Engineering Optimization
Vol. 46, No. 1, 2014, 1–22

Research Article

Customized Evolutionary Optimization Procedure for Generating Minimum Weight Compliant Mechanisms

Deepak Sharma[a*], Kalyanmoy Deb[b] and N. N. Kishore[b]

Department of Mechanical Engineering
[a]*Indian Institute of Technology Guwahati, PIN 781039, India*;
[b]*Indian Institute of Technology Kanpur, PIN 208016, India*

(*v3.9 released December 2010*)

In this paper, a customized evolutionary optimization procedure is developed for generating minimum weight compliant mechanisms. A previously-suggested concept of multi-objectivization in which a helper objective is introduced in addition to the primary objective of the original single-objective optimization problem (SOOP) is used here. The helper objective is chosen in a way so that it is in conflict with the primary objective, thereby causing an evolutionary multi-objective optimization algorithm to maintain diversity in its population from one generation to another. The elitist non-dominated sorting genetic algorithm (NSGA-II) is customized with a domain-specific initialization strategy, a domain-specific crossover operator, and a domain-specific solution repairing strategy. To make the search process computationally tractable, the proposed methodology is made suitable for parallel computing. A local search methodology is applied on the evolved non-dominated solutions found by the above-mentioned modified NSGA-II to further refine the solutions. Two case studies for tracing curvilinear and straight-line paths are performed. Results demonstrate that solutions having smaller weight than the reference design solution obtained by SOOP are found by the proposed procedure. Interesting facts and observations brought out by the study are also narrated and conclusions of the study are made.

Keywords: Helper objective, Multi-objectivization, Reference solution, Compliant mechanisms, Customized evolutionary algorithm

*Corresponding author. Email: dsharma@iitg.ernet.in

ISSN: 0305-215X print/ISSN 1029-0273 online

1. Introduction

Multi-objectivization concept (Knowles *et al.* 2001) was introduced for solving complex single-objective optimization problems (SOOPs) in a computationally efficient manner. In this approach, SOOP is converted into a multi-objective optimization problem (MOOP) by adding a secondary objective which is in conflict with the original objective function. Since diverse trade-off solutions are maintained in an evolutionary multi-objective optimization (EMO) algorithm, it is then demonstrated that multi-objectivization can introduce adequate diversity in an EMO population, thereby enhancing the chance of finding good new solutions with generations.

In addition to introducing a new secondary objective, the original objective function can also be splitted, if possible, to have two separate yet conflicting objectives. In addition to using deterministic objective functions, Jensen (2004) suggested the use of multiple dynamically changing helper objectives that were minimized simultaneously with the primary objective. Although at first, a sequence of helper objectives were chosen at random, later problem-specific knowledge was used to choose them (Lochtefeld and Ciarallo 2011). The idea was tested on the traveling salesman and job-shop scheduling problems. These problems can either be decomposed or be converted to have many helper objectives from the original singular objective. Bleuler *et al.* (2001) and De Jong *et al.* (2001) however used the size of a genetic program as an additional objective to come up with compact yet efficient programs using the genetic programming search algorithm (Koza 1992). Similarly, Abbass and Deb (2003) used the solution age as a supplementary objective, and Toffolo and Benini (2003) considered a distance-based measure of genetic diversity as an objective. Watanabe and Sakakibara (2005) followed a different approach in which additional objectives were added to the primary objective by (i) relaxing constraints of the problem (Coello 2000) and (ii) adding noise to the objective value or to the decision variables. The concept of helper objective was also used to design frames in Greiner *et al.* (2007), in which minimizing the number of different cross-sectional shapes used in a frame was added to the usual minimization of mass of the frame structure.

In this paper, the optimization of a compliant mechanism is considered which usually resorts to a non-linear, high-dimensional and discrete optimization problem (described in Sections 3 and 4). The complexities involved in the problem result in a fewer well-connected practical designs that respect all constraints. Major problems associated with the compliant mechanism design optimization are as follows. The objective function landscape usually has multiple local optima due to inherent non-linearity in constraint and objective functions. Moreover, due to scarcity of feasible solutions, an optimization algorithm is likely to converge to a premature and sub-optimal solution.

A unique approach is followed in this study where a helper objective of maximizing geometrical dissimilarity with a single-objective optimized solution is introduced for optimizing a complaint mechanism. The reference design is generated off-line by using a single-objective optimization procedure discussed in Section 3.1 subjected to satisfaction of constraints outlined in Section 3.2. By forcing population members to have dissimilar shapes than the single-objective optimal solution, diversity in the population is expected to be maintained and the process is expected to arrive at a good set of non-dominated solutions at the end. The obtained trade-off solutions are further refined by a hill-climbing based local search procedure to ensure closeness to locally optimal solutions.

The remainder of the paper is organized as follows. A helper objective based formulation of the complaint mechanism design task is discussed in Section 3. The proposed customized evolutionary algorithm with a hill-climbing based local search procedure is discussed in Section 4. In section 5, two case studies are presented, their minimum weight elastic configurations are presented, and various useful observations are discussed. The paper is concluded in Section 6 with an outline of a few future extensions to this study.

2. Overview of Compliant Mechanism Design

Structural topology optimization (STO) is a fast growing field which is finding numerous applications in automotive, aerospace, and mechanical design processes. It optimizes material distribution or layout within a given design-domain under the applied boundary and support conditions (Hassani and Hinton 1998). Compliant mechanisms (CMs) are also synthesized based on STO approach where the flexible elastic structures of these mechanisms can deform to perform desired task (Howell 2001). Different criteria have been used to design CMs such as strain energy (Frecker *et al.* 1997), geometric and mechanical advantages (Sigmund 1997, Larsen *et al.* 1997), mutual potential energy (Lu and Kota 2006) etc.

Commonly used methods for synthesizing CMs are homogenization method (Nishiwaki *et al.* 1998), material density approach (Yang and Chuang 1994), solid isotropic microstructure with penalization (SIMP) (Bendsøe 1989), level-set method (Wang *et al.* 2003), and evolutionary structural optimization (ESO) (Xie and Steven 1993). Homogenization, material density, and SIMP methods modify discrete problem of structures into a continuous design variable problem. A threshold value is often required to suppress intermediate design variable values. However, an arbitrary threshold value can lead to a non-optimal solution. Although homogenization and ESO methods are computationally effective, their convergence to the global optimal solution for the structural optimization problem is not guaranteed (Rozvany 2001).

Boolean (0-1) representation of genetic algorithm (GA) can be used for material distribution in a design domain. Thus, GA is used as another method for STO (Chapman *et al.* 1994) to preserve discrete nature of the structures. Moreover, GAs are suitable for solving complex non-linear problems for the global optimization and for simultaneously handling multiple objectives (Sharma *et al.* 2011). But, the bottleneck is large computation time. Major portion of the computation time is consumed in the finite element analysis (FEA) of the structures as compared to the time spent in executing GA operations.

Most studies in STO combine multiple objectives into single-objective using classical approaches: (i) weighted-sum method, (ii) ratio method, etc., where the optimization process can get trapped into finding a sub-optimal solution. On the other hand, a multi-objective approach can introduce and maintain diversity in the search process. The concept of helper objective has already been suggested for improving the frame bar structures (Greiner *et al.* 2007). In this paper, the helper objective concept is introduced for maximizing geometric dissimilarity in the population of complaint mechanisms.

It has been observed that problem-specific knowledge given to GA can become a potential tool for global structural topology optimization (Wang *et al.* 2006). The problem-knowledge can be provided at various stages of GA. For example, a domain-specific initial population strategy can help in making a faster convergence of GA (Sharma *et al.* 2011). Similarly, a problem-specific crossover operator (Sharma *et al.* 2011, Deb and Chaudhuri 2005) can be used to enhance the chance of creating better offspring solutions in the GA population. Computation time of running GA can be reduced by performing function evaluations and FEM simulations on a parallel computing platform (Sharma *et al.* 2006, 2008).

3. Problem Formulation

3.1. *Objectives*

Minimizing weight of the structure is chosen as the primary objective. Weight of the structure is evaluated by counting the cells having material (black color) as shown in Fig. 1.

