

# Thermal Driven Placement for Island-style MTCMOS FPGAs

Javid Jaffari and Mohab Anis

Department of Electrical and Computer Engineering, University of Waterloo, Ontario, Canada

Email: {jjaffari, manis}@vlsi.uwaterloo.ca

**Abstract**—Rapid increase in transistor density and operating frequency has led to the increase in power densities, exhibiting itself as a high temperature profile. The high temperature spots over an FPGA impact the power, performance, and reliability of the chip, hence should be addressed during the design process. The logic block placement is targeted as the natural starting point to address the non-uniform thermal profile problem. The proposed placer simultaneously accounts for conventional placement objectives (routability and timing) while increases the temperature profile uniformity by optimally spreading the power sources. As a measure of thermal uniformity in the simulation annealing core of the placer, a cost function is derived by adapting the concept of maximum entropy in a dual electrostatic charge model. The runtime complexity of this cost function is linear with respect to the number of used blocks, regardless of the size of the FPGA, and there is no need to perform the time-consuming thermal extractions. Results show an average of 73% and 51% reductions in the standard deviation and maximum gradient of temperature with less than 4% average wiring and delay penalty.

**Index Terms**—FPGA, Placement, Temperature, Maximum Entropy, Simulation Annealing

## I. INTRODUCTION

Advances in CMOS scaling has increased the power density of chips due to increasing in the number of transistors per unit area and operating frequencies. The increase in the power density has elevated the temperature of die's substrate which brings serious reliability concerns into the future designs. A similar trend is followed by FPGAs as they are increasingly being used for high performance applications rather than only prototyping. In fact, the thermal issue has been recently indicated as an important fact by FPGA manufacturers [1], [2]. Therefore, distributed temperature sensors have been proposed to monitor and study the thermal profile over FPGAs [3], [4].

The uneven power consumption profile in FPGAs due to the existence of unused logic blocks (40% of total blocks [5]) and difference in switching activities generate local hotspots. In addition, the anisotropic heat conduction of the die's sidewalls is another source of thermal non-uniformity. The non-uniform high temperature profile over substrates causes a range of design challenges, as it affects the gate and interconnect delays [6], [7], introduces

new timing faults [8], increases the leakage power [9], and accelerates the chip failure due to electromigration and thermal runaway [10], [11]. Therefore, to achieve a robust design which guarantees satisfaction of system constraints (performance, power, and reliability), one should reduce hot spots during placement which is the natural starting point of a temperature-aware design flow [12].

The thermal placement problem was modeled in [13] as a matrix synthesis problem where the temperature distribution is optimized by uniform distribution of heat sources on the chip. It assumes that a uniform heat distribution leads to a uniform temperature profile which is not accurate due to the anisotropic thermal conductivity of the chip's sidewalls. A compact substrate thermal model was developed in [12], and two algorithms for standard cell and macro cell style design were presented. For the standard cell placement, a simulated annealing procedure tries to match the power distribution with the targeted distribution obtained from a desired temperature profile. However, a close approximation to the targeted power distribution does not necessarily lead to a close approximation of the targeted temperature profile. Also, for the macro cell style design, a simulated annealing procedure is directing by the calculation of temperature profile changes in every move cycle. This approach suffers from low run time efficiency. A partitioning-based thermal placement method was presented in [14] which still runs in quadratic time with the number of mesh nodes on the substrate and needs to compute the inverse of the equivalent resistor network admittance matrix. Another simulated annealing based thermal placement used for 3D ICs was developed in [15]. They made use of an iterative method called Alternate Direction Implicit (ADI) [16] to extract the temperature profile in each cell movement. Although the ADI algorithm runs in linear time, it needs to iteratively solve hundreds of equations to accurately estimate the temperature profile after a move [16]. Hence, due to its long runtime, [15] suggests computing temperature profile in every  $m$  moves which degrades the effectiveness of the method.

### A. Related Works

Besides the listed custom-IC thermal-aware placers, an FPGA thermal-aware placement technique is proposed in [17] which uses the HS3d thermal extraction tool [18] to improve the thermal profile by updating placement constraints and rerunning the placer and router iteratively. In

This paper is based on "Thermal-Aware Placement for FPGAs using Electrostatic Charge Model," by J. Jaffari and M. Anis, which appeared in the Proceedings of the IEEE Symposium on Quality Electronic Design (ISQED), San Jose, USA, March 2007. © 2007 IEEE.

fact, the authors did not integrate any thermal uniformity measure into the placer objective and only reran the placer by new thermal-driven placement constraints until a satisfied maximum temperature is achieved. As a result, this approach suffers from long runtime due to iterative runs of placer and the thermal extractor. Finally, a thermal-aware placement is proposed in [19] for FPGAs where the maximum or variance of temperature is included in the placer's simulated annealing objective. It updates the thermal profile in each iteration in linear time with respect to number of logic blocks; however, the inverse of the equivalent resistor network admittance matrix should be calculated which can be very time consuming when working with industrial large scale FPGAs with over 20K logic blocks [20]. Moreover, as the runtime is a function of FPGA size (not the design size), any attempt to place a rather small design into a large FPGA for prototyping purposes takes a long time.

