CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF MEMS TECHNOLOGY AND ITS APPLICATIONS

MEMS (Micro Electro Mechanical Systems) refer to the technology, which integrates electrical and mechanical components with feature size of 1 to 1000 microns. Due to its small size, low cost, low power consumption and high efficiency, MEMS technology has been widely used in many fields. MEMS technologies have a diverse range of applications in bio-engineering, automotive engineering, telecommunications, environmental monitoring and space exploration [1]. Micro mechanisms are also named MEMS (Micro Electro Mechanical Systems) [2].

MEMS are a processing technology used to create tiny integrated devices or systems that combine mechanical and electrical components. They are fabricated using integrated circuit (IC) batch processing techniques and can range in size from a few micrometers to a millimeter [3]. The reduction in size and power usage of MEMS devices has enabled development of fully implantable medical devices [4]. These devices (or systems) have the ability to sense, control and actuate on a micro scale, and generate effects on macro scale. If semiconductor micro fabrication was seen to be the first micro manufacturing revolution, MEMS would be the second revolution [5].

MEMS range from simple beams and electrostatic gaps to more complex sensors and actuators that include fluidic, magnetic and thermal systems. Modern methodology of MEMS design implies that the entire MEMS can be investigated only at higher abstraction levels such as schematic and system ones, where accurate
macro models can be used [6]. On the other hand, at the component or device levels the physical behavior of three-dimensional continua is described by partial differential equations (PDE) solvable by Finite Element or Finite Difference Element Methods (FEM or FDM)[3]. Micro-Electro-Mechanical Systems consists of mechanical elements, sensors, actuators and electrical and electronics devices on a common silicon substrate [7]. The sensors in MEMS gather information from the environment through measuring mechanical, thermal, biological, chemical, optical, and magnetic phenomena. The electronics then process the information derived from the sensors and through some decision making capability, direct the actuators to respond by moving, positioning, regulating, pumping and filtering, thereby controlling the environment for some desired outcome or purpose [3].

The advantages of semiconductor IC manufacturing such as low cost, mass production, reliability are also integral to MEMS devices. The size of MEMS sub-components is in the range of 1 to 100 micrometers and the size of MEMS device itself measures in the range of 20 micrometers to a millimeter [8]. These have been used as sensors for pressure, temperature, mass flow, velocity, sound and chemical composition, as actuators for linear and angular motions and as simple components for complex systems such as robots, lab-on-a-chip, micro heat engines and micro heat pumps[6]. The Lab on-a-chip in particular is promising to automate biology and chemistry. To some extent the integrated circuit has allowed large-scale automation of computation.

Accelerometers for automobile airbags, keyless entry systems, dense arrays of micro mirrors for high definition optical displays, scanning electron microscope tips to image single atoms, micro heat exchangers for cooling of electronic circuits, reactors for separating biological cells, blood analyzers and pressure sensors for catheter tips are but a few of the current usages. Micro ducts are used in infrared detectors, diode lasers, miniature gas chromatographs and high-frequency fluidic control systems. Micro pumps are used for ink jet printing, environmental testing and electronic cooling. Potential medical applications for small pumps include controlled delivery and monitoring of minute amount of medication, manufacturing of nanoliters of chemicals, and development of artificial
pancreas [5]. Commercial tools like MEMCAD (Microcosm Technologies) [7] and MEMS Modeler (MEMSCAP) use parametric curve-fitting of simulation data to obtain macro models [9]. The primary drawback of these methods is that they do not generate scalable macro models.

However, the greatest potential for MEMS devices lies in new applications within telecommunications (optical and wireless), biomedical and process control areas. Military use of MEMS such as triggers for weapons, micro-gyros, micro-surety systems, and micro-navigation devices give another dimension to the importance of reliability of these devices [8]. Any accidental triggering may claim many lives and, if in a warehouse, may have a domino effect. Initial air bag technology used conventional mechanical “ball and tube” type devices which were relatively complex, weighed several pounds and cost several hundred dollars. They were usually mounted in the front of the vehicle with separate electronics near the airbag. MEMS have enabled the same function to be accomplished by integrating an accelerometer and the electronics into a single silicon chip. Another example of an extremely successful MEMS application is the miniature disposable pressure sensor used to monitor blood pressure in hospitals. These sensors connect to a patient’s intravenous (IV) line and monitor the blood pressure through the IV solution. For a fraction of their cost ($10), the hospitals have replaced the early external blood pressure sensors with MEMS. The early ones cost over $600 and had to be sterilized and recalibrated for reuse [3].

MEMS have several distinct advantages as a manufacturing technology [6]. In the first place, the interdisciplinary nature of MEMS technology and its micromachining techniques, as well as its diversity of applications has resulted in an unprecedented range of devices and synergies across previously unrelated fields (for example biology and microelectronics). Secondly, MEMS with its batch fabrication techniques enables components and devices to be manufactured with increased performance and reliability, combined with the obvious advantages of reduced physical size, volume, weight and cost. Thirdly, MEMS provides the basis for the manufacture of products that cannot be made by other methods. These factors make MEMS potentially a far more pervasive technology than integrated circuit
microchips. However, there are many challenges and technological obstacles associated with miniaturization of MEMS that need to be addressed and overcome before it can realize its overwhelming potential. MEMS is a manufacturing technology; a paradigm for designing and creating complex mechanical devices and systems as well as their integrated electronics using batch fabrication techniques.

A number of actuators operate by thermal actuation that imposes relatively high temperatures and require resistance against thermal cycling, high temperature fatigue or creep. All these issues necessitate rigorous mechanical tests on MEMS scale, to examine the effect of various processing parameters, types of loading, service environments and temperature for different materials [5]. Components such as micro mirrors that have several levels of alignment controls operating at high frequencies suffer from cyclic fatigue accumulation and may fail by crack initiation and propagation under cyclic loading [3].

MEMS accelerometers help in sensing the acceleration experienced by a system. MEMS accelerometers find frequent utilization in airbag deployment systems in automobiles. Here, negative acceleration of the vehicle is being sensed by these accelerometers. A processor will examine the magnitude of acceleration and determines whether the airbags in the vehicle are to be deployed or not. The conventional accelerometers for airbag deployment systems in automobiles are being replaced by MEMS accelerometers in a rapid manner [10].

