currently high performing, computationally inspired, algorithms that first aim to create a diverse set of solutions that push towards the optimal front, with the potential to perform well on discontinuous fronts.

While it may be shown that multi-level selection is not the process by which evolution should be considered, the emergent change in the optimisation process is interesting and beneficial.

The test functions that have been selected are simple and the early results on the ZDT6 function already show that the MLSGA algorithm is starting to struggle to find the optimal solution with the more complex functions. This set is used to allow an easy comparison in performance but the ZDT1-4 functions have a special structure, with f_1 being only determined by a single decision variable, while all other decision variables are zero for all Pareto optimal solutions. Further iterations of the algorithm will move away from evolutionary inspiration and utilise more complex individual and collective reproduction methods to improve performance based on the findings here and this will include determining the sensitivity of performance to the various hyperparameters. Algorithms such as NSGA-II and MOEA/D are shown to outperform the original GA, currently used at the individual level, by a considerable margin, it is proposed that their inclusion at the individual level will increase performance. The effectiveness of the new collective reproduction will also be explored using simple search functions, such as hill climb, and evolutionary algorithm mechanisms, such as MTS, to boost the effectiveness of the collective search.

5. Conclusions

There are a number of evolutionary mechanisms proposed in recent years, which have not been explored in genetic algorithms. One of these, multi-level selection, has been used to inspire a new genetic algorithm but the different variants of this theory have not been compared. Interestingly the two variants of the algorithm, MLS1 and MLS2 show considerable differences in behaviour instigated by the definitions of fitness. The algorithm performs well on the simple ZDT functions with the MLS2 finding a wide diversity of results along the front and the MLS1 algorithm rapidly finding the weighted fitness optimum, demonstrating excellent single-objective performance. These different variants allow the MLSGA methodology to have different behaviour for different sets of problems, allowing strong performance on multi-objective and single objective problems.

6. Acknowledgements

The authors would like to thank the Lloyd's Register Foundation for their continued support of the LRF UTC in Ship Design for Enhanced Environmental Performance at the University of Southampton and without whom this research would not have been possible. We would also like to thank Dr. Yikun Wang for her kind support and effort in helping us prepare this manuscript, LCdr Suttipong Paksuttipol for his early work developing this algorithm and Dr. Richard Watson for his useful interrogation of the underlying concepts.

References

[1] Coello Coello, C.A., personal website. http://delta.cs.cinvestav.mx/ ccoello-/EMOO/EMOOstatistics.html. Accessed: 2016-05-11.

[2] A. Zhou, B. Qu, H. Li, S.-Z. Zhao, P. Suganthan, and Q. Zhang. Multi-objective evolutionary algorithms: a survey of the state of the art. *Swarm Evolutionary Computation*, vol. 1:pp. 32–49, 2011.

[3] A. Menges. Biomimetic design processes in architecture: morphogenetic and evolutionary computational design. *Bioinspiration and Biomimetics*, vol. 7:pp. 1–10, 2012.

[4] D. Bounds. New optimization methods from physics and biology. *Nature*, vol. 329:pp. 215–219, 1987.

[5] J. Barhen, V. Protopopescu, and D. Reister. TRUST: a deterministic algorithm for global optimization. *Science*, vol. 276:pp. 1094–1097, 1997.

[6] D. Wales and H. Scheraga. Global optimization of clusters, crystals, and biomolecules. *Science*, vol. 285:pp. 1368–1372, 1999.

[7] M. Glick, A. Rayan, and A. A. Goldblum. A stochastic algorithm for global optimization and for best populations: A test case of side chains in proteins. *PNAS*, vol. 99:pp. 703–708, 2002.

[8] M. Nowak and K. Sigmund. Evolutionary dynamics of biological games. *Science*, vol. 303:pp. 793–799, 2004.

[9] G. Schoups, J. Hopmans, C. Young, J. Vrugt, W. Wallender, K. Tanji, and S. Panday. Sustainability of irrigated agriculture in the san joaquin valley, california. *PNAS*, vol. 102:pp. 15352–15356, 2005.

[10] V. Menon, N. Spruston, and W. Katha. A state-mutating genetic algorithm to design ion-channel models. *PNAS*, vol. 106:pp. 16829–16834, 2009.

[11] W. Carter. Structure predictions: the genetics of grain boundaries. *Nature Water*, vol. 9:pp. 383–385, 2010.

[12] D. Tran and R. Johnston. Study of 40-atom pt-au clusters using a combined empirical potential-density functional approach. *Proceedings of the Royal Society A*, 2011.

[13] L. McNally, S. Brown, and A. Jackson. Cooperation and the evolution of intelligence. *Proceedings of the Royal Society B*, vol. 279:pp. 3027–3034, 2012.

[14] B. Srinivasan, T. Vo, Y. Zhang, O. Gang, S. Kumar, and V. Venkatasubramanian. Designing DNA-grafted particles that self-assemble into desired crystalline structures using the genetic algorithm. *PNAS*, vol. 110: pp. 18431–18435, 2013.

[15] A. Turing. Computing machinery and intelligence. *Mind*, vol. 49:pp. 433–460, 1950.

[16] J. Holland. *Adaption in natural artificial systems*. University of Michigan Press, Ann Arbor, 1975.

[17] J. Holland. Genetic algorithms. *Scientific American*, vol. 267:pp. 66–72, 2013.

[18] Q. Zhang and P. Suganthan. Final report on CECâĂŹ09 MOEA competition. IEEE Congress on Evolutionary Computation, 2009.