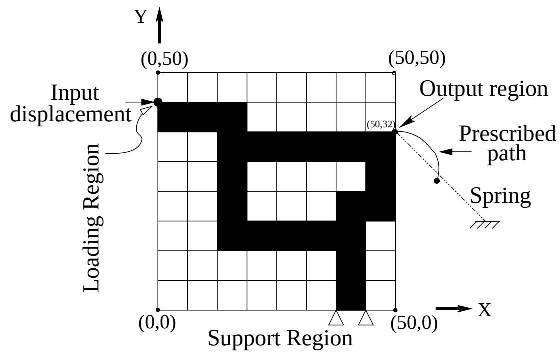


Figure 1. A design-domain for GA population member.

The concept of multi-objectivization (Knowles *et al.* 2001) is used in this paper where the helper objective is coupled with the primary objective. As mentioned in Section 1, multi-objectivization can guide a search of EA to avoid local optima and to maintain diversity in the EA population based on Pareto-ranking. In this study, the helper objective is designed to get evaluated with respect to the reference design. The reference design can be chosen from the available set of optimum elastic structures based on previous practice of designers or from the literature. In this paper, the reference design is generated by a SOOP for weight minimization (cf. equation (1)) subjected to the constraints which are discussed in Section 3.2.

Single-objective optimization (for reference design):
Minimize: Weight of structure

Bi-objective optimization: (1)
Minimize: Weight of structure (primary objective),
Maximize: Geometrical dissimilarity of structure with respect to the reference design (helper objective).

For the given CM problem, the reference design obtained from SOOP application becomes a reference point which specifies the goal or aspiration level for each objective. This reference point methodology can help an EA to find a set of solutions closer to the supplied goal (Deb *et al.* 2006).

For evaluating the helper objective, every cell of the elastic structure evolved from the current EA population is compared with corresponding cell of the previously obtained reference design. For example, if a cell that is present at a particular EA population member is absent in the reference design, the dissimilarity count of the EA population member is increased by one. The total sum of dissimilarity count is a measure of geometrical dissimilarity of EA evolved elastic structure with respect to the reference design.

This dissimilarity count is maximized simultaneously along with minimization of weight of the structure (see equation (1)).

3.2. *Constraints*

The task of path generating compliant mechanisms (PGCMs) is to develop an optimal shape of the elastic structure so that when a force or displacement is applied at a specified point on the mechanism, a user-defined path is generated by another point on the mechanism. These mechanisms can be designed using Euclidean distance-based objective function (Tai *et al.* 2002) or Fourier shape descriptor-based objective function (Rai *et al.* 2007). The former approach cannot ensure closeness of actual path traced by the elastic structure to a user-defined path. It can result an unacceptable gap between the two paths. The later approach depends on many user-defined parameters. Arbitrary values to these parameters can change the optimality of the solution. In this paper, the constraints are imposed at the precision points representing a user-defined path (Sharma *et al.* 2011) (see equation (2)). These constraints restrict a maximum allowed gap between a user-defined path and an actual path. This helps in guiding EA to evolve feasible CMs tracing a user-defined path. A constraint limiting the stress within the material strength is also considered in the formulation.

Constraints:

$$1 - \frac{\sqrt{(x_{ia}-x_i)^2+(y_{ia}-y_i)^2}}{\eta\times\sqrt{(x_i-x_{i-1})^2+(y_i-y_{i-1})^2}} \geq 0, \text{where } i = 1,\ldots,N,$$
$$\sigma_{flexural} - \sigma \geq 0, \tag{2}$$

where N is the number of precision points representing the user-defined path, $\eta = 15\%$ is the permissible deviation (kept fixed in this paper), and $\sigma_{flexural}$ and σ are flexural yield strength of material and maximum stress developed in the structure, respectively.

The physical interpretation of these constraints is shown in Fig. 2. A hypothetical case of user-defined and actual paths is shown in the figure. It suggests that the solution is feasible only when $d_2 \leq d_1$, where $d_1(=\sqrt{(x_{ia}-x_i)^2+(y_{ia}-y_i)^2})$ is Euclidean distance between the precision point (i) and the corresponding point (ia) of actual path. Distance $d_2(=\eta\times\sqrt{(x_i-x_{i-1})^2+(y_i-y_{i-1})^2})$ is $\eta\%$ of Euclidean distance between the current

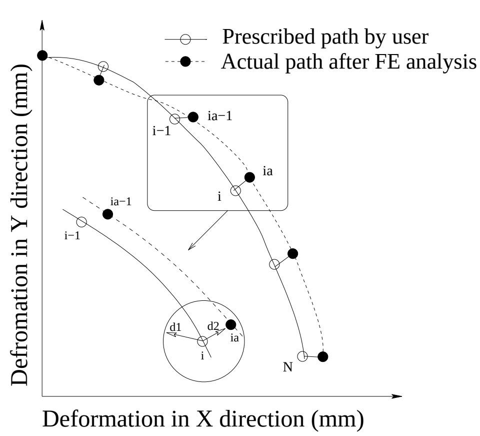


Figure 2. A prescribed path and an actual path traced by the elastic structure.

precision point (i) and previous precision point ($i-1$).

The evolved designs will be referred as path tracing compliant mechanisms (PTCMs) because the term 'path' signifies a user-defined path which is traced by imposing constraints as discussed above. However, for path generating compliant mechanisms (PGCMs) the 'path' suggests an actual path generated by the compliant mechanisms.

4. Customized Evolutionary Algorithm

In this paper, the elitist non-dominated sorting genetic algorithm (NSGA-II, Deb *et al.* (2002)) is modified and used for solving the problem described in Section 3. NSGA-II has been chosen over other algorithms because it has shown to have a good convergence property to the global "Pareto-optimal" front as well as to maintain a good diversity of population members on the "Pareto-optimal" front for various two-objective test and engineering problems (Deb 2001).

The non-dominated solutions generated by the customized NSGA-II are further refined by the hill-climbing based local search method. These solutions are starting points for the local search procedure which minimizes weight of structure. A flow-chart of the optimization procedure is shown in Fig. 3. An earlier NSGA-II algorithm (Sharma *et al.* 2011) is further customized with modifications such as domain-specific crossover operator and a local search for the minimum-weight solution. Next, the customized schemes and operators are discussed.

4.1. *GA Parameters*

Customized NSGA-II parameters for solving the given compliant mechanism problem are given in Table 1. It can been seen that two sets of binary strings are used. The first set is used to represent material distribution in the design domain which is described later in Section 4.3. The second set is used to identify the applied boundary and support conditions. The applied boundary conditions are associated with the location and magnitude of input displacement. For identifying the conditions, the second set of binary string is thus divided into three groups as shown in Fig. 4. The first group of five bits decodes a position of a cell where the nodes of a cell of the elastic structure are restrained with zero displacement. It is referred as "support region" which can be seen at the bottom of design domain shown in Fig. 1. The decoded value of three bits of the second group identifies a loading position, that is, a node of cell of the elastic structure where the input displacement boundary condition is applied. This position is referred as a "loading region" in Fig. 1. The magnitude of the input displacement is also calculated after decoding the four bits of third group which varies in range of 1 mm to 16 mm at a step of 1 mm.

Table 1. NSGA-II parameters used in this study.

