Fuzzy Evolutionary Algorithm for VLSI Placement

Sadiq M. Sait Habib Youssef Junaid A. Khan

Department of Computer Engineering
King Fahd University of Petroleum and Minerals
Dhahran, Saudi Arabia

Presentation Overview

- Introduction
- Problem statement and cost functions
- Proposed scheme
- Results
- Conclusion

Introduction

- A Fuzzy Evolutionary Algorithm for VLSI placement is presented.
- Standard Cell Placement is:
  - A hard multi-objective combinatorial optimization problem.
  - With no known exact and efficient algorithm that can guarantee a solution of specific or desirable quality.
- Simulated Evolution is used to perform intelligent search towards better solution.
- Due to imprecise nature of design information, objectives and constraints are expressed in fuzzy domain.
- The proposed algorithm is compared with Genetic Algorithm.

Problem Statement & Cost Functions

Problem Statement

- <u>Given</u>
  - A set of modules M = {m_1,m_2,m_3,é m_n}
  - A set of signals V = {v_1, v_2, v_3,é v_k}
  - A set of Signals $V_i \subseteq V$, associated with each module $m_i \in M$
  - A set of modules $M_j = \{m_i | v_j \in V_i\}$, associated with each signal $v_j \in V$
  - A set of locations L = {L_1, L_2, L_3é L_p}, where $p \geq n$
- Objectives
  - The objective of the problem is to assign each $m_i \in M$ a unique location L_j, such that
  - Power is optimized
  - Delay is optimized
  - Wire length is optimized
  - Within accepted layout Width (Constraint)

Cost Functions

- Wire length Estimation

$$Cost_{wire} = \sum_{i \in M} l_i$$

Where

l_i ...... is the estimate of actual length of signal net v_i, computed using median Steiner tree technique

- Power Estimation

$$P_t \approx \sum_{i \in M} \frac{1}{2} C_i \cdot V^2{}_{DD} \cdot f \cdot S_i \cdot \beta$$

Where:

S_i ...... Switching probability of module m_i

C_i ...... Load Capacitance of module m_i

V_{DD} ... Supply Voltage

f ...... Operating frequency

β ...... Technology dependent constant

Cost Functions

- Power Estimation (contd.)
  - Also

$$C_i = C_i^r + \sum_{j \in M_i} C_j^g$$

Where

C_i^r is the interconnect capacitance at the output node of cell i.
C_j^g is the input capacitance of cell j.

- In standard cell placement V_{DD}, f, β, and C_j^g are constant and power dissipation depends only on S_i and C_i^r which is proportional to wire-length of the net v_i. Therefore the cost due to power can be written as:

$$Cost_{power} = \sum_{i \in M} S_i \cdot l_i$$

Cost Functions

- Delay Estimation
  - We have a set of critical paths $\{\pi_1, \pi_2, \pi_3$é é $\pi_k\}$
  - $\{v_{i1}, v_{i2}, v_{i3}$é é $v_{iq}\}$ is the set of signal nets traversing path π_i.
  - $T_{\pi i}$ is the delay of path π_i computed as:

$$T_{\pi i} = \sum_{i=1}^{q} (CD_i + ID_i)$$

Where

CD_i ...... is the delay due to the cell driving signal net v_i.

ID_i ...... is the interconnect delay of signal net v_i.

Now

$$Cost_{delay} = \max(T_{\pi i}) \ldots\ldots \forall i \in \{1,2,3,4\ldots\ldots k\}$$

Cost Functions

- Width Constraint

$$Width_{\max} = (1+a) \times Width_{opt}$$

Where

$\text{Width}_{\text{max}}$ is the maximum allowable width of layout

$\text{Width}_{\text{opt}}$ is the optimal width of layout

a denotes how wide layout we can have as compared to its optimal value.

