Appendix A

Techniques for solving the problem

A highly accurate numerical methods, i.e., the hybrid spline difference method (HSDM) is used to solve the direct problem. On the other hand, a customized form of the highly popular and widely applied multi-objective GA, the *Non-dominated Sorting Genetic Algorithm-II* (NSGA-II) [23] is considered for solving the optimization problem.

A.1 Numerical methods for solving the direct problem

A.1.1 Origin of parametric spline

Since in most of the studies [86], the spline function is considered to be a cubic polynomial, its curvature after the second differential is assumed to be a linear relationship as given by Eq. (A.1).

$$\frac{\bar{\theta}''_n(\bar{\xi}) - \bar{\theta}''_{n-1}}{\bar{\xi} - \bar{\xi}_{n-1}} = \frac{\bar{\theta}''_n - \bar{\theta}''_n(\bar{\xi})}{\bar{\xi}_n - \bar{\xi}} \tag{A.1}$$

In Eq. (A.1), $\bar{\theta}_{n-1}(\bar{\xi})$ and $\bar{\theta}_n(\bar{\xi})$ are the cubic spline approximation curves in the interval $[\bar{\xi}_{n-2}, \bar{\xi}_{n-1}]$ and $[\bar{\xi}_{n-1}, \bar{\xi}_n]$ respectively, where $\bar{\xi}$ is the dimensionless radius. $\bar{\theta}''_{n-1}$ and $\bar{\theta}''_n$ are the second order derivatives of $\bar{\theta}_{n-1}$ and $\bar{\theta}_n$ on the computational grid point $\bar{\xi}_{n-1}$ and $\bar{\xi}_n$, respectively.

To increase the accuracy of the values, a random undetermined parameter $\bar{\tau}$ is added in the traditional spline assuming the relationship of quadratic differential to be as given by Eq. (A.2)

$$\bar{\theta}_n''(\bar{\xi},\bar{\lambda}) + \bar{\tau}\bar{\theta}_n(\bar{\xi},\bar{\lambda}) = [\bar{\theta}_n''(\bar{\xi}_{n-1},\bar{\lambda}) + \bar{\tau}\bar{\theta}_n(\bar{\xi}_{n-1},\bar{\lambda})]\left(\frac{\bar{\xi}_n-\bar{\xi}}{\Delta\bar{\xi}_n}\right) + [\bar{\theta}_n''(\bar{\xi}_n,\bar{\lambda}) + \bar{\tau}\bar{\theta}_n(\bar{\xi}_n,\bar{\lambda})]\left(\frac{\bar{\xi}-\bar{\xi}_{n-1}}{\Delta\bar{\xi}_n}\right) \;;\quad \text{for}\ \ \bar{\xi}\in[\bar{\xi}_{n-1},\bar{\xi}_n] \tag{A.2}$$

In Eq. (A.2), $\bar{\xi} \in [\bar{\xi}_{n-1}, \bar{\xi}_n], \bar{\xi}_n$ is the discrete grid points in computation space $[\bar{\xi}_0, \bar{\xi}_N]$, $\bar{\lambda} \geq 0$ is a free parameter, the spacing $\Delta\bar{\xi}$ is defined as $\bar{\xi}_n - \bar{\xi}_{n-1}, \bar{\theta}_n(\bar{\xi},\bar{\lambda})$ is the unknown function defined in $[\bar{\xi}_0, \bar{\xi}_{N+1}]$. The end point relation of Eq. (A.2) is given by Eq. (A.3).

$$\bar{\theta}_n\left(\bar{\xi}_{n-1}\right) = \bar{\theta}_{n-1} \tag{A.3a}$$

$$\bar{\theta}_n\left(\bar{\xi}_n\right) = \bar{\theta}_n \tag{A.3b}$$

Solving Eq. (A.2) with the help of Eq. (A.3), Eq. (A.4) is obtained.

$$\bar{\theta}_n(\bar{\xi},\bar{\lambda}) = \bar{z}\,\bar{\theta}_n + \bar{\bar{z}}\,\bar{\theta}_{n-1} + \Delta\bar{\xi}_n^2\left[\bar{g}(\bar{z})\bar{\theta}_n'' + \bar{g}(\bar{\bar{z}})\bar{\theta}_{n-1}''\right]/\bar{\omega}^2 \tag{A.4}$$

where,

$$\begin{aligned} \bar{\omega} &= \Delta\bar{\xi}/\sqrt{\bar{\lambda}} \\ \bar{z} &= \left(\bar{\xi}-\bar{\xi}_n\right)\Delta\bar{\xi} \\ \bar{\bar{z}} &= 1-\bar{z} \\ \bar{g}(\bar{z}) &= \bar{z} - sin(\bar{\omega}\bar{z})/sin\bar{\omega} \\ \bar{\theta}_n'' &= \bar{\theta}_n''\left(\bar{\xi}_n\right) \end{aligned}$$

Here, $\bar{\theta}_n''$ and $\bar{\theta}_n''\left(\bar{\xi}_n\right)$ are the second differential values at the end points.