[19] N. Lepora, P. Verschure, and T. Prescott. The state of the art in biomimetics. *Bioinspiration and Biomimetics*, vol. 8:pp. 1–11, 2013.

[20] Z. Zhang, C. Gao, Y. Liu, and T. Qian. A universal optimization strategy for ant colony optimization algorithms based on the physarum-inspired mathematical model. *Bioinspiration and Biomimetics*, vol. 9:pp. 1–14, 2014.

[21] E. Sober and D. Wilson. *Unto Others: Evolution and Psychology of Unselfish Behaviour*. Havard University Press, 1999.

[22] C. Dawkin. *The Selfish Gene*. Oxford University Press, 1976.

[23] K. Sterelny and P. Kitcher. The return of the gene. *Journal of Philosophy*, vol. 85:pp. 339–361, 1988.

[24] P. Grudniewski and A. Sobey. Multi-level selection genetic algorithm applied to cec '09 test instances. In *IEEE Congress on Evolutionary Computation (CEC), San Sebastian*, pages pp. 1613–1620, 2017.

[25] Z. Wang, J. Bai, A. Sobey, R. Shenoi, and J. Xiong. Optimal design of triaxial weave fabric composites under tension. *Composite Structures*, vol. 201:pp. 616–624, 2018.

[26] R. Lewontin. The units of selection. *Annual Review of Ecology and Systematics*, vol. 1:pp. 1–18, 1970.

[27] D. McShea. Metazoan complexity and evolution: Is there a trend? *Evolution*, vol. 50:pp. 477–492, 1996.

[28] D. McShea. Possible largest-scale trends in organismal evolution: eight "live hypotheses". *Annual Review of Ecology and Systematics*, vol. 29:pp. 293–318, 1998.

[29] S. Okasha. *Evolution and the Levels of Selection*. Clarendon Press, 2006.

[30] T. Lenaerts, A. Defaweux, P. Remortel, and B. Manderick. Modelling artificial multi-level selection. In *In AAAI Spring Symposium on Computational Synthesis. AAAI Spring Symposium Series*, 2003.

[31] R. Akbari, V. Zeighami, and K. Ziarati. MLGA: A multilevel cooperative genetic algorithm. In *IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA)*, pages pp.271–277, 2010.

[32] K. Ziarati and R. Akbari. A multilevel evolutionary algorithm for optimizing numerical functions. *International Journal of Industrial Engineering Computations*, vol. 2:pp. 419–430, 2011.

[33] S. X. Wu and W. Banzhaf. A hierarchical cooperative evolutionary algorithm. In *In Proceedings of the 12th annual conference on genetic and evolutionary computation*, 2010.

[34] D. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, vol. 6:pp. 182–197, 2002.

[35] H. Seada and K. Deb. U-NSGA-III: A Unified Evolutionary Optimization Procedure for Single, Multiple, and Many Objectives: Proof-of-Principle

Results. *Evolutionary Multi-Criterion Optimization: 8th International Conference, EMO 2015, Guimar{ã}es, Portugal, March 29 –April 1, 2015. Proceedings, Part II*, pages 34–49, 2015.

[36] M. Li, S. Yang, and X. Liu. Pareto or Non-Pareto: Bi-criterion evolution in multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 20(5):645–665, 2016.

[37] Q. Lin, J. Chen, Z.-H. Zhan, W.-N. Chen, C. A. Coello Coello, Y. Yin, C.-M. Lin, and J. Zhang. A Hybrid Evolutionary Immune Algorithm for Multiobjective Optimization Problems. *Ieee Transactions on Evolutionary Computation*, 20(5):711–729, 2016.

[38] D. Whitley, S. Rana, and R. Heckendorn. The island model genetic algorithm: On separability, population size and convergence. *Journal of Computing and Information Technology*, vol. 1:pp. 33–47, 1950.

[39] H. Li and Q. Zhang. MOEA/D: A multiobjective evolutionary algorithm based on decompostion. *IEEE Transactions on Evolutionary Computation*, vol. 11:pp. 712–731, 2007.

[40] H.-l. Liu, F. Gu, and Q. Zhang. Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems. *IEEE Transactions on Evolutionary Computation*, 18(3):450–455, 2014.

[41] J. Branke, H. Schmeck, K. Deb, and M. Reddy.S. Parallelizing multiobjective evolutionary algorithms: cone separation. *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753)*, 2:1952–1957, 2004.

[42] M. Liu, X. Zou, C. Yu, and Z. Wu. Performance assessment of DMOEA-DD with CEC 2009 MOEA competition test instances. *IEEE Congress on Evolutionary Computation*, (1):2913–2918, 2009.

[43] H. L. Liu and X. Li. The multiobjective evolutionary algorithm based on determined weight and sub-regional search. *2009 IEEE Congress on Evolutionary Computation, CEC 2009*, pages 1928–1934, 2009.

[44] R. Cheng, Y. Jin, M. Olhofer, B. Sendhoff, and S. Member. A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation*, 20(5):773–791, 2016.

[45] T. Chugh, N. Chakraborti, K. Sindhya, and Y. Jin. A data-driven surrogate-assisted evolutionary algorithm applied to a many-objective blast furnace optimization problem. *Materials and Manufacturing Processes*, 32(10):1172–1178, 2017.

[46] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength pareto evolutionary algorithm. TIK-Report 103, 2001.

[47] E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorihms: empirical results. *Evolutionary Computation*, vol. 29:pp. 293–318, 2000.