A comparative review between Genetic Algorithm use in composite optimisation and the state-of-the-art in evolutionary computation

ZhenZhou Wang* and Adam Sobey

*Corresponding author. Address: Fluid Structure Interactions Group, Boldrewood Innovation Campus, University of Southampton, Burgess Road, Southampton SO16 7QF, UK. Email: _zw9e14@soton.ac.uk_.

Abstract

The task of providing optimal composite structures is increasingly difficult. New analysis approaches seek to model the material at the fibre/matrix scale and increasingly more control is sought over the material, for example optimising individual tows, and the structure, where new manufacturing techniques are proposed that will allow revolutionary new topologies. This additional complexity will stretch design engineers and as such it is important that state-of-the-art design methods are implemented to help take advantage of these exciting new opportunities, including computational optimisation methods. To determine best practice and the current limitations of the techniques a review of Genetic Algorithms in optimisation of composite materials and structures is performed over the last 10 years. This is compared to a technical review of the developments of Genetic Algorithms in the evolutionary computation literature. By better understanding how Genetic Algorithms are used in composite structures and comparing to evolutionary computational literature, recommendations are provided to help increase the use of Genetic Algorithms in solving composite optimisation problems in the future.

Keywords: Genetic Algorithms (GAs); Evolutionary Computation; Composite structures; Literature review; Multi-objective optimisation; Many-objective optimisation.

1 The importance of Genetic Algorithms to optimisation of composites

Optimisation techniques are widely used across many different disciplines. They are best used to provide innovative solutions or to gain insights into complex problems. Evolutionary algorithms are one of the most popular categories of optimisation techniques, especially in engineering design, as they are capable of finding solutions in large and complex search spaces. A database of applications of evolutionary algorithms, Coello Coello [1], shows 4,983 papers utilising evolutionary algorithms to solve multi-objective optimisation problems from 1988 to 2017. The proportion of different evolutionary algorithms from this database are summarised in Fig. 1, where the papers with titles containing 'evolutionary algorithm' are not counted since it is hard to distinguish and summarise which evolutionary algorithm is used in each of those papers. The main evolutionary algorithms used are: Genetic Algorithms (GAs), Particle Swarm Optimisations (PSOs), Ant Colony (AC), Simulated Annealing (SA), Differential Evolution (DE), Immune Algorithms (IA), Artificial Bee Colony (ABC) and Firefly Algorithm (FA). The largest proportion of algorithms being used are Genetic Algorithms which occupy 56% of applications, showing the continued popularity of this type of optimisation algorithm despite the increasing competition from other methods.
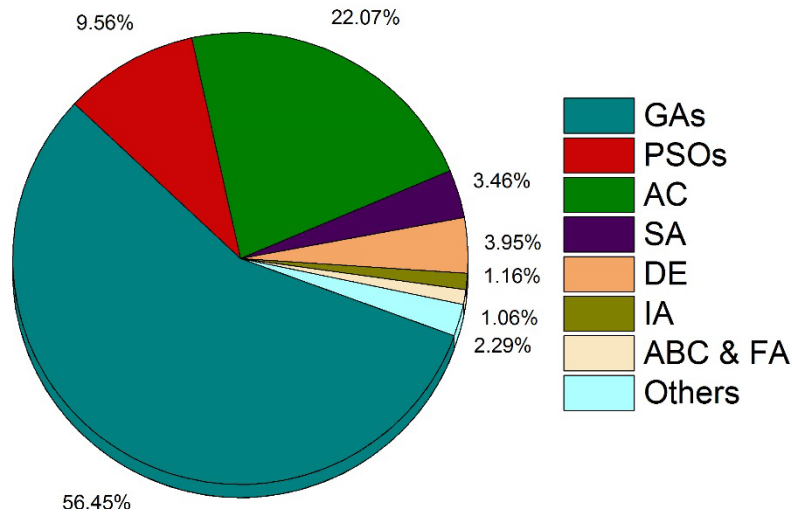
Fig. 1 Popular evolutionary algorithms in the past 30 years

The publications from this database, related to Genetic Algorithms, fall into 23 research broad fields. These fields, and the number of publications in each field, are shown in Fig. 2 where the composite structures optimisation theme is ranked in the lower half, indicating that there might be room for greater uptake of these methods to help understand the behaviour of these materials. As a field optimisation algorithms are used less in composite structures and materials than in similar engineering themes such as Aeronautics & Astronautics and Material Manufacturing, though there might be some overlap between them. Whilst Composite Structures might be considered to be a small field, its complexity and importance require consideration for how optimisation techniques can be better utilised to help future research and industrial applications.
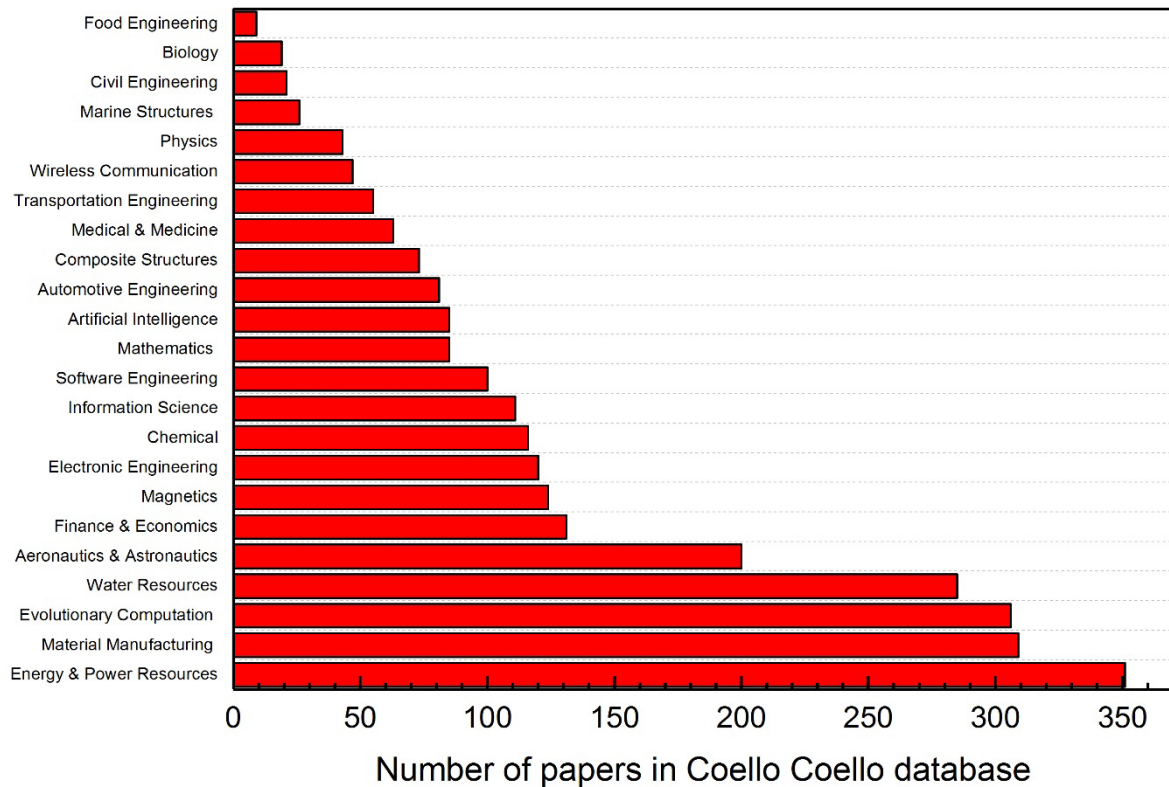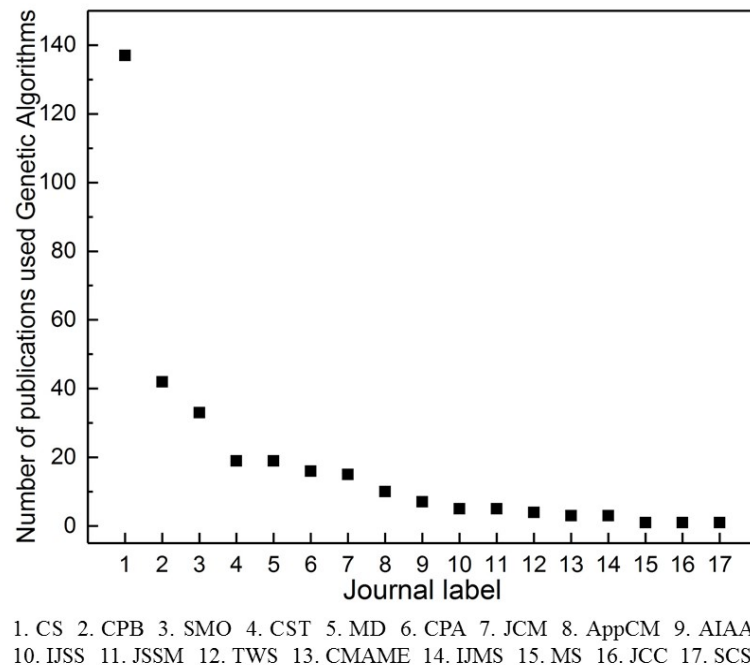

Fig. 2 Multi-objective optimisation papers published in different fields in the last 30 years [1]

Due to the importance of optimisation to the future of composite structures, and the significance of Genetic Algorithms within the field of optimisation, the use of Genetic Algorithms in Composite Structures/Materials is reviewed in this paper. In order to cover all relevant papers on this topic, allowing a more critical review of the state-of-the-art, the publications related to composite material/structure optimisation in 17 journals are reviewed: *Composites Science and Technology (CST), Composites Part A: Applied Science and Manufacturing (CPA), Composites Part B: Engineering (CPB), Composite Structures (CS), Applied Composite Materials (AppCM), International Journal of Solids and Structures (IJSS), Journal of Composite Materials (JCM), Journal of Sandwich Structures & Materials (JSSM), Thin-Walled Structures (TWS), Computer Methods in Applied Mechanics and Engineering (CMAME), International Journal of Mechanical Sciences (IJMS), American Institute of Aeronautics and Astronautics journal (AIAA), Steel and Composite Structures (SCS), Journal of Composites for Construction (JCC), Marine Structures (MS), Materials and Design (MD) and Structural and Multidisciplinary Optimisation (SMO)* are selected over a decade from 2008-2017. These journals are deemed to be representative of best practice in composite optimisation in the rest of the literature. There are 63,780 total publications in these 17 journals from 2008 to 2017 with 1,458 papers found by searching the keywords "Genetic Algorithm for composite material and structure". These are reduced to 321 papers by removing overlapping papers or those that mention Genetic Algorithms but do not use them. The details of these 321 papers are listed in a data attachment. The quantity of composite optimisation papers using Genetic Algorithms over the last decade has been summarised in Fig. 3a. The top five journals that having the highest number of composite optimisation papers using Genetic Algorithms have been broken down by year and is illustrated in Fig. 3b. It shows that the interest in this area has been increasing over the last 5 years.

1. CS 2. CPB 3. SMO 4. CST 5. MD 6. CPA 7. JCM 8. AppCM 9. AIAA
10. IJSS 11. JSSM 12. TWS 13. CMAME 14. IJMS 15. MS 16. JCC 17. SCS
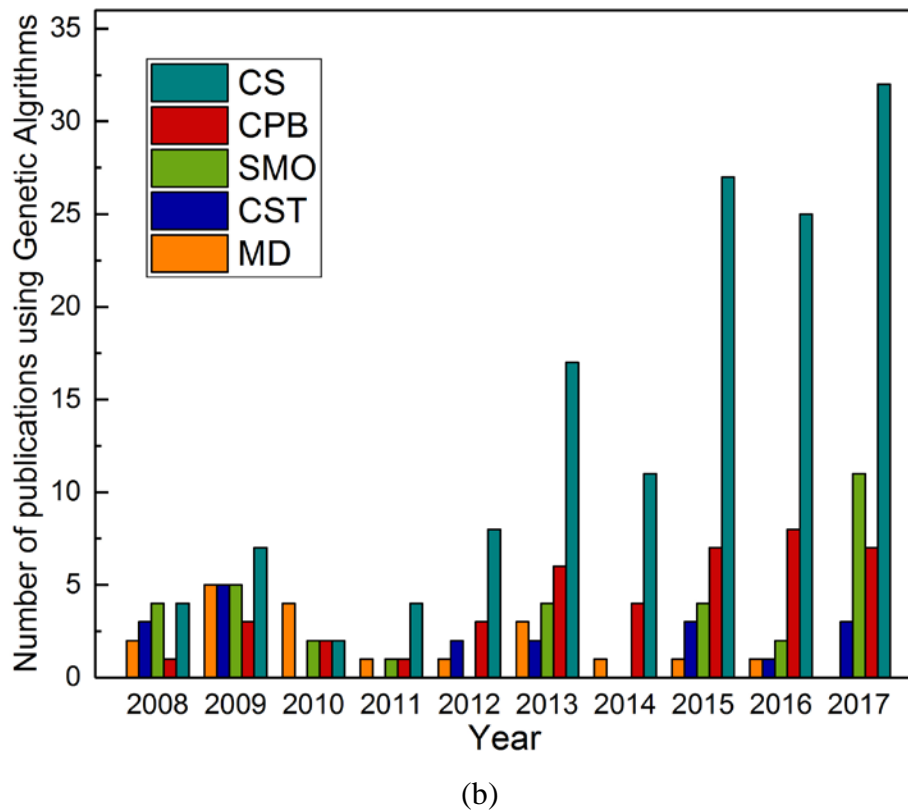
(a)

(b)

Fig. 3 Number of optimisation papers utilising Genetic Algorithms published: (a) in each journal; (b) in the selected journals by year

This growth in publications using Genetic Algorithms indicates that the complexity of the problems being solved is increasing and will continue to do so. To solve these problems the selection of the Genetic Algorithms, and how they are used, are important. The 'no free lunch' theorem states that two optimisation algorithms have the same performance when averaged over all possible optimisation problems. The design space in each problem has different characteristics for example: multimodal, biased, deceptive, concave/convex, continuous/discontinuous, single, multi- and many-objective and dynamic; and general solvers have generally good performance across all of the characteristics but cannot reach the performance spike of a specialist solver when used to solve a problem dominated by that characteristic. Conversely these specialist solvers own performance quickly degrades on problems with different characteristics to its own speciality. This is supported by work in evolutionary computation where there is a difference between the performance of specialist algorithms and the general solvers across a range of constrained and unconstrained problems [2]. This choice between general and specialist solvers creates a problem when selecting the correct algorithm as it is possible that practical problems are dominated by multiple characteristics, meaning that specialist solvers will perform poorly. However, some composite optimisation problems have simpler design spaces, dominated by one characteristic, and as these problems become more complex general solvers may not find the optimal results, or may require a larger number of function evaluations and time to solve. In addition to the type of solver in all optimisation problems there is a balance between convergence, which moves the search towards the optimal solution, and diversity, which ensures the entire search space is covered. Many different types of problems require Genetic Algorithms with strong convergence mechanisms, unconstrained and dynamic problems for example, but increasingly

there are cases where diversity mechanisms are shown to be important; for example discontinuous and constrained problems. In addition the increase in complexity of the problems seems to require solvers with stronger diversity preserving mechanisms. In some cases this diversity seems to dominate even when the problem is posed simply with limited design variables as witnessed in Wang et al. [3] and Mutlu et al. [4].

Therefore, the selection of the correct algorithm is not a simple problem and can greatly affect the performance; it requires an in-depth knowledge of evolutionary computation and composite structures/materials. This paper therefore reviews how Genetic Algorithms are currently being used in the literature before a technical review of some specific composite optimisation problems is performed and compared to the mechanisms of the latest evolutionary computation methods, to find compatible solutions. Recommendations are then made to help improve the performance of Genetic Algorithms on composite optimisation problems.

2 Common applications in the composite optimisation literature

2.1 Common composite optimisation applications

The main applications for optimisation within composite structures are briefly reviewed; although Nikbakt et al. [5] provide a more detailed categorisation of composite structural optimisation problems using similar categories. The literature is split into three main themes: input variables, objectives and applications. Within these there are 12 major categories which composite optimisation can be split into including design variables: laminate stacking sequence (LSS) and shape optimisation (SO); objectives: weight, cost, buckling, deflection, mechanical properties (MP) and natural frequency (NF); and applications: microwave absorption performance (MAP), woven fabric composites (WFC) problem and progressive failure (PF). The proportion of each category are summarised in Fig. 4. The laminate stacking sequence (LSS), weight, buckling and mechanical properties (MP) optimisations occupy the highest proportions.
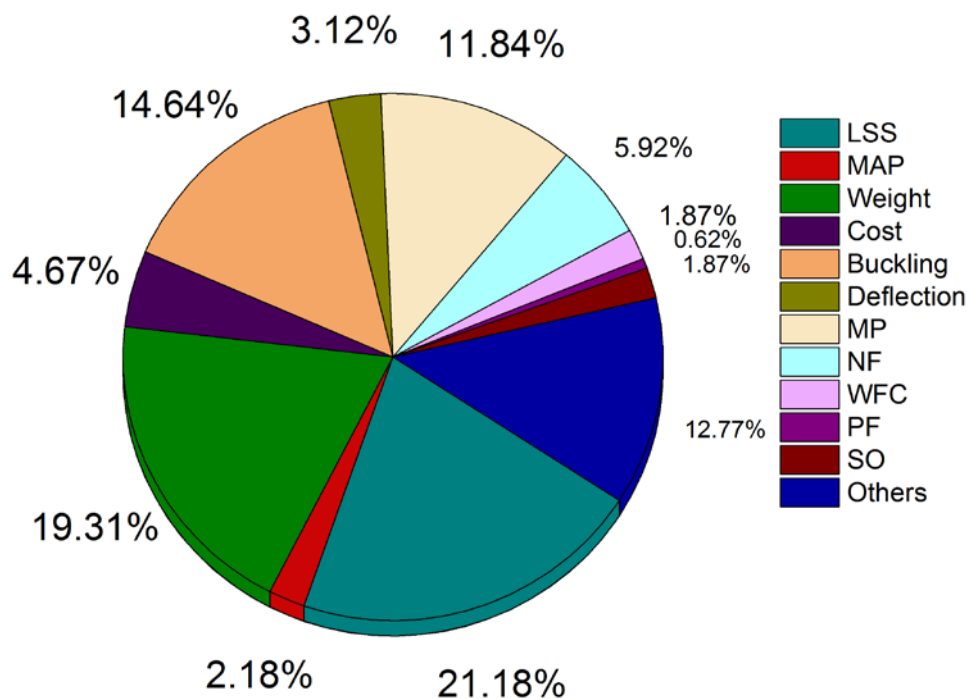