The growing popularity of MEMS accelerometers is mainly due to their small size, light weight and increased reliability. In addition, the price involved in manufacturing the MEMS accelerometer are found to be only a fraction of the cost involved in constructing the conventional massive accelerometers. Numerous novel micromachining approaches are made to work in a joint fashion for achieving a commercially available accelerometer used in low g (gravity acceleration) applications [10].

MEMS accelerometers are employed in a wider range of applications like automobile airbag systems. Other familiar applications include consumer
products, like, computer games, cell phones, pagers, PDAs, advanced robotics, laptop computers, computer input devices, camcorders, digital cameras and SD card accessories. Size and accuracy are the most essential features that decide the performance of the sensor in each of the above-mentioned applications [11].

In recent years, CMOS micromachining has evolved as a chief fabrication technology for VLSI MEMS. CMOS micromachining is utilized for fabricating the design and characterization of a lateral accelerometer. More attention is being paid towards integrated Microsystems, such as inertial measurement unit. Integrated Microsystems yield improved performance with an array of similar topology devices, like accelerometers and gyroscopes, which possess dissimilar performance specifications [12].

1.2 INTRODUCTION TO OPTIMIZATION

“But ask the animals, and they will teach you, or the birds of the air, and they will tell you; or speak to the earth, and it will teach you, or let the fish of the sea inform you”.

-Job 12:7 (adapted from [179])

Optimization is the procedure or procedures used to make a system or design, as effective or functional as possible, especially the mathematical techniques involved; making the best of anything. It is a mathematical technique used for finding a maximum or minimum value of a function for several variables subject to a set of constraints, as linear programming or system analysis.

1.3 TYPES OF OPTIMIZATION

The types of optimization techniques are Unconstrained optimization, Constrained optimization, Multi-objective optimization, Multi modal optimization, Combinatorial optimization, Hill climbing, Intelligence.
1.3.1 Unconstrained Optimization

Unconstrained optimization problem is one where you only have to be concerned about the objective function you are trying to optimize. None of the variables in the objective function are constrained.

1.3.2 Constrained Optimization

Constrained optimization is the process of optimizing an objective function with respect to some variables in the presence of constraints. Constrained optimization is the minimization of an objective function subjected to constraints on the possible values of the independent variable. Constraints can be either equality constraints or inequality constraints.

1.3.3 Multi-Objective Optimization

Multiobjective optimization also known as multi-objective programming, vector optimization, multi criteria optimization, multi attribute optimization or pareto optimization is an area of multiple criteria decision making that is concerned with mathematical optimization problems involving more than one objective function to be optimized simultaneously. It has been applied in many fields of science, including engineering, economics and logistics where optimal decisions need to be taken in the presence of trade-offs between two or more conflicting objectives.

1.3.4 Multimodal Optimization

Multimodal optimization problem is a problem that has more than one local minimum or multimodal optimization deals that with optimization takes, that involve finding all or most of the multiple solutions (as opposed to a single best solution).
1.3.5 Combinatorial Optimization

Combinatorial optimization is a branch of optimization in applied mathematics and computer science, related to operations research, algorithm theory and computational complexity theory. There are many optimization problems for which the independent variables are restricted to a set of discrete values. These types of problems are called combinatorial optimization problems.

1.3.6 Hill Climbing

Hill climbing is a graph search algorithm where the current path is extended with a successor node which is closer to the solution than the end of the current path. In simple hill climbing, the first closer node is chosen whereas in steepest ascent hill climbing all successors are compared and the closest to the solution is chosen.

1.3.7 Intelligence

Intelligence has been defined in many different ways such as in terms of one’s capacity for logical, abstract thought, understanding, self-awareness, communication, learning, emotional knowledge, memory, planning, creativity and problem solving. Some of its characteristics are adaptation, randomness, communication, feedback, exploration, and exploitation.

1.4 INTRODUCTION TO CLASSIC EVOLUTIONARY ALGORITHMS

There are various optimization algorithms that can produce an optimized design of MEMS. The Table 1.1 gives the details of various classic evolutionary algorithms developed for optimization.
### Table 1.1 A List of Evolutionary Algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Author</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Swarm Intelligence Based Algorithms</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Artificial Bee Colony (2005)</td>
<td>Karaboga and Basturk [16]</td>
<td></td>
</tr>
<tr>
<td>Artificial Fish Swarm (2003)</td>
<td>Li et al. [23]</td>
<td></td>
</tr>
<tr>
<td>Firefly Algorithm (2008)</td>
<td>Yang [28]</td>
<td></td>
</tr>
<tr>
<td>Particle Swarm Optimization (1995)</td>
<td>Kennedy and Eberhart [42]</td>
<td></td>
</tr>
<tr>
<td><strong>Bio-inspired(not SI–based) Algorithms</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Biogeography-Based Optimization (2006)</td>
<td>Simon [58]</td>
<td></td>
</tr>
<tr>
<td>Group Search Optimizer (2009)</td>
<td>He et al. [69]</td>
<td></td>
</tr>
<tr>
<td>Shuffled Frog Leaping Algorithm (2003)</td>
<td>Ensuff and Lansey [71]</td>
<td></td>
</tr>
<tr>
<td><strong>Physics and Chemistry Based Algorithms</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gravitational Search (2009)</td>
<td>Rashedi et al. [79]</td>
<td></td>
</tr>
<tr>
<td>Harmony Search (2001)</td>
<td>Geem et al. [97]</td>
<td></td>
</tr>
<tr>
<td>Simulated Annealing (1983)</td>
<td>Kirkpatrick et al. [111]</td>
<td></td>
</tr>
<tr>
<td><strong>Other Algorithms</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Opposition Based Learning (2005)</td>
<td>Tizhoosh [124]</td>
<td></td>
</tr>
<tr>
<td>Tabu Search (1986)</td>
<td>Glover and Mcmillan [130]</td>
<td></td>
</tr>
<tr>
<td>Teaching Learning Based Optimization (2011)</td>
<td>Rao [134]</td>
<td></td>
</tr>
</tbody>
</table>
1.4.1 Ant Colony Optimization (ACO)

Go to the ant, you sluggard; consider its ways, and be wise!