Population	240	**Generation**	100
Crossover probability	0.95	**Mutation probability**	1/string length
String length for a structure	625	**String length for applied boundary and support conditions**	12

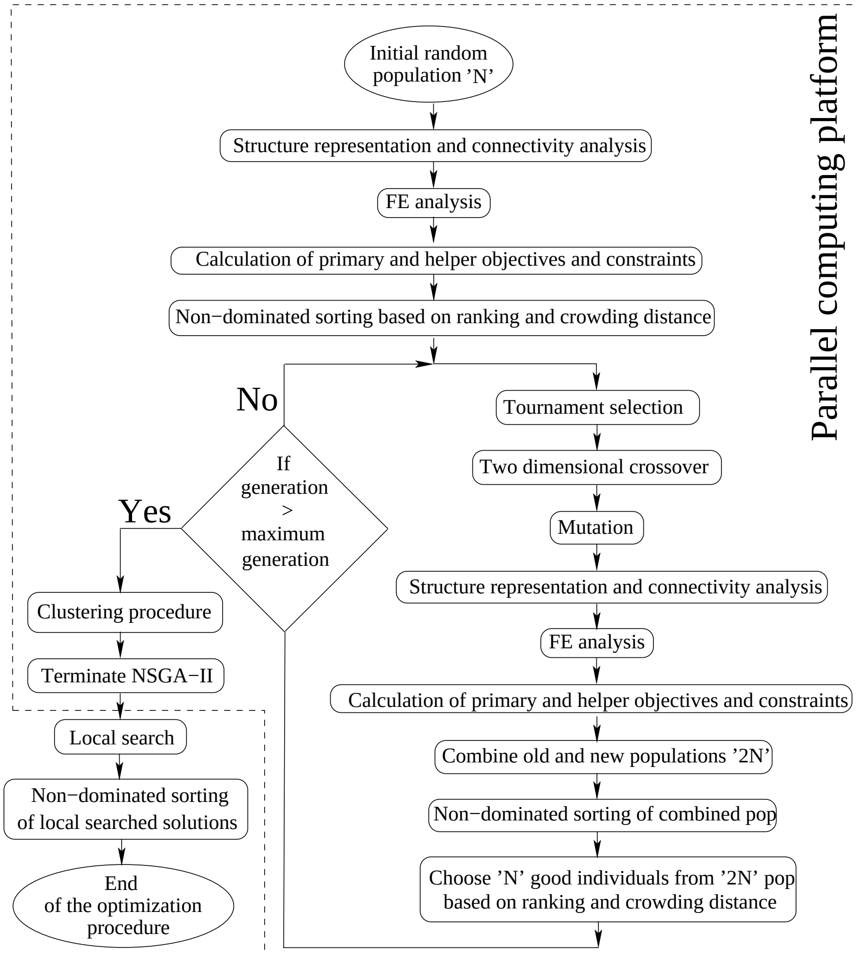


Figure 3. A flow chart of customized NSGA-II algorithm

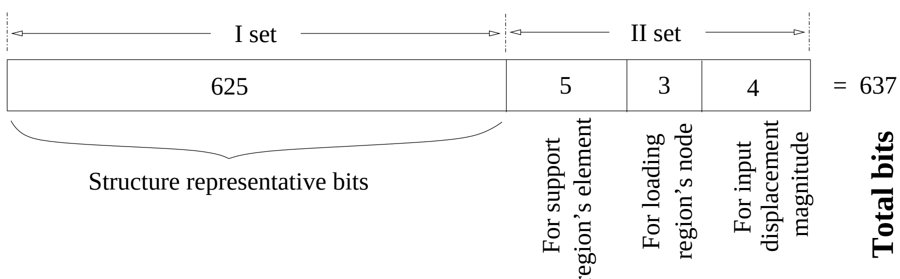


Figure 4. A binary string comprises of two sets.

4.2. *Domain-Specific Initial Population Strategy*

A domain-specific initial population strategy is used and coupled with the customized EA described above. The strategy was found to outperform a random initialization procedure for the compliant mechanism problem (Sharma *et al.* 2011). In this strategy, the material connectivity among the important regions of a design domain is ensured. The structure is geometrically feasible when the support, loading and output regions of a design domain (cf. Fig 1) are connected through the material. The output region is referred as a fixed point on a design domain that traces out a user-defined path. The origin of design domain is fixed on its left side and the output region is positioned at the coordinate $(50, 32)$ of the structure. A spring of constant stiffness ($\kappa = 0.4$ KN/m) is attached at the output node for providing some resistance during the deformation of elastic structure.

Material connectivity between the support and the loading regions is shown in a hypothetical case of Fig. 5. This strategy starts with generating any number of intermediate points between 1 and 5, randomly. In the present case, four random points (P1, P2, P3, and P4) are generated within the design domain. Thereafter, the support (S1) and the loading (L1) positions are connected by straight lines passing through these intermediate points. Material is then assigned to those cells of the design domain where these straight lines pass (cf. Fig. 5). Similarly, a set of piece-wise linear line segments between the support and the output regions and another set between the loading and the output regions are added. Here, the cell positions of support and loading regions are calculated after decoding the binary string of 12 bits (see Fig. 4). This initial population strategy ensures creation of geometrically feasible structures in the initial population.

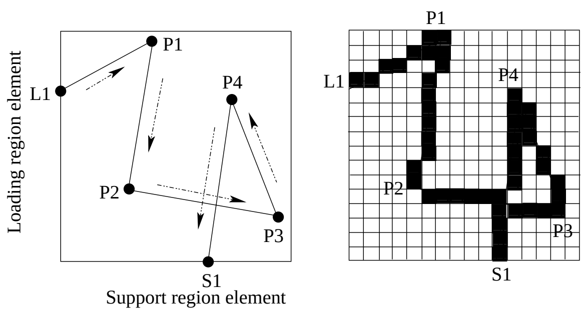


Figure 5. Connectivity between support and loading regions.

4.3. *Structure Representation Scheme*

After an initial population is generated, the binary string of GA population member is converted into two-dimensional representation as shown in Fig. 6. This representation scheme divides the design domain into 25×25 ($= 625$) cells in x and y directions respectively. The bit value 1 of a cell signifies that material is present, whereas 0 represents a void cell.

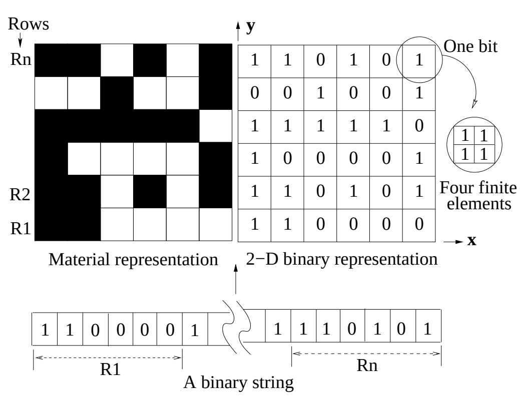


Figure 6. A representation of structure using binary string and material.

4.4. *GA Operators*

A domain-specific crossover operator is introduced here. The given design domain is divided into support, loading and output regions. Thereafter, in this operator, two offspring solutions are created by exchanging sub-domains of the two parent solutions. A common sub-domain between the two parent solutions can exchange with a probability of 0.5. A hypothetical case is shown in Fig. 7 in which three random points (P1, P2,

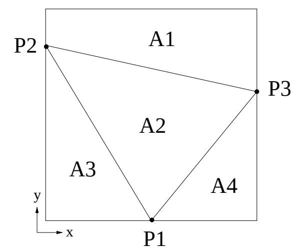


Figure 7. Sub-divided domain for crossover.

and P3) are generated on the respective edges of support, loading and output regions. By joining these points, the design domain is divided into four sub-domains. Each sub-domain then gets swapped between the two parent solutions with 0.5 probability. For a crossover with 12 bits in the second set, a standard single point crossover is used.

A standard bit-wise mutation operator is used for the binary string representing the elastic structure with a mutation probability of $p_m = 1/\ell$ (where ℓ is the string length) on each bit to change from a void to a filled or from a filled to a void cell. For mutating remaining 12 bits of second set, the decoded values of support and loading regions, and magnitude of input displacement are evaluated. These values are then perturbed within the range of $\{-2, 2\}$ at their original values. Here, it is ensured that the perturbed values of above three applied boundary and support conditions do not fall outside their respective bounds. After perturbation, these mutated values are again coded into the binary string of 12 bits.