In Eq. (A.4), the subscript n is replaced by $n+1$ to obtain the relationship of $\bar{\theta}_{n+1}(\bar{\xi},\ \bar{\lambda})$. Since there is a continuity of the first and second derivatives at cross-point, the fundamental relation of the parameter spline function given by Eq. (A.5) can be deduced from the function $\bar{\theta}_n(\bar{\xi},\ \bar{\lambda})$ and $\bar{\theta}_{n+1}(\bar{\xi},\ \bar{\lambda})$.

$$\bar{\theta}_n' = \begin{cases} \frac{\bar{\theta}_{n+1}-\bar{\theta}_n}{\Delta\bar{\xi}} - \Delta\bar{\xi}\left(\bar{\beta}\bar{\theta}_n'' + \bar{\alpha}\bar{\theta}_{n+1}''\right) \;; & n=0 \\ \frac{\bar{\theta}_{n+1}-\bar{\theta}_{n-1}}{2\Delta\bar{\xi}} - \frac{\bar{\alpha}\Delta\bar{\xi}\left(\bar{\theta}_{n+1}''-\bar{\theta}_{n-1}''\right)}{2} \;; & n=1,2,\cdots,N-1 \\ \frac{\bar{\theta}_n-\bar{\theta}_{n-1}}{\Delta\bar{\xi}} + \Delta\bar{\xi}\left(\bar{\alpha}\bar{\theta}_{n-1}'' + \bar{\beta}\bar{\theta}_n''\right) \;; & n=N \end{cases} \tag{A.5a}$$

$$\bar{\alpha}\bar{\theta}'_{n-1} + 2\bar{\beta}\bar{\theta}'_n + \bar{\alpha}\bar{\theta}'_{n+1} = \frac{\bar{\alpha}+\bar{\beta}}{\Delta\bar{\xi}}\left(\bar{\theta}_{n+1} + \bar{\theta}_{n-1}\right) \tag{A.5b}$$

$$\bar{\alpha}\bar{\theta}''_{n-1} + 2\bar{\beta}\bar{\theta}''_n + \bar{\alpha}\bar{\theta}''_{n+1} = \frac{1}{\Delta\bar{\xi}^2}\left(\bar{\theta}_{n+1} - 2\bar{\theta}_n + \bar{\theta}_{n-1}\right) \tag{A.5c}$$

$$\text{where,} \quad \bar{\alpha} = \left[\bar{\omega}csc(\bar{\omega}) - 1\right]/\bar{\omega}^2, \quad \bar{\beta} = \left[1 - \bar{\omega}cot(\bar{\omega})\right]/\bar{\omega}^2. \tag{A.5d}$$

A.1.2 The concept of spline difference

The approximate function of the differential equation is given by Eq. (A.6) assuming it to be composed of multiple different parameter spline $\bar{\Omega}(\bar{\xi}, \bar{\lambda})$

$$\bar{\theta}(\bar{\xi}, \bar{\lambda}) = \sum_{n=-1}^{N+1} p_n \bar{\Omega}\left(\frac{\bar{\xi} - \bar{\xi}_n}{\Delta\bar{\xi}}, \bar{\lambda}\right) \tag{A.6}$$

In Eq. (A.6), p_n is the value of the spline size at the grid point n. The discrete relationship at the grid point given by Eq. (A.7) is obtained by substituting Eq. (A.6) in Eq. (A.5).

$$\bar{\theta}_n = \bar{\alpha}p_{n-1} + 2\bar{\beta}p_n + \bar{\alpha}p_{n+1} \tag{A.7a}$$

$$\bar{\theta}'_n = \frac{p_{n+1} - p_{n-1}}{2\Delta\bar{\xi}} \tag{A.7b}$$

$$\bar{\theta}''_n = \frac{p_{n-1} - 2p_n + p_{n+1}}{\Delta\bar{\xi}^2} \tag{A.7c}$$