Fig. 4 Proportion of each optimisation topic

Within these applications there are a number of ways to evaluate the fitness of the composite structures and each method significantly influences the computational time of the optimisation. The selection of the method can restrict the total number of fitness evaluations that can be implemented during the optimisation or if rapid techniques are used, can increase the importance of the computational expense of the Genetic Algorithm. The methods found in the literature are: finite element method (FEM), surrogate models, artificial neural networks (ANNs), analytical models and constitutive relationships, with the proportions shown in Fig. 5. It is found that over half of the papers, ~59%, utilise FEM with the other methods having a similar representation, at 8-13%. Despite its accuracy the FEM process is computationally expensive when compared with the other four methods as the function evaluation methods and restricts the number of function evaluations that can be used. It is found that when using FEM methods as the objective evaluation method, ~90% utilised less than the average function evaluation number, 60,000 total calls, which is a low value for a complex optimisation problem and is likely to be set due to the computational time constraint. Surrogate modelling or ANNs, used as a surrogate model, are introduced to be the fitness evaluation tool when the problem is too time consuming to utilise it. However, these methods often mean that the complexity of the problem is reduced, and the optimisation becomes more prescriptive, or accuracy might be lost, and the developed Pareto front may not relate to one developed using a more accurate method. It is therefore important to carefully consider the impact of different fitness evaluation methods; whilst FEM, or similarly more accurate methods, might provide more accuracy it reduces the number of function calls in most optimisation trials, and therefore a sub-optimal set of results might be found. Analytical or surrogate models reduce the accuracy or freedom of the model, but this might provide a set of optimal results that can be refined with a smaller number of FEM calls to adjust the final results.
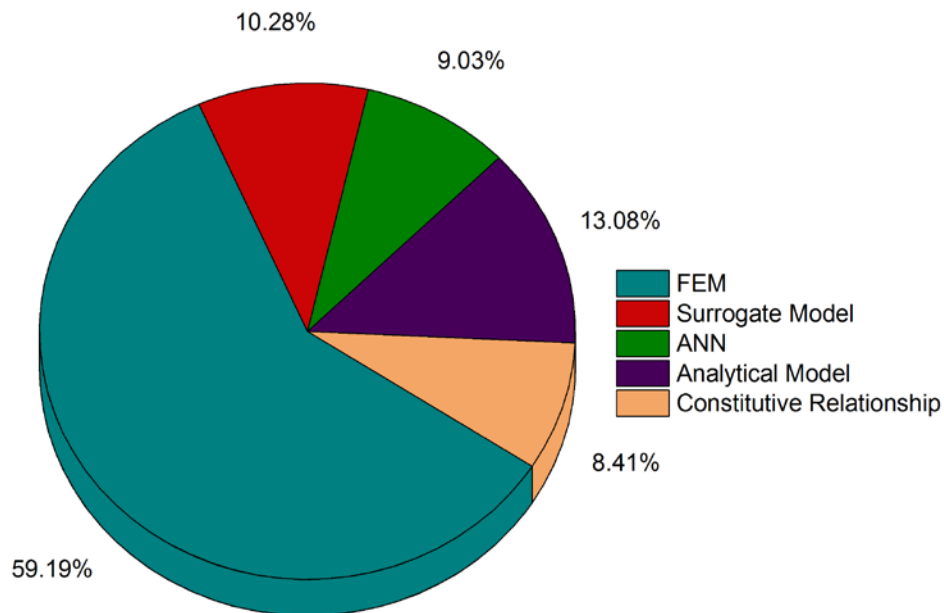

Fig. 5 Proportion of each fitness evaluation method

The most popular application for Genetic Algorithms is the optimal stacking sequence for composite laminates, investigated by ~21% of papers. This is because the ply angle significantly influences the performance of laminated composite structures and is affected by the loading conditions. All of the papers show improved performance of the composite

laminates after achieving the optimal stacking sequence. However, it is found that the optimal stacking sequence achieved by [6–15] are similar since they limit the potential fibre orientation angles to {0°, ±15°, ±30°, ±45°, ±60°, 90°} degrees. The rest of the papers set continuous fibre orientations, so any angle from 0°-90° is possible, or discrete fibre orientations with increments of one or five degrees. However, the fibre orientation is often fixed to guarantee the manufacturing feasibility, which implies that the optimal stacking sequence achieved from continuous or discrete fibre orientations are not applicable in the real-world composite structures [8–13]. This implies that the Genetic Algorithms are less useful for achieving optimal solutions since the best results can be achieved using an exhaustive search as these fixed patterns of the fibre orientations significantly limit the search space. This is unless the stacking sequence optimisation is in conjunction with complex loadings or structural geometries, which is not the case for much of the literature. In these scenarios bi-level optimisation of laminated composite structures will be required to reduce the complexity in these hot spot regions as other the number of variables for the entire structure will become unsearchable.

The second largest proportion of the literature is focused on weight minimisation where most papers solved single objective optimisation problems. In these studies the relationships between the weight and other features of the composite structure remain unknown. This restricts the range of applications for the optimised results, making them less interesting for real-world applications, as the structures may underperform in other areas, though there are a few multi-objective optimisation investigations. The addition of these objectives is important as illustrated by Wang and Sobey [16] who find 643 design schema that achieve an improved strength to stiffness ratio compared with an existing design of a triaxial weave fabric composite when using a multi-objective optimisation. However, this number is reduced to 15 when an additional objective of areal density is added, making the results more realistic. The results also show another 17 design schema which have slightly higher areal densities but achieve significantly improved strength to stiffness ratios that would have been lost in a single objective optimisation but inform the structural designer of more options [3].

The third most popular optimisation topic in composite literature is buckling performance. Many of the studies implement surrogate models to effectively approximate the value of objective functions due to the high computational expense in simulating buckling. The method significantly reduces the computational time caused by FEM method. However, the results from most of them are not compared to those generated with FEM, so the accuracy of these results for optimisation remains uncertain. Optimisation of mechanical properties is the fourth most popular composite optimisation problem. In these cases increasing the performance of one mechanical property can easily reduce the importance of another such as in tensile/compressive strength, tensile/compressive stiffness and shear strength/stiffness. It is therefore important to perform multi-objective optimisation to improve the applicability of the results but this topic has the lowest number of multi-objective optimisation among the four popular topics.

2.2 Bibliometric review of composite optimisation

A key issue in the current composite optimisation literature is a generally poor documentation of the Genetic Algorithm parameters, making it difficult to evaluate the validity and consistency of the optimisation results. In this section a number of parameters considered to be

vital for replication of results or interpretation of performance are investigated and typical values from the literature are given to capture the current state of the art. Of the papers reviewed only ~62% of papers explicitly state the name of their employed Genetic Algorithms, shown in Fig. 6.
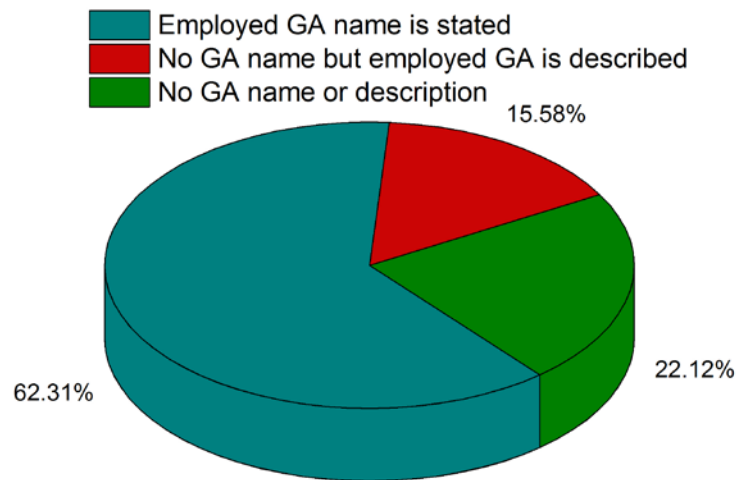

Fig. 6 Proportion of papers documenting the utilised Genetic Algorithm

This lack of documentation is perhaps related to a general consideration that any Genetic Algorithm can be used to solve any problem, and it is therefore suspected that many of the rest of the ~38% papers use old solvers. This is supported by the fact that ~80% of authors only published one composite optimisation paper in the last 10 years with limited authors repeatedly utilising these tools and becoming specialists, illustrated in Fig. 7.
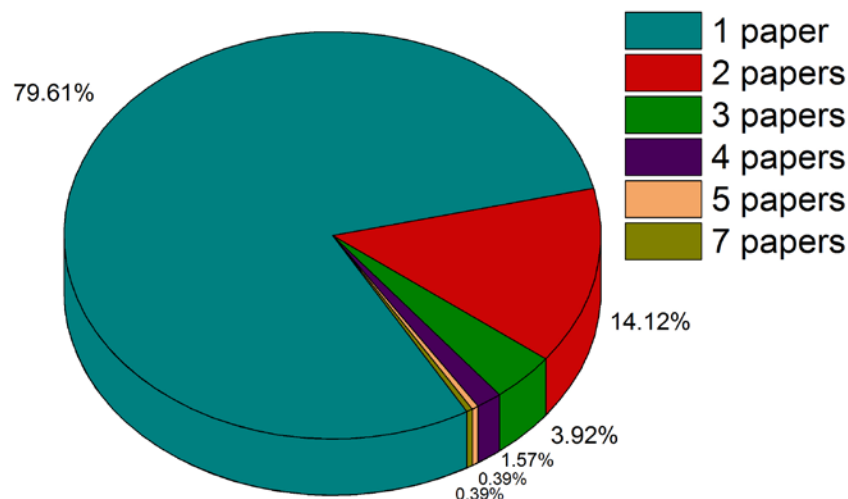

Fig. 7 Papers published on composite optimisation using Genetic Algorithms per author

A range of Genetic Algorithms are used from the most basic versions, which are similar to those developed in the 1970's, to BIANCA which is developed in 2010 and utilised by the same authors [17] in the composite optimisation literature. The proportion of the literature using each algorithm is illustrated in Fig. 8. Of those papers that document the name of the Genetic Algorithm 75% of the papers use: standard GA, MATLAB GA, NSGA-II, Micro-GA, Multi-island GA, BIANCA and hybrid GAs, where more than one mechanism works on the same population. It is shown that the NSGA-II and MATLAB GA toolbox have the highest

proportion making up 48% of all the frequently used Genetic Algorithms. This means that most of the literature uses older GAs or those that are less popular in the computer science literature. NSGA-II shows the top performance on a range of practical problems, making it an excellent selection especially where the optimisation problem has not been categorised or where there is more than one dominant characteristic to the optimisation problem, though there are now more recent algorithms showing both high diversity and convergence of search leading to better performance as general solvers.

Genetic Algorithms are good at solving problems with large search spaces but traditionally require substantial computational effort to find the true Pareto front. Therefore, other search methods are combined with Genetic Algorithms to push the solutions to the real optimal solutions with an acceptable computational cost, such as particle swarm optimisers (PSOs) and local search methods. NSGAIILS [18], BCE [19] and MTS [20] are good examples for hybridisation with local search methods. In the composite literature PSOs are the most frequently used algorithm in the hybridization with Genetic Algorithms, used in six papers of the total 29 papers, followed by a local search method, such as the Levenberg-Marquardt method that is used in three papers. However, of the hybrid algorithms ~59% use unknown Genetic Algorithms. Furthermore, only three papers perform benchmarking of hybrid Genetic Algorithms and therefore to assess the capability of these hybrid Genetic Algorithms for solving composite optimisation problems becomes difficult.
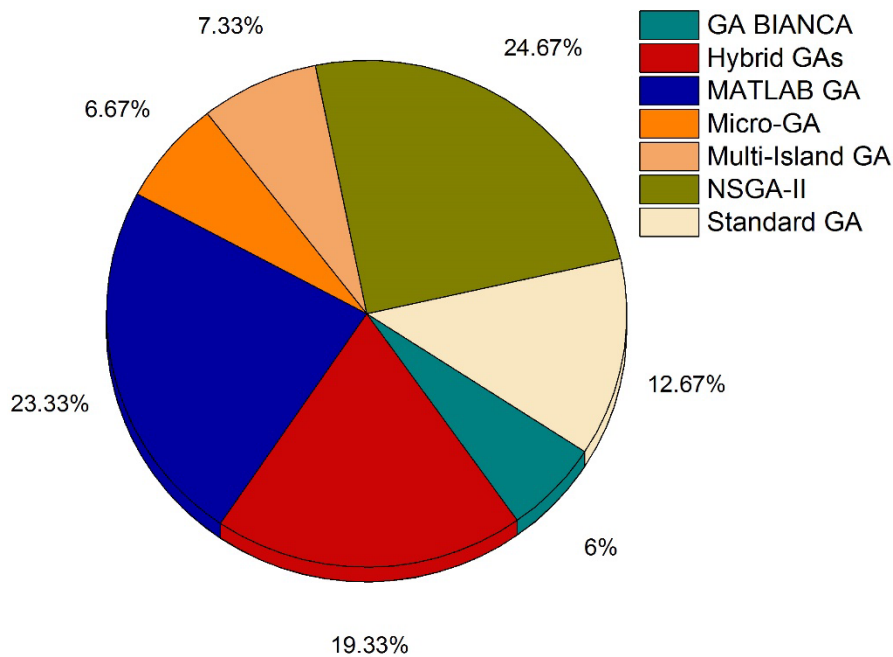

Fig. 8 Frequently utilised Genetic Algorithms in the composite optimisation literature

The genetic operator types and rates have a significant influence on the performance of solvers as it is important to find a balance between the exploration and exploitation ability of the algorithms in order to achieve the global optimum [21]. Exploration means exploring the entire search landscape as comprehensively as possible, which is accomplished by mutation operators and specific diversity retaining mechanisms. While the exploitation is accomplished by the crossover operator, which pushes the search towards the current optimal solution. Most developments in the evolutionary computation field focus on increasing the convergence of the algorithm. More detailed information about the genetic operators are often not included, as

shown in Fig. 9; only ~34% of the papers state the type of selection mechanism, ~26% give the mutation operator, ~27% document the use of elitism and ~29% provide the type of crossover mechanism that they use. In terms of repeatability of these results ~58% of the 321 papers miss the crossover rate and ~61% don't state the mutation rate, though many modern algorithms are not sensitive to these specific values and the tuning performed in the original papers tends to provide close to optimal performance across most problems.

Of those papers that do incorporate this information the most popular used selection, crossover and mutation operators are roulette wheel/tournament selection, single point crossover and gene swap mutation, which indicates binary coded Genetic Algorithms developed in the 1980's and 90's. It is considered likely that the 38% of papers that did not state the Genetic Algorithm were also older solvers, defined here as those developed before 2002 as this was the year NSGA-II was developed which is the oldest Genetic Algorithm that still shows top performance on many practical problems and is frequently considered in the evolutionary computation benchmarking. Approximately 1% of the literature does not include mutation operators, meaning that there are no mechanisms to retain the diversity of the search, and ~73% don't mention elitism mechanisms, a key element in improving the performance of genetic algorithms since 1975. The missing genetic operator type makes it difficult to assess the validity of the results or to characterise why a given algorithm may have been successful. This means that even if the optimisation proves to be successful, with quick convergence and accurate results, the information about the type of solver optimal to this problem is lost and so it is hard to replicate the results or to generate new algorithms to solve similar, but more difficult, problems in the future.
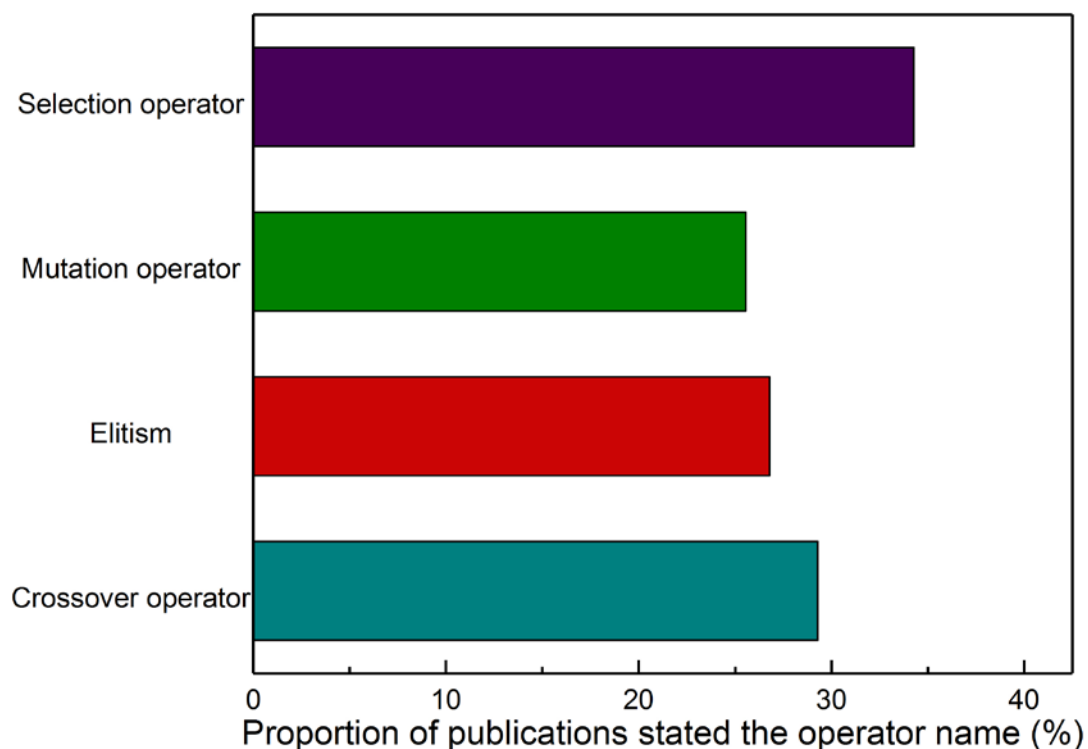

Fig. 9 Proportion of publications stating the name of the utilised genetic operators

In addition to the poor documentation of the mechanisms for the Genetic Algorithms the hyperparameters used within the algorithms are often not recorded. Only around 48% of papers

state the population size and generation number or the total number of fitness evaluations, illustrated in Fig. 10. It is suspected that many of the papers that do not state the information for the number of function evaluations use smaller numbers of calls to reduce the computational time but this may compromise the quality of the solutions. In addition the population size must be appropriate to the problem type and solver, which affects the diversity of the population and convergence rate. These values are vital in allowing a fair benchmarking and demonstrating improvements in methodologies, in addition the use of a small number of function evaluations may reduce confidence in the final result.

Of those papers documenting the hyperparameters the most frequently used number of population size is 50 and the most frequently used generation numbers are jointly 50 and 100 meaning that the most frequently used number of total function evaluations is 5,000 and ~93% of the 321 papers use less than 100,000 total function evaluations for solving complex composite optimisation problems. However, the computational science literature utilises 300,000 total function calls as standard to solve mathematical optimisation problems, where the optimal solutions has not been found even when using the state-of-the-art solvers. This indicates that the problems selected for testing in the composite science literature are less complicated than the benchmarking problems in the evolutionary computation literature or that the optimal designs may not have been found. Recent studies [3,4,22,23] show that the complexity of even simple composite optimisation problems is high in comparison to those found in evolutionary computation, possibly due the problems being dominated by multiple characteristics. However, there must be some realisation that not all of the applied literature will be seeking to find the optimal results and that an improvement on the current performance may be all that is required.