4.5. *Connectivity and Repairing Techniques*

GA operators create new solutions in the population to explore the search space. While implementing them on the binary strings, new solutions may have disconnected or infeasible geometries. For example, in Fig. 8, the support region (**S**) is not connected to the

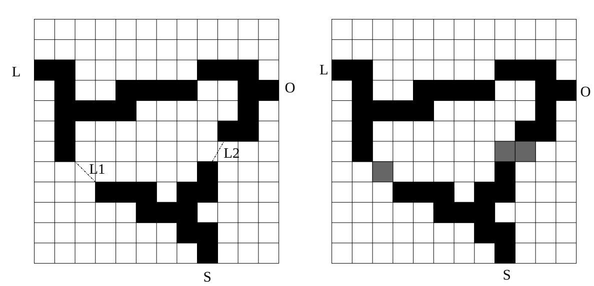


Figure 8. Disconnected topology.

loading (**L**) and the output (**O**) regions. In this scenario, individual distances are calculated from the centroid of each cell of material connected to **S** to the centroid of each cell of material connected to **L** and **O**. The regions are then connected by straight lines where individual distances between the cells of **S-L** and **S-O** are minimum. Material is then assigned to those cells where these straight lines pass.

The initial population strategy, the GA operators, and the connectivity procedure can cause a point singularity between the two cells of material. In this paper, a heuristic repairing technique motivated from the image processing concept (Sigmund 1997) is employed. For any cell of material as shown in Fig. 9(a), a point singularity can arise if

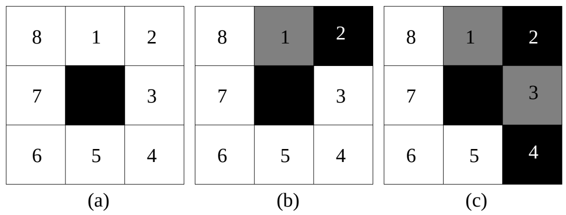


Figure 9. Point singularity case is illustrated.

there is any material at 2nd, 4th, 6th, or 8th position and there is no material at 1st, 3rd, 5th, and 7th positions. Suppose, material at position 2 creates a point connectivity (see Fig. 9(b)), then an extra material is filled at 1st or 3rd position with an equal probability to eliminate the point singularity. Fig. 9(c) illustrates a case for a point connectivity which arises due to material at 2nd and 4th positions.

Due to the mutation operator, a topology with isolated elements may also get created as shown in Fig. 10. In this case, the smallest isolated cells are identified and they are changed to void cells by assigning a value ‘0’ to each of them.

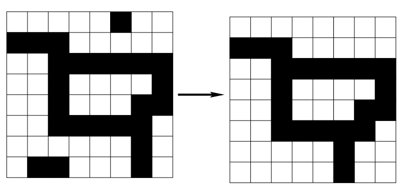


Figure 10. Isolated elements case due to mutation is illustrated.

4.6. *Finite Element Analysis (FEA)*

The above repair mechanisms make sure that the elastic structures are geometrically feasible and are free from checkerboard configuration, point singularity, and isolation. Thereafter, the structures are analyzed using FEM. In this paper, one cell of a structure is further discretized into four finite elements with an identical binary variable value as shown in Fig. 6. Hence, the structure is discretized with 4×625 $(= 2500)$ 4-node rectangular finite elements for non-linear large deformation FEA using ANSYS package. It is noted that the GA operators are executed on 625 bits of the same structure.

4.7. *Objective and Constraint Functions Evaluation*

After FEA, objectives and constraints are evaluated. The primary objective of every individual is calculated as discussed in Section 3.1 using material distribution in the design domain. The helper objective is evaluated by counting dissimilar binary bits with respect to the reference design. In this procedure, the minimum-weight reference design solution becomes the reference point for the given problem where the goal or aspiration level is specified for each objective. In this case, the customized EA is expected to generate diverse Pareto-optimal solutions closer to the reference design solution.

For evaluating constraints at every precision point, points (ia, refer Fig. 2) on actual path is supplied by FE analysis of the elastic structure. Maximum stress developed in the structure is also calculated by the finite element simulation which is then used for stress constraint computation.

4.8. *Ranking*

The elitist non-dominated sorting of NSGA-II (Deb *et al.* 2002) is used for ranking the GA population. Primary and helper objectives are used to evaluate non-dominated ranking of every solution. Primary objective enables the customized EA to generate smaller weight solutions, whereas the helper objective helps to maintain geometrical diversity in solutions, so that NSGA-II's crossover operator is able to exploit the diversity in parent solutions to create new and hopefully useful offspring solutions. Crowding distance operator of NSGA-II is used to preserve the diversity in the population.

4.9. *Parallel Computing*

The function evaluations and FE simulations consume larger proportion of computation time of the optimization process. It can be reduced significantly by using a parallel computing platform. In the parallel implementation, the population of NSGA-II is divided into sub-populations which are sent to multiple available slave processors for function evaluations and FE simulations. The master processor controls the overall procedure and gets back the objective and constraint values from slave processors required for ranking and executing other GA operators. This master-slave parallelization process is repeated till the termination criterion of NSGA-II is met. A Linux cluster with 24 processors is used on which the Message Passing Interface (MPI) environment is used.

4.10. *Clustering Procedure*

When NSGA-II is applied to a problem, it is usually able to find as many solutions as there are population members. For a proper functioning of NSGA-II, as adequate number of population members are needed. However, if NSGA-II population size is large, decision-makers cannot handle too many trade-off solutions, as it is too demanding of their time and efforts. Often, in such problems, there is a discrepancy in the required population size and the expected number of trade-off solutions desired at the end. For this purpose, the use of a clustering procedure for which a few well-diversed solutions are chosen from the final set of non-dominated solutions is suggested. In the proposed approach, the neighboring solutions (in the objective space) are grouped together using the k-mean clustering algorithm and one solution from each group representing that region of the non-dominated front is chosen as final representative solutions (Sharma *et al.* 2011).

4.11. *Minimum-Weight Local Search Method*

Genetic algorithms are good near-optimizers. Their convergence ability can be enhanced if the final NSGA-II solutions are modified further by using a local search procedure. Since the helper objective was used to maintain diversity in the NSGA-II population, its purpose is over with the application of NSGA-II. For the local search, only the primary objective subject to satisfaction of constraints (see equation (2)) is optimized.

In the local search method, binary string of the solution is converted into a two-dimensional array which is then checked for the filled cells. For each material's cell, there are eight possible neighborhood cells, as shown in Fig. 9. One by one, all neighboring bits including its own bit value is mutated to its complement. The new elastic structure is then extracted on which finite element computations are performed for evaluating the objective function and constraints values. If the new structure does not satisfy any of the constraints, then the change in the new string is discarded and the old values are restored. Otherwise, the weight of structure represented by the new string is calculated and compared with that of the old string's value. If mutating a bit brings an improvement in the weight objective and the solution remains to be feasible, then the change is accepted. Else, the change is discarded and the previous values are restored. When all the bits having a material are mutated along with their neighborhoods, the cells of the new elastic structure are again checked for material and are mutated as discussed above. The local search method is terminated when no change in a bit improves the weight objective value and simultaneously keep the solution feasible. In the same way, all chosen non-dominated solutions are mutated one by one. This post-processing method is an exhaustive method that may require considerable computation time to refine the solutions, but is useful to make sure that the obtained solutions are locally optimal.

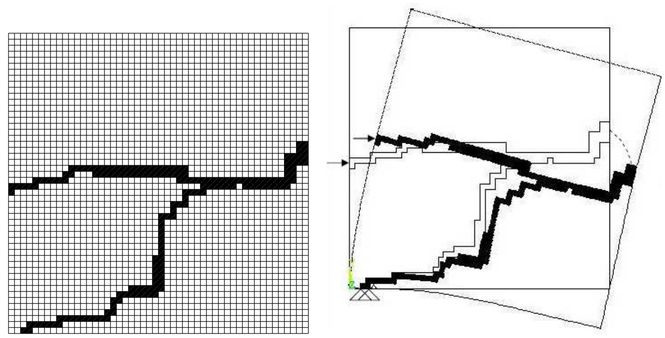
5. Results and Discussion

Two examples of the compliant mechanisms (CMs) tracing: (i) curvilinear path, and (ii) straight line path are solved. These examples have been solved in a previous study (Sharma *et al.* 2011) as well using a different bi-objective formulation of the problem. Parameters that are kept constant for both examples are the Young's modulus ($E = 3.3$ GPa), flexural yield stress ($\sigma_{flexural} = 69$ MPa), density ($\rho = 1.114$ gm/cm^3), and Poisson's ratio ($\nu = 0.40$) for the chosen material. The direction of input displacement is along the x-axis. Five precision points are used to represent the prescribed path. Maximum six representative solutions are chosen from the non-dominated set using the clustering procedure described in Section 4.10.