As described earlier most of the thermal-aware placers reduce a thermal cost function during the annealing process by directly considering the actual temperature profile. However [21] proposed a force-directed algorithm to use in thermal-aware placement of multichip modules. It uses an electrostatic based repulsive force between chips which is proportional to power consumptions of corresponding chips. However, force directed methods usually suffer from overlapping problems, and also, among various existing methods for placement, the simulated annealing is the one that yields the close to global optimum results. In addition, this technique does not take into account the wiring length and timing which is critical for the logic block placement in FPGAs.

In this paper a thermal-aware placement for FPGAs is proposed based on an electrostatic charge model over the VPR framework [22]. The conventional total wiring length and timing objectives are also considered for assuring the routability and performance requirements. By modeling the heat sources as electric charges, the objective of the simulating annealing is altered to minimizing the total electrical potential energy of the model instead of dealing with the temperature profile which is very time consuming. The reason behind choosing this objective is originated from the maximum entropy condition of the electrostatic system. By using this new thermal cost, large charges (heat sources) place fairly far from each other and die sidewalls. This leads to a more smooth temperature profile which has lower temperature variance and gradient as well as the peak temperature. Also, the algorithm runtime complexity is linear with the number of used blocks which is suitable for the FPGA placements and does not need any time-consuming matrix inversion or thermal extraction. In Section II, the thermal extraction and electrostatic charge model are reviewed. The proposed placement algorithm is discussed in Section III. Section IV compares the placement results with and without applying the proposed method. Finally, the paper is concluded in Section V.

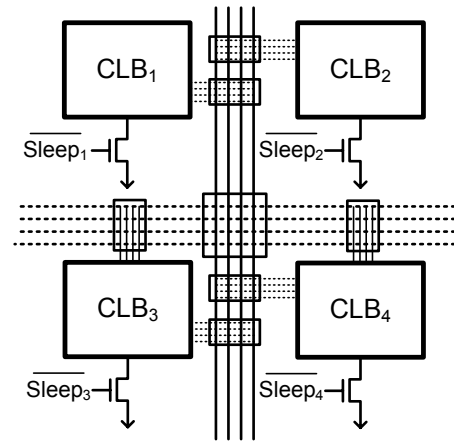


Figure 1. FPGA Architecture

## II. PRELIMINARIES

### A. FPGA Architecture

In this work, the popular island-style MTCMOS FPGA architecture is targeted [23], [24]. In this architecture, the functionality of a given circuit is implemented in Configurable Logic Blocks (CLBs). Each logic block encapsulates Basic Logic Elements (BLE), where each consists of a one Look-Up Table (LUT), a flip-flop, and a multiplexer. The logic blocks are connected to each other through horizontal and vertical routing channels and switches.

To eliminate the static power of unused logic blocks, their power-ground paths are gated through high threshold voltage NMOS switches as shown in Figure 1. MTCMOS architecture has become popular between FPGA vendors recently due to rapid increase in the magnitude of leakage current in advanced technologies. By using this architecture, the power consumption of unused CLBs can be assumed zero.

### B. Thermal Profile of an FPGA

The steady state temperature profile over a solid body is governed by:

$$\nabla^2 T(x, y, z) + \frac{g(x, y, z)}{k} = 0 \quad (1)$$

where  $T$  is temperature ( $^{\circ}C$ ),  $g$  is power density of heat sources ( $W/m^3$ ), and  $k$  is the thermal conductivity ( $W/m^{\circ}C$ ). Typically, the temperature profile is extracted by meshing the substrate and packaging materials, then solving the thermal equation using the Finite Difference Method (FDM) [12].

The mesh structure can be modeled using a circuit composed by resistors and current sources in which each node's voltage represents the corresponding mesh's temperature. The thermal resistance between adjacent meshes  $i$  and  $j$  in horizontal direction is  $R_{ij} = \frac{\Delta x}{k\Delta y\Delta z}$  where  $\Delta x$ ,  $\Delta y$ , and  $\Delta z$  are dimensions of 3-D meshes in  $x$ ,  $y$ , and  $z$  directions. Finally, the power consumption of each mesh is modeled by a current source flowing into the node representing that mesh.

The thermal profile of the island-style FPGA is modeled such that each CLB is covered by one mesh. As a result,  $g(x, y, 0)$  is the power consumption of that CLB in the  $(x, y)$  coordinates.