A.1.3 Concept of hybrid spline

For practical equation solving, the evaluation of the parameter $\{\bar{\alpha}, \bar{\beta}\}$ are difficult. In order to reach the accuracy of fourth order for the first and the second derivative of the approximate function at the same time, the concept of hybrid spline is used and the discrete relationship is defined as given by Eq. (A.8).

$$\bar{\theta}_n = \frac{p_{n-1} + 10p_n + p_{n+1}}{12} \tag{A.8a}$$

$$\bar{\theta}'_n = \frac{p_{n+1} - p_{n-1}}{2\Delta\bar{\xi}} - \Delta\bar{\theta}'_n \tag{A.8b}$$

$$\bar{\theta}''_n = \frac{p_{n-1} - 2p_n + p_{n+1}}{\Delta\bar{\xi}^2} \tag{A.8c}$$

$$\text{where,} \quad \Delta\bar{\theta}'_n = \begin{cases} \frac{\left(-3\bar{\theta}''_n + 4\bar{\theta}''_{n+1} - \bar{\theta}''_{n+2}\right)\Delta\bar{\xi}}{24} \; ; & n = 0 \\ \frac{\left(\bar{\theta}''_{n+1} - \bar{\theta}''_{n-1}\right)\Delta\bar{\xi}}{24} \; ; & n = 1, 2, \cdots, N-1 \\ \frac{\left(3\bar{\theta}''_n - 4\bar{\theta}''_{n-1} + \bar{\theta}''_{n-2}\right)\Delta\bar{\xi}}{24} \; ; & n = N \end{cases} \tag{A.8d}$$

The concept of hybrid spline is as follows: using the parameters $(\bar{\alpha}, \bar{\beta}) \rightarrow (1/12, 5/12)$ and cooperating with Eqs. (A.7a) and (A.7c), Eqs. (A.8a) and (A.8c) are obtained. Eqs. (A.8a) and (A.8c) are substituted in Eq. (A.5a) and considering $(\bar{\alpha}, \bar{\beta}) \rightarrow \left(\frac{1}{6}, \frac{1}{3}\right)$, Eqs. (A.8b) and (A.8d) are obtained through reorganization.

In accordance with the problem discussed in this thesis, Eq. (A.8) is modified as given in Eq. (A.9), where Δr, $\left(N^{(i)} + 1\right)$, n and p represent the grid size, total number of grid points, grid index of space, and spline parameter, respectively.

$$\bar{\theta}_n = \frac{p^{(i)}_{n-1} + 10p^{(i)}_n + p^{(i)}_{n+1}}{12} \tag{A.9a}$$

$$\bar{\theta}'_n = \frac{p^{(i)}_{n+1} - p^{(i)}_{n-1}}{2\Delta r} - \Delta\bar{\theta}'_n \tag{A.9b}$$

$$\bar{\theta}''_n = \frac{p^{(i)}_{n-1} - 2p^{(i)}_n + p^{(i)}_{n+1}}{\Delta r^2} \tag{A.9c}$$

$$\text{where,} \quad \Delta\bar{\theta}'_n = \begin{cases} \frac{\left(-3\bar{\theta}''_n + 4\bar{\theta}''_{n+1} - \bar{\theta}''_{n+2}\right)\Delta r}{24} \; ; & n = 0 \\ \frac{\left(\bar{\theta}''_{n+1} - \bar{\theta}''_{n-1}\right)\Delta r}{24} \; ; & n = 1, 2, \cdots, N^{(i)}-1 \\ \frac{\left(3\bar{\theta}''_n - 4\bar{\theta}''_{n-1} + \bar{\theta}''_{n-2}\right)\Delta r}{24} \; ; & n = N^{(i)} \end{cases} \tag{A.9d}$$

$$i \in \{1, 2\} \; .$$

In Eq. (A.9), $p^{(i)} = p$ and $N^{(i)} = N$ in uniform and variable thickness fins, while i in $p^{(i)}$ and $N^{(i)}$ indicates the steps in the fin with step change in thickness ($i = 1$ in the first step and $i = 2$ in the second step). Further, $\bar{\theta} = \omega$ for uniform and variable thickness fins while $\bar{\theta} = \kappa$ for step fin.

