INDIAN INSTITUTE OF MANAGEMENT CALCUTTA

WORKING PAPER SERIES

WPS No. 661/ August 2010

A Novel Approach for Solving Multi-objective Optimization Problems: A Case of VLSI Thermal Placement

by

Ram Babu Roy
Doctoral student, IIM Calcutta, Joka, Diamond Harbour Road, Kolkata 700104, India

Abstract—We often confront with optimization problems that require optimizing multiple objectives. Such multi-objective optimization problems are often non-linear in nature and are hard to solve. In such situations conventionally heuristic methods are used to arrive at some reasonably good solutions. VLSI standard cell placement problems traditionally have to handle multiple objectives such as area, delay, thermal distribution and so on, to arrive at a reliable design. One of the key concerns in this area is to simultaneously (i) optimize the thermal distribution of the heat dissipated by the logic gates on the chip and (ii) minimize the total wirelength required to interconnect these gates. This in turn helps in reducing the chances of occurrence of hot spots on the chip, on-chip delay and the total chip area. Optimizing these objectives individually is known to be NP-hard and hence simultaneous optimization of the two objectives is a challenging problem. In this work, we have also proposed a game-theoretic formulation to the problem and developed some novel heuristic algorithms for solving this problem. The proposed algorithms have been implemented, and the experimental results are quite encouraging.

***Index Terms*— optimization methods, Integrated Circuit layout design, game theory, heuristic methods**

I. Introduction

Modern business scenario gives us plenty of examples of competition and co-operation as available resources are scarce. On one hand, firms in the market may have to compete with each other on some issues and on the other hand they have to cooperate with each other on some other issues. To survive in the cut throat competition the decision makers need to optimize multiple objectives simultaneously under several constraints and, therefore, they resort to some trade-offs. For designing the retail shop networks for supply chain

*corresponding author
This work was supported in part by a research grant from the Department of Information Technology (DIT), Govt. of India.
Ram Babu Roy is a research student with Management Information Systems Group at the Indian Institute of Management Calcutta, Ko

III. LITERATURE REVIEW

As feature size in a VLSI ch

where *R(i)* and *C(i)* are the row and column coordinate of the cell *i* as shown in Figure 3 and w_{ij} is the interconnect weight between cell i and cell j as shown in Figure 2.
The constraints to be satisfied

The aim of the players is to achieve a collective optimum goal even if the actions are not in the best of their private interest. Such many-player games are called *collective-action games*. The socially optimal outcome is not automatically achievable as the Nash equilibrium of the game [6]. We choose two sub-teams sequentially based on connectivity and proximity between the cells. The players of both the teams play in teams and have a common strategy of accepting the moves or rejecting the moves depending on the collective gain of both the teams. The strategy corresponding to a positive payoff is selected. If the pay-off is negative, we reject the move and select new sub-teams.

B. Choosing the players

The choice of players is an important issue as this ensures the uniformity in the temperature distribution across the chip. The rules for choosing the players for simultaneous optimization of temperature distribution and total wirelength are given below.

After thermal placement using algorithm 1 (discussed in the next section) we obtain approximately uniform thermal distribution and assign colors to all the cells and the T_{max} is very close to the ideal T_{avg}

Compute the total interconnect weight of each cell and arrange the cells in non-increasing order of their total interconnect weight

Choose the cell with the highest interconnect weight as the seed cell

Take the three neighbours of the seed cell from its associated window (t = 2) as players {*p1, p2, p3*} in team 1. Note that these three players will be of three different colors due to the restriction imposed by Algorithm 1 that no two neighbors would have same color

Take three cells heavily connected with the seed cell and each with three different colors matching with the colors of the players of team 1 as the three players {*p4, p5, p6*} in team 2

Play the game of swapping positions amongst the players of these two teams so as to minimize the total wire length

C. Deciding the strategies of the players

We consider two teams each comprising three players. In general, a player has three possible strategies of exchanges with players of the other team. The payoff depends on the sequence of moves of the players. So we need to decide the priority in choosing the players for playing the sequential game. For 3 players of one team we have 3 positions of players of other team to exchange their positions. If we assign first preference to player 1, second preference to player 2, and so on, we have the following 6 possible strategies of exchanges given by $^{3}P_{3}$ as shown in Figure 4. Player P1 can swap its position with any one of the three players from other team. Subsequently, player P2 can swap its position with remaining two and player P3 can swap with the remaining one player of the other team. We can select the one giving the maximum payoff.

3 Possibilities

P1

Algorithm 1
Input: The Temperature matrix
Output: Optimized Temperature matrix such that maximum value of the window (sub-matrix) temperature is minimized.
• Arrange all the n = m*m (assume m to be an even number) cell temperatures in non-decreasing order (T0,T1…….Tn) • Divide them into 4 sets S1,S2,S3,S4 with temperature values say {T0, T1,……….T(n/4-1)}, {Tn/4, Tn/4+1,……….T(n/2-1)}, {Tn/2, Tn/2+1,……….T(3n/4-1)}, {T3n/4, T3n/4+1,……….T(n-1)}, • Assign colors to the cells in the 4 sets as black (B), pink (P), green (G) and red (R) • Mark the top left corner cell of the layout matix as position (0,0) • Starting from position (0,0), place cells from set S1 in an ascending order i.e. starting from T0 in alternate cell positions i.e. (0+2*i, 0+2*j) in the layout matrix • Starting from position (1,1) , place cells from set S2 in a descending order i.e. starting from T(n/2-1) in alternate cell positions i.e. (1+2*i, 1+2*j) in the layout matrix • Starting from position (1,0), place cells from set S3 in a ascending order i.e. starting from Tn/2 in alternate cell positions i.e. (1+2*i, 0+2*j) in the layout matrix • Starting from position (0,1) , place cells from set S4 in a descending order i.e. starting from T(n-1) in alternate cell positions i.e. (0+2*i, 1+2*j) in the layout matrix