Cost Functions

- Fuzzy logic for multiobjective optimization
  - Unlike crisp set theory, members of a fuzzy set have degree of membership in the range [0,1]
  - Each objective cost is mapped to the membership value in the corresponding fuzzy set of ñgood in that objectiveò
  - Some linguistic fuzzy rule is used to combine objectives (AND or OR logic)
  - Linguistic rule is mapped to some fuzzy operator, where membership values are combined into membership in fuzzy set of good overall solution

- Fuzzy Operators Used
- And-like operators
  - Min operator $\mu = \min(\mu_1, \mu_2)$
  - And-like OWA

$$\mu = \beta * \min(\mu_1, \mu_2) + \tfrac{1}{2}(1-\beta)(\mu_1+\mu_2)$$

- Or-like operators
  - Max operator $\mu = \max(\mu_1, \mu_2)$
  - Or-like OWA

$$\mu = \beta * \max(\mu_1, \mu_2) + \tfrac{1}{2}(1-\beta)(\mu_1+\mu_2)$$

SE Algorithm

ALGORITHM SimE(M,L)
/* M: Set of moveable elements */
/* L: Set of locations */
/* B: Selection bias */
INITIALIZAION:
Repeat
  EVALUATION:
    For Each $m \in M$
      compute(g_m)
    End For Each
  SELECTION:
    For Each $m \in M$
      If Selection(m,B) **Then**
        $P_s = P_s \cup \{m\}$
      Else $P_r = P_r \cup \{m\}$
      End If
    End For Each
  Sort the elements of P_s;
  ALLOCATION:
    For Each $m \in P_s$
      Allocation(m)
    End For Each
Until Stopping criteria are met
Return (Best Solution)
End SimE

Proposed Fuzzy goodness evaluation

IF cell i is

near its optimal wire-length AND near its optimal power AND near its optimal net delay OR $T_{max}(i)$ is much smaller than T_{max} **THEN** it has high goodness.

Where

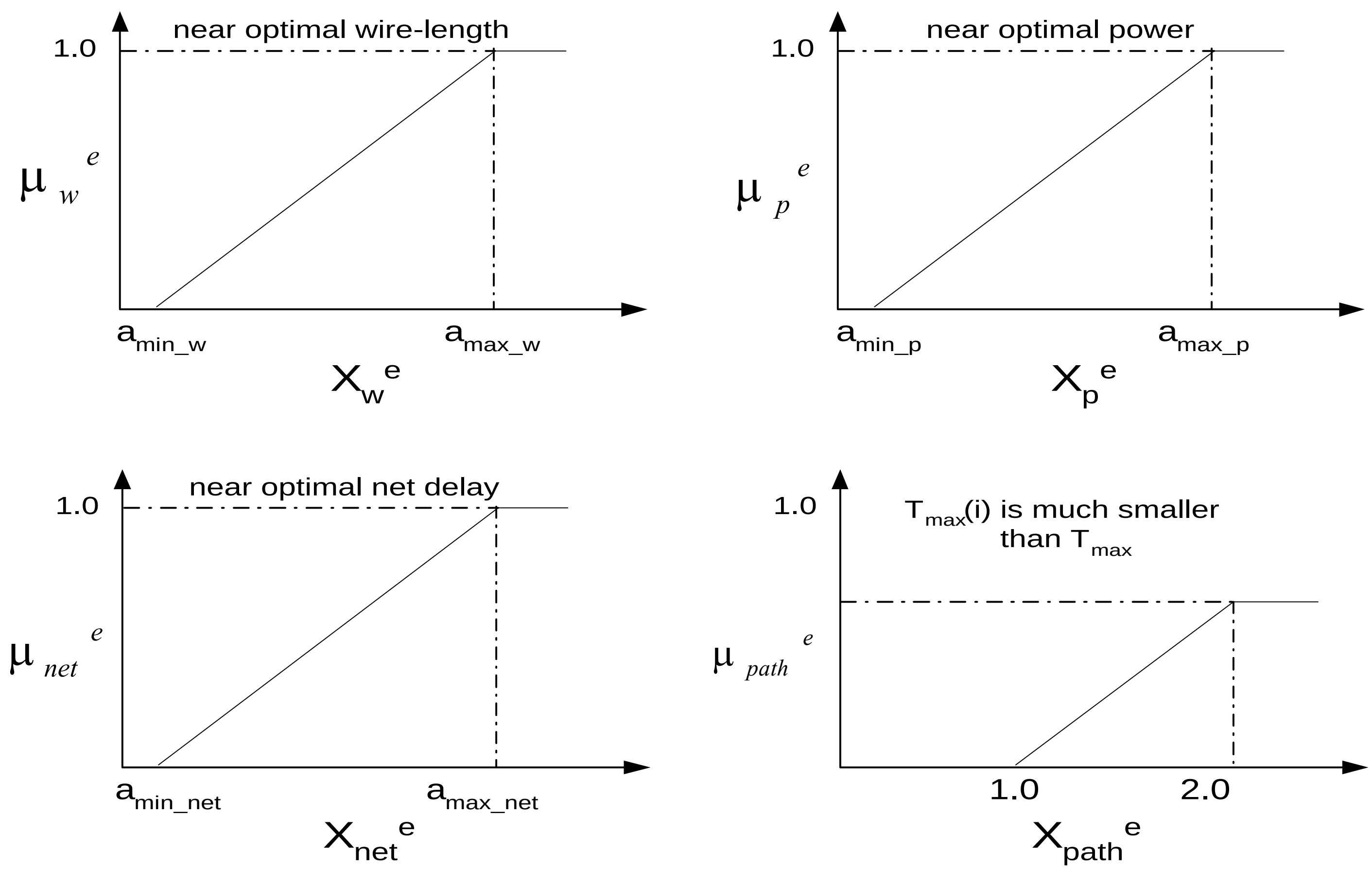
T_{max} is the delay of the most critical path in the current iteration and $T_{max}(i)$ is the delay of the longest path traversing cell i in the current iteration

