Indian Journal of Science and Technology, Vol 8(29), DOI:10.17485/ijst/2015/v8i29/71888, November 2015
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Genetic Algorithm based Optimization of Single Node in Reformed-Digital Micro Fluidic Biochip

G. Brindha[1*], G. Rohini[2] and C. Gnanakousalya[2]

[1]Faculty of Electrical Engineering, Sathyabama University, Chennai - 600119, Tamil Nadu India; brindhag@stjosephs.ac.in
[2]Department of ECE, St. Joseph's Institute of Technology, Chennai - 600119, Tamil Nadu, India; rohini_sneha23@yahoo.co.in, vignesh_kousalya@yahoo.co.in

Abstract

Objectives: Optimization of single node / port in the Reformed Digital Micro Fluidic Biochip (RDMFB) has been carried out in this work. The Digital Micro Fluidic Biochip (DMFB) consists of two reservoirs / ports, one for sample (any biological solution) and the other for reagent (chemical solution). In each port two samples and two reagents, in the form of droplet i.e. nano / micro litre volume have been placed in respective ports. Based on application, at each instance only one solution has to be routed from reservoir to mixer and then to detector. In order to implement this, an optimization technique has been posed to decide which one has to be routed first. **Methods:** The optimization algorithm opted is of an evolutionary algorithm – Genetic Algorithm. It has been implemented for both sample port and reagent port. This algorithm has been opted due to its life science characteristic to get a better and optimal solution for any biological sample when it is being injected or diagnosed. Two samples in the form of droplet i.e. nano / micro litre volume have been used to route, both will be any biological sample which is placed in same node / port, similarly, in reagent port chemical solution has been placed. The port has to decide which one has to be routed based on application with the help of control pin used. **Findings:** The 15 x 15 array DMFB has been reformed into a new one in order to minimize the number of cells in an array. By this way of a modification, a 12 x 12 array DMFBi.e. RDMFB has been designed. The optimized output has been simulated in Xilinx platform and FPGA synthesis has been done in Plan Ahead software.**Application:** DMFB plays a vital role in our day-to-day life for various biological diagnoses in clinical laboratory. It is a lab on chip device.

Keywords: DMFB, FPGA Synthesis, Genetic Algorithm, RDMFB, Single Node, Xilinx

1. Introduction

A DMFB is a 2D array consisting of electrodes and associated peripheral devices such as optical detectors, dispensing ports, etc. The samples and reagents used in this device are in the form of droplets, which is in the order of nano/micro litre volume. These droplets are transported from source point to destination point with the help of electrodes. At each point two electrodes will be placed parallel, by activating (applying voltage) those plates the droplets will be shifted from one plate to other, such that those droplet will be transported from one point to other. The clock pulse does the activation. This technology is termed as Electro Wetting on Dielectrics (EWOD).

With this technology the droplets can be transported to any location within the array. In that routing of those droplets is one major challenge in designing those devices.

Routing of droplet has been optimized by various algorithms. Those algorithms with the published papers were discussed.

In[1] discussed about the four methods of droplet routing.

*Author for correspondence

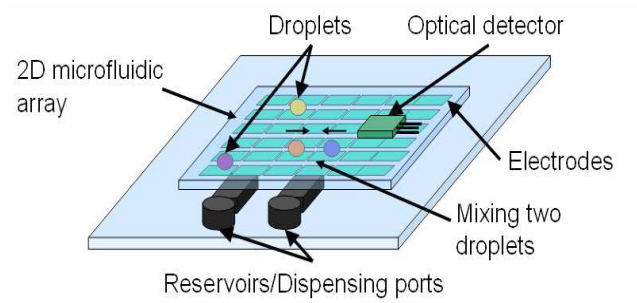


Figure 1. A Schematic view of DMFB[1].