5.1. *Example 1: CMs Tracing Curvilinear Path*

5.1.1. *Reference Design*

The reference design from SOOP is shown in Fig. 11. This design consists of open loops of material joining three regions of interest (support, loading, and output regions). The applied boundary and support conditions evolved by the optimization procedure defined in Section 4.1 are as follows. This reference design is supported at an element which is positioned at 2 mm away from origin. An input load of 5 mm is applied at a node which is located at 24 mm away from origin. Here, origin is at the bottom left corner of the reference design. The weight of reference design is found to be 0.545 gm.

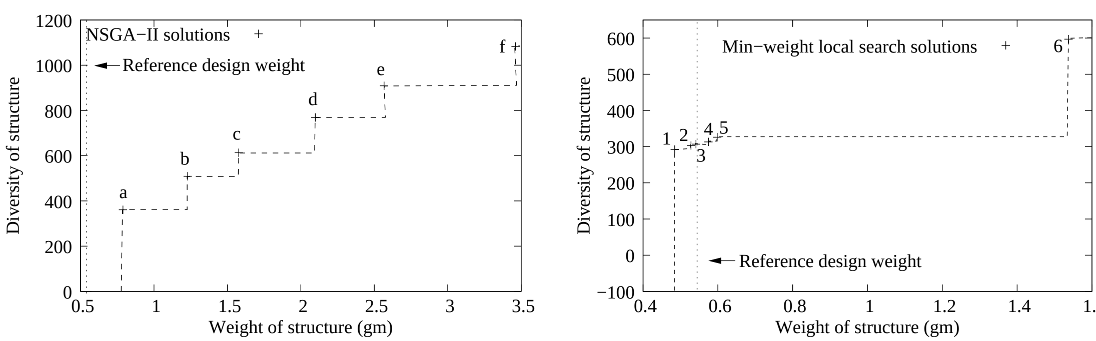


(a) Undeformed topology (b) Final deformed topology

Figure 11. The undeformed and final deformed *reference* design for Example 1 problem.

5.1.2. *Multiple PTCMs*

The above reference design is now used to evaluate the helper objective for the bi-objective optimization task. The representative non-dominated solutions, a to f, evolved by the customized EA are shown in Fig. 12(a). It can be observed from this figure that the customized NSGA-II is able to find a wider range of solutions for which the weight of the structures range from 0.75 gm to 3.5 gm. Due to a trade-off between the two objectives, the minimum weight solution 'a' shows a minimum geometrical dissimilarity with respect to the reference design. On the other hand, the extreme solution 'f' shows the maximum dissimilarity but evolved as the heaviest solution.



(a) NSGA-II solutions (b) Local search solutions

Figure 12. Bi-objective solutions for the curvilinear path tracing CMs (Example 1).

The minimum-weight NSGA-II solution (solution 'a') is heavier than the SOOP solution. But importantly, NSGA-II procedure finds five other diverse solutions along with solution 'a'. These six solutions are then modified using the proposed local search procedure. The obtained solutions are shown in the right plot of Fig. 12. It is interesting that the local search procedure finds a smaller weight solution (solution 1 for example) than the SOOP solution. Importantly, two other local searches find solutions that are better than the SOOP solution. The use of helper objective helps to find a number of good seed solutions which in turn enables the proposed local search procedure to begin its search from them and find better solutions. The discovery of multiple yet similar solutions by the combined NSGA-II and local search procedure gives the user confidence about the near-optimality of obtained minimum-weight solutions by the overall procedure.

A SOOP procedure usually takes its course favouring solutions that are comparatively easier to generate emphasizing a single objective (weight, in this case). Thus, a SOOP approach is likely to get stuck to a sub-optimal solution in the case of a complicated problem. On the other hand, a bi-objective optimization procedure with two conflicting objectives in its search process maintains adequate diversity in the GA population to help overcome sub-optimality. The current case study provides a good testimony to the advantage of the proposed bi-objective approach.

5.1.3. *Designs*

The minimum-weight local search solutions are shown in Fig. 13. The solution 6 is discarded as it is far from the reference design solution due to its higher weight value. The corresponding NSGA-II solution (solution 'f') was too far from the minimum-weight solution for it provide much help to the local search algorithm to find a solution close to the minimum-weight solution. It can be seen that PTCMs are evolved as T-shaped structures. Although topologically they are similar, they are dissimilar due to the shape of material segments from support, loading and output regions, and their connectivity. For example, a material segment from support position of solution 4 is straight as compared to solutions $1, 2$ and 3. Similarly, a material segment joining the common junction and the output region for each solution is different. Junction points of these segments are located at different positions for each solution that result in a different material segment joining the junction and the loading region.

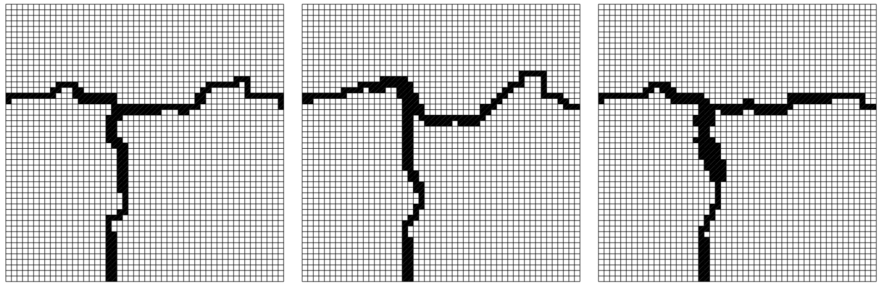
5.1.4. *Posterior Analysis*

5.1.4.1. Conditions:. To compare the obtained solutions with the SOOP solution, the locations of applied boundary and support conditions are tabulated in Table 2. For solutions 1 to 5, identical locations are obtained, however they are different from the those in the reference SOOP design.

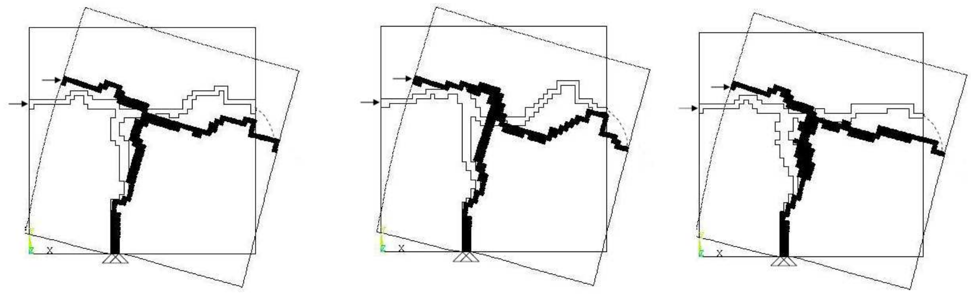
Table 2. Evolved applied boundary and support conditions for curvilinear PTCMs.

Study → Conditions ↓	Single-objective (Ref. design)	Bi-objective (Solutions 1 to 5)
Support position (mm) (*from the origin*)	2	18
Loading position (mm) (*from the origin*)	24	32
Input displacement (mm)	5	7

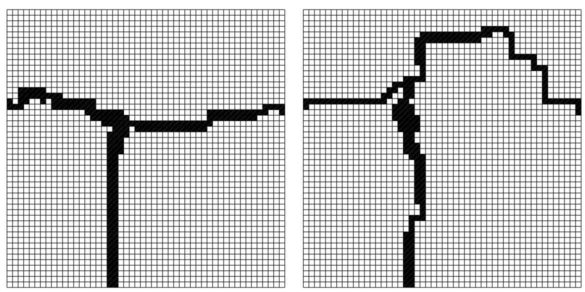
5.1.4.2. Evolution of Conditions:. Evolution of applied boundary and support conditions for the feasible solutions is shown in Fig. 14. Different sets of these conditions are observed during NSGA-II runs. It signifies that only a few sets of applied boundary and support conditions can evolve feasible solutions. This means that any arbitrary set of these conditions cannot evolve feasible solutions that can trace given prescribed path. When NSGA-II terminates, the representative non-dominated solutions are subjected to identical conditions as mentioned in Table 2. It can be observed here that only a set of applied boundary and support conditions is associated with the non-dominated solutions, whereas other sets of conditions may correspond to dominated or infeasible solutions. Such an analysis is useful to understand the dynamics of development of key features of NSGA-II population members with generation.