$$\mu_i^e(x) = \beta^e \times \min(\mu_{iw}^e(x), \mu_{ip}^e(x), \mu i_d^e(x)) + (1-\beta^e)\frac{1}{3}\sum_{j=w,p,d}\mu_{ij}^e(x)$$

Where

$$\mu_{id}^e(x) = \beta_d^e \times \max(\mu^e{}_{inet}(x), \mu^e{}_{ipath}(x)) + (1-\beta_d^e)\frac{1}{2}(\mu^e{}_{inet}(x) + \mu^e{}_{ipath}(x))$$

Goodness (Membership Functions)



Goodness (base values)

$$X_{iw}^{e}(x) = \frac{\sum_{j=1}^{k} l^{*}_{j}}{\sum_{j=1}^{k} l_{j}}$$

$$X_{ip}^{e}(x) = \frac{\sum_{j=1}^{k} S_{j}.l^{*}_{j}}{\sum_{j=1}^{Ki} S_{j}.l_{j}}$$

$$X^{e}{}_{inet}(x) = \frac{ID_{i}^{*} + ID_{p}^{*}}{ID_{i} + ID_{p}}$$

$$X^{e}{}_{ipath}(x) = \frac{T_{\max}}{T_{\max}(i)}$$

Where

- l^{*}_{j} …… lower bound on wire length of signal net v_j
- l_j …… actual wire length of signal net v_j
- S_j …… is the switching probability of v_i

Where

- ID_i^{*} …… is the lower bound on interconnect delay of v_i
- ID_p^{*} …… is the lower bound on interconnect delay of the input net of cell i that is on $\pi_{max}(i)$
- $T_{max}(i)$ …… Delay of longest path traversing cell i
- T_{max} …… Delay of most critical path in current iteration

- Goodness (a_{min_i} and a_{max_i})

$$a_{min\_i} = average(X^e_i)\ \text{ï}\ 2xSD(X^e_i)$$

$$a_{max\_i} = average(X^e_i) + 2xSD(X^e_i)$$

- Selection

A cell i will be selected if

$$Random \geq g_i + bias$$

Range of the random number will be fixed i.e, [0,M]

$$M = average(g_i) + 2 \text{ x } SD(g_i)$$

M is computed in first few iteration, and updated only once when size of selection set is 90% of its initial size

Allocation

- Selected cells are sorted w.r.t. their connectivity to non-selected cells.
- Top of the list cell is picked and its location is swapped with other cells in the selection set or with dummy cells, the best swap is accepted and cell is removed from the selection set.
- Following Fuzzy Rule is used to find good swap.