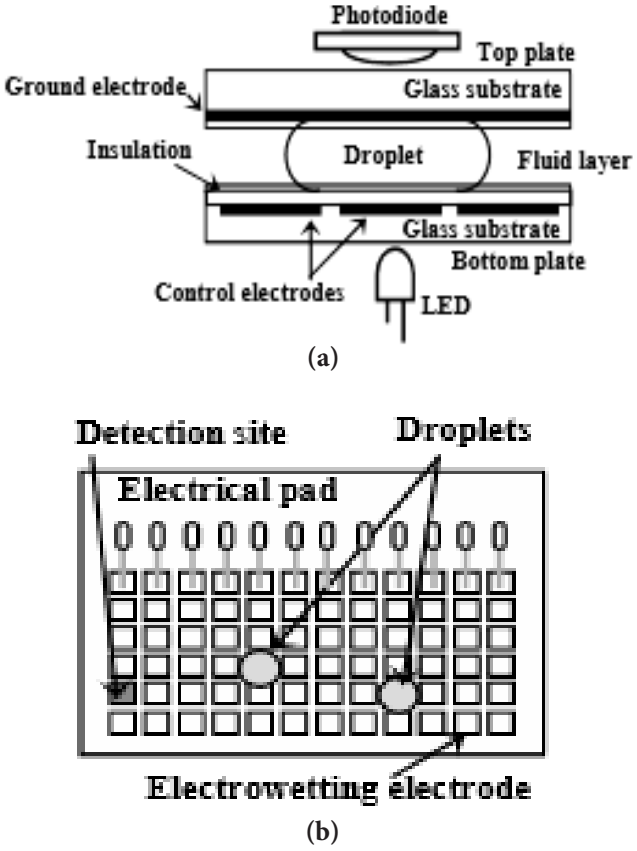


Figure 2. Arrangement of DMFB[2], **(a)** Side view, **(b)** Top view (microfluidic array with two droplets and detecting/ dispensing Port).

- First method: Prioritized A*-search based algorithm, which is highly associated with the priority order of each droplet. The priority assignment offers flexibility but lacks generality.
- Second method: Network based algorithm, which defines layout patterns of a biochip. It modeled these patterns as a network and used the OSPF (Open Shortest Path First) network protocol. Since droplet routing only occurs at these layout patterns, this algorithm failed to exploit the dynamic re-configurability.
- Third method: Two-phase routing algorithm based on maze routing algorithm, which avoids net-routing-order dependence problem and the use of dynamic re-configurability. Since maze routing algorithm needs to apply several times to generate alternative paths for each droplet, it is considered to be inefficiency.
- Fourth method: Network-flow based algorithm, which considers the dynamic re-configurability and can simultaneously perform routing and scheduling. This approach can be further modified to satisfy the different constrained routing problem.

In[2] discussed a literature survey paper on pin configuration and performance of some restricted sized DMFB. It gives the description about DMFB and its design problems with all its terminologies such as droplet creation, routing problem, mixing technique, detecting device and assay with all its constraint like fluidic, time, electrode, area and cross contamination in performing bioassay operations.

In[3] discussed a survey on automation for biochip design. Some optimization techniques like

- Integer Linear Programming (ILP).
- Branch and Bound technique (B&B).
- Divide-and-conquer algorithm.
- Dynamic Programming (DP).
- Greedy algorithms.
- A* algorithm.
- Meta-heuristic techniques are described briefly in various aspects such as bioassay analysis, resource binding and geometry based scheduling are presented. Application of optimization techniques in DMFB is also presented.

In[4] discussed about the design and optimization techniques for DMFB. A routability-driven routing algorithm has been posed to make droplets route on specific tracks orderly. Each droplet has been sequentially routed along these tracks by adopting A* maze searching technique. A Dynamic-Programming (DP) based routing compaction that transforms the sequential routing result (2D) into concurrent manner has been posed. It decodes each 2D routing path into a 1D string and then model the compaction process into a LCS-like recursion (Longest Common Subsequence) to minimize arrival time among

droplets, to achieve fast assay execution time and enhance real-time response.

In[5] discussed an optimization techniques for synchronization of concurrent fluidic operations in pin constrained DMFB. Here Two-phase optimization method has been implemented to identify and synchronize the fluidic operations that can be executed in parallel manner. A fluidic operation without pin-actuation conflict has been implemented. The duration of implementing the outcome sequence after synchronization has been minimized. The method is demonstrated for a 3-plex assay performed on a commercial pin-constrained biochip and multiplexed in-vitro diagnostics performed on an experimental biochip.