C. Analogy Between Electrostatic Charge and Thermal Models

Derived from Gauss' law for electricity, the Laplacian of the voltage profile over a space is proportional to the distribution of charge density, as follows:

$$\nabla^2 V(x, y, z) + \frac{\rho(x, y, z)}{\epsilon} = 0 \quad (2)$$

where  $V$  is the voltage ( $v$ ),  $\rho$  is the charge density ( $C/m^3$ ), and  $\epsilon$  is the permittivity ( $F/m$ ).

This relation between voltage and charge density is similar to the equation between temperature and power density given in (1). Therefore there is an analogy between heat transfer and electrostatic potential equations. By making use of the analogy between (1) and (2), each CLB can be modeled by a point charge ( $\rho$ ) corresponding to the CLB's power density ( $g$ ). Therefore the CLB's temperature ( $T$ ) can be modeled as the charge voltage ( $V$ ).

However, the thermal insulated die sidewalls represented by Neumanns condition ( $\partial T / \partial n_i = 0$ ) are modeled by mirrors which produce images for the charges [21]. The mirroring structure will be described in the next section.

III. PROPOSED METHOD

A. Proposed Thermal Cost and Mirroring Structure

In simulated annealing based approaches, a cost function is minimized during random moves of blocks. Therefore, the run-time of the algorithm is significantly affected by the efficiency of the cost function evaluation as it needs to be executed in every particle move. As mentioned in the introduction, traditionally, simulated annealing based thermal aware placers have used temperature profile directly to minimize the peak, variance, or maximum temperature gradient. This, in fact, leads to have a slow algorithm because of the need to solve finite difference equations generated by 1 in every block move (iteration), or a need to construct the inverse of the resistive equivalent admittance matrix.

To have a fast while efficient cost function which is also only dependent to the size of circuit, the total electrical potential energy of the equivalent electrostatic model of the FPGA. This value is used as an indicator to show how well are the CLBs placed accounting for temperature distribution. This is because when the total electrical potential energy is reduced the entropy of the equivalent electrostatic model is increased. Increase in entropy means more smoothing out the voltage distribution which is equal to the temperature profile in the thermal model. As a result, by using this indicator, during the cooling process of simulation annealing, the large power consuming blocks repel each other in order to

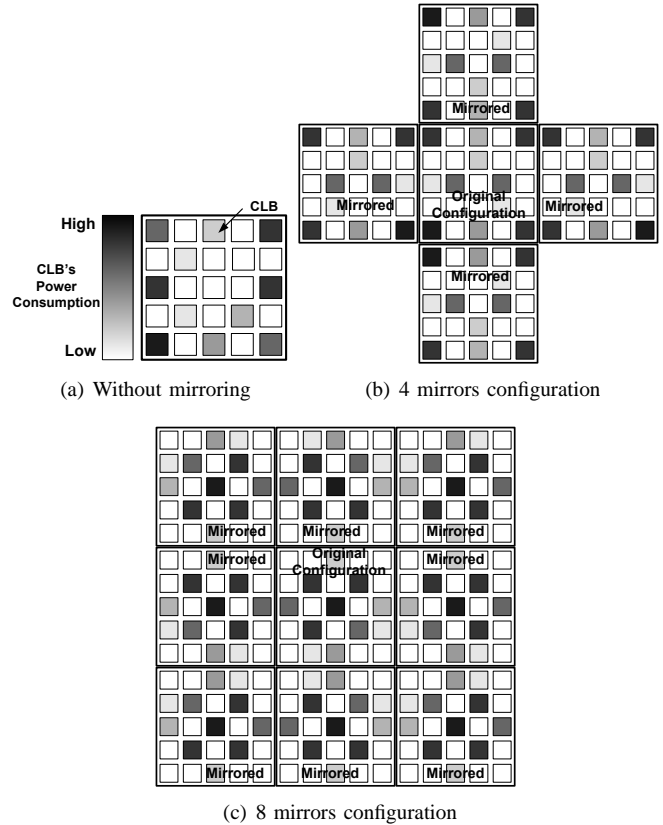


Figure 2. Illustration of different mirroring schemes on placement results

minimize the total potential energy of the system. This cost expression also takes into account insulating die's boundaries effects by adding the potentials caused by 8 symmetrically mirrored configurations surrounding the original FPGA, based on the image charge concept. This energy which can be considered as a thermal cost is:

$$Thermal\_Cost = \sum_{i=1}^u \left( \sum_{j=1}^u \frac{g_i g_j}{r_{ij}} + \sum_{m=1}^{8u} \frac{g_i g_m}{r_{im}} \right) \quad (3)$$

where  $u$  is the number of used CLBs in a FPGA,  $g_i$ ,  $g_j$ , and  $g_m$  are the power densities of actual  $i^{th}$ ,  $j^{th}$  and mirrored  $m^{th}$  CLB. Finally,  $r_{ij}$  and  $r_{im}$  are Euclidean distances between corresponding blocks.