Figure 5: Algorithm 1

We have proposed algorithm 2 (Figure 6) for optimizing the thermal distribution of the cells as well as the total interconnect wirelength. In this algorithm we have implemented game theoretic concept. This is an *n*-player cooperative game. We are choosing a subset of players sequentially to play the game. We choose a seed cell having highest connectivity, and select two sets of players forming two teams. One team consists of neighbouring cells of the seed cell and the other team consists of cells strongly connected to the seed cell. The strategies of the players are to exchange their positions so as to reduce the total wirelength without disturbing the temperature distribution significantly. The payoff of the team is the decrease in the total wirelength L by exchanging their positions simultaneously. The players in the two teams play a cooperative game and try to maximize the total payoff. The exchange is allowed only when it gives a positive payoff. In this algorithm we have not imposed any restriction on the quantum of allowed deviation *DeltaT* of the maximum window temperature T_{max} from T_{avg}. Instead we

Lemma 1: For *t* = 2, each window has four cells with distinct colors.
Proof: We have placed the cells as per algorithm 1 that ensures that each widow has only one cell from a particular temperature band. While exchanging the cells, we are exchanging a cell with only cells of the same color.

Lemma 2: The algorithm always terminates in finite time.
Proof: The algorithm terminates if all the cells have played or the pointer to select the cell not yet played reduces to zero. Since the pointer is decremented by one after every iteration, the algorithm is guaranteed to terminate after a finite number of iterations.

VIII. ILLUSTRATIONS OF THE ALGORITHMS

The effect of algorithm 1 has been illustrated in Fig. 7. The random cell placement before application of algorithm 1 shows that the maximum difference between the temperatures of windows of size 2X2 is 34. This difference reduces to 14 after application of algorithm 1.

Cell placement before application of algorithm 1

Cell placement after application of algorithm 1

Figure 7: Illustration for Algorithm 1

The effect of algorithm 2 has been illustrated in Fig. 8. The cell placement before application of algorithm 2 shows that the maximum difference between the temperatures of windows of size 2x2 is 14. Algorithm 2 brings the heavily connected cells to the seed cell no 7 and of different colors (i.e. 18, 2 and 9) to the same window as occupied by cell no 7 by exchanging these cells with cell no 28, 1 and 8 respectively. This exchange helps in reducing the total wirelength without affecting the maximum difference in window temperature. In this case we observe that the difference reduces from 14 to 13 after application of algorithm 2.

Figure 8: Illustration for Algorithm 2

IX. SIMULATION RESULTS AND OBSERVATIONS

A. Comparison of T_{max} achieved

We have implemented all the algorithms using C and simulated on a computer having AMD Turion64 processor with a clock speed of 1.60 GHz and 512 MB RAM. For the simulation, we have generated random numbers using uniform random distribution representing cell temperatures in the range of 0 to 1000. Similarly the elements of the adjacency matrix for the interconnect weight was generated randomly with weights in the range of 0 to 10. The maximum window temperature T_{max} achieved by both the algorithms have been compared with T_{avg} in Table 1. The percentage deviation from the average window temperature T_{avg} has also been given in the table. As we can see from the histogram in Figure 9 the algorithms reduce the deviation of maximum window temperature from the corresponding T_{avg} value so that the maximum window temperature becomes very close to the ideal average window temperature T_{avg}.

S.No	m	T	Tavg	Tmax (initial)	Tmax (algo1)	Tmax (algo2)	%Deviation (initial)	%Deviation (algo1)	%Deviation (algo2)
1	4	2	1979	3104	2014	2212	56.85	1.77	11.77
2	8	2	1922	2904	2012	2120	51.09	4.68	10.30
3	8	4	7689	8246	7831	7864	7.24	1.85	2.28
4	16	2	1964	3330	1989	2256	69.55	1.27	14.87
5	16	4	7856	9636	7888	8198	22.66	0.41	4.35
6	16	8	31427	32523	31444	31897	3.49	0.05	1.50

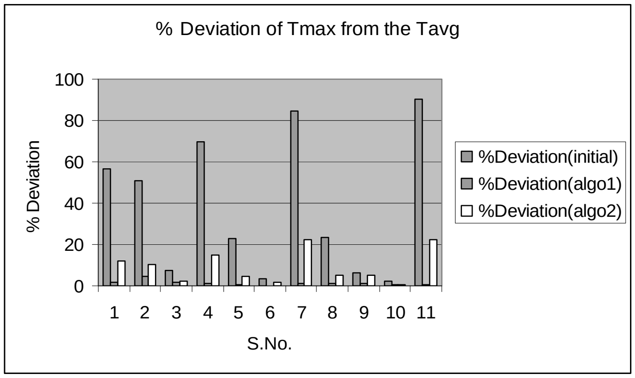

Figure 9: Comparison of deviation (%) of T_{max}