In[6] discussed a simultaneous optimization of droplet routing and control – pin mapping to electrodes in DMFB. Here the co-optimization problem involving droplet routing and pin-mapping has been formulated. An ILP - based optimization method is developed to simultaneously co-optimize the droplet pathway and pin-assignment, in order to minimize the number of control pins. An effective heuristic method has been posed to tackle the co-optimization problem. The method is demonstrated for two commercial biochips and an experimental university chip for multiplexed in-vitro diagnostics.

In[7] discussed a two stage Integer Linear Programming-based droplet routing algorithm for pin constrained DMFB. Basic ILP formulation has been opted to solve the droplet routing problem with simultaneous multi objective optimization. Due to its complexity, a two-stage technique of global routing followed by incremental ILP-based routing to reduce the solution space has been posed. A deterministic ILP formulation that casts the original routing optimization problem into a decision problem and to solve it by a binary solution search method that searches logarithmic time has been posed. A comparative analysis of Direct Addressing, Broadcast Addressing and the posed two-stage ILP Algorithm was done with the in-vitro diagnostics and the colorimetric protein assay.

In[8] discussed a paper on genetic algorithm for the routing of droplets in DMFB: Preliminary results. Here genetic operators are implemented in NSGA-II to deal with the droplet routing problem, to minimize computation time and number of used cells.

1.1 Reformed DMFB (RDMFB)

In[8], Let c = {c_1, c_2, c_3 .....$c_{n\,x\,n}$} be the set of cells of a microfluidic array of size n x n. Here an existing 15 x 15

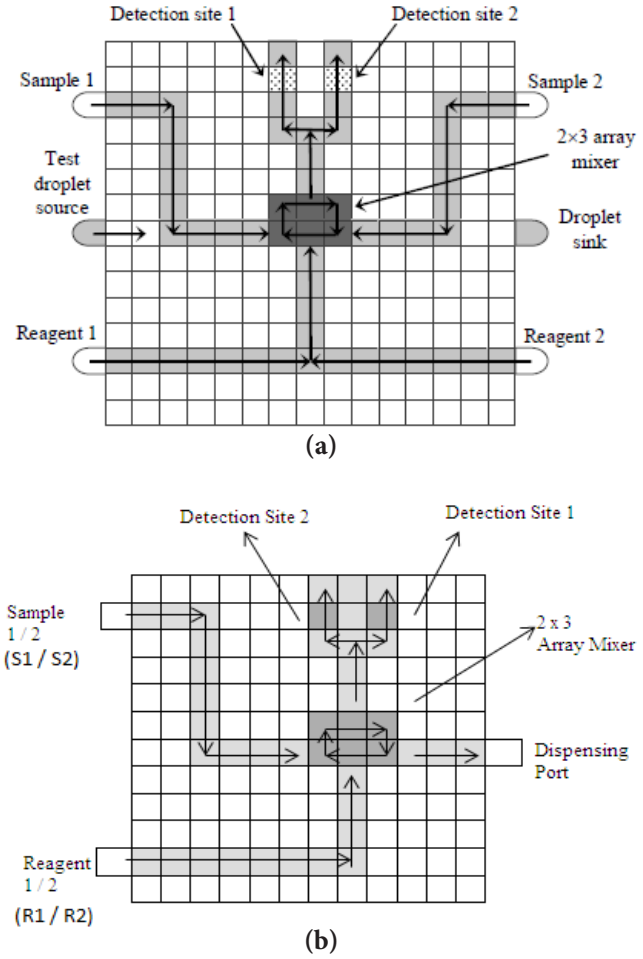


Figure 3. (a) An existing 15 x 15 array DMFB. (b) A 12 x 12 array RDMFB.

array have been chosen, which is given by Figure 3(a)[4]. Here two samples are detected synchronously with two reagents. Using those two samples and reagents six different combinations of mixing are possible such as S1 with R1 or S1 with R2 or S2 with R1 or S2 with R2 or S1 with R1 and R2 or S2 with R1 and R2. Based on application or detection / diagnosis relevant combination can be routed using control pins. At a time only one combination can be detected due to single mixer present in the chip. At each and every instance the routes and mixer has to be washed thoroughly with either water or any fluid based on the samples and reagents used to avoid contamination. This is done using manually[4].