-Proverbs 6:6 (adapted from [179])

ACO initially proposed by Marco Dorigo in 1992 in his Ph.D thesis [13]. Ant colony optimization is a way to solve optimization problems based on the way ants indirectly communicate directions to each other. Ant colony optimization is a probabilistic technique for solving computational problems which can be reduced to find good paths through graphs. Each ant tries to find a route between its nest and a good food source [14].

The behavior of each ant in nature:

(i) Wander randomly at first, laying down a pheromone trial.
(ii) If food is found, return to the nest laying down pheromone trail.
(iii) If pheromone is found, with some increased probability follow the pheromone trial.
(iv) Once back at the nest, go out again in search of food.

However pheromones evaporate over time, such that unless they are reinforced by ants, the pheromones will disappear. Applications of ACO includes scheduling problems (project scheduling, job shop scheduling, open shop, agent based dynamic scheduling), routing problems (TSP, vehicle routing, connection oriented and connection less network routing), assignment problems (quadratic assignment problems, course timetabling, graph coloring), sequential ordering problem, shortest common super sequence problem, constraint satisfaction, classification rules, bayesian networks, protein folding, protein-ligand docking, set problem and device sizing problem in nanoelectronics physical design, machine learning, dynamic problem of data network routing, a shortest path problem where properties of the system such as node availability vary over time, continuous
optimization and parallel processing implementations, digital image processing and classification problem in data mining[15].

1.4.2 Artificial Bee Colony (ABC) Algorithm

ABC algorithm is a swarm based meta-heuristic algorithm that was introduced by karaboga in the year 2005 for optimizing numerical problems [16]. It was inspired by the intelligent foraging behavior of honey bees. The model consists of three essential components: employed and unemployed foraging bees, and food sources. The first two components, employed and unemployed foraging bees search for rich food source, which is the third component, closer to their hive. The model also defines two leading modes of behavior which are necessary for self-organizing and collective intelligence: requirement of foragers to rich food sources resulting in positive feedback and abandonment of poor food sources by foragers causing negative feedback. ABC is developed based on inspecting the behaviors of real bees in finding nectar and sharing the information of food sources to the bees in the hive. ABC algorithm is used to solve unconstrained and constrained optimization problems, multidimensional and multimodal optimization problems [17,18]. Application of ABC includes decoder-encoder and 3-bit parity benchmark problems [19], clustering [20], scheduling problems, resource-constrained project scheduling problem [21], image segmentation, capacitated vehicle routing problem, WSNs, assembly line balancing problem, solving reliability redundancy allocation problem, training neural networks, XOR, pattern classification, p-center problem [22].

1.4.3 Artificial Fish Swarm Algorithm (AFSA)

AFSA is a novel method to search global optimum, which is typically an application of behaviorism in artificial intelligence. AFSA method is one of the swarm intelligence approaches that works based on the population and stochastic search. Fishes show very intelligent and social behavior. This algorithm is one of the best approaches of the swarm intelligence method with considerable advantages like high convergence speed, flexibility, error tolerance and high accuracy. Basic idea of AFSA is to imitate fish behavior such as preying, swarming and following with
local search of fish individual for reaching the global optimum. It is a random and parallel search algorithm [23]. Application of AFSA includes automated design, multi robot task scheduling, UCAV path planning, fault diagnosis in mine hoist, optimum steel making charge plan, target area on simulation robots, function optimization, parameter estimation, combinatorial optimization, least squares support vector machine and geo technical engineering problems [24].

1.4.4 Bacterial Foraging Optimization

BFOA was introduced by Passino in the year 2002. Bacterial foraging optimization algorithm (BFOA) has been widely accepted as a global optimization algorithm of current interest for distributed optimization and control. BFOA is a non-gradient, bio inspired self organizing natural and newly developed efficient optimization technique. In BFOA, the social foraging behavior of Escherichia coli, commonly known as E.coli is mimicked. Application of group foraging strategy of E.Coli bacteria in multi-optimal function optimization is the key idea of the algorithm. Here the natural selection tends to eliminate the animals with poor foraging strategies and favor those having successful foraging strategies. The foraging strategies of E.Coli can be explained by Chemotaxis, Swarming, Reproduction, Elimination and Dispersal [25, 26].

Applications of BFOA includes global optimization, adaptive control, harmonic estimation, optimum power system stabilizers, optimal power flow, optimization over continuous surfaces, PID controller tuning, active power filter for load optimization, optimizing power loss and voltage stability limits, fuzzy controller construction/tuning, neural network training, job shop scheduling, electromagnetics, stock market prediction, motor control, system identification, temperature control, energy efficiency optimization for buildings and distributed energy generation, highly nonlinear and nonconvex problem, inverse airfoil design, transmission loss reduction, implemented as the parameter estimation of nonlinear system model(NSM) for heavy oil thermal cracking, evaluation of independent components to work with mixed signals, solve constrained economic load dispatch
problems, application in the null steering of linear antenna array by controlling the element amplitudes, applications in multi objective optimization [24,25 and 27].

1.4.5 The Firefly Algorithm (FA)

The firefly algorithm was introduced in the year 2007 by Xin-She Yang at Cambridge University [28]. It is based on the attraction of fireflies to one another. Attraction is based on the perceived brightness of a firefly, which exponentially decreases with distance. A firefly is attracted only to those fireflies that are brighter than it. Firefly can be described by using following idealized rules

(i) All fireflies are unisex so that one firefly will be attracted to other fireflies regardless of their sex.

(ii) Attractiveness is proportional to their brightness, thus for any two flashing fireflies, the less bright one will move towards the brighter one. If there is no brighter one than a particular firefly it will move randomly. The brightness of the firefly is affected or determined by the landscape of the objective function. For a maximization problem, the brightness can simply be proportional to the value of the objective function.

Applications of FA includes least computation time for digital image compression [29,30], feature selection [31], efficiently solve highly nonlinear, multimodal design problems [32,33], antenna design optimization [34], efficiently solve NP-hard scheduling problems[35], scheduling and travelling and salesman problem [36-38], clustering [39,40], to train neural networks [41], wireless network design, dynamic market pricing, mobile robotics, image segmentation, real time complex image analysis problems, real time systems and heartbeat synchronization [24].
1.4.6 Particle Swarm Optimization (PSO)

“The particle swarm algorithm imitates human social behavior”.