IF a swap results in

reduced overall wire length AND
reduced overall power AND
reduced overall delay AND within acceptable layout width

THEN it gives good location

Allocation (contd.)

$$\mu^{a}{}_{i\_pwd}(l) = \beta^{a} \min(\mu^{a}{}_{iw}(l), \mu^{a}{}_{ip}(l), \mu^{a}{}_{id}(l)) + (1-\beta^{a})\frac{1}{3}\sum_{j=p,w,d}\mu^{a}{}_{ij}(l)$$

$$\mu_i^{a}(l) = \min(\mu^{a}{}_{i\_width}(l), \mu^{a}{}_{i\_pwd}(l))$$

Where

l ...... represents a location

$\mu_{iw}{}^{a}$ ...... membership in fuzzy set, reduced wire length

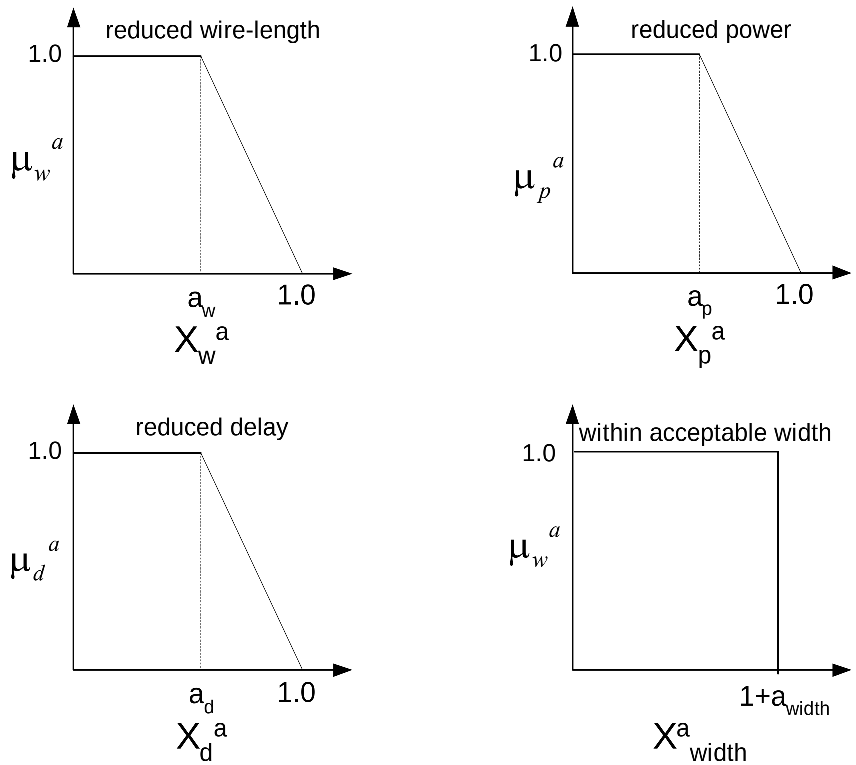
$\mu_{ip}{}^{a}$ ...... membership in fuzzy set, reduced power

$\mu_{id}{}^{a}$ ...... membership in fuzzy set, reduced delay

$\mu^{a}{}_{i\_width}$ ...... membership in fuzzy set, smaller layout width

$\mu_i^{a}(l)$ ...... is the membership in fuzzy set of good location for cell i

Allocation (Membership functions)



$$X^a{}_{iw}(l) = \frac{\left(\sum_{m=1}^{ki} l_{im} + \sum_{m=1}^{kj} l_{jm}\right)_n}{\left(\sum_{m=1}^{ki} l_{im} + \sum_{m=1}^{kj} l_{jm}\right)_{n-1}}$$

$$X^a{}_{ip}(l) = \frac{\left(\sum_{m=1}^{ki} S_{im} l_{im} + \sum_{m=1}^{kj} S_{jm} l_{jm}\right)_n}{\left(\sum_{m=1}^{ki} S_{im} l_{im} + \sum_{m=1}^{kj} S_{jm} l_{jm}\right)_{n-1}}$$

$$X^a{}_{id}(l) = \frac{(ID_i + ID_{ip} + ID_j + ID_{jp})_n}{(ID_i + ID_{ip} + ID_j + ID_{jp})_{n-1}}$$