In the existing design it uses more number of cells. At each instance only one sample will be processed. In proposed design for two samples and two reagents different ports / reservoirs have been specified. To avoid mixing of two droplets adjacent cell on both side of the droplet

routing path need to be empty i.e. deactivated mode. Due to this, size / area of DMFB are quite large enough.

In order to reduce the size of chip both the samples are placed in single port/reservoir, similarly reagents. In this paper just a single node / cell has been optimized using an evolutionary algorithm - Genetic Algorithm. This is the one best optimal algorithm for decision making. So for both the nodes has been implemented. By using the control pin which solution has to be route will be selected by first cell / node. By this way of arrangement, the number of cells in the array has been minimized.

Figure 3 (b). Shows the reduced number of cells in a 15 x 15 array DMFB, which is given by RDMFB.

1.2 Genetic Algorithm

Genetic Algorithms (GAs) are numerical optimization techniques based upon the mechanics of natural selection[9,10]. Pioneered in the 1970's by John Holland and colleagues at the University of Michigan [11], GAs use the concept of "survival of the fittest" to determine, through successive generations of randomized, but directed, information exchange, the optimal sample value or point in a search space [9,10]. Due to its natural solution it has been implemented in our research work which deals with biological solutions or samples.

GAs offer a robust search mechanism as *a priori* knowledge is not generally required for successful application of the algorithm, and they have been successfully applied to a wide range of optimization problems 9. The key operators of a GA are selection[9,10,12], crossover[9,10,13], mutation[9,10,14] and population size[9,10,15], with secondary parameters including variable encoding[9,10,16] and decoding[9,16] and population-update[9, 16], also having a bearing upon the effectiveness of the algorithm. Due to the stochastic nature of operation and the wide range of problems the GAs has been implemented. A formal, analytical description of the interactions and dependencies of the various parameters of a GA is complex. At the same time, selection of GA parameters and its associated settings has to be updated based upon empirical evaluation[17,18].

1.3 GA Parameters

The Simple GA (SGA) can be broken down into three distinct operations:

- Reproduction
- Crossover
- Mutation

The algorithm can be summarized as follows;

Begin
  t=0;
  initialise P(t) // *establish initial population of strings*
  evaluate P(t) // *apply objective function to each string in P(t)*
  fitness P(t) // *determine fitness of each string w.r.t. population as a whole*
  while termination condition not met do
Begin
  t=t+1 // *increment generation*
  select P'(t+1) from P(t) // *form intermediate population of fittest members from P(t)*
  pair off and mate P'(t+1) // *mate individuals in P'(t+1) to produce offspring*
  evaluate offspring P'(t+1) // *apply objective function to offspring of P'(t+1)*
  fitness offspring P'(t+1) // *determine fitness of P'(t+1) offspring*
  update to P(t+1) // *establish P(t+1) population from fittest of P'(t+1)*
  end
End

1.4 Fitness Evaluation

The determination of a string's fitness therefore can be viewed as follows;

$$F(x) = g(f(x))$$

Where, F(x) is the fitness value of string x.

f(x) is the objective function.

g(x) is the transformation of performance to relative fitness.

The measure of fitness of an individual is given by ratio performance value to the average performance value of the population as a whole.

Fitness of an individual = *fi/f '*.

Where, f_i is performance value.

f' is average performance value.

While predominantly used, a drawback to its use is that it can only be applied to non-negative values. In order therefore, to facilitate negative values, a linear transformation of the form,

$$F(x) = af(x) + b \text{ is used}$$

Where, a represents a positive scaling factor

b is an offset factor to ensure non-negative values.

1.5 Selection

Selection consist of three phases:

- Determination of the expected number of trials an individual can expect based on its relative fitness level. This is effectively the fitness evaluation stage as outlined above.
- Assignment of copies to the intermediate group ("Roulette Wheel" method).
- Random partner allocation of intermediate group members.