- James Kennedy and Russell Eberhart [42]

PSO is a population based on stochastic optimization algorithm to find a solution and solve an optimization problem in a search space. It is a simple, computationally efficient optimization method. It is based on a social-psychological model of social influence and social learning.

Inspired by the flocking and schooling patterns of birds and fish, Particle Swarm Optimization (PSO) was invented by Russell Eberhart and James Kennedy in 1995. Originally, these two started out developing computer software simulations of birds flocking around food sources, and then later realized how well their algorithms worked on optimization problems.

Particle Swarm Optimization might sound complicated, but it's really a very simple algorithm. Over a number of iterations, a group of variables have their values adjusted closer to the member whose value is closest to the target at any given moment. Imagine a flock of birds circling over an area where they can smell a hidden source of food. The one who is closest to the food chirps the loudest and the other birds swing around in his direction. If any of the other circling birds comes closer to the target than the first, it chirps louder and the others veer over toward him. This tightening pattern continues until one of the birds happens upon the food. It's an algorithm that's simple and easy to implement [43-46].

PSO has been successfully applied in many areas: function optimization, artificial neural network training, fuzzy system control, and other areas where GA can be applied. The various application areas of PSO include power systems operations and controlling-Hard combinatorial problems, job scheduling problems, vehicle routing problems, mobile networking, modeling optimized parameters, batch process scheduling, multiobjective optimization problems, image processing and
pattern recognition problems, multimodal biomedical image registration and the
iterated prisoner’s dilemma, classification of instances in multiclass databases,
feature selection, web service composition course composition, power system
optimization problems, edge detection in noisy images, finding optimal machining
parameter assembly line balancing problem in production and operations
management, anomaly detection, color image segmentation, sequential ordering
problem, constrained portfolio optimization problem, selective particle regeneration
for data clustering, machinery fault detection, unit commitment computation, and
signature verification [24, 47- 49].

1.4.7 Genetic Algorithm (GA)

“GAs are NOT function optimizers”

- Kenneth De Jong (adapted from [179])

Genetic Algorithms in particular became popular through the work of
John Holland in early 1970s [50]. Genetic Algorithms are inspired by Darwin’s
theory about evolution. Solution to a problem by Genetic Algorithms evolved.
Genetic Algorithms and evolutionary strategies mimic the principle of natural
genetics and natural selection to construct and search optimization procedures. GA
belong to the larger class of EA which generate solutions to optimization problems
using technique inspired by natural evolution, such as inheritance, mutation,
selection and crossover.

Algorithm is started with a set of solutions (represented by
chromosomes) called population. Solutions from one population are taken and used
to form new population. This is motivated by a hope, that the new population will be
better than the old one. Solutions which are selected to form new solutions
(offspring) are selected according to their fitness. The more suitable they are, the
more chances they have to reproduce. This is repeated until some condition is
satisfied [51, 52].
GA can even be faster in finding global maxima than conventional methods, in particular when derivatives provide misleading information. The enormous potential of GA lies elsewhere in optimization of non-differentiable or even discontinuous functions. The Table 1.2 gives the details of various applications of Genetic Algorithms.

**Table 1.2 Applications of Genetic Algorithm**

<table>
<thead>
<tr>
<th>S.No</th>
<th>Domain</th>
<th>Application types</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Control</td>
<td>Gas pipeline, pole balancing, missile evasion, pursuit.</td>
<td>[53-57]</td>
</tr>
<tr>
<td>2</td>
<td>Design</td>
<td>Semiconductor layout, aircraft design, keyboard configuration, communication networks.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Scheduling</td>
<td>Manufacturing, facility scheduling, resource allocation.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Robotics</td>
<td>Trajectory planning</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Machine Learning</td>
<td>Designing neural networks, improving classification algorithms, classifier systems.</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>others</td>
<td>Mechanical sector, electrical engineering, civil engineering, image processing, data mining, wireless networks, VLSI, production planning, air traffic problems, automobile, signal processing, communication networks, environmental engineering, bioinformatics, phylogenetics, computational science, economics, chemistry, manufacturing, mathematics, physics, pharmacometrics and other fields.</td>
<td></td>
</tr>
</tbody>
</table>
1.4.8 Biogeography Based Optimization (BBO)

BBO was introduced by Simon in the year 2006. BBO is based on the science and study of species movement from one habitat to another. BBO is an evolutionary algorithm that optimizes a function by stochastically and iteratively improving candidate solutions with regard to a given measure of quality or Fitness function (F). BBO belongs to the class of metaheuristics since it includes many variations, and since it does not make any assumptions about the problem and can therefore be applied to a wide class of problems [58]. It is modeled after the emigration and immigration of species between habitats’s to achieve information sharing. The two key concepts involved in the optimization are HSI and SIV. If a habitat with high HSI consists of a large number of species and habitat with low HSI tend to have low number of species. Features that correlate with HSI include rain fall, vegetative diversity, topographic diversity, land area, temperature, and others. Here SIV is an independent variable and HSI is the dependent variable. Species will immigrate to, and emigrate from, a habitat with a probability that is determined by the HSI. A habitat with high HSI will tend to have low immigration rate and high emigrate rate. If a habitat with low HSI will tend to have high immigration rate and low emigration rate. Nature will optimize the number of species living in each habitat to achieve equilibrium. The Figure 1.1 represents BBO Migration Curves.

![BBO Migration Curves](image)

**Figure 1.1** BBO Migration Curves (Adapted from [179])
In the Figure 1.1 the immigration rate $\lambda$ and the emigration rate $\mu$ are the functions of the number of species in a habitat. Here each habitat is a candidate solution to an optimization problem, each species is an independent variable of that candidate solution.

In BBO, each candidate solution shares its features with other candidate solutions, and this sharing process is analogous to migration in biography. If migration occurs for many iterations, the habitat becomes more suitable for their species, which corresponds to candidate solutions providing increasingly better solution to an optimization problem.

Application of BBO includes Economic Load Dispatch (ELD) problem [59] with generator constraints in power plants. BBO has also solved real-world application problems such as Block based motion estimation in video coding [60], Implementing color image segmentation [61], color image quantization [62], Face recognition [63], Feature selection [64], ECG signal classification, power system optimization, ground water detection, and satellite image classification [65], general benchmark functions, constrained optimization, the sensor selection problem for aircraft engine health estimation, web based BBO graphical user interface, global numerical optimization, and optimal meter placement for security constrained state estimation [24].