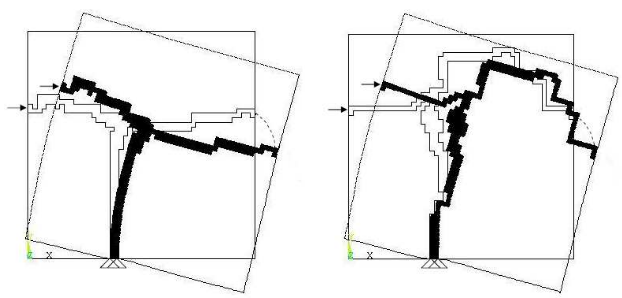
(a) Sol. 1: Undeformed (b) Sol. 2: Undeformed (c) Sol. 3: Undeformed



(d) Sol. 1: Final deformed (e) Sol. 2: Final deformed (f) Sol. 3: Final deformed



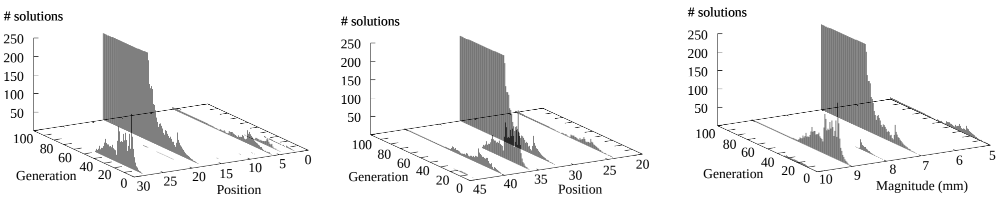
(g) Sol. 4: Undeformed (h) Sol. 5: Undeformed



(i) Sol. 4: Final deformed (j) Sol. 5: Final deformed

Figure 13. Lightweight CMs tracing curvilinear path.

5.1.4.3. Path:. The prescribed path and the path traced by local search solutions from single and bi-objective studies are shown in Fig. 15. As constraints are imposed at each precision point, a gap between the prescribed path and the traced paths are limited to d_1 as described in Fig. 2. Numerical values of d_1 and d_2 for all local search solutions are shown in Table 3. It can be seen from this table that d_2 value of the reference design first increases till the precision point 2 and then it decreases at precision point 3. Finally, it



(a) Support position (b) Loading position (c) Magnitude of input displacement

Figure 14. Evolution of applied boundary and support conditions for feasible solutions during NSGA-II run.

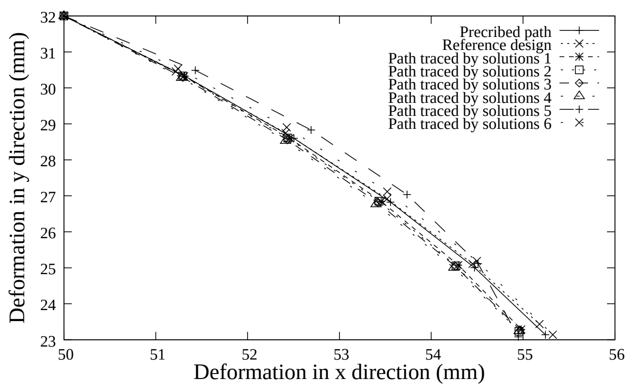


Figure 15. Prescribed path and path traced by the *reference* design and by the minimum weight local search solutions.

makes the precision point 5 critical because d_2 value is close to maximum allowed d_1 value. A similar behavior can be seen in d_2 value of solution 5, but its traced path intersects the prescribed path after precision point 4 (refer Fig. 15). Other solutions show an increasing trend of d_2 value that makes the precision point 5 critical. This indicates that constraint at precision point 5 is active for all solutions. In this example, the prescribed path is designed in such a way that an output node of each structure deforms 10.48% in x-direction and 17.72% in y-direction with respect to the size of design domain (Sharma *et al.* 2011).

Table 3. Deviation at precision points.

Precision points (PP)	1	2	3	4	5
Maximum allowed $\mathbf{d}_1$	0.3196	0.3142	0.3093	0.3056	0.3027
Reference design					
$\mathbf{d}_2$	0.1662	0.1714	0.1017	0.1088	0.3026
Two-objective study					
Solution 1: $\mathbf{d}_2$	0.0097	0.0439	0.1027	0.1903	0.3025
Solution 2: $\mathbf{d}_2$	0.0233	0.0664	0.1278	0.2086	0.3026
Solution 3: $\mathbf{d}_2$	0.0417	0.0841	0.1374	0.2110	0.3027
Solution 4: $\mathbf{d}_2$	0.0586	0.1145	0.1672	0.2269	0.3026
Solution 5: $\mathbf{d}_2$	0.1994	0.2987	0.2788	0.1214	0.3012
Solution 6: $\mathbf{d}_2$	0.2208	0.3142	0.2941	0.1851	0.0809

5.1.4.4. Computational Time:. Time consumed by the customized EA followed by the local search method is given in Table 4. Objectives and constraint functions along with FE simulations are done in parallel on 24 homogeneous processors. It is observed that a proportion of time consumed in GA operations, ranking and communication among the processors is very small as compared to function evaluations and finite element simulations. This signifies that the computational time of customized EA is reduced almost in proportion to the number of available processors. Local search is performed serially on different processors that consumes a considerable amount of time to refine the representative non-dominated solutions.

5.2. *Example 2: CMs Tracing Straight Line Path*

In this example, CMs are generated for the same design domain as shown in Fig. 1. However, the point on the output region has to trace a straight line path. The reference design tracing straight line path is generated using SOOP. The weight of reference design is found to be 0.702 gm. It consists of four closed loops of material as shown in Fig. 16. This reference design is supported at an element located at 48 mm away from the origin. The input load of 8 mm is applied at a node positioned at 40 mm away from the origin. Here, origin is at the bottom left corner of the reference design.

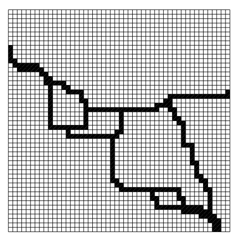


Figure 16. Straight line tracing reference design obtained by SOOP.

Using the above reference design, the bi-objective optimization problem is solved for generating straight line path tracing CMs. The evolved non-dominated solutions, a to f, are shown in Fig. 17(a). Due to the helper objective, the customized NSGA-II is again evolved a wider range of solutions for which the weight of the structures range from 1.5 gm to 3.5 gm. These solutions are refined by the local search procedure which are shown in Fig. 17(b). It can be seen that the local search procedure finds smaller weight solutions, 1 to 4, that are better than the SOOP solution. In this example, the solutions 3 and 4 are almost converged to the same point. Thus, the solution 3 is chosen as the representative solution and the solution 4 is discarded. This case study provides another good evidence to the advantage of the proposed bi-objective approach.

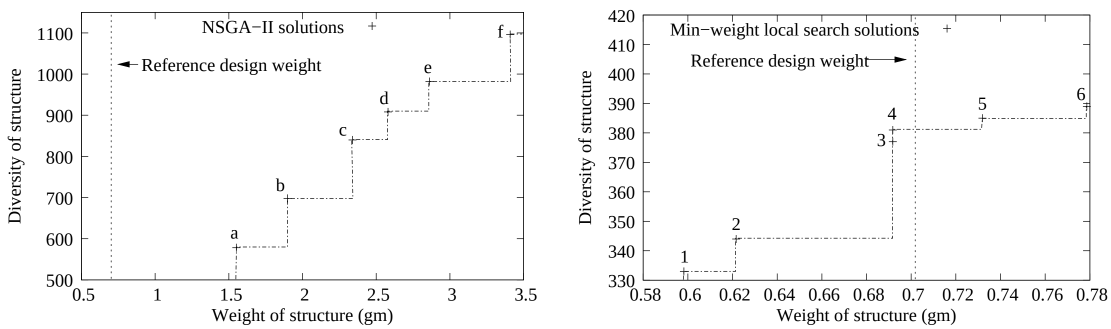
Designs of minimum-weight local search solutions are shown in Fig. 18. Similar looking designs are evolved which possess a bigger closed-loop of material except solution 6. Solutions 1 and 2 show a similar closed loop of material but their material segments joining the loop are slightly different. Solutions 3 and 5 have differently shaped closed loop which result in different shapes of segments joining the closed-loop with the loading

Table 4. Computational time taken by given optimization procedure.