Flowchart:

Figure 4(a) Shows the flow diagram of general genetic algorithm. It represents the processing steps involved in genetic optimization technique.

Each cell in the Biochip will be optimized using GA. Below represents architecture of single node i.e. first node for both sample port and reagent port, which decides which one to be transported from sample and reagent node to 2 x 3 array mixer. Using those two samples and reagents six different combinations of mixing are possible such as S1 with R1 or S1 with R2 or S2 with R1 or S2 with R2 or S1 with R1 and R2 or S2 with R1 and R2. Based on our requirement either S1 or S2 and R1 or R2 will be transported.

Above Figure 4(b) Represents the architecture of single node in the DMFB. Each and every node in the chip architecture will be of this, i.e. Each and every node in the chip will be optimized with GA and then the solution or sample will be routed. This concept will be implemented for above given modified 12 x 12 array chip.

Pseudo-code:

let A<= 11110000 →38 total fitness // (A is greater)
let B<= 11001111→23 total fitness // A and B are 8-bit wide.
for i in 1 to 7 loop // x,y are 3-bit wide
X<= A (((i-1) * 1) downto (2+ (i-1) * 1));
Y<= B (((i-1) * 1) downto (2 + (i-1) * 1));
end loop;

X => 000;
for j in 1 to 3 loop
if (X(j) = '1') then
X(j) := '0';
else
X(j):= '1' ;
end if ;
end loop;

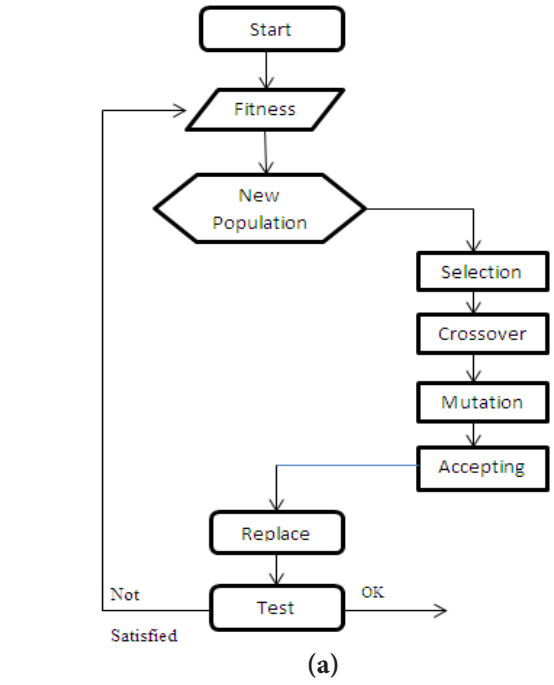


Figure 4. **(a)** Single Genetic Node Process flow chart, (b) Single Genetic Node Architecture.

Now X=>111;
Now A => 11110000 => 33

Y => 111;
for k in 1 to 3 loop;
if (Y(k)= '1') then

Y(k) := '0';
else
Y(k) := '1';
end if;
end loop;
Now Y=>000;
Now B=>11001000=>16
From above A > B

then repeat for loop

Loop repeats for all mutation and crossover till all are done. Then finally compare the fitness of each individual. From LSB 3 bits have been chosen for iteration and fitness has been evaluated and this repeats till all 8 bits are iterated. Based upon greater fitness value input solution will be routed. From above A > B, so for control signal '1', solution A will be delivered first and then B. Similarly for control signal '0', B will be routed first and then A. Here solution A is sample 1 and solution B is sample 2.