1.4.9 Differential Evolution (DE)

Compared to several existing EAs, DE is much simpler and straightforward to implement ……Simplicity of programming is important for practitioners from other fields, since they may not be experts in programming ….

-S. Das, P. Suganthan, and C. Coello Coello (adapted from [179])

Global optimization is necessary in the field of engineering, statistics and finance. But many practical problems have objective functions that are non-differentiable, non-continuous, non-linear, noisy, flat, multi dimensional or have
18

many local minima, constraints or stochasticity. Such problems are difficult if not impossible to solve analytically. DE can be used to find approximate solution to such problems. DE was developed by Rainer Stron and Kenneth V. Price around 1995 [66]. It is a stochastic, population-based optimization algorithm developed to optimize real parameter, real valued functions. It is a method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. Such methods are commonly known as metaheuristics as they make few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions.

An application of DE includes optimal operation of multipurpose reservoir [67], design of digital filter [68], optimization of strategies for checkers, maximization of profit in a model of a beef property, optimization of fermentation of alcohol, unsupervised image classification, clustering, optimization of non-linear functions, global optimization of non-linear chemical engineering processes and multi-objective optimization [24].

1.4.10 Group Search Optimizer (GSO)

GSO has been proposed by S. He in the year 2009. GSO, which was inspired by animal behavior especially animal searching (foraging) behavior. Animals normally search for food in group. They get benefited sharing the information among themselves. The frame work mainly based on the producer-scrounger model, which assumes that group members search either for finding or for joining opportunities. Based on this frame work, concepts from animal searching behavior are employed metaphorically to design optimum searching strategies for solving continuous optimization problems [69]. The population of GSO is called group and each individual in the population is called a member. In the search space, each member knows its own position, its head angle and a head direction, which can be calculated from the head angle via polar to cartesian co-ordinate transformation.

A group constitutes three types of members: producers, scroungers and rangers. Producers: perform producing strategies, searching for food. Scroungers:
perform scrounging strategies, joining resources uncovered by others. Rangers: perform random walk motions and will be dispersed from their current positions.

Applications of GSO includes truss structure design [70], benchmark functions applied for optimal power flow problems, mechanical design optimization problems, multi objective optimization, optimal placement of FACT devices, machine condition monitoring, optimal location and capacity of distributed generations [24].

1.4.11 Shuffled Frog Leaping Algorithm (SFLA)

SFLA was introduced by Muzaffar Ensuff and Kevin Lansey in 2003. SFLA is a memetic meta-heuristic and a population based cooperative search metaphor inspired by natural memetics. The algorithm contains elements of local search and global information exchange. The SFLA consists of a set of interactive virtual population of frogs partitioned into different memeplexes. The virtual frogs act as hosts or carriers of memes where meme is a unit of cultural evolution. The algorithm performs simultaneously an independent local search in each memeplex. The local search is completed using a particle swarm optimization-like method adapted for discrete problems but emphasizing a local search. To ensure global exploration, the virtual frogs are periodically shuffled and reorganized into new memeplexes in a technique similar to that used in the shuffled complex evolution algorithm. In addition, to provide the opportunity for random generation of improved information, random virtual frogs are generated and substituted in the population [71]. The steps in SFLA include the following: initial population, sorting and distribution, memeplex evolution, shuffling and terminal condition.

Application of SFLA includes color image segmentation [72], solving TSP [73], fuzzy controller design [74], mobile robot path planning [75], grid task scheduling [76], combined economic emission dispatch [77], job scheduling [78], automatic recognition of speech emotion water, unit commitment problem, optimal viewpoint selection for volume rendering, multi-user detection in DS-CDMA distribution, optimal reactive power flow, a web document classification,
classification rule mining, ground water model calibration problems and multicast routing optimization [24].

1.4.12 Gravitational Search Algorithms (GSA)

GSA was introduced by E. Rashedi in the year 2009. GSA is constructed based on the law of gravity and the notion of mass interactions. The GSA algorithm uses the theory of Newtonian physics and its searcher agents are the collection of masses. In GSA, we have an isolated system of masses. Using the gravitational force every mass in the system can see the situation of other masses. The gravitational force is therefore a way of transferring information between different masses. In GSA, agents are considered as objects and their performance is measured by their masses.

All these objects attract each other by a gravity force, and this force causes a movement of all objects globally towards the objects with heavier masses. The heavy masses correspond to a good solution of the problem. The position of the agent corresponds to a solution of the problem, and its mass is determined using a Fitness function (F) [79].

By lapse of time, masses are attracted by the heaviest mass. We hope that this mass would present an optimum solution in the search space. The GSA could be considered as an isolated system of masses. It is like a small artificial world of masses obeying the Newtonian laws of gravitation and motion.

Application of GSA includes in the following fields: NN training [80], robotics [81], optical [82], bioinformatics [83], software engineering [84], networking [85], image processing [86], classification [87], clustering [88], scheduling [89], business [90], computer engineering [91], civil engineering [92], control engineering [93], mechanical engineering [94], power engineering [95], telecommunication engineering [96].
1.4.13 Harmony Search (HS)

Harmony search (HS) was introduced in 2001 by Geem and further explained by Lee [97]. HS is based on musical processes. Each musician in a choir or band sounds a note within some allowable domain. If all the notes result in good harmony, the positive experience is saved in the choir’s collective memory and the possibility of achieving continued good harmony is increased. In HS, a choir or band is analogous to a candidate problem solution, and a musician is analogous to an independent variable or candidate solution feature.

Applications of HS includes school bus routing problems[98], Sudoku puzzle [99], water distribution network design [100], satellite heat pipe design [101], structural design [102], ecological conservation [103], multiple dam operation [104], music composition [105], vehicle routing [106], university course timetabling [107], oceanic oil structuring mooring [108], hydrologic parameter calibration [109], heat exchanger design [110], six-hump back function, multimodal function, ANN, web-based parameter calibration, robotics, internet searching, visual tracking, management science, project scheduling, medical physics, bio informatics [24].

1.4.14 Simulated Annealing (SA)

“We conjecture that the analogy with thermodynamics can offer new insight in to optimization problems and can suggest efficient algorithms for solving them”.