A.2 NSGA-II for solving the optimization problem

An individual or a solution representation which represent a complete solution of a problem is the basic component of an EA which is usually an array of some elements. A single element or a sub-array of elements constitutes design variable or variables of the problem. For coding the design variables in a solution, there are four techniques which are {0,1} binary-coded, real-coded, integer-coded, and permutation representation. For binary-coded representation, a sub-array of elements is used to represent a variable, the size of whose depends on the desired accuracy of the variable. For the other three representations, a single element is used to represents a variable, the only difference being the element be allowed to have a real value, an integer value, and a unique integer value, respectively. In any of the above mentioned four representation, a population is formed by a set of solutions, which by applying some algorithm-specific operators, is gradually improved toward the optima. A termination criteria is then applied to obtained the solutions which is usually the desired objective values are obtained or a predefined maximum number of generations (iterations) are performed [21].

Each of the EA differing from each other in the specific mechanisms through which a population is evolved toward the optima. Genetic algorithm (GA) is one of the search technique, wherein by repeated application of three major operators, namely selection, crossover, and mutation operators, a population is evolved toward the optima. A mating pool with above-average solutions of the GA population is formed by the selection operator, with the help of the solutions of the mating pool, the offspring (children solutions) are generated by the crossover operator whereas the neighborhood of an offspring are explored by the mutation operator. The literature is available with a good number of variants of GA. A very popular and widely applied multi-objective GA, the *Non-dominated Sorting Genetic Algorithm-II* (NSGA-II) proposed by Deb et al. [23] is considered in the present work. The following subsections is dedicated to address the different steps of NSGA II, customized for solving the present problem at hand.

A.2.1 Solution representation and population initialization

In the formulation, all the design variables being the real valued variables, they are represented by two real-coded elements of the solution array. After the total number of elements in the solution array is determined, each of them is initialized by a random real number in the range specified in Chapters 3 and 4.

A.2.2 Solution comparison in multi-objective optimization

In a multi-objective optimization, two solutions are compared through the concept of dominance. The solution z_i is said to be dominated by solution z_j if and only if z_j is not worse than z_i in any objective value and z_j is strictly better than z_i in at least one objective value. If there is a violation of any of these two conditions the solutions are called non-dominated solutions.

However, irrespective of their objective values, the presence of constraints makes an infeasible solution always dominated by any feasible solution. On the other hand, between two infeasible solutions, the solution with lesser total amount of constraint violation will dominate the other solution. However, if their total amounts of constraint violation are equal, the solutions are non-dominated with respect to one another.

Once the dominance relations between each pair of solutions is obtained, the sorting of the population is done according to the non-domination levels of the solutions. The set of the best non-dominated solutions is called the first non-dominated front. This front is also known as the Pareto front. The non-dominated fronts of the subsequent solutions are identified from the rest of the population. This is done by excluding the solutions of the preceding non-dominated fronts (in a non-dominated front, the characteristic of the solutions are such that the value of an objective function cannot be improved until unless the value of at least one of the other objective functions is degraded). Finally, in accordance to their levels of non-domination, the solutions are ranked. As for example, each of the solutions of the first front has a rank of 1, the second front has a rank of 2, and so on [22].

In addition to the non-dominated ranking, maintaining the diversity among the solutions of a non-dominated front is another primary requirement in multi-objective optimization. The present study adopts the crowding distance-based diversity measure, proposed by Deb et al. [23]. By applying Eq. (A.10) to the perimeters of the cuboid formed by the nearest neighbors of a solution, crowding distance-based diversity measure can be estimated.

$$d_i^{(\mathcal{I})} = \sum_{j=1}^{q} \frac{\bar{f}_j^{\left(\mathcal{I}_{i+1}^{(j)}\right)} - \bar{f}_j^{\left(\mathcal{I}_{i-1}^{(j)}\right)}}{\bar{f}_j^{\max} - \bar{f}_j^{\min}} \tag{A.10}$$

In Eq. (A.10),

$\mathcal{I}$ = index of the non-dominated front
$\mathcal{I}_i^{(j)}$ = ith solution of $\mathcal{I}$ in the direction of the jth objective function
$\bar{f}_j^{(\mathcal{I}_i^{(j)})}$ = jth objective value of $\mathcal{I}_i^{(j)}$
$d_i^{(\mathcal{I})}$ = crowding distance (diversity) of the ith solution of $\mathcal{I}$
$\left(\bar{f}_j^{\min}, \bar{f}_j^{\max}\right)$ = range of the jth objective function in the entire population
q = number of objective functions in the problem

If z_i has a better (smaller) non-dominated rank than that of z_j, or both have the same rank but z_i has a better (higher) crowding distance than that of z_j, then solution z_i can be said to be better than solution z_j.