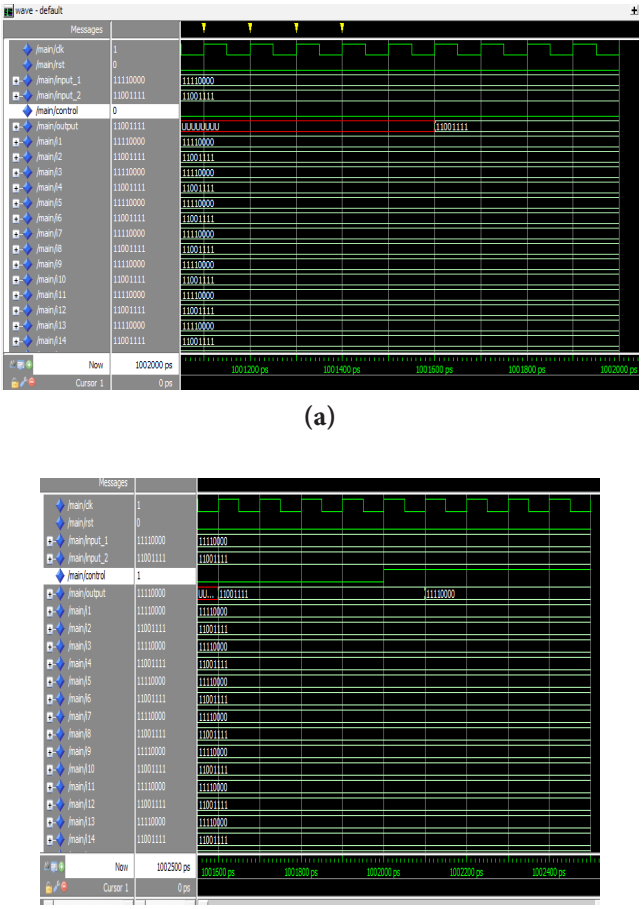
The same has been implemented for Reagent 1 and reagent 2. The reagent node will also be optimized using this algorithm. Based on the control signal the reagent 1 or 2 will be routed into the chip. The same pseudo-code has been implemented at the same time based on the utility. Here solution A is reagent 1 and solution B is reagent 2.

2. Experimental Results and Discussion

A random combination of inputs of 8 bit can be optimized with this algorithm. Which in sense any sample and reagent can be routed into this chip. Here for simulation the input combination has been chosen as A-11110000 and B-11001111 and control signal for A is 1 and control signal for B is 0. The given solution should be digitalized and then it will be optimized using this algorithm. For this optimized cell FPGA synthesis has been also done in order to analyze the block utilization.

2.1 Simulated Output

- When control pin is '1', Sample 1 (S1) and reagent 1 (R1) [here it is given by input_1] will be transported. The solution or sample which is placed in port S1 will be routed into the chip, which is shown in Figure 5(a).
- When control pin is '0', Sample 2 (S2) and Reagent 2 (R2) [here it is given by input_2) will be transported.



(a)

(b)

Figure 5. **(a)** Simulated Output, when control pin is '1', **(b)** Simulated Output, when control pin is '0'.

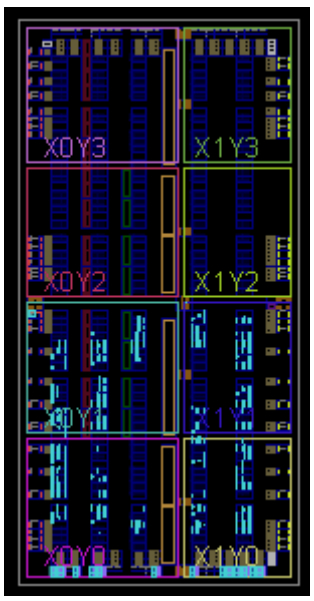


Figure 6. FPGA Layout.

Table 1. Utilization Table

Estimated Values			
Logic	**Used**	**Available**	**Utilization**
Number of Slice Registers	520	4800	10%
Number of Slice LUTs	329	2400	13%
Number of Fully used LUT-FF pairs	296	553	53%
Number of bonded IOBs	27	102	26%
Number of BUFG / BUFGCTRLs	1	16	6%

The solution or sample which is placed in port S2 will be routed into the chip, which is shown in Figure 5(b).

2.2 FPGA Synthesis

- The optimized cell has been synthesized in FPGA tool kit using plan a-head software, which is shown in Figure 6. This is same for reagent cell and sample cell.

2.3 Device Utilization

It gives about the single node/port utilization. Table 1, gives the description about the registers and IO utilization.