In terms of population size then the standard from most of composite optimisation literature is 50 and most of the evolutionary computation is 600 with some authors recently showing better results with 1500 population sizes [3]. In many cases engineers use a much smaller population size, but this may not contain the necessary diversity of species to ensure a full search of the design space in many cases, though it may help convergence. In many cases it is likely that the optimisation converges early in the generation cycle, supported by many of the documented Pareto fronts in the literature looking unresolved or where graphs of the fitness function remain static for long periods towards the end. This means these function calls are wasted and a larger diversity is required to search more of the space, a more powerful algorithm is required or the optimal results have indeed been obtained. For biological systems it is estimated that at least 10,000 population sizes are required to retain genetic diversity, so even these numbers, 600-1500, are small by evolutionary standards.
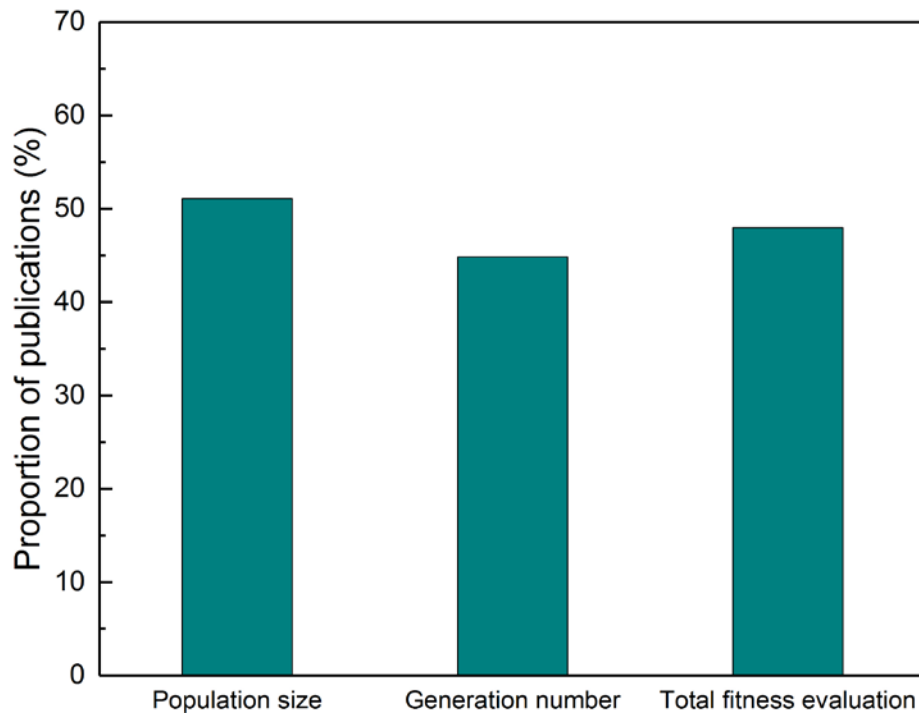
Fig. 10 Documentation status of hyperparameters

It addition to ensuring the function calls are utilised to match diversity and convergence it is important to have several independent run cycles to ensure that the final solutions have been achieved as Genetic Algorithms are stochastic solvers and the final solution is dependent on the initial population in a number of algorithms. However, only ~14% of papers state the number of repeated run cycles as shown in Table 1, where it is shown that a range of values are used from 2-100. It is suspected that many of the papers that do not provide a value for the number of repetitions give results from a single run meaning that the authors of these papers are not able to evaluate the robustness of the algorithm and it is difficult to determine the accuracy of the final results. Of those papers with a higher number of run cycles, two papers with high values are not independent cycles to determine statistical distributions of the solutions [24,25] but are used to generate the Pareto front using a weighted sum approach to solve a multi-objective problem by changing the weights 50 and 100 times to find the Pareto front, requiring many more function calls than a standard multi-objective approach. In evolutionary computation the standard value for benchmarking of Genetic Algorithms is 30 run cycles, of which 15 papers used this value or higher, to ensure enough data to determine the performance of a methodology. This is perhaps too high where the solution is the output, rather than a justification of the method, but recent papers are showing that many solvers require multiple attempts to find the entire Pareto front [3].

Table 1 Documented distribution for the number of independent run cycles

No. of independent run cycles	No. of papers	References
Unrecorded	276	-
1	2	[24,25]
2	2	[26,27]
3	2	[28,29]
4	3	[30–32]
5	8	[9,33–38][39]
6-20	12	[40–47][48–51]
25	1	[27]
30	3	[15,53,54]
40	1	[55]
50	3	[56–58]
100	8	[59–66]

The optimisation problems are categorised into single objective, multi-objective and weighted multi-objective problems, which reduces the multiple objectives down to one objective by using a weighted sum approach. Multi-objective is generally considered to be the most interesting category since the results can be used by engineers to better understand the design space and from which engineers can pick designs that match to other considerations that may not be included in the optimisation. In addition the lessons for other researchers are increased by having the whole front, allowing more areas of interest to be published and allowing any areas of the search space that are difficult or impossible to find to be highlighted. An additional disadvantage is that the results from a weighted sum approach are highly dependent on the selection of the weights and it is hard to accurately select the correct weights between objectives to precisely achieve solutions, even for experts. However, many authors select this approach as, for the same optimisation problem, using single objective and weighted sum approaches can help to simplify the problem and more traditional Genetic Algorithms are able to solve these problems, especially if the design variable spaces are small and/or simple.
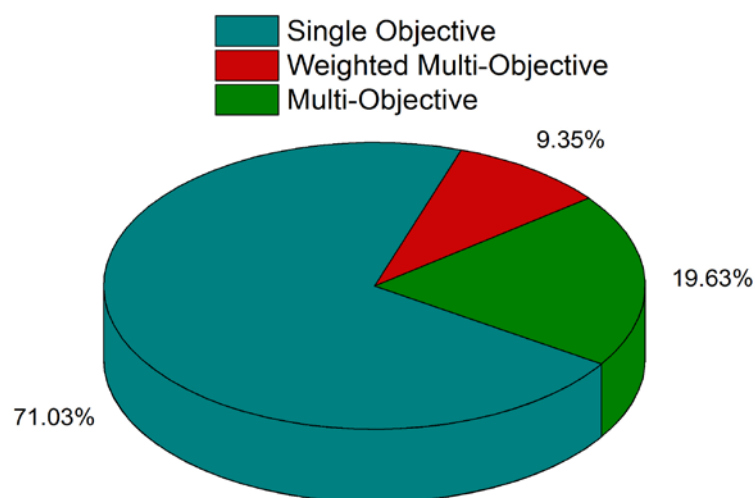

Fig. 11 Proportion of publications solving single objective and multi-objective problems

The composite optimisation literature relies heavily on single objective optimisation problems, illustrated in Fig. 11. In total ~80% of papers focus on single objective and weighted multi-

objective problems. This may explain the lack of more modern Genetic Algorithms in the composite optimisation literature, as these are designed with diversity retaining mechanisms rather than the convergence dominated algorithms required for single objective problems. For the single objective and weighted multi-objective problems the convergence information is not comprehensively documented, summarised in Fig. 12, where over half of the papers do not state the converge information. It is suspected that the final solutions in much of this ~56% of the single objective optimisation may stop at a local optima. In total ~29% of papers solve problems with multiple objectives, where ~9% of the total are weighted multi-objective optimisation and ~20% of the total are multi-objective optimisation.
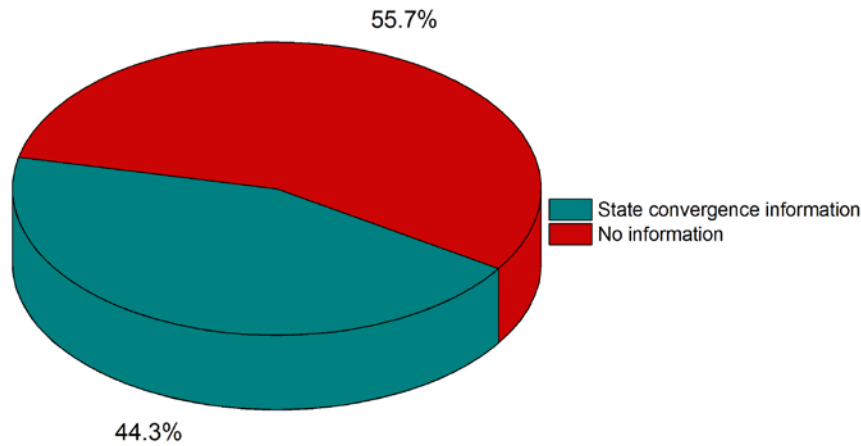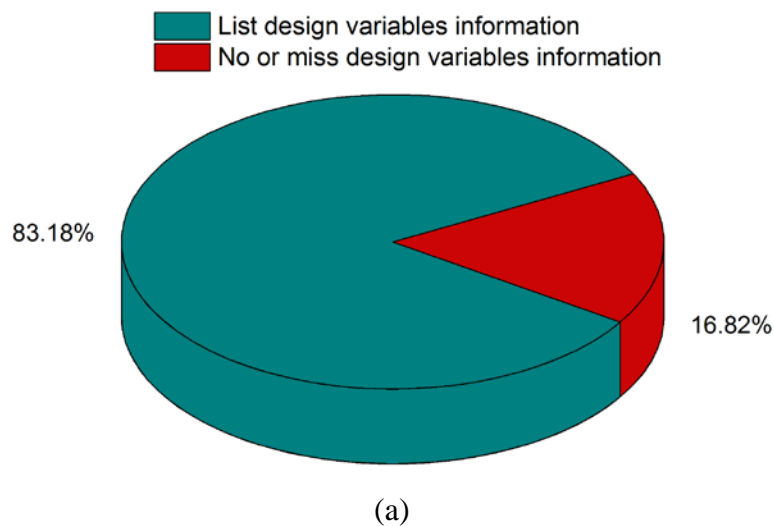

Fig. 12 Proportion of papers providing convergence information for single objective problems

In terms of the variable space then ~17% of papers don't have or miss documentation of the variables used or the range over the simulations are performed and ~31% of papers don't document the constraints, illustrated in Fig. 13. The maximum and median number of variables through all the 321 papers are 217 and 5 respectively and the maximum and median number of constraints are 16 and 3. These numbers indicate the complexity of composite optimisation problems, requiring state-of-the-art Genetic Algorithms with the correct hyperparameters. The missing design variables and constraint information makes it difficult to determine the accuracy of the optimisation results and the lack of details defining the relationship between the variable space and objective space makes it impossible to replicate.

In large-scale real-world composite structures there are potentially thousands of dimensions in the variable space. The main difficulties in solving these problems are the computational cost and the multi-modal shape of the search space, which contain many local optima. The computer science literature is developing methods to improve Genetic Algorithm performance on these problems using different methodologies, such as applying compression and decompression on the encoding chromosome to reduce the search space [67], parallel implementations of Genetic Algorithms on Central Processing Unit (CPU) and Graphic Processing Unit (GPU) [68–70] and developing a customised Genetic Algorithm for finding near-optimal solutions [71], to reduce the complexity of the problems and to improve the computational efficiency. The problems being tackled can contain billions of variables but are still flat in comparison to real-world problems, with limited multi-modality, and reduce the complexity by looking at single objective problems. In the composite optimisation literature there is still limited literature

discussing and developing methodologies for solving these large-scale optimisation problems, limiting the practicality of these problems.

In the 29% of papers that solve problems with multiple objectives 15 papers solved problems with more than 2 objectives. The highest number of objectives is 21 and second highest is 6 but in both cases one fitness function is formed using a weighted average approach, the rest of the papers use 3 objectives. 8 of the 15 papers generate a Pareto front surface to illustrate the optimal solutions for the 3 objectives including one paper which generates a Pareto front surface by tuning the weights. There are 8 papers which use a weighted average approach to solve these problems. Solving many-objective optimisation problems, defined as problems with 4+ objectives, provides additional challenges as the solutions must simultaneously satisfy several design requirements. The main difficulty in solving many-objective optimisation problems is that with the additional objectives it is less likely that one solution dominates the other solutions in the population. While the complexity of solving these problems is difficult, they provide the most interest as the additional complexity means that humans struggle to make rational decisions about the design but where computational analysis can be most helpful in visualising the trade-off between objectives. There is a strong focus in computer science literature for improving the performance of algorithms on problems with four and more objectives and several Genetic Algorithms for solving many-objective optimisation problems have been developed, such as cMLSGA [23], U-NSGA-III [72], HEIA [73] and BCE [19]. However, the test functions for these problems are still relatively few and biased towards certain types of search space. This means the performance of algorithms on these problems is particularly difficult to relate to composite structural optimisation and there must be more emphasis on developing benchmarking problems of this complexity in the composite structures field in the future.
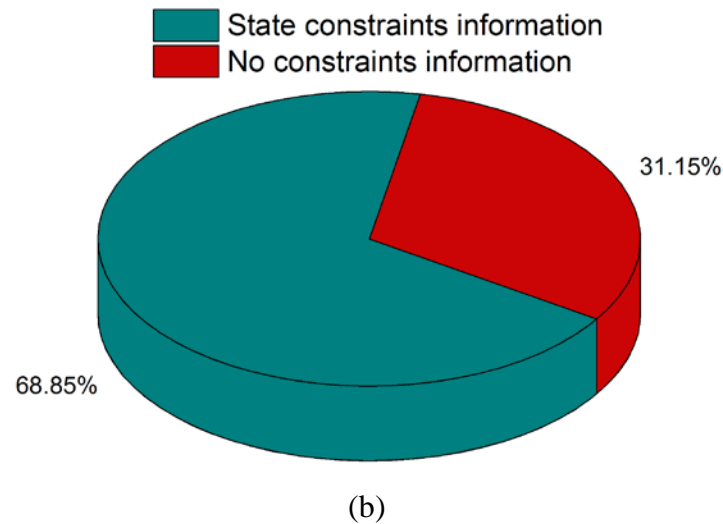

(a)

State constraints information
No constraints information

31.15%

68.85%

(b)

Fig. 13 Proportion of papers stating (a) design variables information (b) constraints information

2.3 Multi-objective optimisation in composites structures

Pareto fronts provide the set of non-dominated solutions for a multi-objective problem, with an example illustrated in Fig. 14. This is based on Pareto optimality first proposed by Pareto [74] in 1906 which is the set of optimal results for a given problem. Each point represents an optimal design point, each with a different weighting for the objectives. This allows the objective space to be visualised and for users to quickly determine how changes to a composite design are reflected in performance. They can also provide a clear link between how the objective space relates to the variable space, for instance are points next to each other in objective space next to each other in variable space? If they are then small changes to the variables relate to small changes in the objectives and the space is easy to interpret for an engineer.

Finding the Pareto front reflects best practice in multi-objective optimisation with 61 papers illustrating the final Pareto fronts [15,24,31,36,46,75–130]. Of these 55 papers perform multi-objective optimisation and 8 papers utilise a weighted average approach to generate Pareto fronts, including two papers which present both methods across different problems, preferring the weighted sum approach for the 3 objective problem. The proportion of papers finding the Pareto front in solving multi-objective optimisation and weighted multi-objective optimisation problems are summarised in Fig. 15. Weighted multi-objective optimisation generate Pareto fronts by repeatedly changing weights in repeated runs, for example the weights 1 and 0 are run followed by 0.99 and 0.01 through to 0 and 1, and all of these points are plotted as the Pareto front. Although the weighted sum approach is a straightforward implementation since single objective Genetic Algorithm can be utilised through minor modifications and weighted multi-objective Genetic Algorithms are efficient, they are not recommended as all of the Pareto-optimal solutions cannot be achieved through this method especially when the real Pareto front is non-convex and the process is slow to achieve a good density of points. For example a typical Pareto front would contain 400-1000 points from one simulation which would require the same number of repetitions from a weighted sum approach. The reason why multi-objective optimisation papers are performed that don't provide Pareto front is unknown. However, these Pareto Fronts are complex to find and require advanced Genetic Algorithms to

produce good results on modern problems, for example NSGA-II or more recent but most modern Genetic Algorithms are developed to find these Pareto Fronts. These modern algorithms will not work properly with single-objective or weighted multi-objective approaches, so it is important that more optimisation is performed using multi-objective problems so that more modern algorithms can be used in addition to the beneficial information these Pareto Fronts provide.
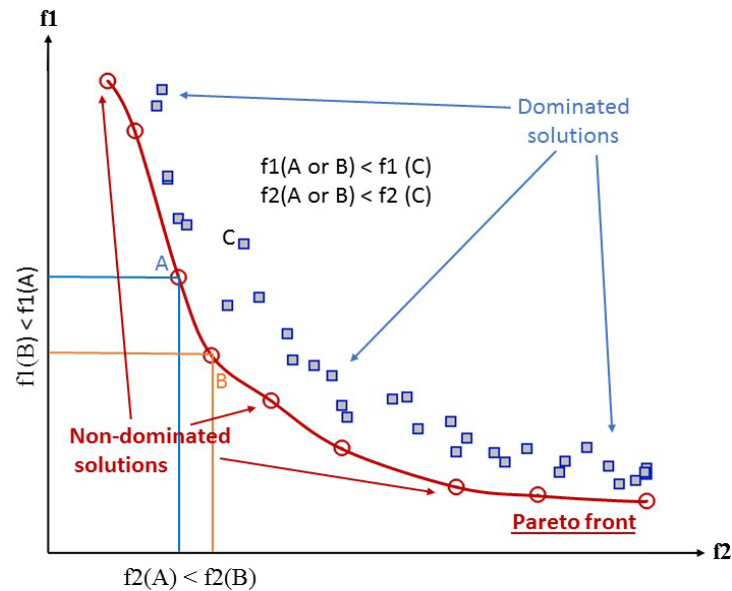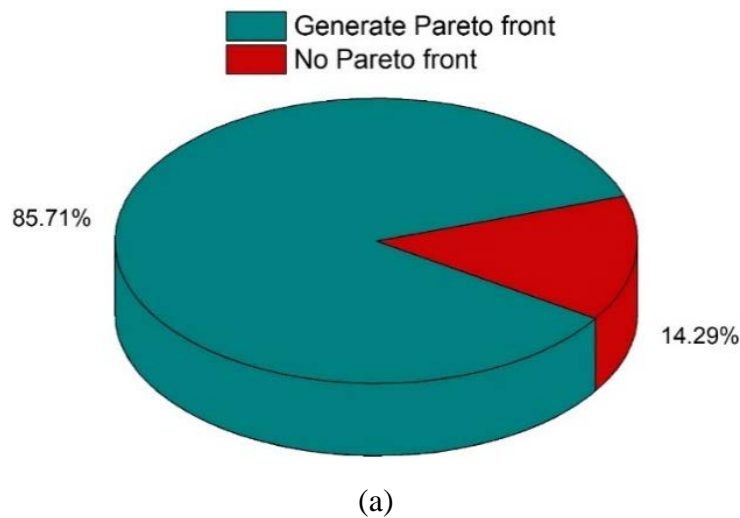

Fig 14. Illustration of a Pareto front, including non-dominated and dominated solutions, on a bi-objective minimisation example
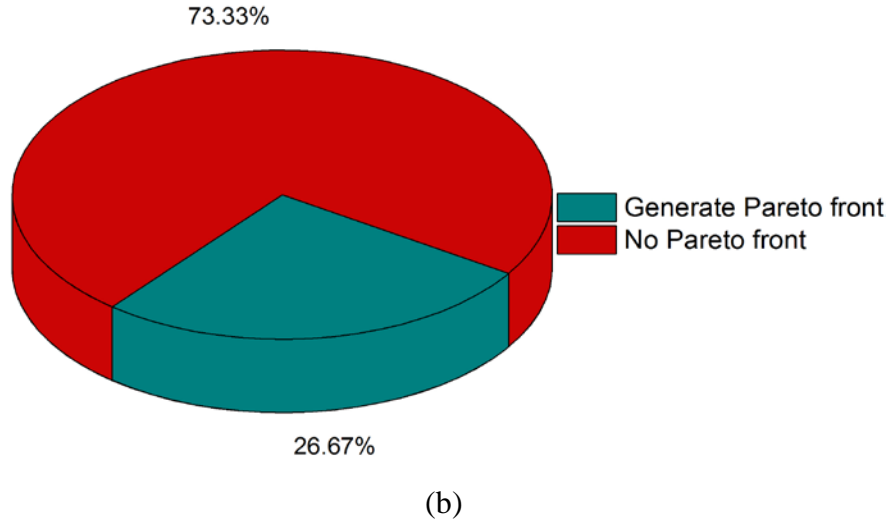

(a)

(b)

Fig 15. Proportion of papers generating Pareto fronts when solving (a) multi-objective problems (b) weighted multi-objective problems

The final Pareto fronts should have three basic elements [131]: (1) the best achieved Pareto front should be as close as possible to the true Pareto front; (2) the best achieved Pareto front should be uniformly distributed and diverse as wider as possible in order to provide the decision-maker a true picture of trade-offs; (3) the best achieved Pareto front should capture the entire spectrum of the Pareto front. However, for engineering optimisation, as for most real world optimisation, the true Pareto front is not known. Wang et al. [3,132] propose a mimicked inverted generational distance (mIGD) approach inspired by the IGD value calculated in the evolutionary computation literature to evaluate the accuracy and diversity of Pareto fronts. They combine all of the Pareto front solutions from all of the independent run cycles, from each of the algorithms that are benchmarked. Non-domination and duplication checks are implemented on this combined Pareto front set to filter it and create a 'true Pareto front'. They then calculate the mIGD values of the obtained Pareto fronts from each algorithm to this mimicked 'true' Pareto front to assess the performance of each benchmarking algorithm and whether the obtained Pareto front has been resolved, mIGD is defined in equation 1,

$$mIGD(O, M^*) = \frac{\sum_{v \in M} d(v, O)}{|M^*|},$$ (1)

where M^* is a set of points along the mimicked Pareto front, O is a set of points on the currently obtained Pareto front, v represents each point in the set M^* and $d(v, O)$ is the minimum Euclidean distance between v and the points in O; lower mIGD values reflect a better quality and diversity of the obtained Pareto front.