Problem	NSGA-II Time (hrs)	Local search Time (hrs)
Single-objective	5.81	8.65
Two-objective	7.23	Solution 1: 14.71 Solution 2: 16.18 Solution 3: 9.43 Solution 4: 6.34 Solution 5: 24.21 Solution 6: 32.45

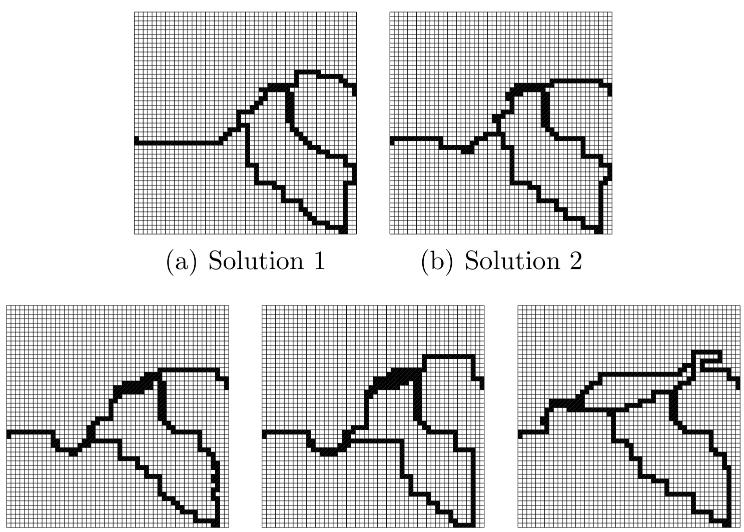
and output regions. Solution 6 shows topologically different design from others. It has two closed-loops of material.

Values of applied boundary and support conditions evolved by the customized NSGA-II are given in Table 5. It can be seen that the location of support position is same for the reference design and other local search solutions. However, different loading positions are evolved which influence the magnitude of input displacement. It is interesting to note that the magnitude of input displacement increases when the loading region is positioned away from the origin. This information can be useful for designing CM in order to restrict the magnitude of input displacement boundary condition.



(a) NSGA-II solutions (b) Local search solutions

Figure 17. Bi-objective solutions for straight line path tracing CMs



(a) Solution 1 (b) Solution 2

(c) Solution 3 (d) Solution 5 (e) Solution 6

Figure 18. Non-dominated lightweight compliant mechanisms tracing straight path.

Table 5. Evolved applied boundary and support conditions for CMs tracing straight line path.

Study → Conditions ↓	Single-objective (Ref. Design)	Bi-objective (Solutions 1 to 6)
Support position (mm) (*from the origin*)	48	48
Loading position (mm) (*from the origin*)	40	20
Input displacement (mm)	8	4

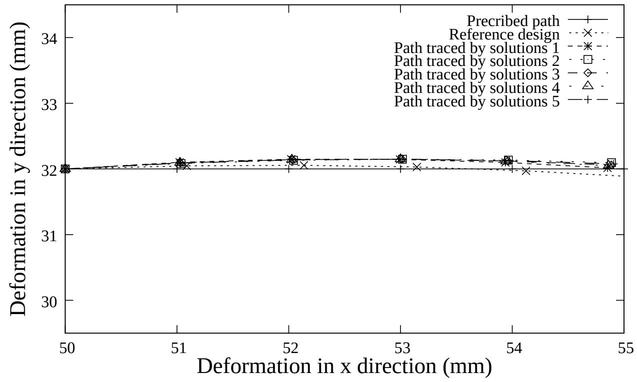


Figure 19. Prescribed path and path traced by all local search solutions.

Table 6. Deviation at precision points.

Precision points (PP)	1	2	3	4	5
Maximum allowed $\mathbf{d}_1$	0.15	0.15	0.15	0.15	0.15
Reference design					
$\mathbf{d}_2$	0.1006	0.1473	0.1496	0.1267	0.1350
Two-objective study					
Solution 1: $\mathbf{d}_2$	0.0918	0.1385	0.1489	0.1398	0.1485
Solution 2: $\mathbf{d}_2$	0.1039	0.1497	0.1460	0.1179	0.1487
Solution 3: $\mathbf{d}_2$	0.0939	0.1421	0.1499	0.1303	0.1207
Solution 4: $\mathbf{d}_2$	0.0982	0.1457	0.1500	0.1274	0.1287
Solution 5: $\mathbf{d}_2$	0.0974	0.1443	0.1499	0.1350	0.1492
Solution 6: $\mathbf{d}_2$	0.1002	0.1472	0.1498	0.1302	0.1473

The prescribed straight line path and those traced by the local search solutions are shown in Fig. 19. It can be seen that the continuum elastic structures cannot trace a straight line path exactly. It is because the categorization of the given design domain (cf. Fig. 1) favors curvilinear paths. But, imposing constraints on the precision points guide the customized NSGA-II to evolve feasible straight line path tracing CMs. Numerical value of d_1 and d_2 are given in Table 6. It can be seen that d_2 value of all local search solutions first increases till the precision point 3 and then, it decreases at precision point 4. At precision point 5, d_2 value is again increased. For solutions 1, 2, 5, and 6, the precision points 3 and 5 are active thereby making these solutions critical. The straight line path in this example is designed to deform 10.00% in $x-$direction and 0.0% in $y-$direction with respect to the size of design domain (Sharma *et al.* 2011).

5.3. *Discussion*

A minimum-weight CM tracing a user-defined path can be generated using a single-objective optimization procedure, as described in Section 3. In this paper, the concept of multi-objectivization is exploited where the helper objective is coupled with the primary objective of weight minimization. The helper objective is evaluated with respect to the reference design in order to preserve geometrically dissimilar structures. As the reference design itself is generated from SOOP, this reference point methodology sets a desired goal or aspiration level for each objective. The non-dominated solutions from the bi-objective study are then used as starting points for the local search method. The results on two examples show that lesser weight structures are evolved than the single-objective reference design solution. Hence, it can be concluded that the proposed reference based multi-objectivization approach is a viable approach for solving complex real-world problems where the optimum solution is difficult to obtain.

Although a bi-objective approach is used in the proposed procedure, the overall focus is still to obtain a minimum-weight solution. Therefore, the obtained solutions at the end are similar, despite having some differences in their detailed shape. Importantly, the use of two conflicting objectives was helpful in maintaining a diverse set of solutions in intermediate populations so that a better minimum-weight solution can be found. A local search from these solutions are then able to find structures that are much lighter than the single-objective solution. Due to the application of the local search procedure from a diverse set of solutions, the overall procedure is able to generate slightly dissimilar solutions thereby finding near-optimal solutions.

Imposing constraints on the precision points has allowed CM designs to be feasible in terms of tracing a path within the allowable proximity to the specified path. From the results, the constraints are found to be active at some of the precision points. These constraints are found to be more useful for the second example problem, where the elastic structures are required to generate a straight line path, instead of a curvilinear paths. If an optimization procedure is not guided by these constraints, then a wide difference between a user-defined path and the actual obtained path is expected.

Apart from the above facts, it is interesting to observe that CMs tracing a curvilinear path are supported on the left side of the design domain. However, for the straight line path generation, a support at the right side of the design domain is obtained. This aspect is intuitive for the given design domains, but it was not explicitly specified in the optimization process and the proposed methodology is able to evolve these fundamental features.