-Valdo Cerny 1985 (adapted from [179])

Simulated annealing was independently described by Scoot Kirkpatrick, C. Daniel Gelatt and Mario P. Vecchi in 1983 [111]. Simulated annealing is an optimization algorithm that is based on the cooling and crystallizing behavior of chemical substances. Simulated annealing is a single individual stochastic algorithm. Simulated annealing mimics the cooling phenomenon of molten metal’s to constitute a search procedure. Slowly cool down a heated solid, so that all
particles arrange in the ground energy state. At each temperature wait until the solid reaches its thermal equilibrium. Probability of being in a state with energy $E$ can be represented by the Equation (1.1),

$$\Pr \{ E = E \} = \frac{1}{Z(T)} \cdot e^{(-E / K_b)T} \quad (1.1)$$

where,

- $E$ - Energy
- $T$ - Temperature
- $K_b$ - Boltzmann constant
- $Z(T)$ - Normalization factor

SA is a good solution method that is easily applicable to large number of problems. Tuning of parameter is relatively easy. Quality of results of SA is good although it take a lot of time. Results are generally not reproducible. SA can leave an optimal solution and not find it again. Proven to find the optimum under certain conditions; one of these conditions is that you must run forever.

Its applications include combinatorial optimization problem with permutation property [112], basic problems, engineering design, facilities layout, image processing and code design in information theory, structural optimization, fluid transportation systems, water distribution systems, chip floor planning, GPP (Graph Partitioning Problem), GCP (Graph Coloring Problem), NPP (Number Partitioning Problem), TSP (Travelling Salesman Problem) [113].

1.4.15 Cultural Algorithm (GA)

“Culture optimizes cognition”

- James Kennedy (adapted from [179])
Cultural algorithms (CA) are a branch of evolutionary computation where there is a knowledge component that is called the belief space in addition to the population component. It was derived from models of cultural evolution in anthropology. It provides a framework to accumulate and communicate knowledge so as to allow self-adaptation in an evolving model. Cultural algorithms can be seen as extension to a conventional Genetic Algorithm. It was introduced by Reynolds [114]. The Figure 1.2 shows the cultural algorithm frame work.

![Diagram of Cultural Algorithm Framework](figure1.png)

**Figure 1.2 Cultural Algorithm Frame Work (Adapted from [135])**

The cultural algorithm components consist of a belief space and a population space. The component interacts through a communication protocol. Applications of cultural algorithms includes various optimization problems, social...
simulation, constrained optimization in ammonia synthesis, real world application, cloud computing applications, multi objective optimization, bioinformatics, eco system modeling and virtual world, distributed computing applications [115,116].

1.4.16 Estimation of Distribution Algorithm (EDA)

“That is what learning is. You suddenly understand something you’ve understood all your life, but in a new way”.

- Doris Lessing (adapted from [179])

EDAs or PMBGAs (Probabilistic Model Building Genetic Algorithms), are stochastic optimization methods that guide the search for the optimum by building and sampling explicit probabilistic models of promising candidate solutions. Optimization is viewed as a series of incremental updates of a probabilistic model, starting with the model encoding the uniform distribution over admissible solutions and ending with the model that generates only global optima [117]. EDA is a new class of EA that does not use conventional crossover, mutation operators; instead it estimates the distribution of the selected parent population and uses a sampling step for offspring generation.

Application of EDA includes power system controller design [118], linear and combinatorial optimization [119], design of process sensor networks [120], HW/SW partition [121], graph matching problems [122] and bioinformatics [123].

1.4.17 Opposition Based Learning (OBL)

The concept of OBL was first introduced by Tizhoosh. The main idea behind OBL is the simultaneous consideration of an estimate and its corresponding opposite estimate in order to achieve a better approximation of the current candidate solution [124]. OBL has first been utilized to improve learning and back propagation in neural networks. OBL includes three different kinds of concepts namely opposite point, quasi-opposite point and quasi reflected point. OBL has been
applied to many evolutionary algorithms such as DE [125], BBO [126], PSO [127], ACO [128], and GA [129].

1.4.18 Tabu Search

Tabu search was introduced in 1986 by Glover and McMillan [130]. It is a meta-heuristic super imposed on another heuristic. The overall approach is to avoid entrainment in cycles by forbidding or penalizing moves which takes the solution, in the next iteration, to points in the solution space previously visited. The method is still actively researched, and is continuing to evolve and improve. Tabu or Taboo means forbidden, banned, or not allowed. Forbidden items, speech, or practices can be based on culture, religion, morality or politics. Tabu search is a higher level heuristic procedure for solving optimization problems, design to guide other methods to escape the trap of local optimality [131,132]. Tabu search has obtained optimal and near optimal to a wide variety of classical and practical problems in application ranging from scheduling to telecommunications and from character recognition to neural networks.

Applications of Tabu search includes employee scheduling, maximum satisfiability problems, character recognition, space planning and architectural development, telecommunications path assignment, probabilistic logic problems, NN pattern recognition, quadratic assignment problems, network topology design, computer channel balancing, TSP, graph coloring, graph partitioning, nonlinear covering, maximum stable set problems, flow shop sequencing, design, location and allocation, logic and artificial intelligence, technology, general combinational optimization, graph optimization, routing, production, inventory and investment [133].

1.4.19 Teaching-Learning-Based Optimization (TBLO)

TLBO was introduced in the year 2011 by R.V. Rao. TLBO is based on the teaching and learning process in a classroom. Teaching-Learning is an important process where every individual tries to learn something from other individuals to
improve themselves. The algorithm simulates two fundamental modes of learning. Through the teacher (Teacher phase) and interacting with other learners (Learning phase). In TLBO analogies group of students considered as population, different subjects as different design variables, result scores as fitness value of the problem and teacher is best solution [134]. The applications of TLBO includes clustering [135], multi objective optimization [136], optimal power flow [137], discrete optimization of truss structure [138], global optimization problems [139], economic dispatch problems [140], design of IIR based digital hearing aids [141], reconfiguration in radial distribution systems for loss reduction [142], optimal scheduling [143].