Another indicator which can be utilised to evaluate the performance of the obtained Pareto fronts is Hyper Volume (HV). This metric is more representative in determining the diversity of the population that has been found and using the IGD and HV together give a strong indication of how an algorithm has performed on a problem both in terms of convergence and diversity. The Hyper Volume demonstrates the volume of the objective space between the

obtained Pareto front points and a predefined reference point [133]. A fast way of calculating the HV is demonstrated in [134]. Both the HV and IGD are sensitive to the selection of reference points. This is especially the case for the IGD indicator as it requires a large number of uniformly distributed reference points to guarantee the accuracy. However, HV has problems when sorting objectives with different ranges [134]. In the benchmarking of Genetic Algorithms it is best practice to use both indicators to evaluate the solvers with IGD indicating the accuracy of the obtained Pareto fronts and HV the diversity.

2.4 Review of available Genetic Algorithm benchmarks on composite structures

The proportion of papers performing benchmarking is shown in Fig. 16. Of these papers only ~13% of papers compare different methods, or 43 papers [25,27,28,34,36,37,41,45,46,48,51,54,56,57,59,60,62–66,75,122,135–154], which means that the characteristics of most problems are difficult to determine and the selection of future algorithms on similar problems is not possible. This lack of documentation makes algorithm selection and development difficult, moving the literature towards general solvers.

Fig. 16 Proportion of papers performing benchmarks

The evolutionary computation literature benchmarks algorithms across a range of problems, for static optimisation this is normally dominated by whether problems are constrained or unconstrained. Wang et al [3] demonstrates that the triaxial weave fabric problem is not dominated by these characteristics and that algorithms performing well on these problems perform extremely poorly. The highest performing solvers are general solvers with good diversity of searches, indicating that a number of characteristics are important to these problems and/or that diversity of search is important. This indicates that the benchmarking in the evolutionary computation literature cannot be used for selecting the correct algorithms in the composite optimisation problems until these problems are better understood. Therefore, it is necessary to perform benchmarking on composite problems to allow the selection of the

correct Genetic Algorithms and increase their use. Of those that perform benchmarking 26% of them compare different Genetic Algorithms and 74% compare Genetic Algorithms with other non-GA solvers, such as Particle Swarm Optimisation (PSO) and Simulated Annealing (SA), shown in Fig. 16. Of the 12 papers that benchmark different Genetic Algorithms, micro-GA is presented as the best algorithm three times but all of these benchmark the micro-GA against only the standard Genetic Algorithm. NSGA-II and hybrid NSGA-II are stated as the best algorithms in two papers respectively. Montemurro et al. [140] benchmarked GA BIANCA with the MATLAB GA and found that the GA BIANCA achieved better optimisation results but that the MATLAB GA is more efficient. It is difficult to determine best practices through these benchmarking results.

The rest of the 31 papers benchmark Genetic Algorithms with non-Genetic Algorithms. Four papers state that hybrid Genetic Algorithms outperform the Genetic Algorithms and the non-GA algorithms that the hybrid algorithms are made from [37,137,148,150]. There are five papers benchmarking non-Genetic algorithms with the standard Genetic Algorithm [34,45,56,59,143] and 10 papers benchmarking against unspecified Genetic Algorithms [60,62,63,138,144,147,151–154]. Of these papers 14 [25,27,45,59,62,66,122,141–143,146,152–154] state that the non-Genetic algorithms perform better. Eight papers [28,34,41,56,57,75,145,151] result in a draw between the Genetic Algorithms and the non-GA optimisation approaches according to the outcomes from the benchmarking metrics in each paper. Two of the Genetic Algorithms win the benchmarking among these papers [48,63]. However, 11 of the 24 papers are missing the GA hyperparameters and only four papers include a modern Genetic Algorithm, NSGA-II, making it hard to assess whether these benchmarks are fair tests. It is therefore necessary to extend these benchmarking exercises to composite optimisation problems utilising a more modern selection of Genetic Algorithms and a full documentation of the hyperparameters.

3 State-of-the-art in the Evolutionary Computation literature

3.1 A brief timeline of major Genetic Algorithm developments

A brief timeline of the major developments in Genetic Algorithms is summarised in Fig. 17. The frequently utilised Genetic Algorithms in composite optimisation literature are highlighted in red, those that are less common in this literature are highlighted in green and those missing from the literature are in black. To support this review we would refer the reader to [21] for an excellent introduction and clear explanation of the standard Genetic Algorithm mechanisms and Konak et al. [155] for a detailed and clear explanation of the mechanics for most of the main multi-objective Genetic Algorithms developed before 2006.

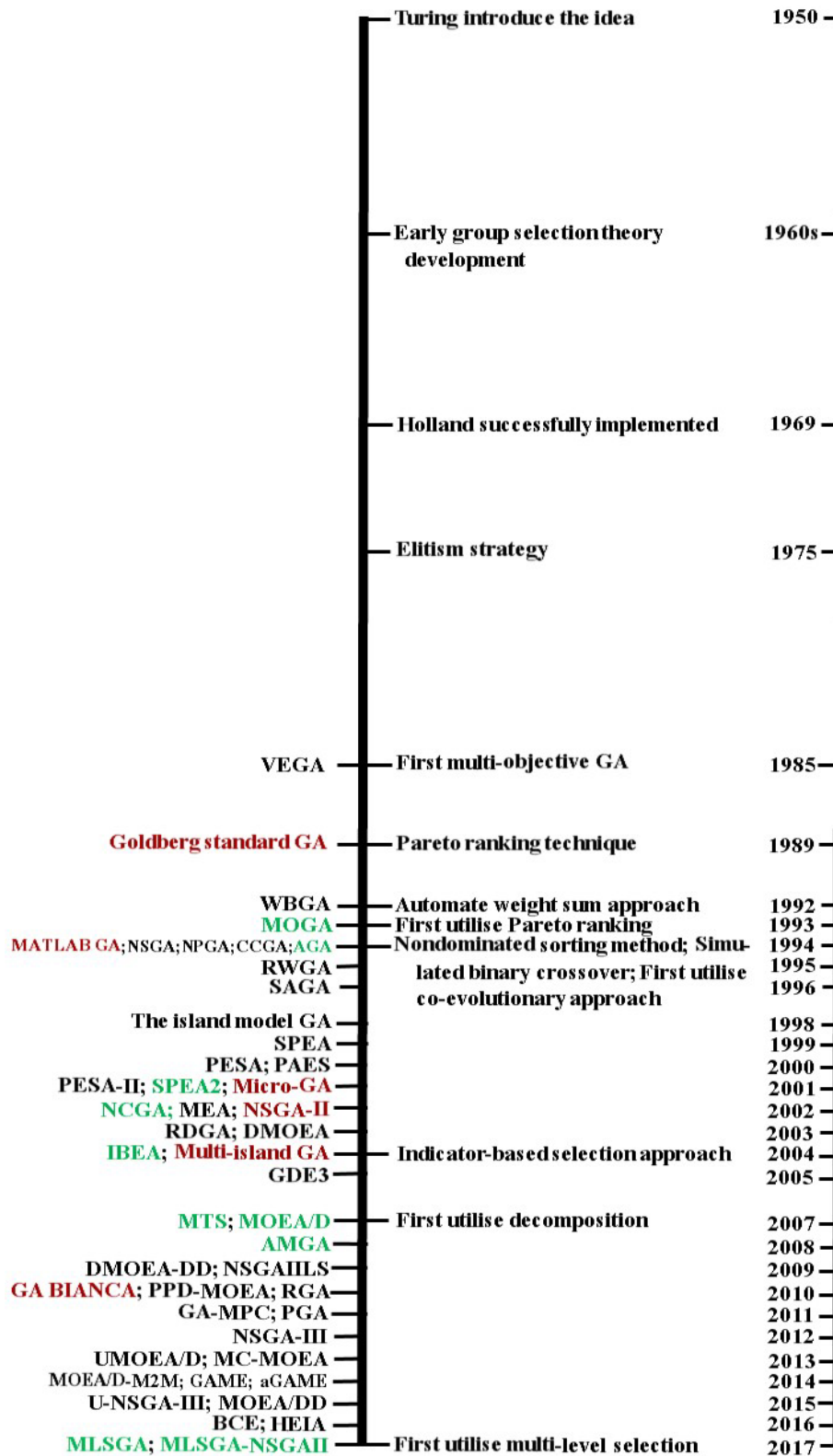| Algorithms (left)		Methods (right)	Year
		Turing introduce the idea	1950
		Early group selection theory development	1960s
		Holland successfully implemented	1969
		Elitism strategy	1975
VEGA		First multi-objective GA	1985
Goldberg standard GA		Pareto ranking technique	1989
WBGA		Automate weight sum approach	1992
MOGA		First utilise Pareto ranking	1993
MATLAB GA; NSGA; NPGA; CCGA; AGA		Nondominated sorting method; Simulated binary crossover; First utilise co-evolutionary approach	1994
RWGA			1995
SAGA			1996
The island model GA			1998
SPEA			1999
PESA; PAES			2000
PESA-II; SPEA2; Micro-GA			2001
NCGA; MEA; NSGA-II			2002
RDGA; DMOEA			2003
IBEA; Multi-island GA		Indicator-based selection approach	2004
GDE3			2005
MTS; MOEA/D		First utilise decomposition	2007
AMGA			2008
DMOEA-DD; NSGAIILS			2009
GA BIANCA; PPD-MOEA; RGA			2010
GA-MPC; PGA			2011
NSGA-III			2012
UMOEA/D; MC-MOEA			2013
MOEA/D-M2M; GAME; aGAME			2014
U-NSGA-III; MOEA/DD			2015
BCE; HEIA			2016
MLSGA; MLSGA-NSGAII		First utilise multi-level selection	2017

Fig. 17 Timeline of the main Genetic Algorithm developments with Algorithms on the left and methods on the right

Genetic Algorithms are inspired by Darwin's theory of evolution and Mendel's inheritance theory. The idea is that by mating the fittest individuals in a population the children of the next generation are fitter, on average, than the last. The initial inspiration was provided by Turing

in 1950 [156] who outlines the potential for a biologically inspired learning machine including the ability to mutate and which would exhibit survival of the fittest. In the 1960s early group selection theories were developed in an attempt to simulate and study natural selection. The first successful implementation of a Genetic Algorithm was performed by Holland and is summarised in 1969 [157]. These initial studies inspired others to develop algorithms for use in optimisation, or implement additional mechanisms, often moving away from the initial biological inspiration. Elitism [158], which was developed in 1975, is one of the most important mechanisms as it is used to retain the best solutions between generations, substantially improving the performance of these algorithms. This increases the search speed as the best solution in each generation is guaranteed to survive but is also a core strategy in multi-objective solvers to keep non-dominated solutions.

The first multi-objective Genetic Algorithm, Vector Evaluated Genetic Algorithm (VEGA) [159] was developed in 1985, which utilises a single objective GA to approximate the Pareto-optimal set, where the population is divided into K subpopulations for K objectives with each subpopulation corresponding to one objective. All of the subpopulations are then combined to perform fitness-proportional selection, crossover and mutation using a single objective to determine the fitness before creating a new population which is then randomly sorted into subpopulations. These different processes were combined by Goldberg [160] who developed what can be considered to be the algorithm closest to a standard version of a multi-objective Genetic Algorithm utilising classical mechanisms. This was developed alongside the first Pareto ranking approach proposed for multi-objective optimisation in 1989, which is at the core of many modern Genetic Algorithms, the niching algorithms, and this process is based on Pareto optimality.

In 1993 Hajela and Lin [161] developed a simple method to find the Pareto optimal front using a Weight-Based Genetic Algorithm (WBGA) to automate the process of changing weights to find the Pareto front in multi-objective optimisation. Fonseca and Fleming [162] later develop the first Genetic Algorithm utilising Pareto ranking and niching techniques to solve multi-objective optimisation problems, called multi-objective Genetic Algorithm (MOGA), in 1993. This is integrated into MATLAB through the popular MATLAB GA Toolbox by Chipperfield and Fleming [163] in 1994, which is still currently in many optimisation problems. In the same year, an Adaptive Genetic Algorithm (AGA) [164] is developed to reduce the influence of crossover and mutation probability settings which changes the parameters itself to adapt the optimisation problem. It achieves much higher performance in solving multimodal problems when benchmarked with the standard GA in a series of cases. This self-tuning approach inspires the Self-Adaptive Genetic Algorithm (SAGA) [165], developed in 1996. However, both AGA and SAGA need extra internal parameters, such as the frequency of tuning for crossover and mutation probability, where the performance of the algorithms are sensitive to these factors. Extending the ideas from Goldberg, the non-dominated sorting method [166] and simulated binary crossover [167] are first developed in 1994 as NSGA. These improve the Pareto ranking technique and create a shift from binary encoding to real value encoding. In 1998, the island model Genetic Algorithm [168] is developed that divides the population into sub-population islands and allows the migration of individuals between islands, improving the diversity of the solutions and is among the first hierarchical algorithms. An effective Genetic Algorithm which combines the characteristics from several previous multi-objective evolutionary algorithms is developed in 1999, which provides leading performance in the 1990s, called Strength Pareto

Evolutionary Algorithm (SPEA) [169]. It is upgraded to SPEA2 [170], which is still used in the composite optimisation literature, and updated to the Neighbourhood Cultivation Genetic Algorithm (NCGA) [171]. micro-GA [172] is introduced in 2001 to reduce the number of function calls, which is based on small population sizes with an external population for updating and re-initialisation. The micro-GA is improved in 2008 as the Adaptive Micro-GA (AMGA) [173].

Modern Genetic Algorithms are considered here to be from NSGA-II, as this is the oldest currently competitive algorithm. NSGA-II [174] is an improvement of the non-dominated sorting method, which is developed in 2002. Multi-island Genetic Algorithm [175], developed in 2004, extends the concepts from the island model Genetic Algorithm. It is frequently used in the composite optimisation literature since there is a commercial optimisation software based on this algorithm produced by SIMULIA, Isight, which easily links to ABAQUS, though the software contains NSGA-II, AMGA and NCGA. In 2004, the indicator-based selection approach is first developed in Indicator-based Evolutionary Algorithm (IBEA) [176], which improves the efficiency of finding the final Pareto front. Rather than using dominance to evaluate the achieved solutions, the indicator is used to reflect the diversity and quality of current Pareto front in each generation and pushes the solutions to the true Pareto front. The indicator-based selection approach is increasingly investigated in recent years, and is included in the top performing algorithms Bi-Criterion Evolution (BCE) [19] and Hybrid Evolutionary Immune Algorithm (HEIA) [73].

Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) is developed by Zhang and Li [177] in 2007 based on the Tchebycheff decomposition approach [178]. This algorithm demonstrated top performance on unconstrained optimisation problems at the Congress on Evolutionary Computation 2009 benchmarking competition [2], on a set of problems developed by the same authors, and a number of subsequent benchmarking exercises. Various groups around the world are working on upgrading these mechanisms leading to a range of top performing algorithms based on these mechanisms for specific characteristics such as MOEA/D-M2M [179], which improves the diversity of the population by decomposing multi-objective problems into a number of simple multi-objective sub-problems which performs well on solving multimodal problems. Two other approaches are MOEA/D-PSF and MOEA/D-MSF [180] that propose two improved scalarizing functions: penalty scalarizing function (PSF) and multiplicative scalarizing function (MSF), to replace the Tchebycheff scalarizing approach in MOEA/D to improve the balance between the convergence and diversity of the algorithm. In 2017, a Genetic Algorithm based on Multi-Level Selection theory, an evolutionary theory first proposed in 1999 [181], which uses individual and collective evolution mechanisms is developed by Sobey and Grudniewski [182]. The algorithm is hybridised with a range of top performing algorithms to achieve top performance across a wide range of test problems including constrained, unconstrained and discontinuous problems and importantly is one of a limited number of diversity first convergence second Genetic Algorithms [183]. The addition of a co-evolutionary inspired mechanisms lead it to being benchmarked as the top performing general solver [23].

3.2 Categorisation and description of mechanisms for the leading Genetic Algorithm methodologies

Of the frequently used Genetic Algorithms in the composite optimisation literature some of those considered to be state-of-the-art in various benchmarking exercises in evolutionary computation are not visible. Therefore, a brief review of the current state-of-the-art in evolutionary computation is performed to encourage further benchmarking of these algorithms. These are performed by splitting the algorithms into 4 broad categories: niching, decomposition, co-evolutionary and multi-level selection algorithms. Codes for: cMLSGA, HEIA, NSGA-II, U-NSGA-III, MOEA/D, MOEA/D-MSF, MOEA/D-PSF, MOEA/D-M2M, BCE, IBEA and MTS are available for benchmarking in C++ and python from multiple sources including[1].

3.2.1 Niching algorithms

In ecology a niche is defined as the fit of a species living under specific environmental conditions. In Genetic Algorithms, niching describes the formulation of sub-populations in a population, where each sub-population responds to a sub-task of the optimisation problem. The introduction of niching techniques increases the diversity of the population and helps the Genetic Algorithms to improve the ability of solving multi-peak optimisation problems.

The niching technique was first proposed by Cavicchio [184] in 1970, where this method is based on the preselection mechanism meaning that the parent individuals can only be replaced when the newly created offspring individuals are higher in terms of fitness than the parent individuals, otherwise the parent individuals are retained. De Jong [185] further develops this preselection mechanism and proposes a niching technique based on the crowding mechanism. In this method 1/CF individuals, where CF is the Crowding Factor, are randomly selected to expel the individuals that are similar to the newly generated members. The similarity is calculated according to the Hamming distance, which is the number of different symbols at corresponding positions between the encoded strings of the two individuals. In 1987, Goldberg and Richardson propose a fitness sharing based niching technique [186]. The fitness of the individuals is adjusted in the population by using a sharing function that reflects the similarity between individuals to maintain the diversity of the population. The algorithm performs selection operations by adjusting the fitness in the later generations. This method is one of the most widely utilised niching techniques. Sequential niching [187] is developed to improve the performance of searching all of the optimal solutions since the technique is able to be included in any kind of GA. It is more efficient than the fitness sharing based niching technique and capable of locating all of the maxima in multimodal problems, requiring fewer function evaluations. Clearing based niching technique [188] is beneficial where there are significantly complex search spaces since the technique has a reduced complexity, meaning more efficiency, than the fitness sharing technique and can achieve significant improvements in the probability of finding all of the peaks over the fitness sharing technique, and is used to solve complex multimodal problems with several million local maxima.