To clarify the die's boundaries effects as reflecting mirrors, it should be noted that for a substrate, the heat conduction from the sidewalls is insignificant compared to the top and bottom sides of it, and hence the temperature gradient normal to the boundary is almost zero. This phenomenon has been modeled as a reflected mirror image of the charge in the electrostatic model [21] where the electrical fields normal to the sidewalls are zero ( $\partial T / \partial n_i = 0$ ). This implies that higher local electrical potentials (temperatures) are generated where charges (heat sources) are so close to any mirror (die's sidewall). Therefore, the model can take into account the boosted temperature by the heat sources close to the insulated boundaries of the die. To consider the effects of insulated sidewalls, the CLB configuration is mirrored with respect to the die

sidewalls in an 8 mirrored configuration. For illustrative purposes, a sample of a 5×5 CLB FPGA is shown in Figure 2(c) after placement. It can be seen that no high power consumption blocks (dark block) are placed near each other or sidewalls as it may cause increment in the defined *Thermal\_Cost*. Without including mirrored charges in (3), the placer solution may converge to the configuration shown in Figure 2(a) where the high consumption blocks repel each other, and hence, place in the boundaries of the FPGA, which raises the temperature profile near the sidewalls. However, if 4 mirrored configuration had been used, the high consumption blocks might have been placed in corners rather than sidewalls as shown in Figure 2(b). This would reduce the temperature near sidewalls but increase the temperature of the four corners of the FPGA significantly. Therefore, the best structure is when all 8 mirrored configurations are included, so repelling of the high power consumption blocks around the sidewalls prevent any misplacement of the high power consumption blocks.

The computational complexity of *Thermal\_Cost* is quadratic with respect to the number of used CLBs ( $u$ ) which is not appropriate for large circuits. Considering the fact that in each CLB movement, the potential energy of the equivalent electrostatic model due to other fixed CLBs are constant, the exact *Thermal\_Cost* can be calculated by finding its changes ( $\Delta Thermal\_Cost$ ) after a move. As shown in Figure 3, after moving the CLB#3, the *Thermal\_Cost* is just altered due to changes in  $r_{13}$  and  $r_{23}$ , distance of CLB#3 from CLB#1 and CLB#2. Therefore, there is no need to recompute *Thermal\_Cost* from scratch. In other words,  $\Delta Thermal\_Cost$  which represents the amount of work needed to replace the charge  $i$  can be computed faster and used to direct the simulation annealing process. This modification reduces the complexity of calculations in each move (simulation iteration) to linear with the number of used CLBs over the FPGA.

To calculate the cost change in every move, the following equation is used:

$$\Delta Thermal\_Cost = \sum_{j=1}^u g_i g_j \left( \frac{1}{r_{ij(new)}} - \frac{1}{r_{ij(old)}} \right) + \sum_{m=1}^{8u} g_i g_m \left( \frac{1}{r_{im(new)}} - \frac{1}{r_{im(old)}} \right) \quad (4)$$

where distances after and before each move are shown by *new* and *old* indices. A negative value for  $\Delta Thermal\_Cost$  is desirable, indicating that the replacement has caused a reduction in total potential energy, and hence, increase in entropy which results in smoother electrostatic potential (thermal) profile. It should be noted that during the simulated annealing process the change rather than actual value of a cost function is used to reject or accept a random move.

### B. Implementation

An integrated final cost factor containing the new defined thermal cost and the traditional wiring and timing

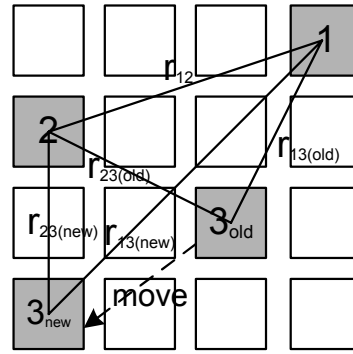


Figure 3. Computation scheme of  $\Delta Thermal\_Cost$

costs is formulated. The placement cost function of VPR is a weighted sum of two defined costs on wiring and timing [22]. Our proposed final cost change ( $\Delta C$ ) merges the traditional VPR costs with the new thermal cost by a coefficient ( $0 \leq \alpha \leq 1$ ) determining the emphasis on thermal smoothening during placement:

$$\Delta C = \alpha \cdot \left( \frac{\Delta Thermal\_Cost}{Previous\_Thermal\_Cost} \right) + (1 - \alpha) \left( \lambda \cdot \left( \frac{\Delta Timing\_Cost}{Previous\_Timing\_Cost} \right) + (1 - \lambda) \cdot \left( \frac{\Delta Wiring\_Cost}{Previous\_Wiring\_Cost} \right) \right) \quad (5)$$

where  $\lambda$  denotes the trade off between wiring and timing, and the previous value of costs are used to normalize the changes in each iteration. Hence, the weights of the three components are automatically adjusted such that the algorithm always allocates the required importances to each factor. The  $\Delta Timing\_Cost$  and  $\Delta Wiring\_Cost$  are calculated based on a timing analyzer and a net semi-perimeter metric wire length estimator [22].

