CHAPTER 6

CONCLUSION AND FUTURE SCOPE OF WORK

In this thesis, Evolutionary algorithms in the form of Genetic Algorithm (GA) and its variants have been used to optimize the solutions for a class of VLSI problems ranging from test vector generation for combinational and sequential circuits, minimizing the mincut i.e. number of signal crossover between partitions for circuit partitioning problem, minimizing the bounding rectangle area for a group of modules in the floorplanning and placement and reducing the interconnect length and number of vias for a set of terminal using a 3D channel and the interconnection space.

The proposed approaches rely on the main components like efficiency, robustness and complexity reduction for the problems targeted. To meet these goals effectively, advanced search techniques, clustering and hybridization is used. The proposed work is related to the design and development of CAD tools for a class of problems ranging from Test vector generation for digital circuits, floorplanning, partitioning and circuit routing using evolutionary optimization techniques viz. Simple Genetic Algorithm (SGA), Modified Genetic Algorithm (MGA), Hybrid Genetic Algorithm (HGA). Though not separately addressed, but compaction of circuits is also addressed in addition to circuit partitioning and
Floorplanning. To verify the effectiveness of the proposed approaches the techniques are applied to a class of benchmark circuits and the results are compared with published results existing in the literature. The fault simulation is targeted with the objective of a minimum number of test vector set to detect the possible number and types of faults in combinational and sequential circuits. The circuit partitioning problem is addressed with the objective of minimizing the number of crossovers between the partitions and thus minimizing interconnects and their composite length and finally the delay in the signal path. The floorplanning problem has an objective of packing a set of modules in a given area, thus minimizing the silicon area for fabricating or manufacturing a set of modules and the interconnect length between them. Finally, the channel routing problem minimizes the length of interconnecting wire required to interconnect terminals in a 3d routing space and number of vias between two layered channel routing space.

### 6.1 CONCLUSIONS

Four important process steps of physical design are addressed in this thesis. All the processes have been evaluated using a class of evolutionary algorithms known as Genetic Algorithm (GA) and its variants.

All the design steps addressed have different optimization constraints and different modeling properties as described in the sections below.

#### 6.1.1 Fault Simulation

Fault simulation is targeted in both combinational and sequential circuits. Different types of faults such as stuck-at faults, bridge faults, and delay faults are considered. The fault simulation in combinational circuits is targeted first. *For a test circuit with 30 stuck at faults and 20 delay faults, it requires 15 iterations.* After this, a Modified Genetic Algorithm (MGA) is proposed and tested for the fault simulation problem. *The 74153 test circuit is considered and in 9 generations, it detects all the stuck at and delay faults.*

Further to this, the same technique is extended to sequential circuits having memory elements in them. There due to the dependency of present output to the present state and past output, a two-phase functional simulation is used to detect the possible faults in them.
The technique is tested on 4-bit parallel load shift register and vending machine controller. After this, MGA is used for sequential fault simulation. *In comparison with GA, MGA uses 0.73 fewer seconds for 4-bit register and 0.48 fewer seconds for vending machine controller.*

Later, the technique is applied over a class of benchmark problems, i.e. ISCAS 85 74 series of benchmark problems using Hierarchical approach. Here, instead of applying the algorithm in one run to the whole circuit, the hierarchy is expanded and only the block under test is evaluated for possible fault occurrences. The desired mapping is done using a library of blocks developed using the structural style of modeling done in HDL-like VHDL. *The technique is tested on 74 series of benchmark problems with 11 inputs and 33 gates. It detects all 154 stuck at and 52 delay faults in 35 iterations.*

### 6.1.2 Circuit Partitioning

The requirement of minimizing the number of interconnections between a set of logic is called **Circuit Partitioning or Mincut Partitioning.** It serves a dual purpose of reducing the delay and making the circuit simple to manufacture and fabricate. This objective of minimizing the number of interconnections is targeted using various approaches viz. Modified Genetic Algorithm (MGA) and Hybrid Genetic Simulated Annealing Algorithm (HGSAA). The genetic operators and local search methods evolve over the initially generated better population members and results into mincut reduction over several benchmark problems. *For a 6 input and 2 output test circuit, the technique successfully performs partitioning with mincut as 5 in 12 iterations. Using local search method, the algorithm achieves local minima in 3 iterations with respect to 22 iterations in case of without local search. Using adaptive crossover and mutation, the algorithm converges to local minima in 7 iterations as compared to 22 in the case of without adaptive crossover and mutation.** The effect of probability of mutation, i.e. $P_m$ is also studied and found that mincut as 17 is obtained with $P_m$ as 0.02. The lowest mincut i.e. 14 for variation in population size is also found with a number of population elements as 30.