NSGA-II is the most popular Genetic Algorithm using niching as it is a robust general solver with few hyperparameters and good diversity of population based on the fast non-dominated ranking. There have been a number of updates to this algorithm and the most recent version, currently version 1.1.6 [189], should be used to ensure the best performance. Simple Genetic Algorithms face difficulties in solving optimisation problems with multiple peaks since fitness proportional selection methods are utilised which reduce the diversity of the solutions. This is

---

[1] https://www.bitbucket.org/Pag1c18/cmlsga

because competition between individuals allows the best individual to rapidly dominate the entire population. However, the focus is at one global or local optimum which can lead to some optimal solutions being missed.
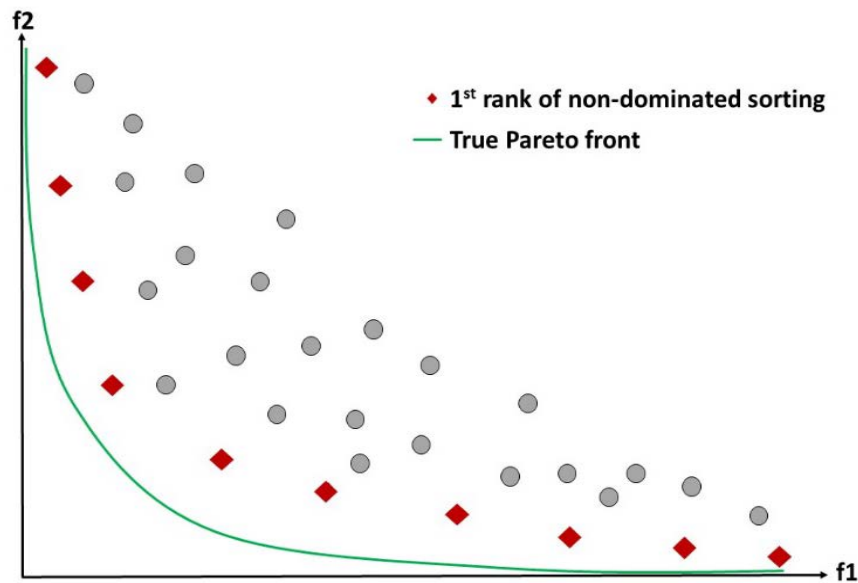

Fig. 18 Mechanism of NSGA-II [174]: non-dominated sorting

In the fast non-domination of solutions, the parent and offspring population are combined together and sorted by the fast non-dominated sorting procedure. The solutions that are not dominated by each other are categorised into the same non-domination rank. All individuals are classified into different non-domination ranks, where the previous ranks are temporarily discounted when formulating a new rank. The best half ranks are reserved in each generation, where the best rank of non-dominated solutions are shown in Figure 18. However, the total number of individuals in the reserved best half is often larger than the population size. Therefore, the crowding distance niching technique is implemented in the reserved rank to eliminate the individuals in the crowded area and maintain the same number of individuals. The crowding distance is evaluated on each individual in the rank. The objective functions are normalised and the crowding distance of the ith solution is the sum of the distances of two individuals on either side of the individual along each of the objectives, which is shown in Figure 19. These procedures are continued until the termination criterion is achieved. This algorithm has been upgraded to a variety of versions, such as Local Search based NSGA-II (NSGAIILS) [18] that effectively and accurately converges to the Pareto front by hybridising local search methods, NSGA-III [190] for solving many-objective optimisation problems and U-NSGA-III [72] that unifies NSGA-III to be suitable for mono-objective, multi-objective and many-objective problems.
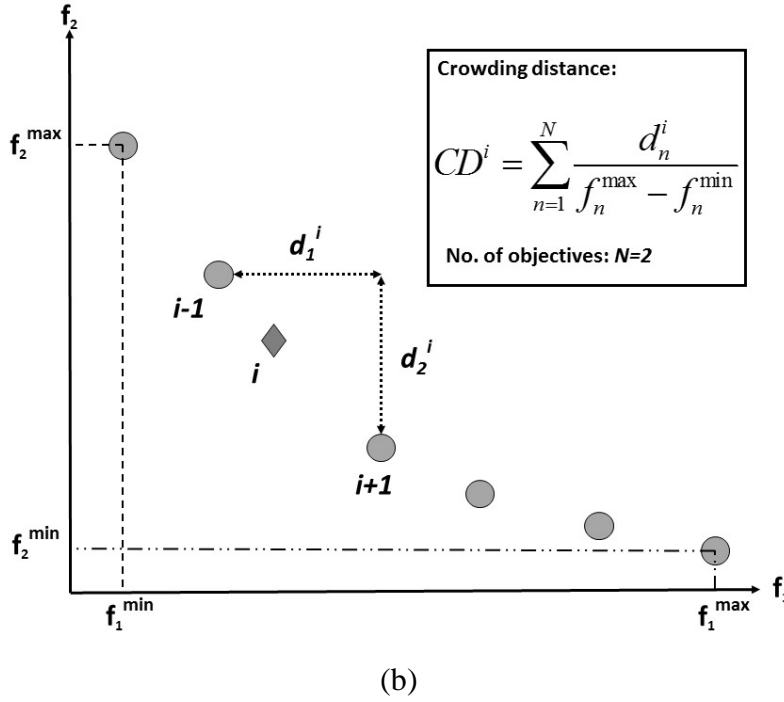
The figure shows a crowding distance diagram with f_2 on the vertical axis and f_1 on the horizontal axis. Points are marked at $i-1$, i, and $i+1$, with distances d_1^i and d_2^i indicated. Axis limits are labeled f_2^{max}, f_2^{min}, f_1^{min}, and f_1^{max}.

Crowding distance:

$$CD^i = \sum_{n=1}^{N} \frac{d_n^i}{f_n^{max} - f_n^{min}}$$

No. of objectives: *N=2*

(b)

Fig. 19 Mechanism of NSGA-II [174]: crowding-distance calculation

3.2.2 Decomposition algorithms

In decomposition algorithms, the population is divided into sub-groups that search different sub-regions of the entire search space. Decomposition methods are a popular approach in the recent evolutionary computation literature dominating the benchmarking of unconstrained problems; examples include: MOEA/D [177], MOEA/D-M2M [179], CS-NSGA-II [191], DMOEA-DD [192] and LiuLi [193]. These algorithms require additional setting parameters, such as the weight vectors in MOEA/D. These parameters influence the ability of the algorithm to find the optimal solutions and it is important to determine the optimal settings of these hyperparameters, or the performance is extremely poor.

As an example MOEA/D still shows top performance on unconstrained benchmarking problems. It works by initially creating a set of homogeneous distributed weight vectors and an empty external population for keeping and updating the non-dominated solutions. The multi-objective optimisation problem is divided into *N* weighted multi-objective optimisation sub-problems. These sub-problems are synchronously optimised in the algorithm. The Euclidean distances between any two weight vectors are calculated and the closest *M* weight vectors of the weight vector, γ, are defined as its neighbourhood. The neighbourhood can overlap and each weight vector has its neighbourhood. An initial population of solutions can then be randomly generated. Two individuals in the neighbourhood of weight vector γ are randomly selected to generate the offspring solutions through crossover and mutation. The offspring solutions are compared with their parents and neighbourhood of parents. If the newly generated solutions are better than their parents and neighbourhood of parents, they replace the previous solutions and the reference points of the weight vector and neighbouring solutions are updated. The non-dominated solutions of each sub-problem are combined to achieve the Pareto front. The mechanism of MOEA/D is illustrated in Figure 20.
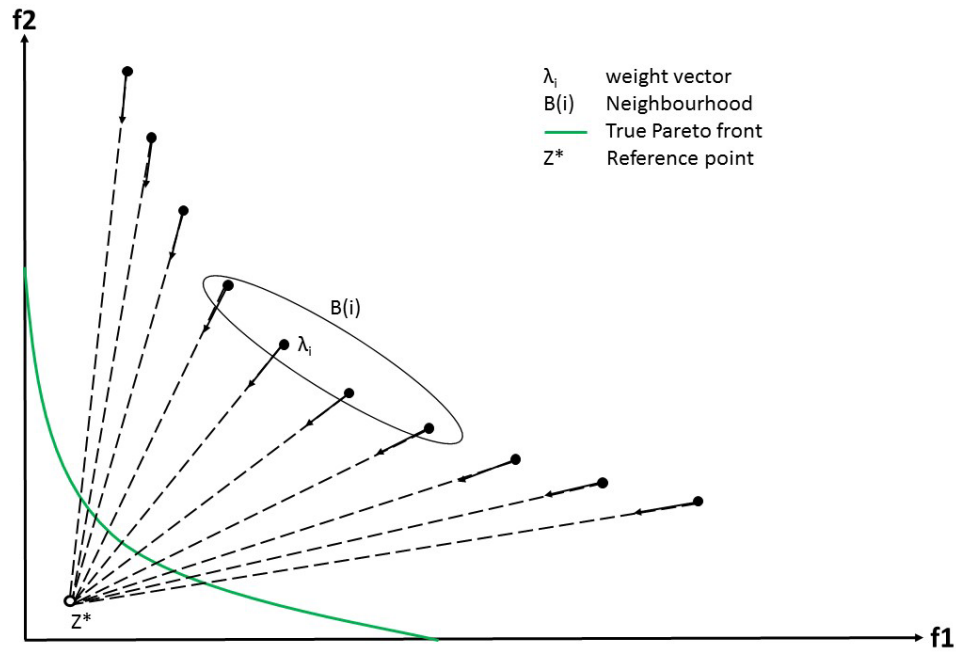
Fig. 20 Mechanism of MOEA/D [177]

MOEA/D, and its variants, requires a priori knowledge of the objective space or it can result in extremely poor performance and is not able to find points in negative regions. It is hard to obtain the required knowledge of the objective space for engineering optimisation problems before solving them, making these algorithms difficult to use, and this knowledge, and likelihood of using these algorithms, would be improved through better documentation of the literature. Therefore, these algorithms may not be (currently) suitable for solving composite optimisation problems. When solving discontinuous problems or constrained problems, these algorithms struggle with large gaps where there are no feasible solutions as the weight vectors point straight through the gaps and the individuals struggle to go around these spaces, resulting in a waste of computational power. In order to utilise these decomposition algorithms for engineering optimisation problems it will be necessary to simplify the tuning process or increase the knowledge about the current problems, focusing on problems which require strong convergence but where diversity of the population is less important.

3.2.3 Co-evolutionary algorithms

The origins of co-evolutionary theory are in the works of Darwin [194] who clearly describes the interactions between plants and insects. The term is formalised much later by Ehrlich and Raven in 1964 [195] to describe the coexistence of plants and butterflies, as one group cannot survive without the other, meaning that their evolution is intertwined. Groups of organisms can co-evolve in a number of different ways [196] such as: symbiosis [197], coadaptation [198], host-parasite [199] and hunter-prey [200]. These form into two broad forms: cooperation [201] , where organisms coexist and "support" each other or competition [202], where an "arms race" occurs between species and only the strongest may survive in the given environment.

The idea of introducing co-evolution into GAs is first proposed by Potter and De Jong in 1994 [201]. In these approaches multiple populations coexist and evolve in parallel. This usually takes the form of using distinct reproduction mechanisms in each population, effectively using two different algorithms at the same time, with a method to exchange data between them. The

sub-populations can operate on the same search space [203], or can be divided into several regions by problem decomposition or additional separation mechanisms [180]. The form of data exchange between groups depends on the type of co-evolution: competitive or cooperative. In cooperative co-evolution the information is shared by different species to form a valid solution when the problem is decomposed e.g. along the decision variable space [201]; or different sub-populations may cooperate to form the Pareto objective front with different subpopulations focusing on different regions [204]. In competitive algorithms different groups compete in the creation of new sub-populations [205], or populations [73], with fitter sub-populations gain a wider proportion of children in the next generation or via "arms race" where losing subpopulations try to counter the winning ones by adaptation [206]. The mechanisms are illustrated in Figure 21a and an example of a Pareto front from a co-evolutionary genetic algorithm is illustrated in Figure 21b.

In the literature multiple algorithms can be found which are inspired by co-evolution. Competitive-Cooperative Coevolutionary algorithm (COEA) [205] is a methodology inspired by symbiosis where cooperation and competition can occur. The cooperation improves the convergence while the competition retains the diversity in the population. The resulting methodology shows high performance on various dynamic problems [207] but the adaptability is limited. The utilisation of a NSGA-II like Pareto-dominance principle and decomposition along the variable space means that performance on irregular or discontinuous spaces and problems with strong variable-linkage is likely to be low. Two new co-evolutionary Genetic Algorithms were developed in 2016, Bi-Criterion Evolution (BCE) [19] and Hybrid Evolutionary Immune Algorithm (HEIA) [73] for solving multi-objective optimisation problems with both showing strong performance. In these algorithms cooperation is introduced between sub-populations. The population is divided into 'solution' sub-populations and 'test' sub-populations where different operations are performed on each of them. The individuals from the sub-population can be exchanged and there is cooperation between the sub-groups. The fitness of individuals in each sub-population is determined by comparing with the randomly selected individuals in the other sub-population. In BCE the sub-populations operate on the same search space and individuals are selected based on two fitness indicators: Pareto-based criterion and Non-Pareto-based. In this case the Pareto criterion increases the convergence using the standard Pareto dominance and the Non-Pareto fitness increases the divergence using the Hyper-Volume metric. The combination of these mechanisms leads to an overall improvement in diversity and provides particularly good performance on irregular search spaces, those with variable linkages and many-objective cases. A similar approach is utilised in Hybrid Evolutionary Immune Algorithm (HEIA) [73] where two different evolutionary methods are used in each sub-population, rather than separate quality indicators. At each generation the best individuals are moved to a shared pool and cloned before the sub-populations are recreated. This method shows strong convergence but good performance across quite a broad range of different problem types, though this is poor on imbalanced problems.
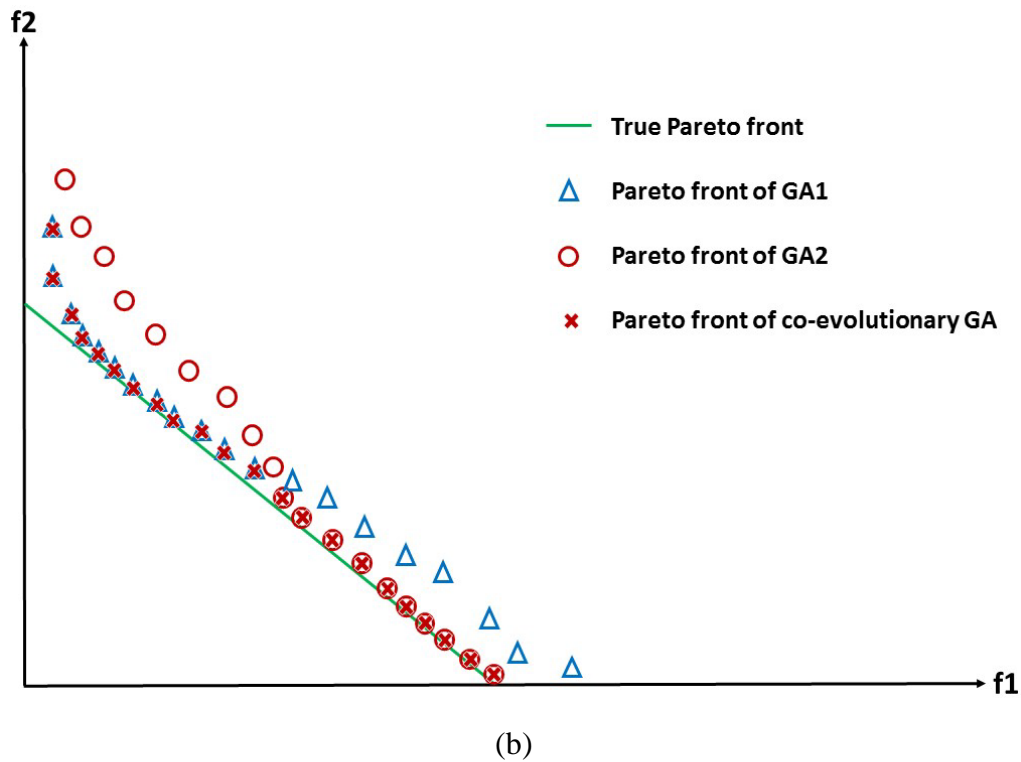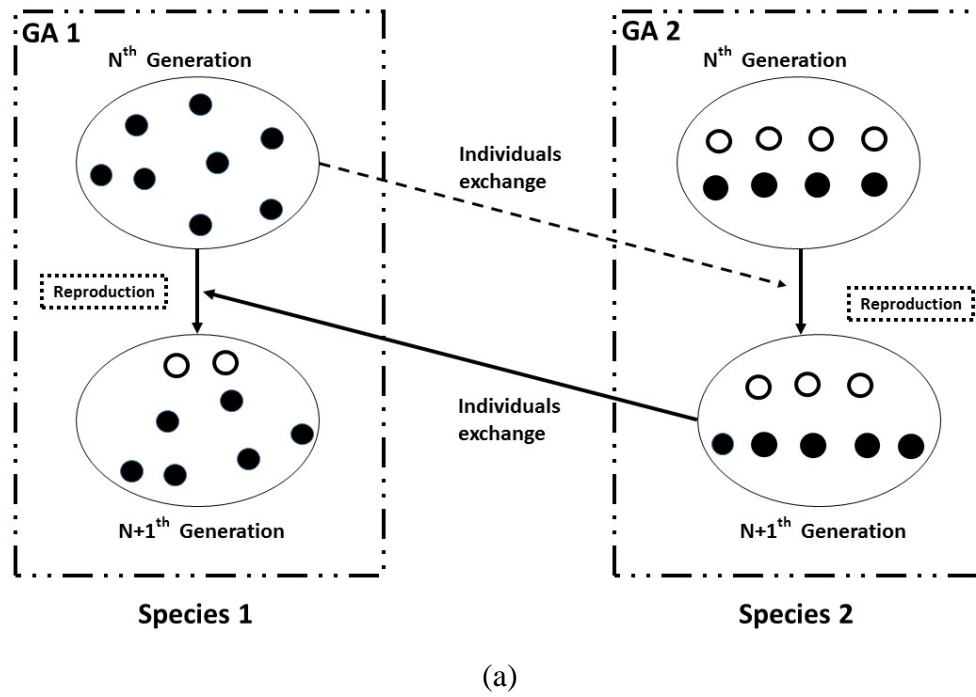
Fig. 21 Summary co-evolutionary genetic algorithm: (a) example co-evolutionary mechanism; (b) an example of a Pareto front from a co-evolutionary genetic algorithm

3.2.4 Multi-Level selection algorithms

Multi-Level Selection Genetic Algorithm (MLSGA) was first introduced by Sobey and Grudniewski in 2017 [182,183]. The algorithm mimics the Multi-Level Selection Theory introduced by Wilson and Sober [181,208] where the different theories are discussed in an excellent review by Okasha [209]. This leads to the introduction of a collective level reproduction mechanism, in addition to the individual level used in the standard Genetic

Algorithm, and a split in the fitness function between the two levels. It is unusual as it is a diversity first optimisation approach and also provides a general solver, working well across all problem types.