A.2.3 Working principle of NSGA-II

The binary tournament selection operator, a crossover operator, a mutation operator and an elite preserving mechanism are the major operators used in NSGA-II.

At the first instance, two random solutions from the GA population are picked at a time by the binary tournament selection operator out of which a copy of the best one, obtained using the measures addressed in Section A.2.2, is stored in a temporary population. The process is repeated until the size of the temporary population, known as a mating pool equals to that of the GA population.

Once the mating pool is formed, the crossover operator is applied, which select two random parent solutions at a time from the mating pool and generates two children solutions. This is done by crossing the two selected parent solutions with a predefined crossover probability ($\bar{p}_c$). In case of real-coded variables, the simulated binary crossover (SBX) operator is applied as given by Eq. (A.11).

$$\left.\begin{aligned} \bar{x}_j^{(c_1)} &= \tfrac{1}{2}\left[\left(1+\bar{\bar{\beta}}\right)\bar{x}_j^{(\bar{p}_1)} + \left(1-\bar{\bar{\beta}}\right)\bar{x}_j^{(\bar{p}_2)}\right] \quad ; \quad j = 1, 2, \cdots, \bar{n}_r \\ \text{and} \quad \bar{x}_j^{(c_2)} &= \tfrac{1}{2}\left[\left(1-\bar{\bar{\beta}}\right)\bar{x}_j^{(\bar{p}_1)} + \left(1+\bar{\bar{\beta}}\right)\bar{x}_j^{(\bar{p}_2)}\right] \quad ; \quad j = 1, 2, \cdots, \bar{n}_r \end{aligned}\right\} \tag{A.11}$$

In Eq. (A.11),

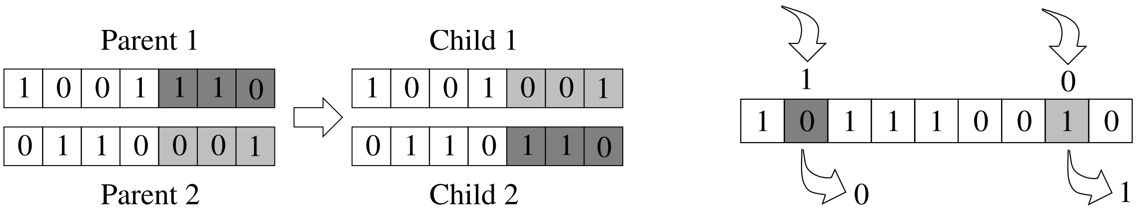
$\bar{x}_j$ = j-th real-coded variable
$\bar{n}_r$ = number of real-coded variables
p = parents
c = children

$\bar{\bar{\beta}}$ = ordinate of a probability distribution

The ordinate of a probability distribution $\bar{\bar{\beta}}$ for a random number $u \in (0,1)$ and a non-negative probability distribution index n is expressed by Eq. (A.12).

$$\bar{\bar{\beta}} = \begin{cases} (2u)^{\frac{1}{1+n}} \; ; & \text{if } u \leqslant 0.5 \\ \left(\frac{1}{2(1-u)}\right)^{\frac{1}{1+n}} \; ; & \text{otherwise.} \end{cases} \tag{A.12}$$

A single-point crossover operator is applied in case of {0,1} binary-coded variables where by crossing two parents two children are generated as follows: As shown in Figure A.1(a), the portions of the parents on the right side of a randomly chosen crossing site are exchanged.



(a) Single-point crossover operator. (b) Mutation operator.

Figure A.1: GA operators for {0,1} binary-coded variables.