In order to speed up the algorithm, a lookup table matrix ( $D$ ) containing the inverted values of distances between CLB's is used. At the start of the algorithm, each  $D_{x,y}$  element of this matrix is filled with  $1/\sqrt{x^2 + y^2}$ . Therefore, during the placement procedure and  $\Delta Thermal\_Cost$  evaluations there is no need to do the time consuming square root, divide, and multiplications in evaluating  $1/r$  as the inverted values of distances have already been extracted and saved in a lookup table.

## IV. RESULTS AND DISCUSSION

As mentioned earlier, the VPR placement tool was used for implementing our proposed technique. In order to verify the output configuration temperature profile, the equivalent circuit of the thermal model [12] is simulated. The surface meshing is done based on the CLBs, where each mesh represents one CLB, and the thickness meshing is done by discretizing the die thickness ( $200\mu m$ ) into 4 meshes.

The CLBs' switching activities are randomly generated between '0' and '1' as been done in [12], [14].  $2 \times 10^6 (W/m^2)$  is assigned to the peak power dissipation density. Thermal conductivity for the packaging is

TABLE I.  
EXPERIMENTAL RESULTS FOR APEX2

Size (Util.)	$\alpha$	Wiring (Penalty)	Delay (Penalty)	$T_{max}$ (Drop)	$\sigma_T$ (Drop)	$\delta_T$ (Drop)
23×23 92%	0	185.4	40.18	123.5	4.97	5.17
	0.33	(0%)	(0%)	(1%)	(37%)	(13%)
	0.5	(0.5%)	(0%)	(2%)	(52%)	(28%)
	0.75	(0.6%)	(0.6%)	(2.7%)	(64%)	(39%)
26×26 72%	0	187.9	40.67	115.9	13.5	7.78
	0.33	(0%)	(1%)	(6.5%)	(34%)	(15%)
	0.5	(1.6%)	(0.5%)	(7%)	(43%)	(22%)
	0.75	(3.1%)	(4%)	(12%)	(72%)	(46%)
31×31 50%	0	187.72	41.03	115.8	26.6	9.19
	0.33	(0.5%)	(2.3%)	(11%)	(26%)	(7.5%)
	0.5	(0.7%)	(5%)	(18%)	(48%)	(10%)
	0.75	(16%)	(14%)	(27%)	(80%)	(53%)

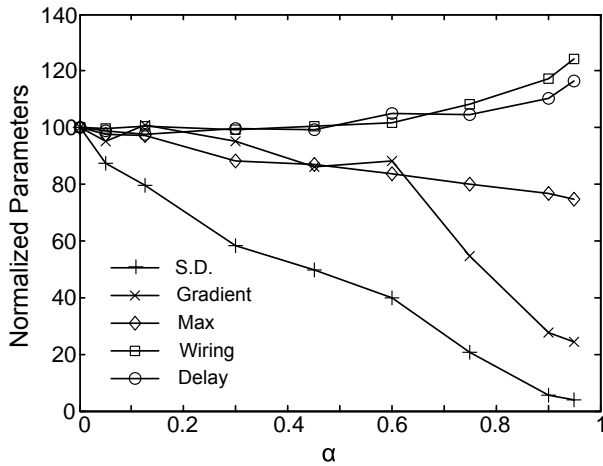


Figure 4. Normalized temperature's standard deviation, maximum gradient, and maximum vs. delay and wiring with respect to trade off factor  $\alpha$ , for Apex2 with 62% utilization.

assumed to be  $10^4(W/m^2C)$  and  $25^{\circ}C$  is assigned to ambient temperature.

Our method is applied to the MCNC benchmark circuits. The circuits are partitioned into clusters containing 4 BLEs where each has a 4-input LUT. The FPGAs are assumed to have equal CLBs in its rows and columns. The simulation results for **apex2** with various FPGA utilization percentage and  $\alpha$  is given in Table I.  $\alpha=0$  means the placement is done without considering the temperature (Eq. (5)). The trade off factor between wiring and timing cost ( $\lambda$ ) is set to 0.5, to provide the best results between wiring and timing [22]. All temperatures are in ( $^{\circ}C$ ), and "Delay" represents critical path delay in ( $ns$ ). The "Wiring" is the estimated cost of wiring in CLB count unit.  $T_{max}$  is the maximum temperature across the chip,  $\sigma_T$  is the standard deviation of the temperature, and  $\delta_T$  is the maximum on-chip temperature gradient which is equal to maximum adjacent CLBs' temperature difference. It should be noted that since the proposed thermal-aware placement increase the thermal smoothness, all variance, maximum temperature gradient, and maximum temperature are reduced.