The algorithm starts with a randomly generated initial population which is classified into collectives, via SVM, according to the design variables. At the individual level, each individual is evaluated through their objective function and standard genetic operators are utilised to perform individual reproduction. Simultaneously, each collective is evaluated through all of its individuals by calculating the collective objective function at the collective level. The competition is performed among the collectives and the lowest fitness collectives are eliminated. These collectives are replaced by copying the best individuals from each of the remaining collectives into a new collective. The process is stopped when the termination condition is satisfied. This mechanism is illustrated in Figure 22. There are two main variants, each with a different fitness definition at the individual and collective level, MLS1 and MLS2(R) [209], where MLS2R is the reverse of MLS2. MLS1 defines the fitness as the aggregate of the individuals in the population to calculate the fitness of a generation. However, the MLS2(R) calculates different objectives at each level. Therefore, MLS1 focuses its search at the middle of the Pareto front and MLS2 and MLS2R enhance the search at the two sides of the real Pareto front. MLSU combines these two variants, simplifying the multi-objective optimisation process in MLSGA and reducing the hyperparameter tuning. An example of the MLSU search is shown in Figure 23.
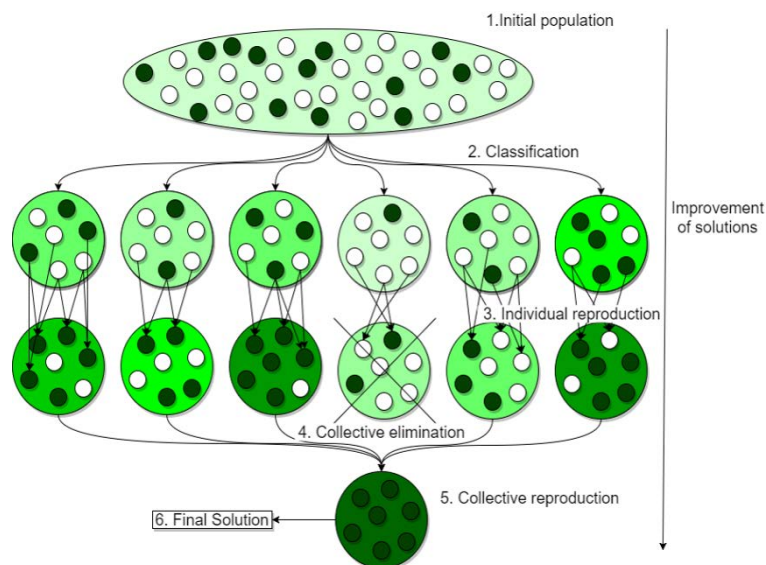

Fig. 22 Mechanism of MLSGA [182,183]: individual and collective reproduction
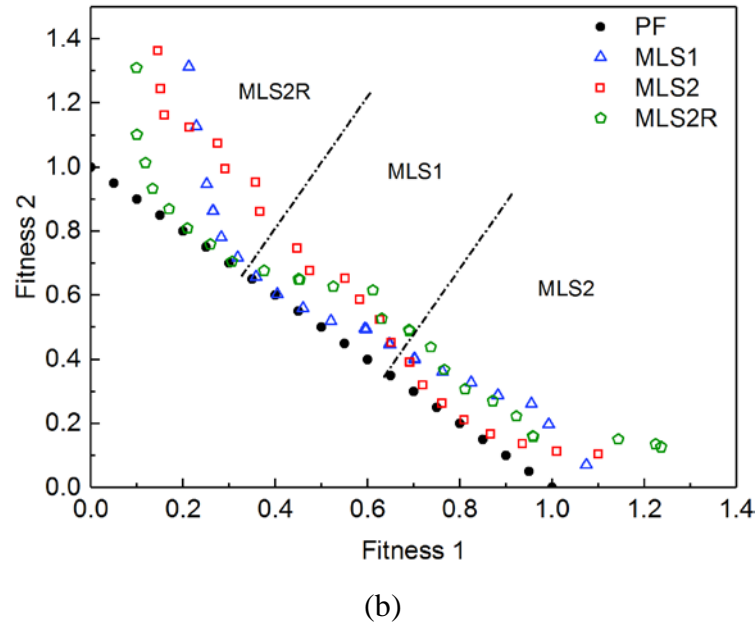
(b)

Fig. 23 Mechanism of MLSGA [182,183]: an example of MLS1, MLS2 and MLS2R search preference

MLSGA can be combined with any other Genetic Algorithm by using these algorithms at the individual reproduction level which has been demonstrated on NSGA-II, MOEA/D and Multiple Trajectory Search (MTS) [183]. The hybrid MLSGAs achieve higher performance than their original algorithms when benchmarked on the CEC'09 test functions [2] except for MOEA/D on unconstrained problems [183]. The MLSGA-MTS exhibits the best performance when solving both the constrained and unconstrained test problems among all the benchmarked algorithms, showing excellent performance as a general solver. This approach has been demonstrated practically on the optimisation of triaxial weave fabric patterns where MLSGA-NSGAII is shown to be more suitable for solving discontinuous Pareto fronts with large gap between sub-fronts [3]. The collectives are distributed in the search space, which maintains the diversity of the entire population and improves the ability to find all of the sub-fronts. This approach is combined with the co-evolutionary approach, which improves the convergence while retaining the increase in diversity while also improving the generality of the algorithm. This leads to the best performing general solver in a recent benchmarking [23].

3.3 Test functions in the computer science literature

The state-of-the-art Genetic Algorithms are frequently benchmarked using mathematical optimisation problems to evaluate the performance of the new solvers. Understanding these problems is a key aspect in selecting the correct algorithm for a composite problem. A variety of these test functions have been reviewed by Grudniewski and Sobey [210]. These problems are increasingly close to the level of complexity in engineering optimisation problems but often focus on one dominant characteristic. The problems in both literatures have similar features including:

- nonlinearities,
- multi-modal,
- imbalanced,
- discontinuous objective spaces,

- non-uniform real Pareto fronts where the density of the Pareto front points are different at different parts of the Pareto front and where the lower density part has a significantly lower possibility of being found,
- multiple constraints or complicated constraints which increase the complexity of the fitness function,
- large boundaries for the variable space
- variable linkages where the objectives are determined by a set of variables and the variables for different objectives overlap [211,212].

The test functions are categories into four types according to the types of dynamics. In the static problems, Grudniewski and Sobey [210] classify 42 bi-objective test functions and 31 three-objective problems into nine categories. Additionally, most test functions have secondary properties, such as multi-modal search spaces, where there are many local optima in the objective space and some local optima are close to the global optimum. Wider benchmarking exercises are performed each year on dynamic and static problems and best practice is frequently found in the *IEEE CEC* competitions, for instance CEC09 for constrained and unconstrained optimisation problems[2], CEC11 for real-world numerical optimisation problems [22], CEC13 and CEC15 for multimodal optimisation problems [213,214] and CEC17 for many-objective optimisation problems [215].

4 Recommendations

A series of recommendations are listed in this section based on the findings from the paper. The common problems are summarised as follows:

1) <u>Poor documentation</u>: ~38% of the 321 papers reviewed don't state the name of the Genetic Algorithm used, ~71% have incomplete information about the genetic operators and ~41% miss the genetic operator rates. Around ~52% do not list the hyperparameters, including population size and generation number or total function evaluations. Only ~14% of papers document the use of several independent cycles of Genetic Algorithms, which have been shown to be dependent on the initial population. It is therefore difficult to assess the validity of the optimisation results and to be confident that the optimal solution is found. This makes benchmarking performance of different methodologies on these problems impossible and there is no consensus on which Genetic Algorithms are best aligned to different problems or what type of mechanisms should be developed; for example is convergence or diversity most important in these problems? Categorising the success, or indeed failure, of a given methodology in the optimisation leads to insight about the characteristics of the problem, for example determining whether it is discontinuous or concave, but also the type of mechanisms most suited to solving it. Defining the necessary documentation to understand the performance of an algorithm will allow the determination of which Genetic Algorithms are best suited to different types of problems, allowing more rapid progress. Since many algorithms converge on a solution early in the generation cycle, using the computational time for a higher numbers of simulations, rather than fewer, longer runs, is likely to result in better solutions and resolved Pareto fronts. A similar argument can be made about how best to use the function calls, whether for population or generations. In composite structures the tendency is towards generations but this will

reduce the diversity of search. The field cannot develop its own best practice without the ability to learn from past examples.

2) <u>Little focus on categorising the optimisation problems:</u> related to the previous issues of documentation ~80% of authors only published one paper on optimisation of composites through Genetic Algorithms indicating that there is often a lack of experience on using Genetic Algorithms. A number of outdated Genetic Algorithms are still prevalent in the composite optimisation literature with ~56% of single objective optimisation papers and ~58% of weighted multi-objective optimisation papers missing information about convergence. When multi-objective optimisation is performed ~14% of multi-objective optimisation papers and ~73% of weighted multi-objective optimisation papers don't generate Pareto fronts, which is the standard approach in most fields to make the most use of multi-objective optimisation techniques. Only ~13% of the papers perform benchmarking and normally compare just two or three different algorithms, without justifying the selection for why these are the state-of-the-art for this problem type and stating what measures are used to guarantee fair tests are performed. Only eight of the 321 papers develop specialist Genetic Algorithms for specific composite optimisation problems; however, only three of these eight papers performed benchmarking, with two papers comparing against two algorithms and the other paper showing that the specialist algorithm didn't achieve the best performance in the benchmarking. This lack of categorisation of the problems makes tackling the problems that have already been solved more efficiently a difficult task, helping new authors understand the current state-of-the-art in composite optimisation must be a key feature in an area where so many authors are using these tools for the first time.

3) <u>A small range of Genetic Algorithms from the computational science literature are used:</u> A range of Genetic Algorithms are used from the most basic versions similar to those developed in the 1970's and the latest frequently used version is GA BIANCA. However, the state-of-the-art Genetic Algorithms, such as MOEA/D for unconstrained problems, cMLSGA, BCE, HEIA as general solvers, cMLSGA for constrained and discontinuous problems and cMLSGA, U-NSGA-III, HEIA and BCE for many-objective problems are not documented. The most frequently used Genetic Algorithm is NSGA-II, which performs well on a wide range of problems as a general solver where it is likely that there are multiple dominant characteristics for the optimisation problem. However, its performance will be compromised on specialist problems and its performance is now below a number of different algorithms in the computational evolution literature, meaning it would be ideal to test some of these algorithms on real problems. The small range of Genetic Algorithms used in composite optimisation literature restricts the probability of achieving optimal designs and stops progress in this field. But the poor documentation of the problem types means that specialist solvers for these problems can't be developed.

4) <u>A focus on simplifying the problems:</u> ~71% of papers solve single objective optimisation problems. In addition ~9% of papers reduce the multi-objective optimisation problems down to single objective problems by using the weighted sum approach. ~17% of papers don't provide information of design variables and ~31%

don't document the constraints, indicating that they solve simple optimisation problems with limited numbers of variables and constraints. The optimisation results of these may have limited use in real-world applications and these problems are all within the bounds of complexity that Genetic Algorithms are capable of solving. Much of the literature focuses on solving problems which have previously been tackled, such as layup optimisation, with incremental steps. While exciting and complex problems such as bi-level optimisation of the structure and material together, multi-disciplinary optimisation and fibre weave optimisation are less well explored. Moving the boundaries of what is possible in optimisation will result in more useful results and make it easier to define what is not possible.

Based on these findings a series of recommendations are made in the hope of improving research into optimisation of composite structures.

1. **Accurate documentation:** better documentation, standardisation, of the information of the Genetic Algorithm and the methods utilised in the optimisation are required to allow readers to evaluate the results, repeat their research and make improvements.

2. **Treat Genetic Algorithms as a specialisation:** understanding the features and mechanisms of the state-of-the-art Genetic Algorithms and select the most suitable solvers for the specific composite optimisation problems. There is limited technical understanding of evolutionary computation demonstrated in the literature. Understanding the search space and dominant characteristics through benchmarking of state-of-the-art Genetic Algorithms. Adequate benchmarking processes will help to select the correct Genetic Algorithms and increase their use.

3. **Perform rigorous benchmarkings**: it is necessary to benchmark state-of-the-art Genetic Algorithms on composite optimisation problems as these engineering problems are dominated by different characteristics. The current computational benchmarking is simple, making it difficult to select the correct algorithm in these cases. Benchmarking helps to determine the dominant characteristics of the problem and investigate the best practices of Genetic Algorithms for the specific problem rather than using them as optimisation tools. The problem type (e.g constrained or unconstrained, single, multi-objective or many objectives and continuous or discontinuous variable or objective space) should be considered when selecting a Genetic Algorithm. Though it is possible to utilise general solvers such as NSGA-II, HEIA and cMLSGA to ensure a reasonable result and these often perform well on problems with more than one characteristic. Additionally, due to the stochastic nature of Genetic Algorithms, it is necessary to run several cycles to ensure the final solutions have been found depending on the nature of the objective space. Using metrics to determine the quality of the solution, such as HV and IGD, will improve the performance. These should be used along with the robustness of solutions and number of generations to convergence to improve the understanding generated from the literature being tackled.

4. **Solve problems with multiple (many) objectives:** Adding more objectives will make the optimisation more complicated and more realistic for use in industrial applications. Multi-objective or many objectives problems, those with 4+ objectives, Genetic

Algorithms are recommended over weighted multi-objective Genetic Algorithms when solving these problems. Better documentation and using state-of-the-art algorithms will allow these problems to be tackled, with a subsequent evolution in the composite optimisation literature.

5 Conclusions

The application of Genetic Algorithms to solving composite optimisation problems is increasing, especially over the last five years. In this paper, 321 papers published between 2008-2017 from 17 journals which cover composite structures and materials are reviewed to understand the technical challenges related to Genetic Algorithm use and to determine the implications for the future. Also included is a detailed review of the Genetic Algorithms from the evolutionary computation literature, with a description of their mechanisms and a prediction for their suitability to different composite optimisation problems. The review shows that in general old algorithms are used and that the hyperparameters are poorly selected, necessitating the need for simple optimisation problems. The database of the papers collected for this analysis is made available in an attached datasheet. A number of problems are presented in the literature and can be summarised as follows:

1. poor documentation of the Genetic Algorithms used;
2. little focus on categorising the optimisation problems;
3. a small range of Genetic Algorithms from the computational science literature are compared;
4. a focus on simplifying the problems.

This has made determining which methods are most appropriate and effective difficult. In addition the results are impossible to replicate and best practice has not been collated over this period, meaning that the field has not evolved. Many of the best Genetic Algorithms in the computational science literature are not utilised, meaning there is the potential for more complex problems to be solved, and the focus on simple problems means that the real benefits of Genetic Algorithms are not seen. Importantly there is no documentation linking the mechanisms of the Genetic Algorithm to the problem type, making it impossible to solve difficult and exciting problems of interest to industry. Therefore, a series of recommendations are listed in this review, categorised corresponding to the problems found in literature, to help improve composite optimisation. They are summarised as follows:

1. a need for accurate documentation of hyperparameters and mechanisms in each paper;
2. treat Genetic Algorithms as a specialisation, utilising current state-of-the-art algorithms with sensible hyperparameters;
3. perform rigorous benchmarkings of different algorithms to help characterise the problems being solved and to generate a body of best practice;
4. solve problems with multiple (many) objectives, reducing single-objective and weighted objective instances.

Acknowledgements

References

[1]   Coello CAC. Journal Papers on Evolutionary Multiobjective Optimisation. J Pap Evol

Multiobjective Optim 2017.
http://delta.cs.cinvestav.mx/~ccoello/EMOO/EMOOjournals.html (accessed August 20, 2018).

[2]   Zhang Q, Suganthan PN. Final Report on CEC ' 09 MOEA Competition. Congr. Evol. Comput. (CEC 2009), 2009.

[3]   Wang Z, Bai J, Sobey AJ, Xiong J, Shenoi RA. Optimal design of triaxial weave fabric composites under tension. Compos Struct 2018;201:616–24. doi:https://doi.org/10.1016/j.compstruct.2018.06.090.

[4]   Mutlu U, Grudniewski PA, Sobey AJ, Blake JIR. Selecting an optimisation methodology in the context of structural design for leisure boats. Des. Constr. Super Mega Yachts, Genoa, Italy: 2017.

[5]   Nikbakt S, Kamarian S, Shakeri M. A review on optimization of composite structures Part I: Laminated composites. Compos Struct 2018;195:158–85. doi:10.1016/j.compstruct.2018.03.063.

[6]   Deveci HA, Artem HS. Optimum design of fatigue-resistant composite laminates using hybrid algorithm. Compos Struct 2017;168:178–88. doi:10.1016/j.compstruct.2017.01.064.

[7]   Naik GN, Gopalakrishnan S, Ganguli R. Design optimization of composites using genetic algorithms and failure mechanism based failure criterion. Compos Struct 2008;83:354–67. doi:10.1016/j.compstruct.2007.05.005.

[8]   Paluch B, Grédiac M, Faye A. Combining a finite element programme and a genetic algorithm to optimize composite structures with variable thickness. Compos Struct 2008;83:284–94. doi:10.1016/j.compstruct.2007.04.023.

[9]   Marques FD, Natarajan S, Ferreira AJM. Evolutionary-based aeroelastic tailoring of stiffened laminate composite panels in supersonic flow regime. Compos Struct 2017;167:30–7. doi:10.1016/j.compstruct.2017.01.062.

[10]   Roos C, Bakis CE. Multi-physics design and optimization of flexible matrix composite driveshafts. Compos Struct 2011;93:2231–40. doi:10.1016/j.compstruct.2011.03.011.

[11]   An H, Chen S, Huang H. Simultaneous optimization of stacking sequences and sizing with two-level approximations and a genetic algorithm. Compos Struct 2015;123:180–9. doi:10.1016/j.compstruct.2014.12.041.

[12]   An H, Chen S, Huang H. Optimal design of composite sandwich structures by considering multiple structure cases. Compos Struct 2016;152:676–86. doi:10.1016/j.compstruct.2016.05.066.

[13]   Kim DH, Choi DH, Kim HS. Design optimization of a carbon fiber reinforced composite automotive lower arm. Compos Part B Eng 2014;58:400–7. doi:10.1016/j.compositesb.2013.10.067.

[14]   Irisarri FX, Bassir DH, Carrere N, Maire JF. Multiobjective stacking sequence optimization for laminated composite structures. Compos Sci Technol 2009;69:983–90. doi:10.1016/j.compscitech.2009.01.011.

[15]   Irisarri FX, Laurin F, Leroy FH, Maire JF. Computational strategy for multiobjective optimization of composite stiffened panels. Compos Struct 2011;93:1158–67.

doi:10.1016/j.compstruct.2010.10.005.

[16]   Wang Z, Sobey AJ. Optimal design of triaxial weave fabric composites for specific strength and stiffness under tension. 18th Eur. Conf. Compos. Mater., Athens, Greece: 2018.

[17]   Vincenti A, Ahmadian M, Vannucci P. BIANCA : a genetic algorithm to solve hard combinatorial optimisation problems in engineering. J Glob Optim 2010;48:399–421. doi:10.1007/s10898-009-9503-2.

[18]   Sindhya K, Sinha A, Deb K, Miettinen K. Local search based evolutionary multi-objective optimization algorithm for constrained and unconstrained problems. 2009 IEEE Congr. Evol. Comput., 2009, p. 2919–26. doi:10.1109/CEC.2009.4983310.

[19]   Li M, Yang S, Liu X. Pareto or Non-Pareto : Bi-Criterion Evolution in Multiobjective Optimization. IEEE Trans Evol Comput 2016;20:645–65. doi:10.1109/TEVC.2015.2504730.

[20]   Tseng LY, Chen C. Multiple Trajectory Search for Multiobjective Optimization. 2007 IEEE Congr. Evol. Comput., 2007, p. 3609–16. doi:10.1109/CEC.2007.4424940.

[21]   Coley DA. An introduction to genetic algorithms for scientists and engineers. Singapore: World Scientific; 1999.

[22]   Suganthan APN. Testing Evolutionary Algorithms on Real-World Numerical Optimization Problems. 2011.

[23]   Sobey AJ, Blanchard J, Grudniewski PA, Savasta T. There's no free lunch: A study of Genetic Algorithm use in maritime applications. Int. Conf. Comput. Appl. Inf. Technol. Marit. Ind. (COMPIT 19), Tullamore, Ireland: 2019.

[24]   Almeida FS, Awruch AM. Design optimization of composite laminated structures using genetic algorithms and finite element analysis. Compos Struct 2009;88:443–54. doi:10.1016/j.compstruct.2008.05.004.

[25]   Jing Z, Sun Q, Silberschmidt VV. Sequential permutation table method for optimization of stacking sequence in composite laminates. Compos Struct 2016;141:240–52. doi:10.1016/j.compstruct.2016.01.052.

[26]   Maes VK, Pavlov L, Simonian SM. An efficient semi-automated optimisation approach for (grid-stiffened) composite structures: Application to Ariane 6 Interstage. Compos Struct 2016;209:1042–9. doi:10.1016/j.compstruct.2016.02.082.

[27]   Ju S, Shenoi RA, Jiang D, Sobey AJ. Multi-parameter optimization of lightweight composite triangular truss structure based on response surface methodology. Compos Struct 2013;97:107–16. doi:10.1016/j.compstruct.2012.10.025.

[28]   Kamarian S, Shakeri M. Thermal buckling analysis and stacking sequence optimization of rectangular and skew shape memory alloy hybrid composite plates. Compos Part B Eng 2017;116:137–52. doi:10.1016/j.compositesb.2017.01.059.

[29]   Pelegri AA, Kedlaya DN. Design of composites using a generic unit cell model coupled with a hybrid genetic algorithm. Compos Part A Appl Sci Manuf 2008;39:1433–43. doi:10.1016/j.compositesa.2008.05.006.

[30]   Antoine GO, Batra RC. Optimization of transparent laminates for specific energy

dissipation under low velocity impact using genetic algorithm. Compos Struct 2015;124:29–34. doi:10.1016/j.compstruct.2014.12.066.

[31] Knipprath C, McCombe GP, Trask RS, Bond IP. Predicting self-healing strength recovery using a multi-objective genetic algorithm. Compos Sci Technol 2012;72:752–9. doi:10.1016/j.compscitech.2012.02.002.