1.5 RESEARCH MOTIVATION

Motivation of this research work is to optimize the parameters $L_1, L_2, L_3$, $y_m$ and Fitness function (F) or Die Area (DA) values of a MEMS accelerometer using Artificial Bee Colony (ABC) algorithm with Particle Swarm Optimization algorithm methods. The main problem in MEMS is to get an optimal design. The significance of MEMS optimization relating to concert, power utilization, and consistency increases. There are various optimization algorithms that can produce an optimized design of MEMS.

Some of the optimization algorithms are methods without derivatives (e.g. Nelder-Mead-Simplex), methods using derivatives (e.g. Conjugate Gradient or Quasi-Newton). These methods have various disadvantages like, with an rising amount of parameters these surfaces turn out to be more and more problematical and it is almost not possible to compute them.

An additional universal difficulty is the typically complex relationship among structure parameters and the structure performance. Probably the most important drawback is finding a global optimum. FEM simulation is also another important method to achieve the optimized design of MEMS. Here a simplified spring-mass model is used to predict the device sensitivity. This method also will not give a efficient optimized design of MEMS. The spring constant of the beams should be further reduced by using some more complaint flexure structures (e.g.
four-fold beam) to get a better optimized method. All the drawbacks mentioned above can be overcome by means of using a Genetic Algorithm. By using Genetic Algorithm, MEMS design can be optimized in a better way. But GA having some draw backs such as representation, population size and mutation rate, selection and deletion policies, crossover and mutation operators, termination criterion. The disadvantages of GA can be overcome by means of using ABC algorithm. 

In general, the cost and the Die Area (DA) of the accelerometer are directly proportional. Thus, the cost associated with the design of accelerometer increases slowly with the increase in Die Area (DA) of the accelerometer. This behavior has lead to the need for minimizing the Die Area (DA) along the design parameter force (N) and thereby, optimization of these parameters comes into effect. The optimization algorithm is centered on objective function or Fitness function (F). The proposed method uses a combination of Artificial Bee Colony (ABC) optimization algorithm and Particle Swarm Optimization (PSO) algorithm to optimize the design parameters of MEMS accelerometer.

1.6 RESEARCH OBJECTIVES

MEMS based accelerometers used for airbag deployment in automobile industry as alternate for conventional accelerometer provide advantages in terms size, weight and cost. Therefore, the problem was identified to optimize the Die Area (DA) of MEMS based accelerometer with a range of value can be between 90000 to 160000 μm² and it can be relaxed up to 240000 μm² (adopted from [11]).

The research objectives were to:

(a) Suggest a best optimization algorithm to optimize parameters of MEMS based accelerometer.

(b) By applying the Genetic Algorithm we have obtained the optimal parameters L₁, L₂, L₃, yₘ and Fitness function (F) or Die Area (DA) of a MEMS accelerometer.
(c) In the second method we have utilized the ABC optimization algorithm for optimizing the parameters $Z_1, Z_2, Z_3, y_k$ and Fitness function (F) or Die Area (DA) of MEMS Accelerometer.

(d) The major intension of this work is to mainly focus on optimization of the design parameter like Fitness function (F) or Die Area (DA) along with a new parameter force (N). For optimization of these parameters we have incorporated two optimization algorithms like ABC and PSO. The primary optimization is done using ABC and the resultant fitness solution from the ABC is further optimized using the PSO algorithm. By combining two algorithms we can get better optimized parameters which help in efficient design of MEMS accelerometer.

(e) The optimization of parameters of a MEMS accelerometer using GA, ABC, and ABC with PSO is carried out in MATLAB 7.12 environment.

(f) Based on the three different types of optimization techniques the obtained parameter values of a MEMS accelerometer are compared and optimal parameters are reported.

1.7 LITERATURE REVIEW

The various works related to the MEMS design optimization has been presented here.

John K. Sakellaris [144] has presented the design of a vibration control mechanism for a beam with bonded piezo electric sensors, actuators and an application of the arising smart structure for vibrations suppression too. The mechanical modeling of the structure and the subsequent finite element approximation were based on Hamilton’s principle and classical engineering theory of bending of beams in connection with simplified modeling of piezoelectric sensors and actuators. Two control schemes LQR and H2 were considered.
Aniket Singh et al. [145] have presented a study of MEMS RF Power Sensors. An optimized sensor with low reflection loss parameters was identified. Two designs, one with a cantilever bridge and another with a fixed bridge were compared in terms of reflection and transmission losses. The designs were simulated with different dielectric layers and varied thickness to get a series of results. A fairly optimized design was realized with minimum reflection losses.

Xiaolin Chen et al. [146] have presented a study on multi-level simulation that proved to be an efficient way to accelerate the design process and improve the device performance. It can be used effectively to optimize the gyroscope system. The device design was automatically generated based on mask layout and fabrication process restrictions. Design verification was performed at the device-level for detailed analysis and at the system-level for behaviour characterization.

Chaitanya Chandrana et al. [147] have presented the structure for an integrated transducer that used a non-conductive epoxy for mechanical backing of the transducer and a thin film electrode for backside contact as part of the integrated process for the transducer. The desired outcome was a single integrated MEMS PVDF transducer chip, combining a high input impedance preamplifier and focused transducer. It showed an approach for building integrated PVDF transducers with minimal parasitic that could be widely used in clinical IVUS applications.

Adam Długosz [148] has presented the MOOPTIM algorithm that had been used for multiobjective shape optimization of MEMS structures. The effectiveness of MOOPTIM had been compared to NSGAII on several benchmark test problems. The obtained results showed the effectiveness of MOOPTIM for both un-constrained and constrained optimization tasks. To reduce the time of the optimization, parallel computation or approximate surrogate evaluations were used.

Rohit Pathak and Satyadhar Joshi [149] have presented a novel way to approach reliability calculations and shown how properties at different levels and types needed to be linked up in a multi scale analysis, where HPC can benefit
reliability calculations for MEMS devices. They have calculated various parameters of different scale for a MEMS device and proceeded with the analysis of reliability using MATLAB distributive computing Toolbox.

Sujata N. Naduvinamani et al. [150] have demonstrated the design of cantilever based switch using CoSolve-EM and it was observed that pull in voltage for RF MEMS switch varies for different dimensions. They have recently developed CoSolve-EM, a coupled solver for 3D quasi-static electro-mechanics. With the help of CoventorWare, the switch was designed. Initially in the process editor, required gap (i.e between the beam and the substrate) was set to some desired value. Then the 2-D layout was drawn. Then with the layout editor, the 3-D layout was drawn. Then the meshing of the structure was done. After meshing the MEM-Mech set up was done for the CoSolve-EM and existing loads were removed. It was solved for the CoSolve-EM and the mechanical deflections that takes place solely due to the electrostatic force were observed. Then different voltages were applied and the corresponding deflections towards the substrate were noted down.