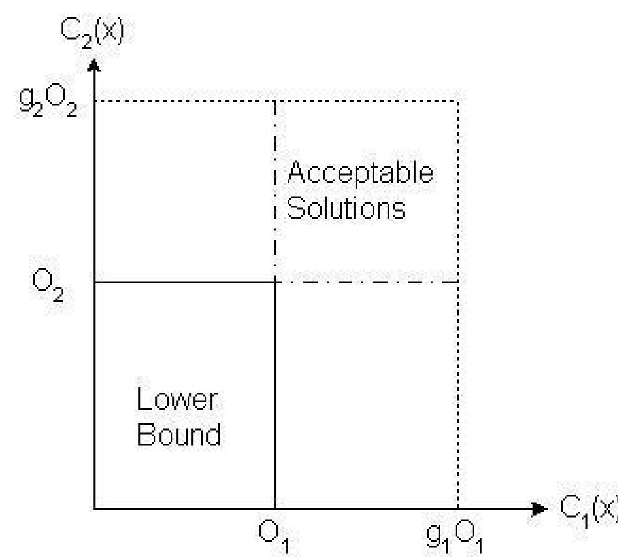
$$X^a{}_{i\_width(l)} = \frac{Cost_{Width_n}}{Width_{opt}}$$

These values are computed when cell i swap its location with cell j, in nth iteration

Fuzzy Cost Measure

- Set of solutions is generated by SimE
- Best solution is one, which performs better in terms of all objectives and satisfies the constraint
- Due to multi-objective nature of this NP hard problem fuzzy logic (fuzzy goal based computation) is employed in modeling the single aggregating function

- Range of acceptable solutions



Fuzzy Cost Measure (contd.)

- Following fuzzy rule is suggested in order to combine all objectives and constraint

$$\mu^c{}_{pdl}(x) = \beta^c \times \min(\mu^c{}_p(x), \mu^c{}_d(x), \mu^c{}_l(x)) + (1-\beta^c) \times \frac{1}{3} \sum_{i=p,d,l} \mu^c{}_i(x)$$

IF a solution is within acceptable wire-length AND acceptable power AND acceptable delay AND within acceptable layout width

THEN it is an acceptable solution

$$\mu^c(x) = \min(\mu^c{}_{pdl}(x), \mu^c{}_{width}(x))$$

Fuzzy Cost Measure (contd.)

where

X ... is the solution

μ^c_{pdl} ... is membership in fuzzy set, acceptable power and delay and wire-length

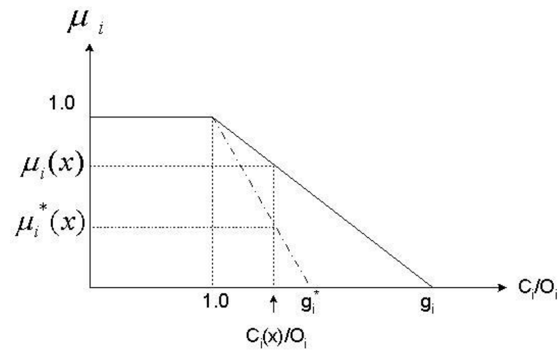
μ^c_p ... is membership in fuzzy set, acceptable power

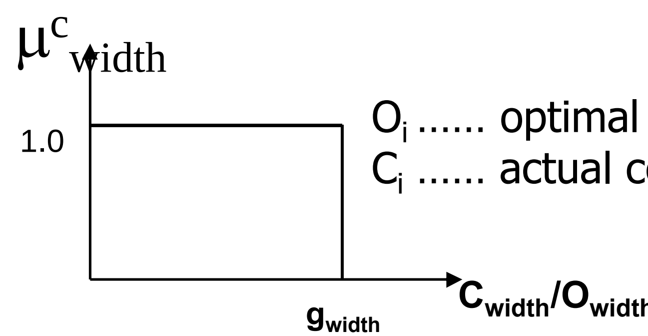
μ^c_d ... is membership in fuzzy set, acceptable delay

μ^c_l ... is membership in fuzzy set, acceptable wire-length

μ^c_{width} ... is membership in fuzzy set, acceptable width

μ^c ... is membership in fuzzy set, acceptable solution





O_i ...... optimal costs

C_i ...... actual costs

Membership functions

Genetic Algorithm

Å Membership value $\mu^c(x)$ is used as the fitness value.

Å Roulette wheel selection scheme is used for parent selection.

Å Partially Mapped Crossover.