[32] Axinte A, Taranu N, Bejan L, Hudisteanu I. Optimisation of fabric reinforced polymer composites using a variant of Genetic Algorithm. Appl Compos Mater 2017;24:1479–91. doi:10.1007/s10443-017-9594-8.

[33] Barnes RH, Morozov EV. Structural optimisation of composite wind turbine blade structures with variations of internal geometry configuration. Compos Struct 2016;152:158–67. doi:10.1016/j.compstruct.2016.05.013.

[34] Taetragool U, Shah PH, Halls VA, Zheng JQ, Batra RC. Stacking sequence optimization for maximizing the first failure initiation load followed by progressive failure analysis until the ultimate load. Compos Struct 2017;180:1007–21. doi:10.1016/j.compstruct.2017.08.023.

[35] Ho-Huu V, Do-Thi TD, Dang-Trung H, Vo-Duy T, Nguyen-Thoi T. Optimization of laminated composite plates for maximizing buckling load using improved differential evolution and smoothed finite element method. Compos Struct 2016;146:132–47. doi:10.1016/j.compstruct.2016.03.016.

[36] Badalló P, Trias D, Marín L, Mayugo JA. A comparative study of genetic algorithms for the multi-objective optimization of composite stringers under compression loads. Compos Part B 2013;47:130–6. doi:10.1016/j.compositesb.2012.10.037.

[37] Lieu QX, Lee J. Modeling and optimization of functionally graded plates under thermo-mechanical load using isogeometric analysis and adaptive hybrid evolutionary firefly algorithm. Compos Struct 2017;179:89–106. doi:10.1016/j.compstruct.2017.07.016.

[38] Hwang SF, Wu JC, He RS. Identification of effective elastic constants of composite plates based on a hybrid genetic algorithm. Compos Struct 2009;90:217–24. doi:10.1016/j.compstruct.2009.03.021.

[39] Stodieck O, Cooper JE, Weaver PM, Kealy P. Optimization of tow-steered composite wing laminates for aeroelastic tailoring. AIAA J 2015;53:2203–15. doi:10.2514/1.J053599.

[40] Petrone G, Meruane V. Mechanical properties updating of a non-uniform natural fibre composite panel by means of a parallel genetic algorithm. Compos Part A Appl Sci Manuf 2017;94:226–33. doi:10.1016/j.compositesa.2016.12.017.

[41] Ihesiulor OK, Shankar K, Zhang Z, Ray T. Delamination detection with error and noise polluted natural frequencies using computational intelligence concepts. Compos Part B Eng 2014;56:906–25. doi:10.1016/j.compositesb.2013.09.032.

[42] Cai H, Aref AJ. On the design and optimization of hybrid carbon fiber reinforced polymer-steel cable system for cable-stayed bridges. Compos Part B Eng 2015;68:146–52. doi:10.1016/j.compositesb.2014.08.031.

[43] Macquart T, Bordogna MT, Lancelot P, De Breuker R. Derivation and application of blending constraints in lamination parameter space for composite optimisation.

Compos Struct 2016;135:224–35. doi:10.1016/j.compstruct.2015.09.016.

[44] Fan HT, Wang H, Chen XH. An optimization method for composite structures with ply-drops. Compos Struct 2016;136:650–61. doi:10.1016/j.compstruct.2015.11.003.

[45] Ashjari M, Khoshravan MR. Mass optimization of functionally graded plate for mechanical loading in the presence of deflection and stress constraints. Compos Struct 2014;110:118–32. doi:10.1016/j.compstruct.2013.11.025.

[46] Kalantari M, Dong C, Davies IJ. Multi-objective robust optimisation of unidirectional carbon/glass fibre reinforced hybrid composites under flexural loading. Compos Struct 2016;138:264–75. doi:10.1016/j.compstruct.2015.11.034.

[47] Keller D. Optimization of ply angles in laminated composite structures by a hybrid, asynchronous, parallel evolutionary algorithm. Compos Struct 2010;92:2781–90. doi:10.1016/j.compstruct.2010.04.003.

[48] Karakaya Ş, Soykasap Ö. Natural frequency and buckling optimization of laminated hybrid composite plates using genetic algorithm and simulated annealing. Struct Multidiscip Optim 2011;43:61–72. doi:10.1007/s00158-010-0538-2.

[49] Herath MT, Natarajan S, Prusty BG, John NS. Isogeometric analysis and genetic algorithm for shape-adaptive composite marine propellers. Comput Methods Appl Mech Eng 2015;284:835–60. doi:10.1016/j.cma.2014.10.028.

[50] Baran I, Tutum CC, Hattel JH. Optimization of the thermosetting pultrusion process by using hybrid and mixed integer genetic algorithms. Appl Compos Mater 2013;20:449–63. doi:10.1007/s10443-012-9278-3.

[51] Kim SH, Kim CG. Optimal design of composite stiffened panel with cohesive elements using micro-genetic algorithm. J Compos Mater 2008;42:2259–73. doi:10.1177/0021998308094967.

[52] Perera R, Ruiz A. Design equations for reinforced concrete members strengthened in shear with external FRP reinforcement formulated in an evolutionary multi-objective framework. Compos Part B Eng 2012;43:488–96. doi:10.1016/j.compositesb.2011.10.013.

[53] Satheesh R, Narayana NG, Ganguli R. Conservative design optimization of laminated composite structures using Genetic Algorithms and multiple failure criteria. J Compos Mater 2010;44:369–87. doi:10.1177/0021998309347579.

[54] Montemurro M, Vincenti A, Vannucci P. The automatic dynamic penalisation method (ADP) for handling constraints with genetic algorithms. Comput Methods Appl Mech Eng 2013;256:70–87. doi:10.1016/j.cma.2012.12.009.

[55] Perera R, Tarazona D, Ruiz A, Martín A. Application of artificial intelligence techniques to predict the performance of RC beams shear strengthened with NSM FRP rods. Formulation of design equations. Compos Part B Eng 2014;66:162–73. doi:10.1016/j.compositesb.2014.05.001.

[56] de Almeida FS. Stacking sequence optimization for maximum buckling load of composite plates using harmony search algorithm. Compos Struct 2016;143:287–99. doi:10.1016/j.compstruct.2016.02.034.

[57] da Silva RF, Teófilo FA, Parente JE, de Melo AM, de Holanda ÁS. Optimization of

composite catenary risers. Mar Struct 2013;33:1–20. doi:10.1016/j.marstruc.2013.04.002.

[58] Gillet A, Francescato P, Saffre P. Single- and multi-objective optimization of composite structures: the influence of design variables. J Compos Mater 2010;44:457–80. doi:10.1177/0021998309344931.

[59] Jing Z, Fan X, Sun Q. Stacking sequence optimization of composite laminates for maximum buckling load using permutation search algorithm. Compos Struct 2015;121:225–36. doi:10.1016/j.compstruct.2014.10.031.

[60] Aymerich F, Serra M. Optimization of laminate stacking sequence for maximum buckling load using the ant colony optimization (ACO) metaheuristic. Compos Part A Appl Sci Manuf 2008;39:262–72. doi:10.1016/j.compositesa.2007.10.011.

[61] Lopez RH, Luersen MA, Cursi ES. Optimization of laminated composites considering different failure criteria. Compos Part B Eng 2009;40:731–40. doi:10.1016/j.compositesb.2009.05.007.

[62] Rao ARM, Lakshmi K. Optimal design of stiffened laminate composite cylinder using a hybrid SFL algorithm. J Compos Mater 2012;46:3031–55. doi:10.1177/0021998311435674.

[63] Ghiasi H, Pasini D, Lessard L. Constrained globalized Nelder-Mead method for simultaneous structural and manufacturing optimization of a composite bracket. J Compos Mater 2008;42:717–36. doi:10.1177/0021998307088592.

[64] Danzi F, Frulla G, Cestino E. Constrained combinatorial optimization of multi-layered composite structures by means of Stud GA with proportionate selection and extinction. Struct Multidiscip Optim 2017;55:2239–57. doi:10.1007/s00158-016-1638-4.

[65] An H, Chen S, Huang H. Laminate stacking sequence optimization with strength constraints using two-level approximations and adaptive genetic algorithm. Struct Multidiscip Optim 2015;51:903–18. doi:10.1007/s00158-014-1181-0.

[66] Rao ARM, Lakshmi K. Multi-objective optimal design of hybrid laminate composite structures using scatter search. J Compos Mater 2009;43:2157–82. doi:10.1177/0021998309339221.

[67] Kunasol N, Suwannik W, Chongstitvatana P. Solving one-million-bit problems using LZWGA. 2006 Int. Symp. Commun. Inf. Technol. Isc., 2006, p. 32–6. doi:10.1109/ISCIT.2006.339882.

[68] Sastry K, Goldberg DE, Llor X. Towards Billion Bit Optimization via Efficient Genetic Algorithms. llliGAL Rep., 2007.

[69] Goldberg DE, Sastry K, Llor X. Toward Routine Billion-Variable Optimization Using Genetic Algorithms. Complexity 2007;12:27–9. doi:10.1002/cplx.

[70] Iturriaga S, Nesmachnow S. Solving very large optimization problems (up to one billion variables) with a parallel evolutionary algorithm in CPU and GPU. 7th Int. Conf. P2P, Parallel, Grid, Cloud Internet Comput., 2012, p. 267–72. doi:10.1109/3PGCIC.2012.63.

[71] Deb K, Myburgh C. Breaking the Billion-Variable Barrier in real-world optimization using a customized evolutionary algorithm. Genet. Evol. Comput. Conf., 2016, p. 653–

60. doi:10.1145/2908812.2908952.

[72] Seada H, Deb K. U-NSGA-III: A unified evolutionary optimization procedure for single, multiple, and many objectives. InInternational Conf. Evol. Multi-Criterion Optim., Springer, Cham; 2015, p. 34–49. doi:10.1007/978-3-319-15892-1_3.

[73] Lin Q, Chen J, Zhan Z-H, Chen W, Coello CAC, Yin Y, et al. A Hybrid Evolutionary Immune Algorithm for Multiobjective Optimization Problems. IEEE Trans Evol Comput 2016;20:711–29. doi:10.1109/TEVC.2015.2512930.

[74] Pareto V. Manuale di Economia Politica. Soc. Ed. Libr., 1906.

[75] Aumjaud P, Fieldsend JE, Boucher MA, Evans KE, Smith CW. Multi-objective optimisation of viscoelastic damping inserts in honeycomb sandwich structures. Compos Struct 2015;132:451–63. doi:10.1016/j.compstruct.2015.05.061.

[76] Lee DS, Morillo C, Bugeda G, Oller S, Onate E. Multilayered composite structure design optimisation using distributed/parallel multi-objective evolutionary algorithms. Compos Struct 2012;94:1087–96. doi:10.1016/j.compstruct.2011.10.009.

[77] Kalantari M, Dong C, Davies IJ. Effect of matrix voids, fibre misalignment and thickness variation on multi-objective robust optimization of carbon/glass fibre-reinforced hybrid composites under flexural loading. Compos Part B Eng 2017;123:136–47. doi:10.1016/j.compositesb.2017.05.022.

[78] Kalantari M, Dong C, Davies IJ. Multi-objective robust optimization of multi-directional carbon/glass fibre-reinforced hybrid composites with manufacture related uncertainties under flexural loading. Compos Struct 2017;182:132–42. doi:10.1016/j.compstruct.2017.09.019.

[79] De Munck M, De Sutter S, Verbruggen S, Tysmans T, Coelho RF. Multi-objective weight and cost optimization of hybrid composite-concrete beams. Compos Struct 2015;134:369–77. doi:10.1016/j.compstruct.2015.08.089.

[80] Vo-Duy T, Duong-Gia D, Ho-Huu V, Vu-Do HC, Nguyen-Thoi T. Multi-objective optimization of laminated composite beam structures using NSGA-II algorithm. Compos Struct 2017;168:498–509. doi:10.1016/j.compstruct.2017.02.038.

[81] Woods BKS, Hill I, Friswell MI. Ultra-efficient wound composite truss structures. Compos Part A Appl Sci Manuf 2016;90:111–24. doi:10.1016/j.compositesa.2016.06.022.

[82] Rouhi M, Ghayoor H, Hoa SV, Hojjati M. Multi-objective design optimization of variable stiffness composite cylinders. Compos Part B Eng 2015;69:249–55. doi:10.1016/j.compositesb.2014.10.011.

[83] Joglekar S, Von Hagel K, Pankow M, Ferguson S. Exploring how optimal composite design is influenced by model fidelity and multiple objectives. Compos Struct 2017;160:964–75. doi:10.1016/j.compstruct.2016.10.089.

[84] Shaw AD, Dayyani I, Friswell MI. Optimisation of composite corrugated skins for buckling in morphing aircraft. Compos Struct 2015;119:227–37. doi:10.1016/j.compstruct.2014.09.001.

[85] Choi SW, Kim Y, Park HS. Multi-objective seismic retrofit method for using FRP jackets in shear-critical reinforced concrete frames. Compos Part B Eng 2014;56:207–

16. doi:10.1016/j.compositesb.2013.08.049.

[86]   Qi C, Yang S, Yang LJ, Wei ZY, Lu ZH. Blast resistance and multi-objective optimization of aluminum foam-cored sandwich panels. Compos Struct 2013;105:45–57. doi:10.1016/j.compstruct.2013.04.043.

[87]   Honda S, Igarashi T, Narita Y. Multi-objective optimization of curvilinear fiber shapes for laminated composite plates by using NSGA-II. Compos Part B Eng 2013;45:1071–8. doi:10.1016/j.compositesb.2012.07.056.

[88]   Stanford BK, Jutte C., Wu KC. Aeroelastic benefits of tow steering for composite plates. Compos Struct 2014;118:416–22. doi:10.1016/j.compstruct.2014.08.007.

[89]   António CC, Hoffbauer LN. Bi-level dominance GA for minimum weight and maximum feasibility robustness of composite structures. Compos Struct 2016;135:83–95. doi:10.1016/j.compstruct.2015.09.019.

[90]   Fayazbakhsh K, Nik MA, Pasini D, Lessard L. Defect layer method to capture effect of gaps and overlaps in variable stiffness laminates made by Automated Fiber Placement. Compos Struct 2013;97:245–51. doi:10.1016/j.compstruct.2012.10.031.

[91]   Okabe T, Oya Y, Yamamoto G, Sato J, Matsumiya T, Matsuzaki R, et al. Multi-objective optimization for resin transfer molding process. Compos Part A Appl Sci Manuf 2017;92:1–9. doi:10.1016/j.compositesa.2016.09.023.

[92]   Akbarzadeh AH, Nik MA, Pasini D. Vibration responses and suppression of variable stiffness laminates with optimally steered fibers and magnetostrictive layers. Compos Part B Eng 2016;91:315–26. doi:10.1016/j.compositesb.2016.02.003.

[93]   Dal Monte A, De Betta S, Castelli MR, Benini E. Proposal for a coupled aerodynamic–structural wind turbine blade optimization. Compos Struct 2017;159:144–56. doi:10.1016/j.compstruct.2016.09.042.

[94]   Gupta A, Kelly PA, Ehrgott M, Bickerton S. A surrogate model based evolutionary game-theoretic approach for optimizing non-isothermal compression RTM processes. Compos Sci Technol 2013;84:92–100. doi:10.1016/j.compscitech.2013.05.012.

[95]   Struzziero G, Skordos AA. Multi-objective optimisation of the cure of thick components. Compos Part A Appl Sci Manuf 2017;93:126–36. doi:10.1016/j.compositesa.2016.11.014.

[96]   Lee DS, Morillo C, Oller S, Bugeda G, Oñate E. Robust design optimisation of advance hybrid (fiber-metal) composite structures. Compos Struct 2013;99:181–92. doi:10.1016/j.compstruct.2012.11.033.

[97]   Yarasca J, Mantari JL, Petrolo M, Carrera E. Multiobjective Best Theory Diagrams for cross-ply composite plates employing polynomial, zig-zag, trigonometric and exponential thickness expansions. Compos Struct 2017;176:860–76. doi:10.1016/j.compstruct.2017.05.055.

[98]   Rocha IBCM, Parente E, Melo AMC. A hybrid shared/distributed memory parallel genetic algorithm for optimization of laminate composites. Compos Struct 2014;107:288–97. doi:10.1016/j.compstruct.2013.07.049.

[99]   Yarasca J, Mantari JL, Petrolo M, Carrera E. Best Theory Diagrams for cross-ply composite plates using polynomial, trigonometric and exponential thickness

expansions. Compos Struct 2017;161:362–83. doi:10.1016/j.compstruct.2017.05.055.

[100] Choi SW. Investigation on the seismic retrofit positions of FRP jackets for RC frames using multi-objective optimization. Compos Part B Eng 2017;123:34–44. doi:10.1016/j.compositesb.2017.05.026.

[101] Caccese V, Ferguson JR, Edgecomb MA. Optimal design of honeycomb material used to mitigate head impact. Compos Struct 2013;100:404–12. doi:10.1016/j.compstruct.2012.12.034.

[102] Paz J, Díaz J, Romera L, Costas M. Size and shape optimization of aluminum tubes with GFRP honeycomb reinforcements for crashworthy aircraft structures. Compos Struct 2015;133:499–507. doi:10.1016/j.compstruct.2015.07.077.

[103] Murugan S, Friswell MI. Morphing wing flexible skins with curvilinear fiber composites. Compos Struct 2013;99:69–75. doi:10.1016/j.compstruct.2012.11.026.

[104] Elham A, van Tooren MJL. Weight indexing for wing-shape multi-objective optimization. AIAA J 2014;52:320–37. doi:10.2514/1.J052406.

[105] Todoroki A, Sekishiro M. Modified efficient global optimization for a hat-stiffened composite panel with buckling constraint. AIAA J 2008;46:2257–64. doi:10.2514/1.34548.

[106] Soury E, Behravesh AH, Esfahani ER, Zolfaghari A. Design, optimization and manufacturing of wood-plastic composite pallet. Mater Des 2009;30:4183–91. doi:10.1016/j.matdes.2009.04.035.

[107] Cui X, Wang S, Hu SJ. A method for optimal design of automotive body assembly using multi-material construction. Mater Des 2008;29:381–7. doi:10.1016/j.matdes.2007.01.024.

[108] Ahmadian MR, Vincenti A, Vannucci P. A general strategy for the optimal design of composite laminates by the polar-genetic method. Mater Des 2011;32:2317–27. doi:10.1016/j.matdes.2010.08.036.

[109] Johnson KL, Chowdhury S, Lawrimore WB, Mao Y, Mehmani A, Prabhu R, et al. Constrained topological optimization of a football helmet facemask based on brain response. Mater Des 2016;111:108–18. doi:10.1016/j.matdes.2016.08.064.

[110] Qi C, Sun Y, Hu HT, Wang DZ, Cao GJ, Yang S. On design of hybrid material double-hat thin-walled beams under lateral impact. Int J Mech Sci 2016;118:21–35. doi:10.1016/j.ijmecsci.2016.09.009.

[111] Tran T, Hou S, Han X, Tan W, Nguyen N. Theoretical prediction and crashworthiness optimization of multi-cell triangular tubes. Thin-Walled Struct 2014;82:183–95. doi:10.1016/j.tws.2014.03.019.

[112] Studzinski R. Optimal design of sandwich panels with hybrid core. J Sandw Struct Mater 2017;0:1–13. doi:10.1177/1099636217742574.

[113] Sheyka MP, Altunc AB, Taha MR. Multi-objective genetic topological optimization for design of blast resistant composites. Appl Compos Mater 2012;19:785–98. doi:10.1007/s10443-011-9244-5.

[114] Airoldi A, Bertoli S, Lanzi L, Sirna M, Sala G. Design of a motorcycle composite

swing-arm by means of multi-objective optimisation. Appl Compos Mater 2012;19:599–618. doi:10.1007/s10443-011-9227-6.