As can be seen in Table I, by increasing  $\alpha$ , the temperature metrics are improved while the wiring and critical delay become worse. However, to have better

TABLE II.  
EXPERIMENTAL RESULTS FOR UTILIZATION $\approx$ 75% AND  $\alpha=0.75$

Circuit	(Wiring Pen.)	(Delay Pen.)	$T_{max}$ (Drop)	$\sigma_T$ (Drop)	$\delta_T$ (Drop)
alu4	(3%)	(6.7%)	103.7 (9.4%)	3.17 (77%)	3.38 (57%)
apex2	(3.1%)	(4%)	102.5 (11.5%)	3.78 (72%)	4.23 (46%)
apex4	(3.6%)	(4.3%)	101 (11%)	2.76 (65%)	3.4 (33%)
diffeq	(4.6%)	(0%)	100.2 (5%)	1.68 (73%)	2.6 (63%)
misex3	(3.3%)	(2.6%)	105.4 (9.1%)	3.5 (70%)	3.23 (57%)
seq	(2.7%)	(4.5%)	106.7 (7%)	3.45 (66%)	3.97 (44%)
tseng	(6%)	(0%)	99.2 (6.7%)	1.03 (87%)	2.87 (56%)
AVG	3.75%	3.15%	8.5%	73%	51%

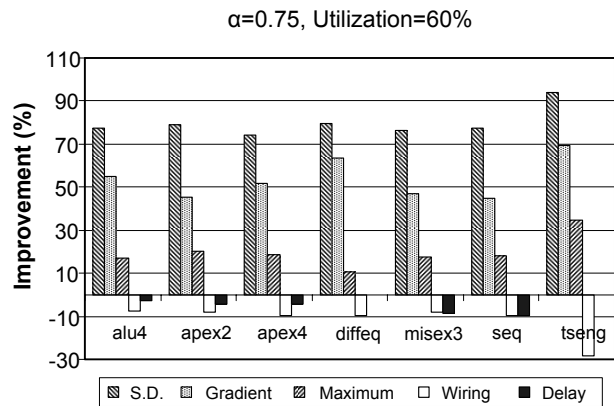


Figure 5. Experimental results for utilization  $\approx$ 60% and  $\alpha = 0.75$  for different benchmarks

results both for traditional costs and temperature, it is beneficial to increase  $\alpha$  for higher utilization percentages while for lower utilizations assigning lower values to  $\alpha$  avoids any undesirable wiring and delay penalty. This is due to the fact that for the lower utilizations, as there are many unused blocks, they can be placed among used blocks to make the temperature profile more uniform. This will impact the total wiring length of the circuit. On the other hand, when the FPGA's CLBs are mostly used, higher values can be set to  $\alpha$  with insignificant wiring and delay overhead.

Figure 4 depicts the normalized standard deviation, maximum gradient, and maximum temperature versus the delay and timing of Apex2 benchmark when  $\alpha$  is swept from 0 to 0.95. It can be concluded that by careful setting of  $\alpha$ , one can keep the wiring and delay costs close to the traditional (non-thermal placement) method while providing a huge smoothness in the temperature profile. In this case (utilization=62%) such  $\alpha$  would be between 0.6 and 0.75.

Table II shows the penalty and dropped percentages of traditional and new factors for different benchmarks where the utilization is set to 75%. The actual temperature performance metrics after thermal-aware placements have also been given. It can be seen that even without any

fine adjustment of  $\alpha$  and setting it to 0.75, an average of 73% and 51% reductions are achieved in the temperature standard deviation and maximum gradient with less than 4% wiring and delay penalties. Also, in our experiments **tseng** and **diffeq** did not show any penalty in their critical delay. The reason is that these circuits contain long paths which makes them less sensitive to the replacement of any CLB among them causing less overall delay and wiring penalty.

Figure 5 also gives the similar information this time using the typical FPGA logic utilization of 60% [5], but the same  $\alpha$  which gives more temperature's standard deviation and maximum gradient improvement (80% and 54% in average, respectively) due to lower utilization. However, the increased wiring penalty of **tseng** suggests using a lower value of  $\alpha$  as was expected from the lower utilization and discussed earlier.