To further improve the results of the GA, a Hybrid approach using GA and SA is used. It improves the best solution of GA. For a test circuit as *spp_N199_E232-R11_154*, the lowest mincut as 27 in obtained in 800 iterations with fitness as 0.0983 using GA, while using HGSAA the mincut as 13 and fitness as 0.143 is obtained in 800 iterations. In comparison
with the existing results by other techniques, it has been found that in the case of the hybrid approach the average improvement of the proposed algorithm when compared to UCLA Branch and Bound partitioner for a class of netlist problems varies from -17.556 to 9.690 while for LIFO-FM method the average improvement varies from -20.000 to 38.018. The average runtimes are better or comparable with those of compared such as Branch and Bound partitioner and LIFO-FM method. It is noticed that for smaller netlists UCLA Branch and Bound partitioner is fastest and LIFO-FM is slower, however, the proposed algorithm execution times lie in between the two techniques. The execution times of Branch and Bound and LIFO-FM reverses as the circuit becomes complex, but the proposed technique still remains in between the two.

6.1.3 Floorplanning

In this step, the VLSI circuit is seen as a set of rectangular blocks interconnected by signal nets. It has multiple objective functions such as, minimize area, minimum interconnect length, maximize routability, minimize delays, etc. The floorplanning problem is addressed using the GA and its variants. The problem is initially targeted using a netlist where the net positions are fixed on rectangular blocks and the task is to place them such that the bounding rectangle area using semi-perimeter approach reduces to a minimum in a set of iterations and the connected blocks can be considered as a composite block. As an initial effort, the size dimensions are not specified for blocks. The algorithm is tested over a test circuit with 10 cells and 10 nets, whose optimization results shows that the GA is successful in reducing the cost of placement from 648 to 514 in 0.43 seconds. For another test circuit, it produces a reduction from 480 to 442 in 0.434 seconds.

Further to this, the netlist with rectangular blocks and size dimensions in the form of width and height are considered such that the area of each block can be calculated. Various functions like series maker, merge, and the minimum effective area along with GA operators are used to optimize the placement in a given area. The crossover operation and variants of mutation are used to improve the results as compared to GA.

In a test case with 25 modules, the effective Area utilization has increased up to 48% in just 10 iterations and the contribution of dead space reduced up to 51%. The Dead space got reduced from 17900 to 2753 with the effective reduction of the placement area from 20449 to 5302. The combined Blocks Area is 2549 with a simulation time of 1.66 seconds.
The technique is then further improved using a recursive approach where rotation of blocks is also considered to decide the final orientation of blocks in the final placement. As claimed earlier, the compaction is achieved by using 2 blocks, 3 blocks, and 4 blocks based composite modules as one unit in optimization. To test its performance, the algorithm is applied to benchmark problems reported in the literature, i.e. apte, ami33, ami49 and xerox and results obtained are better than the existing ones. The dead space in the benchmark class of problems ranges from 11.43 to 0.76 in executing times as small as less than 2 Sec. The comparison results show that area utilization from 88 to 99 % can be achieved using the proposed method for the benchmark set of problems. Moreover, the execution can be further improved by omitting the time required in plotting the result and writing the details in the text files.

6.1.4 Channel Routing

The problem is targeted from the channel routing approach where the channel is a 3D rectangular area consisting of two layers and terminals are present on its opposite sides. The parameters addressed by using GA are minimum routing area, minimum Net length, and a minimum number of via. Of all the problems addressed in this research work, the encoding of chromosomes for this technique is the most challenging one as the solution here represents a completely routed solution which is not so easy for two layered 3D channel. So, unless one complete set of connections between sources and targets is performed, it cannot be considered as a solution. So, the initial population generation is taken as a separate block as it is time-consuming and a onetime process. The 3D encoding process is effective to model all the node values in two layered channel structure using its 11 valued integer based representation.