3. Conclusion

Thus the single node has been optimized using Genetic Algorithm and synthesized using FPGA. This optimization is for both sample and reagent port. By Genetic Algorithm optimization the logic utilization has been minimized. The register utilization is 10%, LUT's utilization is 13%, bonded IOB's utilization is 26% and buffer gate and buffer gate control signal utilization is 6%. For future work, this 12 x 12 array can be reduced into 10 x 10 arrays for routing of sample from input port to detection or dispensing port. Each and every node will be optimized using this GA and finally the solution will be routed.

4. References

1. Chao WC. Paper Survey of Biochip Routing. IEEE transaction, Computer Aided Design, ICCAD, Double Tree Hotel San Jose, CA. 2009 Nov; 2-5: 1–7.
2. Dhal D, Roy S, Datta P, Chakraborty A, Pal RK. A Literature Survey on Pin Configuration and Performance of Some Restricted Sized Digital Microfluidic Biochip. IJIET, 2015 Feb;5(1):20–8.
3. Pan I, Mukherjee R, Rahaman H, Samanta T, Dasgupta P. Optimization Algorithms for the Design of Digital Microfluidic Biochips: A Survey. ELSEVIER, Computers and Electrical Engineering. 2013 Jan; 39(1):112–21.
4. Huang TW, Tsung-Yi Ho. Design and Optimization for Digital Microfluidic Biochips. *IEEE* TCAD. 2011; 30: 215–228.
5. Zhao Y, Chakrabarty K, Sturmer R, Pamule VK. Optimization Techniques for Synchronization of concurrent fluidic operations in pin constrained Digital Microfluidic Biochips. IEEE transactions on VLSI systems. 2012 Jun; 20(6):1132 - 45.
6. Zhao Y, Chakrabarty K. Simultaneous optimization of droplet routing and control – pin mapping to electrodes in Digital Microfluidic Biochips. IEEE transactions on Computer Aided Design of Integrated Circuits and systems. 2012 Feb; 31(2):242–54.
7. Hueng TW, Ho TY. A two stage Integer Linear Programming-based droplet routing algorithm for pin constrained digital microfluidic biochips. IEEE transactions on Computer Aided Design of Integrated Circuits and systems. 2011 Feb; 30(2):215–28.
8. Juarez J, Brizuela C, Martinez I, Velazquez K, Lozano E. A Genetic Algorithm for the Routing of Droplets in DMFB: Preliminary Results. IEEE International Conference on Systems, Man and Cybernetics, San Diego, CA; 2014 Oct. p. 3808–15.
9. Goldberg DE. Genetic Algorithms in Search, Optimisation and Machine Learning. Addison- Wesley;1989.
10. Darrell W. A Genetic Algorithm Tutorial. Statistics and Computing. 1994; 2(4):65–85.
11. Synopsis of John Holland's early development of Genetic Algorithms; 2013 May. Available from : www.brunel.ac.uk/depts./AI/alife/ga-holla,
12. John GJ, James BE. How Genetic Algorithms Work: A critical look at Implicit Parallelism. Proceedings of the Third International Conference on Genetic Algorithms. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA;1989. p. 20–7.
13. Gilbert S. Uniform Crossover in Genetic Algorithms. Proceedings of the Third International Conference on Genetic Algorithms. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA; 1989. p. ;2–9 .
14. Robert H, Harry G, Peachey TC. The Nature of Mutation in Genetic Algorithms', Proceedings of the Sixth International Conference on Genetic Algorithms. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA; 1995. p. 65-72.
15. Colin RR. Using Genetic Algorithms with Small Populations. Proceedings of the Fifth International Conference on Genetic Algorithms. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.1993.92 –9.

16. Andrew C, Peter F, Harmut P, Carlos F. Genetic Algorithm Toolbox For use with Matlab v5 – User Guide. Department of automatic control and systems engineering. University of Sheffield, United Kingdom; 1994.
17. Kalyanmoy D, Samir A. Understanding Interactions among Genetic Algorithm Parameters. Foundations of Genetic Algorithms 5 conference, Leiden, Holland, San Mateo, CA: Morgan Kauffman; 1998. p. 265–86.
18. Sathish BS, Thirusakthimurugan P. Fuzzy Position Control of PMBLDC Motor using Adaptive Genetic Algorithm. Indian Journal of Science and Technology. 2015 Apr; 8(s.8): 600–6.