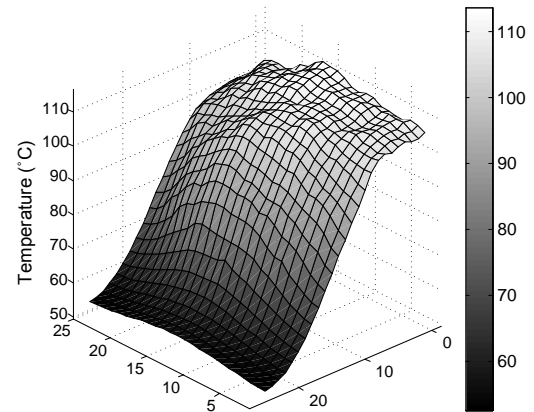
The reduced maximum gradient and temperature standard deviation yield to smoother temperature profile over the chip, and hence, the hot spots are reduced. Figure 6 shows two temperature profiles of **alu4** circuit placed in a  $25 \times 25$  CLBs FPGA (Utilization=61%) and  $\alpha=0.75$ . It shows how the proposed method yields to a smoother temperature profile with reduced hotspots temperature in comparison to the traditional method which does not consider temperature during placement.

V. CONCLUSION

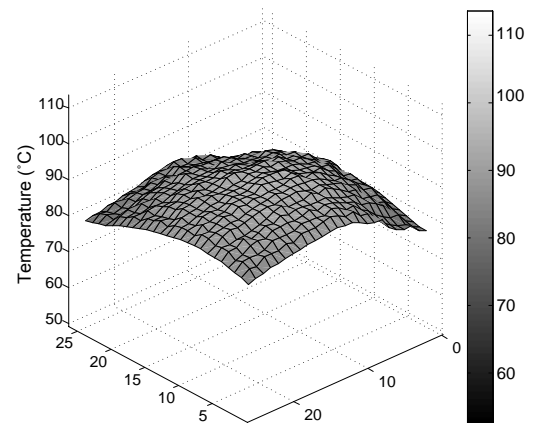
In this paper, a thermal-aware CLB placement is proposed for FPGAs. A new thermal cost factor is defined for the objective function of the simulation annealing core based on an electrostatically equivalent charge model with 8 mirrors modeling insulated sidewalls. By adapting the concept of maximum entropy in electrostatic systems and analogy between potential and thermal profile equations, the thermal profile is optimized through minimizing the total potential energy of the equivalent electrostatic system. Therefore, by avoiding the use of actual temperature profile as a cost function, there is no need to extract the whole chip temperature profile in each simulation annealing iteration which is runtime inefficient. In fact, the computational complexity of the proposed cost function has been kept linear with the number of used CLBs, by modeling the changes in the electrical potential energy of the model. The placement objective was then set to make a trade off among temperature, wiring, and timing costs. For 75% utilization of CLBs in sample benchmarks, the method showed an average of 73% and 51% reduction in temperature's standard deviation and maximum gradient, respectively, with less than 4% delay and wiring penalties. The reduction of the standard deviation improves peak temperature 8.5% in average and yields smoother temperature profile.

REFERENCES

[1] "Thermal management for FPGAs," Application Note 358, Altera Corporation.



(a) Without thermal consideration



(b) With thermal consideration

Figure 6. Temperature profile of alu4

[2] A. Telikepalli, "Designing for power budgets and effective thermal management," *Xilinx Xcell Journal*, vol. 56, pp. 24–27, 2006.

[3] S. Lopez-Buedo, J. Garrido, and E. I. Boemo, "Dynamically inserting, operating, and eliminating thermal sensors of FPGA-based systems," *IEEE Trans. Comp., Packag., Manuf. Technol. A*, vol. 25, pp. 561–566, 2002.

[4] S. Velusamy, H. Wei, J. Lach, M. Stan, and K. Skadron, "Monitoring temperature in FPGA based socs," in *Proc. IEEE ICCD*, 2005, pp. 634–637.

[5] A. Gayasen, Y. F. Tsai, V. Narayanan, M. Kandemir, M. J. Irwin, and T. Tuan, "Reducing leakage energy in FPGAs using region-constrained placement," in *Proc. ACM FPGA*, 2004, pp. 51–58.

[6] R. Kumar and V. Kursun, "Reversed temperature-dependent propagation delay characteristics in nanometer CMOS circuits," *IEEE Trans. on Circuit and Systems-II*, vol. 53, pp. 1078–1082, Oct. 2006.

[7] A. H. Ajami, K. Banerjee, and M. Pedram, "Modeling and analysis of nonuniform substrate temperature effects on global ULSI interconnects," *IEEE Trans. Computer-Aided Design*, vol. 24, pp. 849–861, June 2005.

[8] S. Bota, M. Rosales, J. Rosello, A. Keshavarzi, and J. Segura, "Within die thermal gradient impact on clock-skew: a new type of delay-fault mechanism," in *Proc. IEEE Int. Test Conference*, 2004, pp. 1276–1283.