Å Extended Elitism Random Selection is used for the creation of next generation.

Å Variable mutation rate in the range [0.03- 0.05] is used depending upon the standard deviation of the fitness value in a population.

Experiments and Results

Technology Details

- .25 μ MOSIS TSMC CMOS technology library is used
- Metal1 is used for the routing in horizontal tracks
- Metal2 is used for the routing in vertical tracks

Metal Type	ω (μm)	R_{sh} $\Omega/\square$	C_p aF/μ^2	C_f aF/μ
Metal 1	0.36	0.07	39	26
Metal 2	0.36	0.07	19	60

0.25 μ technology parameters

Circuits and Layout Details

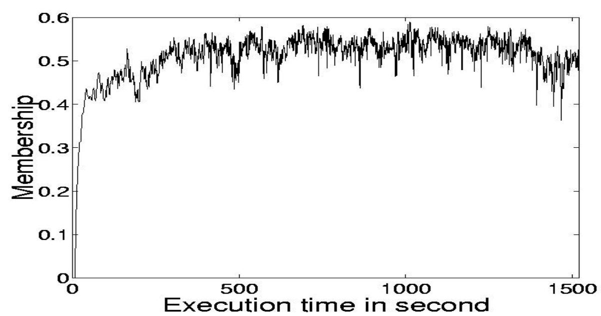
Circuit		Layout	
Name	Number of Cells	Number of Rows	Average Channel Height in μm
S2081	122	4	6.66
S298	136	5	7.08
S386	172	5	7.68
S641	433	7	9.72
S832	310	7	9.78
S953	440	8	11.76
S1196	561	9	11.58
S1238	540	9	11.64
S1488	667	11	10.92
S1494	661	11	10.56
C3540	1753	16	13.68

Comparison b/w GA and FSE

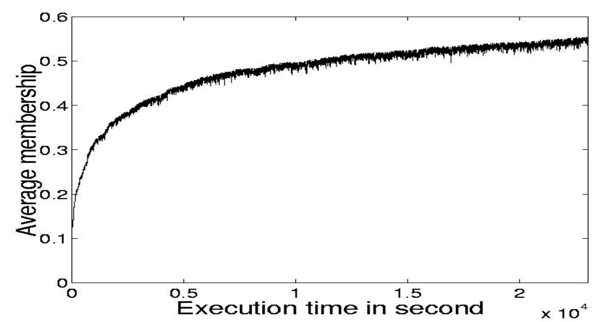
Circuit	GA					FSE				
	L (μm)	P (μm)	D (ps)	W (μm)	T (s)	L (μm)	P (μm)	D (ps)	W (μm)	T(s)
S2081	2426	388	113	142	2341	2693	462	112	152	43
S298	4062	838	130	171	2922	4989	1013	133	181	104
S386	6824	1665	193	181	3945	7088	1640	197	186	152
S832	21015	4787	395	232	7206	24705	5827	390	258	1643
S641	18320	4365	736	254	21982	13906	3321	702	296	618
S953	32031	5156	230	262	11221	32340	5242	245	262	1278
S1238	52679	15473	410	279	16208	39629	12377	371	310	1168
S1196	51804	15259	370	292	23070	42426	12745	364	325	1521
S1494	71021	17497	803	336	26032	56961	14071	719	360	3378
S1488	69792	17346	784	334	21434	57091	13887	710	358	3529
C3540	310996	109850	924	427	43232	164897	58055	734	507	18318

Table 1: Layout found by FSE and GA, "L", "P" "D" and "W" represent the wire-length, power, delay, and width costs and "T" represents execution time in seconds