The proposed technique is different in terms of GA operators, as traditional crossover and mutation methods cannot be used here. For each set of parents selected, the Steiner points are found from each parent and unwanted routing paths are backtracked. Similarly, the mutation is not a simple alteration or exchange of values between solutions. Its two variants are implemented, one selects a Net at random and removes all its connections, while the other considers a random size rectangle in a channel and removes all connections in it. It is observed, that if plotting or display of results is skipped for each iteration, then the execution times increases considerably. For the encoding of the chromosome, an 11 valued node is used at each position of a two layered channel, thus making it as the longest encoded
chromosome in this research work. In comparison with existing solutions for Benchmark routing problems, the improvement obtained both in terms of net length and number of vias is multifold. For Joo6_16, in terms of net length, the improvement in the percentage when compared with already reported results varies from -12.06 to 0.763 and in terms of a number of vias, the improvement in percentage varies from 42.10 to 72.72. For Burnstein’s difficult channel, in terms of net length, the improvement in the percentage with already reported results varies from 2.469 to 4.81 and in terms of a number of vias, the improvement in the percentage varies from 66.66 to 80.

6.2 MAIN CONTRIBUTIONS AND RECOMMENDATIONS

The goal of doing this research is to assist in the development of possible optimization techniques for various physical design problems which have nowadays become unmanageable due to their high circuit density and their interconnections. The various EA based strategies are used for different VLSI applications. All the possible variants of EA perform well with a set of conditions applied, but most of the approaches have their domains in which they can work effectively like GA are more suitable for solving problems where the function to be determined is expressed simply in function form. The MGA algorithm of EA is similar to GA but due to their differential nature helps in evading the local minima and by varying the control parameters results in an optimized solution. As in the VLSI design process, each optimizing step has a serious effect on final design implemented and its performance also. So, the work assists in having a set of heuristic based algorithms which, when applied, helps in having better and fast solutions for the different problems faced during design and testing.

The main contributions and recommendations of the present work are as given below

- An application of various evolutionary algorithmic technique is explored for VLSI applications. Various methods like SGA, MGA, and HGA are applied on various problem successfully to solve them. The various classes of problems are fault simulation, circuit partitioning, floorplanning and placement, and circuit routing.
- The development of a class of the EA which are quite efficient and robust as compared to existing one in the literature in terms of minimum test vector generation, minimum bounding rectangle area, minimum number of mincuts, thus
the minimum signal delay, minimum length of interconnect and minimum number of vias in a 3D channel structure are the main contributions of this research work.

- Clustering or Compaction, which is a separate process in physical design is also performed, in addition, as the emphasis is given in both floorplanning and circuit partitioning to place connected blocks close to each other.
- Efficiency and robustness are provided by testing the algorithms on benchmark problems of varying sizes and on other practical circuits.

6.3 SCOPE OF FUTURE WORK

The work undertaken involves finding a software optimization algorithm, simulated in MATLAB for various classes of VLSI problems. Though sufficient amount of work has been done in this field of optimization of VLSI based problems by numerous researchers using various techniques but, still lots of scope is left to improve the existing results or introduce a new way of solving the problems.

- To increase the speed/performance, a mix of hardware and software based co-simulation technique can be used where more memory or time intensive tasks can be implemented in hardware like FPGA and the less computationally complex blocks can be implemented on software platform like MATLAB.
- Another high-level programming language like C or model-based implementation using Simulink can be explored to speed up the optimization algorithms implemented, it also helps in porting of logic on reconfigurable hardware.
- The performance of the algorithms can be improved by using hybridization of two or more algorithms such that drawbacks of one algorithm can be overpowered by the strength of another. An example of such implementation is already tried and verified using a mix of GA and SA called as HGSAA in circuit partitioning approach.
- In place of GA or its variants other EA techniques such as MGA, GP can also be tried to test their effectiveness for such class of problems.
- Different architectures such as distributed approach, parallel approach or pipelining techniques can be used to increase the throughput of the system and to reduce the overall load on the implementing hardware or software.
- Other Nature based optimization algorithms can be explored and implemented to test their performance for this class of problems.