[9] H. Su, F. Liu, A. Devgan, E. Acar, and S. Nassif, "Full chip leakage estimation considering power supply and temperature variations," in *Proc. IEEE Int. Symp. on Low Power Electronics and Design*, 2003, pp. 78–83.

[10] T. Y. Wang, J. L. Tsai, and C. C. P. Chen, "Thermal and

- power integrity based power/ground networks optimization,” in *Proc. IEEE/ACM Design, Automation and Test in Europe*, 2004, pp. 830–835.
- [11] A. Vassighi and M. Sachdev, “Thermal runaway in integrated circuits,” *IEEE Trans. Device and Materials Reliability*, vol. 6, pp. 300–305, June 2006.
- [12] C. Tsai and S. Kang, “Cell-level placement for improving substrate thermal distribution,” *IEEE Trans. Computer-Aided Design*, vol. 19, pp. 253–266, Feb. 2000.
- [13] C. C. N. Chu and D. F. Wong, “A matrix synthesis approach to thermal placement,” *IEEE Trans. Computer-Aided Design*, vol. 17, pp. 1166–1174, Nov. 1998.
- [14] G. Chen and S. Sapatnekar, “Partition-driven standard cell thermal placement,” in *Proc. IEEE ISPD*, 2003, pp. 75–80.
- [15] K. Balakrishnan, V. Nanda, S. Easwar, and S. K. Lim, “Wire congestion and thermal aware 3D global placement,” in *Proc. ACM ASPDAC*, 2005, pp. 1131–1134.
- [16] T. Wang and C. Chen, “3-D thermal-adi: A linear-time chip level transient thermal simulator,” *IEEE Trans. Computer-Aided Design*, vol. 21, pp. 1434–1445, Dec 2002.
- [17] P. Sundararajan, A. Gayasen, N. Vijaykrishnan, and T. Tuan, “Thermal characterization and optimization in platform FPGAs,” in *Proc. IEEE/ACM ICCAD*, 2006, pp. 443–447.
- [18] G. M. Link and N. Vijaykrishnan, “Thermal trends in emerging technologies,” in *Proc. IEEE ISQED*, 2006.
- [19] S. Bhoj and D. Bhatia, “Thermal modeling and temperature driven placement for FPGAs,” in *Proc. IEEE ISCAS*, 2007.
- [20] “Xilinx products documentation,” <http://www.xilinx.com/literature>.
- [21] J. Lee, “Thermal placement algorithm based on heat conduction analogy,” *IEEE Trans. Comp. Packag. Technol.*, vol. 26, pp. 473–482, June 2003.
- [22] V. Betz, J. Rose, and A. Marquardt, *Architecture and CAD for deep-submicron FPGAs*. Kluwer Academic Publishers, 1999.
- [23] B. H. Calhoun, F. A. Honore, and A. Chandrakasan, “A leakage reduction methodology for distributed MTCMOS,” *IEEE J. Solid-State Circuits*, vol. 39, pp. 818–826, May 2004.
- [24] A. Rahman, S. Das, T. Tuan, and S. Trimberger, “Determination of power gating granularity for FPGA fabric,” in *Proc. IEEE CICC*, 2006, pp. 9–12.

**Javid Jaffari** received the B.Sc. degree in bioelectrical engineering from Science and Research Branch, Azad University, Tehran, Iran, in 2002, and the M.Sc. degree in electrical engineering from the University of Tehran, Iran, in 2005. He is currently working toward the Ph.D. degree in Electrical and Computer Engineering in University of Waterloo, ON, Canada. His current research interests include computer aided design of integrated circuits for process variation and thermal issues in deep sub micron regime.

**Mohab Anis** received the B.Sc. degree (honors) in electronics and communication engineering from Cairo University, Cairo, Egypt in 1997, and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of Waterloo, Ontario, ON, Canada, in 1999 and 2003, respectively. Currently, he is an Assistant Professor and Co-Director of the VLSI Research Group at the University of Waterloo. He has authored/coauthored over 60 papers in international journals and conferences, and is the author of the book *Multi-Threshold CMOS Digital Circuits - Managing Leakage Power* (Norwell, MA: Kluwer, 2003). His research interests include integrated circuit design and design automation for VLSI systems in the deep submicron regime. Dr. Anis is Associate Editor of *IEEE Transactions on Circuits and Systems - II*, *Journal of Circuits, Systems and Computers (JCSC)*, *ASP Journal of Low Power Electronics (JOLPE)*, and *VLSI Design*. He is also a member of the program committee for several IEEE conferences. He was awarded the 2004 Douglas R. Colton Medal for Research Excellence in recognition of excellence in research leading to new understanding and novel developments in Microsystems in Canada, and the 2002 International Low-Power Design Contest.