[115] Irisarri FX, Peeters DMJ, Abdalla MM. Optimisation of ply drop order in variable stiffness laminates. Compos Struct 2016;152:791–9. doi:10.1016/j.compstruct.2016.05.076.

[116] Asgari M. Material optimization of functionally graded heterogeneous cylinder for wave propagation. J Compos Mater 2016;50:3525–38. doi:10.1177/0021998315622051.

[117] Vosoughi AR, Nikoo MR. Maximum fundamental frequency and thermal buckling temperature of laminated composite plates by a new hybrid multi-objective optimization technique. Thin-Walled Struct 2015;95:408–15. doi:10.1016/j.tws.2015.07.014.

[118] Baier H, Huber M, Langer H. Design optimization of hybrid material structures. Struct Multidiscip Optim 2008;36:203–13. doi:10.1007/s00158-007-0204-5.

[119] Arikoglu A. Multi-objective optimal design of hybrid viscoelastic/composite sandwich beams by using the generalized differential quadrature method and the non-dominated sorting genetic algorithm II. Struct Multidiscip Optim 2017;56:885–901. doi:10.1007/s00158-017-1695-3.

[120] Dayyani I, Friswell MI. Multi-objective optimization for the geometry of trapezoidal corrugated morphing skins. Struct Multidiscip Optim 2017;55:331–45. doi:10.1007/s00158-016-1476-4.

[121] Zhou Y, Saitou K. Topology optimization of composite structures with data-driven resin filling time manufacturing constraint. Struct Multidiscip Optim 2017;55:2073–86. doi:10.1007/s00158-016-1628-6.

[122] Ghiasi H, Pasini D, Lessard L. Pareto frontier for simultaneous structural and manufacturing optimization of a composite part. Struct Multidiscip Optim 2010;40:497–511. doi:10.1007/s00158-009-0366-4.

[123] Schatz ME, Hermanutz A, Baier HJ. Multi-criteria optimization of an aircraft propeller considering manufacturing: Structural design optimization simultaneously considering a knowledge-based manufacturing and a structural model. Struct Multidiscip Optim 2017;55:899–911. doi:10.1007/s00158-016-1541-z.

[124] Antonio CAC. Local and global Pareto dominance applied to optimal design and material selection of composite structures. Struct Multidiscip Optim 2013;48:73–94. doi:10.1007/s00158-012-0878-1.

[125] Ghasemi AR, Hajmohammad MH. Multi-objective optimization of laminated composite shells for minimum mass/cost and maximum buckling pressure with failure criteria under external hydrostatic pressure. Struct Multidiscip Optim 2017;55:1051–62. doi:10.1007/s00158-016-1559-2.

[126] Velea MN, Wennhage P, Zenkert D. Multi-objective optimisation of vehicle bodies made of FRP sandwich structures. Compos Struct 2014;111:75–84. doi:10.1016/j.compstruct.2013.12.030.

[127] Araújo AL, Martins P, Soares CMM, Soares CAM, Herskovits J. Damping optimization of viscoelastic laminated sandwich composite structures. Struct

Multidiscip Optim 2009;39:569–79. doi:10.1007/s00158-009-0390-4.

[128] Kalhor R, Akbarshahi H, Case SW. Numerical modeling of the effects of FRP thickness and stacking sequence on energy absorption of metal-FRP square tubes. Compos Struct 2016;147:231–46. doi:10.1016/j.compstruct.2016.03.038.

[129] Dal Monte A, Castelli MR, Benini E. Multi-objective structural optimization of a HAWT composite blade. Compos Struct 2013;106:362–73. doi:10.1016/j.compstruct.2013.05.038.

[130] Visweswaraiah SB, Ghiasi H, Pasini D, Lessard L. Multi-objective optimization of a composite rotor blade cross-section. Compos Struct 2013;96:75–81. doi:10.1016/j.compstruct.2012.09.031.

[131] Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results. Evol Comput 2000;8:173–95. doi:10.1162/106365600568202.

[132] Wang Z, Sobey AJ. Optimal design of triaxial weave fabric composites for specific strength and stiffness under tension. 18th Eur. Conf. Compos. Mater., Athens: 2018.

[133] Auger A, Bader J, Brockhoff D, Zitzler E. Theory of the hypervolume indicator: Optimal μ-Distributions and the Choice of the Reference Point. Found. Genet. Algorithms - FOGA '09, Orlando, Florida, United States: 2009. doi:10.1145/1527125.1527138.

[134] While L, Bradstreet L, Barone L. A fast way of calculating exact hypervolumes. IEEE Trans Evol Comput 2012;16:86–95. doi:10.1109/TEVC.2010.2077298.

[135] Nguyen HX, Lee J, Vo TP, Lanc D. Vibration and lateral buckling optimisation of thin-walled laminated composite channel-section beams. Compos Struct 2016;143:84–92. doi:10.1016/j.compstruct.2016.02.011.

[136] Şen I, Alderliesten RC, Benedictus R. Design optimisation procedure for fibre metal laminates based on fatigue crack initiation. Compos Struct 2015;120:275–84. doi:10.1016/j.compstruct.2014.10.010.

[137] Shi S, Sun Z, Ren M, Chen H, Hu X. Buckling resistance of grid-stiffened carbon-fiber thin-shell structures. Compos Part B Eng 2013;45:888–96. doi:10.1016/j.compositesb.2012.09.052.

[138] Lemaire E, Zein S, Bruyneel M. Optimization of composite structures with curved fiber trajectories. Compos Struct 2015;131:895–904. doi:10.1016/j.compstruct.2015.06.040.

[139] Nguyen XH, Kim NI, Lee J. Optimum design of thin-walled composite beams for flexural-torsional buckling problem. Compos Struct 2015;132:1065–74. doi:10.1016/j.compstruct.2015.06.036.

[140] Montemurro M, Koutsawa Y, Belouettar S, Vincenti A, Vannucci P. Design of damping properties of hybrid laminates through a global optimisation strategy. Compos Struct 2012;94:3309–20. doi:10.1016/j.compstruct.2012.05.003.

[141] Akoussan K, Hamdaoui M, Daya EM. Improved layer-wise optimization algorithm for the design of viscoelastic composite structures. Compos Struct 2017;176:342–58. doi:10.1016/j.compstruct.2017.05.047.

[142] Mazloomi MS, Ranjbar M, Boldrin L, Scarpa F, Patsias S, Ozada N. Vibroacoustics of 2D gradient auxetic hexagonal honeycomb sandwich panels. Compos Struct 2017;187:593–603. doi:10.1016/j.compstruct.2017.10.077.

[143] Do DTT, Lee S, Lee J. A modified differential evolution algorithm for tensegrity structures. Compos Struct 2016;158:11–9. doi:10.1016/j.compstruct.2016.08.039.

[144] Wang J, Simacek P, Advani SG. Use of Centroidal Voronoi Diagram to find optimal gate locations to minimize mold filling time in resin transfer molding. Compos Part A Appl Sci Manuf 2016;87:243–55. doi:10.1016/j.compositesa.2016.04.026.

[145] Aumjaud P, Smith CW, Evans KE. A novel viscoelastic damping treatment for honeycomb sandwich structures. Compos Struct 2015;119:322–32. doi:10.1016/j.compstruct.2014.09.005.

[146] Badalló P, Trias D, Lindgaard E. Damage tolerance optimization of composite stringer run-out under tensile load. Compos Struct 2015;133:98–104. doi:10.1016/j.compstruct.2015.07.025.

[147] Le-Anh L, Nguyen-Thoi T, Ho-Huu V, Dang-Trung H, Bui-Xuan T. Static and frequency optimization of folded laminated composite plates using an adjusted Differential Evolution algorithm and a smoothed triangular plate element. Compos Struct 2015;127:382–94. doi:10.1016/j.compstruct.2015.02.069.

[148] Vosoughi AR, Gerist S. New hybrid FE-PSO-CGAs sensitivity base technique for damage detection of laminated composite beams. Compos Struct 2014;118:68–73. doi:10.1016/j.compstruct.2014.07.012.

[149] António CC. A study on synergy of multiple crossover operators in a hierarchical genetic algorithm applied to structural optimisation. Struct Multidiscip Optim 2009;38:117–35. doi:10.1007/s00158-008-0268-x.

[150] Karakaya Ş, Soykasap Ö. Buckling optimization of laminated composite plates using genetic algorithm and generalized pattern search algorithm. Struct Multidiscip Optim 2009;39:477–86. doi:10.1007/s00158-008-0344-2.

[151] Moussavian H, Jafari M. Optimum design of laminated composite plates containing a quasi-square cutout. Struct Multidiscip Optim 2017;55:141–54. doi:10.1007/s00158-016-1481-7.

[152] Zeng F, Xie H, Liu Q, Li F, Tan W. Design and optimization of a new composite bumper beam in high-speed frontal crashes. Struct Multidiscip Optim 2016;53:115–22. doi:10.1007/s00158-015-1312-2.

[153] Wang W, Guo S, Chang N, Zhao F, Yang W. A modified ant colony algorithm for the stacking sequence optimisation of a rectangular laminate. Struct Multidiscip Optim 2010;41:711–20. doi:10.1007/s00158-009-0447-4.

[154] Abachizadeh M, Tahani M. An ant colony optimization approach to multi-objective optimal design of symmetric hybrid laminates for maximum fundamental frequency and minimum cost. Struct Multidiscip Optim 2009;37:367–76. doi:10.1007/s00158-008-0235-6.

[155] Konak A, Coit DW, Smith AE. Multi-objective optimization using genetic algorithms: A tutorial. Reliab Eng Syst Saf 2006;91:992–1007. doi:10.1016/j.ress.2005.11.018.

[156] Turing A. Computing machinery and intelligence. MIND 1950;LIX:433–60.

[157] Holland JH. Adaptive plans optimal for payoff-only environments. 1969.

[158] De Jong KA. Analysis of the behaviour of a class of genetic adaptive systems 1975.

[159] Schaffer JD. Multiple objective optimization with vector evaluated genetic algorithms. First Int. Conf. Genet. Algorithms their Appl., 1985, p. 93–100.

[160] Goldberg DE. Genetic algorithms in search, optimisation and machine learning. Reading, MA Addison-Wesley 1989.

[161] Hajela P, Lin C. Genetic search strategies in multicriterion optimal design. Struct Optim 1992;4:99–107. doi:10.1007/BF01759923.

[162] Fonseca CM, Fleming PJ. Multiobjective genetic algorithms. IEE Colloq. Genet. algorithms Control Syst. Eng., 1993, p. 6–1.

[163] Chipperfield A, Fleming P, Pohlheim H, Fonseca CM. Genetic Algorithm Toolbox for use with MATLAB 1994. doi:10.1243/0954405021519915.

[164] Srinivas M, Patnaik LM. Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms. IEEE Trans Syst Man Cybern 1994;24:656–67.

[165] Hinterding R, Michalewicz Z, Peachey TC. Self-adaptive genetic algorithm for numeric functions. Int. Conf. Parallel Probl. Solving from Nat., Springer Berlin Heidelberg; 1996, p. 420–9.

[166] Srinivas N, Deb K. Muiltiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. Evol Comput 1994;2:221–48. doi:10.1162/evco.1994.2.3.221.

[167] Deb K, Agrawal RB. Simulated Binary Crossover for Continuous Search Space. Complex Syst 1994;9:1–5.

[168] Whitley D, Rana S, Heckendorn RB. The island model genetic algorithm: On separability, population size and convergence. J Comput Inf Technol 1999;7:33–47. doi:10.1.1.36.7225.

[169] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Trans Evol Comput 1999;3:257–71. doi:10.1109/4235.797969.

[170] Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the Strength Pareto Evolutionary Algorithm 2001:95–100. doi:10.1.1.28.7571.

[171] Watanabe S, Hiroyasu T, Miki M. NCGA : Neighborhood Cultivation Genetic Algorithm for Multi-Objective Optimization Problems. GECCO Late Break Pap 2002:458–65.

[172] Coello CAC, Pulido GT. A Micro-Genetic Algorithm for Multiobjective Optimization. First Int Conf Evol Multi-Criterion Optim 2001:126–40. doi:10.1007/3-540-44719-9.

[173] Tiwari S, Fadel G, Koch P, Deb K. Performance Assessment of the Hybrid Archive-based Micro Genetic Algorithm (AMGA) on the CEC09 Test Problems. IEEE Congr. Evol. Comput., 2009, p. 1935–42.

[174] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 2002;6:182–97.

doi:10.1109/4235.996017.

[175] Chen H, Ooka R, Kato S. Study on optimum design method for pleasant outdoor thermal environment using genetic algorithm (GA) and coupled simulation of convection radiation and conduction: Optimum arrangement of trees for design of pleasant outdoor thermal environment. J Environ Eng 2004;69:65–71. doi:10.3130/aije.69.65_1.

[176] Zitzler E, Kunzli S. Indicator-Based Selection in Multiobjective Search. 8th Int Conf Parallel Probl Solving from Nat (PPSN VIII) 2004;3242:832–42. doi:10.1007/b100601.

[177] Zhang Q, Li H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Trans Evol Comput 2007;11:712–31. doi:10.1109/TEVC.2007.892759.

[178] Paquete L, Stutzle T. A two-phase local search for the biobjective traveling salesman problem. Int. Conf. Evol. Multi-Criterion Optim., Springer Berlin Heidelberg; 2003, p. 479–93. doi:10.1007/3-540-36970-8.

[179] Liu H, Gu F, Zhang Q. Decomposition of a Multiobjective Optimization Problem into a Number of Simple Multiobjective Subproblems. IEEE Trans Evol Comput 2014;18:450–5. doi:10.1109/TEVC.2013.2281533.

[180] Jiang S, Yang S, Wang Y, Liu X. Scalarizing Functions in Decomposition-Based Multiobjective Evolutionary Algorithms. IEEE Trans Evol Comput 2018;22:296–313. doi:10.1109/TEVC.2017.2707980.

[181] Sober E, Wilson D. Unto others: The evolution and psychology of unselfish behaviour. Harvard University Press; 1999.

[182] Grudniewski PA, Sobey AJ. Multi-level selection genetic algorithm applied to CEC '09 test instances. IEEE Congr. Evol. Comput., 2017, p. 1613–20.

[183] Sobey AJ, Grudniewski PA. Re-inspiring the genetic algorithm with multi-level selection theory : multi-level selection genetic algorithm. Bioinspir Biomim 2018;13. doi:10.1088/1748-3190/aad2e8.

[184] Cavicchio D. Adaptive search using simulated evolution 1970.

[185] De Jong KA. Analysis of the behavior of a class of genetic adaptive systems 1975.

[186] Goldberg DE, Richardson J. Genetic algorithms with sharing for multimodal function optimization. Genet. algorithms their Appl. Proc. Second Int. Conf. Genet. Algorithms, 1987, p. 41–9.

[187] Beasley D, Bull DR, Martin RR. A sequential niche technique for multimodal function optimization. Evol Comput 1993;1:101–25. doi:10.1016/j.neucom.2006.02.016.

[188] Petrowski A. A clearing procedure as a niching method for genetic algorithms. Proc IEEE Int Conf Evol Comput ICEC-96 1996:798–803. doi:10.1109/ICEC.1996.542703.

[189] Deb K. Multi-objective NSGA-II code in C Revision 1.1.6. Softw Dev by KanGAL 2011. https://www.egr.msu.edu/~kdeb/codes.shtml.

[190] Deb K, Jain H. Handling Many-Objective Problems Using an Improved NSGA-II Procedure. 2012 IEEE Congr. Evol. Comput., 2012, p. 1–8.

[191] Branke J, Schmeck H, Deb K, Reddy M. Parallelizing Multi-Objective Evolutionary Algorithms : Cone Separation. CEC2004 Evol. Comput., 2004, p. 1952–7. doi:10.1109/CEC.2004.1331135.

[192] Liu M, Zou X, Chen Y, Wu Z. Performance assessment of DMOEA-DD with CEC 2009 MOEA competition test instances. 2009 IEEE Congr. Evol. Comput., 2009, p. 2913–8. doi:10.1109/CEC.2009.4983309.

[193] Liu HL, Li X. The multiobjective evolutionary algorithm based on determined weight and sub-regional search. 2009 IEEE Congr. Evol. Comput. CEC 2009, 2009, p. 1928–34. doi:10.1109/CEC.2009.4983176.

[194] Darwin C. On the origins of species by means of natural selection, Murray, London: 1859.

[195] Ehrlich PR, Raven PH. Butterflies and plants: a study in coevolution. Evolution (N Y) 1964;18:586–608. doi:10.1111/j.1558-5646.1964.tb01674.x.

[196] Nuismer S. Introduction to coevolutionary theory. Macmillan Learning; 2017.

[197] Currie C, Wong B, Stuart A, Schultz T, Rehner S, Mueller U, et al. Ancient Tripartite Coevolution in the Attine Ant-Microbe Symbiosis. Science (80-) 2003;299:386–8.

[198] Potter MA, Jong K. Cooperative coevolution: an architecture for evolving coadapted subcomponents. Evol Comput 2000;8:1–29. doi:10.1162/106365600568086.

[199] Anderson R, May R. Coevolution of hosts and parasites. Parasitology 1982;85:411–26.

[200] Downes S, Shine R. Sedentary snakes and gullible geckos: Predator-prey coevolution in nocturnal rock-dwelling reptiles. Anim Behav 1998;55:1373–85. doi:10.1006/anbe.1997.0704.

[201] Potter MA, De Jong KA. A cooperative coevolutionary approach to function optimization. Int. Conf. Parallel Probl. Solving from Nat., Springer Berlin Heidelberg; 1994, p. 249–57.

[202] Hill J. The three C's - competition, coexistence and coevolution - and their impact on the breeding of forage crop mixtures. Theor Appl Genet 1990;79:168–76. doi:10.1007/BF00225947.

[203] De Jong KA. Evolutionary computation: a unified approach. MIT; 2006.

[204] Coello CAC, Sierra MR. A coevolutionary multi-objective evolutionary algorithm. 2003 Congr. Evol. Comput. CEC 2003, vol. 1, 2003, p. 482–9. doi:10.1109/CEC.2003.1299614.

[205] Goh C, Tan KC. A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. IEEE Trans Evol Comput 2009;13:103–27. doi:10.1109/TEVC.2008.920671.

[206] Rosin C, Belew R. New methods for competitive coevolution. Evol Comput 1997;5:1–29. doi:10.1162/evco.1997.5.1.1.

[207] Jiang S, Yang S. Evolutionary dynamic multiobjective optimization: benchmarks and algorithm comparisons. IEEE Trans Cybern 2017;47:198–211. doi:10.1109/TCYB.2015.2510698.

[208] Wilson D, Sober E. Reviving the superorganism. J Theor Biol 1989;136:337–56.

[209] Okasha S. Evolution and the levels of selection. Oxford University Press; 2006.

[210] Grudniewski PA, Sobey AJ. cMLSGA : Co-evolutionary Multi-Level Selection Genetic Algorithm (Under Review). Swarm Evol Comput n.d.

[211] Huband S, Barone L, While L, Hingston P. A Scalable Multi-objective Test Problem Toolkit. Int. Conf. Evol. Multi-Criterion Optim., 2005, p. 280–95. doi:10.1007/978-3-540-31880-4_20.

[212] Liu HL, Chen L, Deb K, Goodman ED. Investigating the effect of imbalance between convergence and diversity in evolutionary multiobjective algorithms. IEEE Trans Evol Comput 2017;21:408–25. doi:10.1109/TEVC.2016.2606577.

[213] Li X, Engelbrecht A, Epitropakis MG. Results of the 2013 IEEE CEC Competition on Niching Methods for Multimodal Optimization. IEEE Congr. Evol. Comput., 2013.

[214] Li X, Engelbrecht A, Epitropakis MG. Results of the 2015 IEEE CEC Competition on Niching Methods for Multimodal Optimization. IEEE Congr. Evol. Comput., 2015.

[215] Cheng R, Li M, Tian Y, Zhang X, Yang S, Jin Y, et al. Final report of the CEC 2017 Competition. IEEE CEC Compet. Evol. Many-Objective Optim., 2017.