6. Conclusions

In this paper, the notion of helper objective has been used to exploit the "multi-objectivization" concept and has been coupled with the primary objective for evolving minimum-weight path tracing compliant mechanisms. The helper objective has not only enabled in maintaining geometrically dissimilar structures in the population, but also allowed an automatic way to set the reference point in the optimization run. The local search method has exploited NSGA-II solutions generated from the bi-objective optimization to generate multiple lighter weight solutions than the single-objective reference design solution. Moreover, instead of one, the proposed approach is able to find multiple near-optimal solutions, thereby providing the user a flexibility in choosing a single preferred minimum-weight solution. The use of multiple parallel processors has also helped in completing the overall optimization runs tractable. Various other observations discussed in Section 5.3 have been made from this study. Further, other forms of helper objectives and other NSGA-II operators can be investigated to obtain better and faster optimization procedure.

References

Abbass, H.A. and Deb, K., 2003. Searching Under Multi-Evolutionary Pressures. *In*: *Proceedings of the 2nd international conference on Evolutionary multi-criterion optimization*, EMO'03, Faro, Portugal. Berlin, Heidelberg: Springer-Verlag, 391–404.

Bendsøe, M.P., 1989. Optimal Shape Design as a Material Distribution Problem. *Structural and Multidisciplinary Optimization*, 1 (4), 193–202.

Bleuler, S., *et al.*, 2001. Multiobjective Genetic Programming: Reducing Bloat using SPEA2. *In*: *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, Vol. 1 IEEE, 536–543.

Chapman, C.D., Saitou, K., and Jakiela, M.J., 1994. Genetic Algorithms as an Approach to Configuration and Topology Design. *ASME, Journal of Mechanical Design*, 116, 1005–1012.

Coello, C.A.C., 2000. Treating Constraints as Objectives for Single-Objective Evolutionary Optimization. *Engineering Optimization*, 32 (3), 275–308.

De Jong, E.D., Watson, R.A., and Pollack, J.B., 2001. Reducing Bloat and Promoting Diversity using Multi-Objective Methods. *In*: L. Spector, E.D. Goodman, A. Wu, W.B. Langdon, H.M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M.H. Garzon and E. Burke, eds. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, 7-11 July San Francisco, California, USA: Morgan Kaufmann, 11–18.

Deb, K. and Chaudhuri, S., 2005. Automated Discovery of Innovative Designs of Mechanical Components Using Evolutionary Multi-Objective Algorithms. *In*: N. Nedjah and L. de Macedo M, eds. *Evolutionary Machine Design: Methodology and Applications*. New York: Nova Science Publishers, Inc, chap. 6, 143–168.

Deb, K., *et al.*, 2002. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *Evolutionary Compuation, IEEE Transactions on*, 6 (2), 182 –197.

Deb, K., *et al.*, 2006. Reference Point based Multi-Objective Optimization using Evolutionary Algorithms. *International Journal of Computational Intelligence Research (IJCIR)*, 2 (6), 273–286.

Deb, K., 2001. *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley: Chichester, UK.

Frecker, M.I., *et al.*, 1997. Topological Synthesis of Compliant Mechanisms Using Multi-Criteria Optimization. *ASME, Journal of Mechanical Design*, 119, 238–245.

Greiner, D., *et al.*, 2007. Improving Computational Mechanics Optimum Design Using Helper Objectives: An Application in Frame Bar Structures. *In*: S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu and T. Murata, eds. *Evolutionary Multi-Criterion Optimization.*, Vol. 4403 of *Lecture Notes in Computer Science* Springer Berlin / Heidelberg, 575–589.

Hassani, B. and Hinton, E., 1998. *Homogenization and Structural Topology Optimization: Theory, Practice and Software*. Verlag, UK: Springer.

Howell, L.L., 2001. *Compliant Mechanisms*. Wiley: New York.

Jensen, M.T., 2004. Helper-Objectives: Using Multi-Objective Evolutionary Algorithms for Single-Objective Optimisation. *Journal of Mathematical Modelling and Algorithms*, 3, 323–347 10.1023/B:JMMA.0000049378.57591.c6.

Knowles, J.D., Watson, R.A., and Corne, D., 2001. Reducing Local Optima in Single-Objective Problems by Multi-Objectivization. *In*: *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, EMO '01 London, UK: Springer-Verlag, 269–283.

Koza, J.R., 1992. *Genetic Programming : On the Programming of Computers by Means of Natural Selection*. MIT Press: Cambridge, MA.

Larsen, U.D., Sigmund, O., and Bouwstra, S., 1997. Design and Fabrication of Compliant Micromechanisms and Structures With Negative Poisson's Ratio. *Journal of Microelectromechanical Systems*, 6 (2), 99–106.

Lochtefeld, D.F. and Ciarallo, F.W., 2011. Helper-Objective Optimization Strategies for the Job-Shop Scheduling Problem. *Applied Soft Computing*, 11 (6), 4161 – 4174.

Lu, K.J. and Kota, S., 2006. Topology and Dimensional Synthesis of Compliant Mechanisms Using Discrete Optimization. *ASME, Journal of Mechanical Design*, 128, 1080–1091.

Nishiwaki, S., *et al.*, 1998. Topology Optimization of Compliant Mechanisms Using the Homogenization Method. *International Journal for Numerical Methods in Engineering*, 42, 535–559.

Rai, A.K., Saxena, A., and Mankame, N.D., 2007. Synthesis Of Path Generation Compliant Mechanisms Using Initially Curved Frame Elements. *ASME, Journal of Mechanical Design*, 129, 1056–1063.

Rozvany, G.I.N., 2001. Aims, Scope, Methods, History and Unified Terminology of Computer-Aided Topology Optimization in Structural Mechanics. *Structural and Multidisciplinary Optimization*, 21 (2), 90–108.

Sharma, D., Deb, K., and Kishore, N., 2008. Towards Generating Diverse Topologies of Path Tracing Compliant Mechanisms using a Local Search Based Multi-Objective Genetic Algorithm Procedure. *In*: *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on*, june IEEE, 2004 –2011.

Sharma, D., Deb, K., and Kishore, N.N., 2006. Evolving Path Generation Compliant Mechanisms (PGCM) Using Local-Search Based Multi-Objective Genetic Algorithm. *In*: *International Conference on Trends in Product Life Cycle, Modeling, Simulation and Synthesis (PLMSS)*, December, 227–238.

Sharma, D., Deb, K., and Kishore, N.N., 2011. Domain-Specific Initial Population Strategy for Compliant Mechanisms using Customized Genetic Algorithm. *Structural and Multidisciplinary Optimization*, 43, 541–554.

Sigmund, O., 1997. On the Design of Compliant Mechanisms Using Topology Optimization. *Mechanics Based Design of Structures and Machines*, 25 (4), 493–524.

Tai, K., Cui, G.Y., and Ray, T., 2002. Design Synthesis of Path Generating Compliant Mechanisms by Evolutionary Optimization of Topology and Shape. *ASME, Journal of Mechanical Design*, 124, 492–500.

Toffolo, A. and Benini, E., 2003. Genetic Diversity as an Objective in Multi-Objective Evolutionary Algorithms. *Evolutionary Computation*, 11, 151–167.

Wang, M.Y., Wang, X., and Guo, D., 2003. A Level Set Method for Structural Topology Optimization. *Computer Methods in Applied Mechanics and Engineering*, 192 (1–2), 227–246.

Wang, S.Y., Tai, K., and Wang, M.Y., 2006. An Enhanced Genetic Algorithm for Structural Topology Optimization. *International Journal for Numerical Methods in Engineering*, 65 (1), 18–44.

Watanabe, S. and Sakakibara, K., 2005. Multi-Objective Approaches in a Single-Objective Optimization Environment. *In*: *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, Vol. 2, sept. IEEE, 1714 – 1721 Vol. 2.

Xie, Y.M. and Steven, G.P., 1993. A Simple Evolutionary Procedure for Structural Optimization. *Computers & Structures*, 49 (5), 885–896.

Yang, R.J. and Chuang, C.H., 1994. Optimal Topology Design Using Linear Programming. *Computers & Structures*, 52 (2), 265–275.