Prince Nagpal et al. [151] have described the capacitive pressure sensor design for biomedical applications like blood pressure measurement. The described pressure sensors provided high sensitivity even at low pressure. This makes it suitable for biomedical applications. Effects of varying different parameters on the pressure sensor performance have been studied. From the results, the pressure sensors with compatible parameters can be selected for specific requirements. These compact pressure sensors are made up of biocompatible materials and can be implanted easily inside body to be used for RF telemetry purpose.

Shveta Jain et al. [152] have presented the Performance Study of RF MEMS Ohmic Series Switch. The effect of different geometrical parameters was studied and simulated using CoventorWare. It showed that varying the anchors length improves the contact force thereby reducing the insertion loss. Thus the trade-off between the parameters and switch performance can be enhanced by maintaining the parameters.
Zhang et al. [153] implemented a hierarchical MEMS synthesis and optimization architecture, integrating an object-oriented data structure with SUGAR and two types of optimization: Genetic Algorithms (GA) and local gradient-based refinement. They noted that the MOGA approach needed a means for automating the starting populations for MOGA that would enable a larger sampling of the solution space of MEMS design.

Jain et al. [154] developed a MEMS transducer for an ultrasonic flaw detection system. This experiment appears to be the first attempt to detect ultrasonic signal by MEMS transducers in direct contact with solids.

Attoh-Okine et al. [155] proposed the potential applications of MEMS in pavement engineering and highlighted some of the potential applications. They highlighted both the advantages and disadvantages of MEMS within pavement engineering applications. They also developed an experimental protocol for the use of MEMS resonator sensors in monitoring micro cracking in concrete.

Kamalian et al. [156] extended Zhou’s work to more advanced MEMS problems and explored interactive evolutionary computation (IEC), integrating human expertise into the synthesis loop to lever the strength of human expertise with computational efficiencies.

Li and Anton son [157] applied GAs to the mask-layout aspect of MEMS synthesis. Ma and Anton son [158] also used GAs for automated mask-layout synthesis, but extended their work to include process synthesis for MEMS. Given a desired MEMS device topology and fabrication process, their tool could produce mask layouts and associated fabrication steps for a particular MEMS device. GAs was used to evolve an optimal mask layout given a user-defined shape. Li et al. [159] also concentrated on the development of automatic fabrication process; planning for MEMS devices for the later stages of product development.

Obadat et al. [160] developed full-scale MEMS-based biaxial strain transducers for monitoring the fatigue state of railway tracks. A unique feature of
this work involves the combined use of finite element method (FEM) and MEMS. FEM analysis was used to determine the critical fatigue locations where the MEMS transducers were to be attached.

Mourad et al. [161] have designed a micro machined accelerometer that relies on area variation capacitive sensing. This can be used in many applications to enhance the efficacy and sensitivity of a capacitive accelerometer. Here, the capacitive accelerometer depends on the area of variation capacitive sensing, regarded to be a micro electro mechanical system (MEMS) that was existing and realizable. MATLAB software was used for simulation. Optimization of some accelerometer parameters and a single direction, which possesses movable fingers and fixed fingers, ensures that the system damping is carried out using MATLAB.

Hamid et al. [162] have suggested a capacitive micro machined MEMS acceleration sensor that was immune to normal-to-plane shock. When compared to the springs of the structure, the suspended cantilevers were more beneficial because it reduces the spring length for normal motion of the proof mass, after covering a certain distance. Thus, the springs were made even stronger to avoid normal movements and dangerous failures. Optimization of the consumed area, which was a significant parameter for determining the price associated with the on-chip device fabrication, was performed using Genetic Algorithm here.

M. S. Allen et al. [163] have expressed the input waveform OUU for a highly nonlinear, electrostatically actuated RF MEMS switch. The MCS, which helps in envisaging the maximum impact velocity experienced by an ensemble of switches subjected to an input waveform, utilizes a reduced-order model for the switch, that incorporates an uncertainty model based on experimental data and expert’s viewpoint. The contact velocity for the ensemble of switches can be decreased to a larger extent with the optimization of the shape of the waveform. On comparing the unshaped waveform, the overall contact velocity was minimized to 50% with the optimization in shape. The optimization steps aid in foretelling the amount of contact velocity reduction produced with the change in the design of the switch.
1.8 ORGANIZATION OF THESIS

Chapter 1 discusses the overview of MEMS technology and its applications, MEMS accelerometer sensor for airbag deployment in automobile industry. The various types of optimization techniques and classical evolutionary algorithms are reviewed. Research motivation and research objective are also discussed. Literature review of various works related to the MEMS design optimization has been described.

The parameters of MEMS accelerometer, as optimized using Genetic Algorithm are discussed in Chapter 2. The optimization of parameters of a MEMS accelerometer using Artificial Bee Colony (ABC) algorithm is described in Chapter 3. The Fitness function (F) or Die Area (DA) using a Genetic Algorithm and Artificial Bee Colony (ABC) algorithm are compared in Chapter 3.

Chapter 4 discusses the optimization of parameters of a MEMS accelerometer using Artificial Bee Colony (ABC) algorithm with Particle Swarm Optimization (PSO) algorithm method. Chapter 5 shows the comparison of optimal parameters $L_1$, $L_2$, $L_3$, $y_m$ and Fitness function (F) or Die Area (DA) values of accelerometer using GA, ABC algorithm and ABC with PSO algorithm. Finally Chapter 6 focuses on the highlights of the work done, conclusion, and suggestions for the further research.

1.9 SUMMARY

The significance of MEMS technology and its applications are discussed. Among various optimization evolutionary algorithm techniques GA, ABC, PSO are widely used, advantages and disadvantages of these optimization techniques are discussed. The motivation for optimizing the parameter values of MEMS based accelerometer and objectives of the research work are discussed. Literature reviews on various works related to the MEMS design optimization are carried out. Algorithms for optimization are identified.