Once a children population is generated, the neighborhood of a child solution is explored with the application of a mutation operator. For a random number $r \in (0,1)$ and a polynomial distribution index $\bar{\eta} > 0$, the polynomial mutation operator is applied to the real-coded variables to evolve real-valued variable $\bar{x}_j$ in the range of $\left(\bar{x}_j^{\min}, \bar{x}_j^{\max}\right)$ as given by Eq. (A.13).

$$\bar{x}_j \leftarrow \bar{x}_j + \left(\bar{x}_j^{\max} - \bar{x}_j^{\min}\right) d_q \tag{A.13}$$

where,

$$d_q = \begin{cases} \left[2r + (1-2r)\left(1 - \frac{\bar{x}_j - \bar{x}_j^{\min}}{\bar{x}_j^{\max} - \bar{x}_j^{\min}}\right)^{\bar{\eta}+1}\right]^{\frac{1}{\bar{\eta}+1}} - 1 \; ; & \text{if } r < 0.5 \\ 1 - \left[2(1-r) + 2(r-0.5)\left(1 - \frac{\bar{x}_j^{\max} - \bar{x}_j}{\bar{x}_j^{\max} - \bar{x}_j^{\min}}\right)^{\bar{\eta}+1}\right]^{\frac{1}{\bar{\eta}+1}} \; ; & \text{otherwise.} \end{cases} \tag{A.14}$$

The binary mutation operator, on the other hand, is applied to {0,1} binary-coded variables for altering the elemental values from 0 to 1 or from 1 to 0 as shown in Figure A.1(b).

An important point to be noted here that because of the stochastic nature of the GA, there is no assurance that the children population, generated by the application of the crossover and mutation operators, as expressed above, would be superior than the parent population. Under such a situation, simply considering the children population of a generation as the parent population for next generation may sometime lead the search in the opposite direction to the optima. A convergence and diversity based elite preserving mechanism is applied in NSGA-II to avoid such a possibility and thus retaining the best solutions of a generation. Applying this mechanism, at first both the parent and children populations of a generation are combined upon which, based on their non-dominated ranks and crowding distances as stated in Section A.2.2, sorting is done. By selecting the first 50% of the best solutions of this combined population, finally the parent population for the next generation is formed. Even if no good solution is generated at a generation, this mechanism guarantees that from the current position, NSGA-II never moves opposite to the optima.

A.2.4 Working steps of NSGA-II in the context of the present study

The working steps (NSGA-II) in the context of the present study is summarized below:

(1) Form a GA population with $\bar{N}$ number of solutions, where a solution is a real-valued array representing the design variable vector $\boldsymbol{x} \equiv \left(\bar{d}_1, \bar{d}_2, \bar{d}_3 \ldots\ldots\right)^{\mathrm{T}}$ where $\bar{d}_1, \bar{d}_2, \bar{d}_3 \ldots\ldots$ are the design variables.

(2) Initialize the solutions of the GA population with design variable values generated randomly within their user-specified ranges.

(3) Evaluate the objective functions for the GA population.

(4) Employ the binary tournament selection operator to the GA population to form a mating pool of size $\bar{N}$.

(5) Apply the simulated binary crossover operator to generate a children population of size $\bar{N}$ by exploiting the mating pool.

(6) Apply the polynomial mutation operator to explore the neighborhood of the children population.

(7) Evaluate the objective functions for the children population repeating Step (3).

(8) Form the population for the next generation by applying the elite-preserving mechanism to the original GA population and the children population.

(9) Repeat Steps (4)–(8) until the user-specified maximum number of generations is performed.

Appendix B

Publications

B.1 List of publications

(i) Journal publications

(a) Deka, A. and Datta, D. Multi-objective optimization of annular fin array subject to thermal load. Accepted for publication in *Journal of Thermophysics and Heat Transfer*, subject to minor revisions.

(b) Deka, A. and Datta, D. Multi-objective optimization of annular fin array with B-spline curve based fin profiles. *Journal of Thermal Stresses*, 41:247-261, 2018.

(c) Deka, A. and Datta, D. Geometric size optimization of annular step fin using multi-objective genetic algorithm. *Journal of Thermal Science and Engineering Applications*, 9:0210131–0210139, 2017.

(d) Deka, A. and Datta, D. B-spline curve based optimum profile of annular fins using multi-objective genetic algorithm. *Journal of Thermal Stresses*, 40:733-746, 2017.

(ii) Conference publication

(a) Deka, A. and Datta, D. A comparative investigation of annular fins of different profiles using multi-objective genetic algorithm. In IEEE *International Conference on Advances in Mechanical, Industrial, Automation and Management Systems* (AMIAMS-2017), pages 74–79, MNNIT Allahabad, India, 2017.

(iii) Book chapter

(a) Deka, A. and Datta, D. Geometric size optimization of annular step fin array for heat transfer by natural convection. Accepted for publication in *Metaheuristic Optimization Methods: Algorithms and Engineering Applications*, Springer.