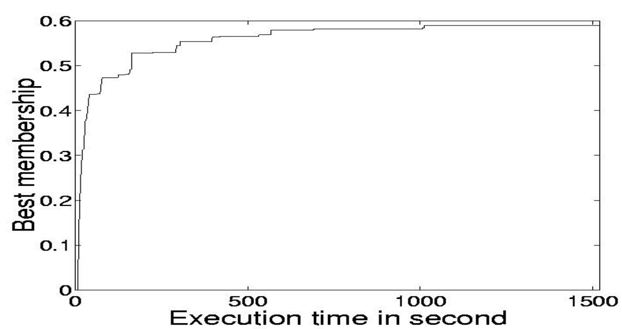
Comparison b/w GA and FSE



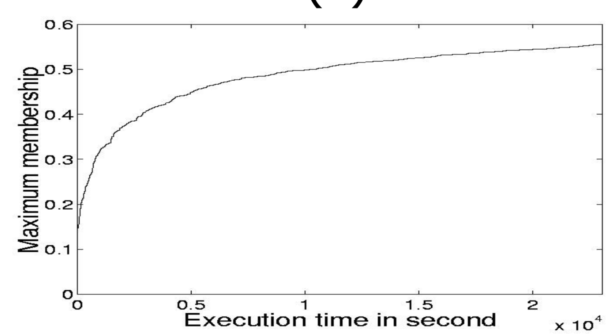
(a)



(b)



(c)

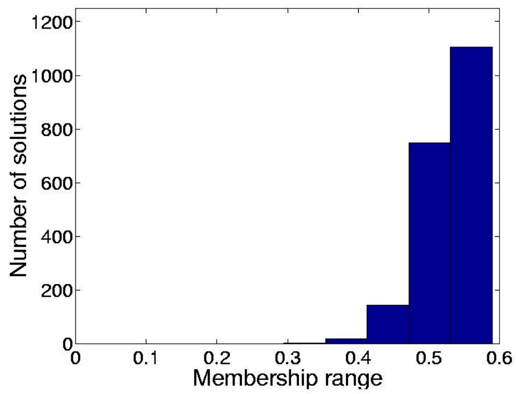


(d)

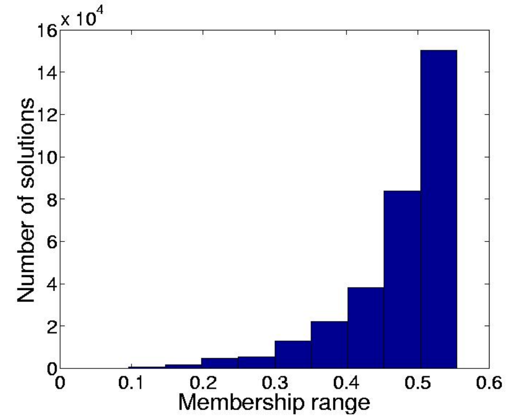
(a) And (c) represents current and best fitness of solution by FSE. (b) and (d) represent average and best membership by GA for S1196

Comparison b/w GA and FSE

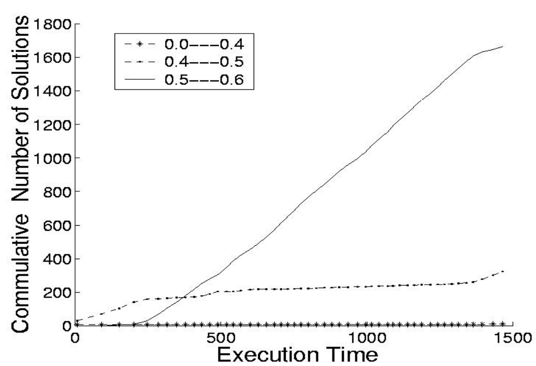
(a)



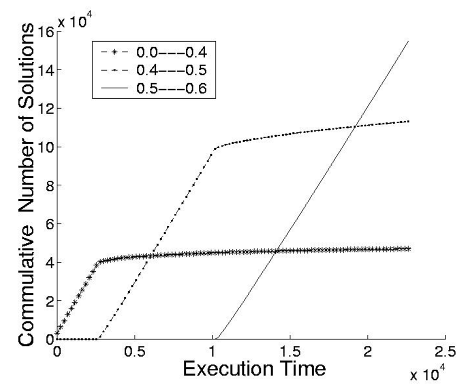
(b)



(c)



(d)



(a) and (b) show number of solution visited in a particular membership range. (c) and (d) show cumulative number of solutions in specific membership ranges vs. execution time for FSE and GA respectively for S1196.

Conclusion

- An evolutionary algorithm (FSE) for low power high performance VLSI standard cell placement is presented
- Fuzzy logic is used to overcome the multiobjective nature of the problem
- FSE performs better than GA with less execution time and better quality of final solution
- FSE has better evolutionary rate as compared to GA