

COOLER- A Fast Multiobjective Fixed-outline Thermal Floorplanner

Debarshi Chatterjee
Stanford University
debarshi@stanford.edu

Theodore W. Manikas
University of Tulsa
theodore-manikas@utulsa.edu

Igor Markov
University of Michigan
imarkov@eecs.umich.edu

Abstract

In this work, we study multiobjective thermal-aware floorplanning in the fixed-outline context. Our baseline implementation demonstrates a 14% average interconnect improvement over Parquet for ami49, and can additionally optimize peak on-chip temperature. To circumvent the expense of Compact Thermal Models, we develop a novel approach to power-density aware floorplanning, but rigorously evaluate final layouts using a standard methodology based on the Hotspot tool. With these techniques, our multiobjective floorplanner COOLER, reduces peak temperature on MCNC circuits by 15.3°C, which is similar to the 15.1°C reduction achieved by Hotfloorplan. However, our tool is 11-146 times faster than Hotfloorplan. We also show that solutions found by Hotfloorplan lie on the Pareto front generated by COOLER.

1. Introduction

Automated floorplanning performs placement of on-chip blocks to facilitate early estimation, as well as to improve circuit performance and yield. These blocks typically include circuit modules generated by partitioning, as well as embedded memories, IP blocks, and analog devices. The Classical Floorplanning Problem (CFP) seeks to optimize area and interconnect, but has recently undergone several paradigm shifts due to major changes in micro-architecture design flows. The increasing scale and complexity of Integrated Circuits, as well as time-to-market pressures in commercial IC design, necessitate a top-down hierarchical approach [1]. Such an approach can be enabled by fixed-outline floorplanning of circuit modules if module outlines are viewed as outlines of nested floorplans. However, in computational terms, fixed-outline floorplanning is considerably harder than the corresponding CFP [2]. Another major shift in floorplan design technology is the introduction of maximum on-chip temperature as a key optimization objective. This has become crucial in view of the increasing power dissipation in portable and stationary electronics, due to considerable leakage

current at the 65nm technology node and below, as well as increasing clock frequencies. If this trend continues unchecked, power density may reach 10,000 W/cm² by the year 2015 [3].

Operating temperature may increase due to greater local power density, triggering several harmful effects, such as electromigration and exponential increase in leakage, which decreases mean time-to-failure and increases the probability of thermal runaways. Additionally, the dependence of carrier mobility and interconnect resistivity on temperature causes the current-driving capability of transistors to decrease and the interconnect delay to increase as chips get hotter. As maximum on-chip temperature (T) increases, burn-in tests used to weed out devices with latent defects are becoming more and more time-consuming [4]. This is because the difference between maximum burn-in temperature (limited by thermal runaway) and maximum on-chip temperature is decreasing steadily, making it difficult to simulate accelerated use. Thermal optimization is particularly important for future ICs with memory stacked on top of random logic and for true-3D ICs, since in both cases air cooling is less effective. As the peak on-chip temperature increases, the costs of packaging and cooling grow at the rate of \$1/W above 40W [5]. Without proper design methodology, this burgeoning overhead alone may hamper future device scaling.

In light of thermal challenges, we consider the following Modern Floorplanning Problem (MFP): Given a set of blocks $\Pi = \{B_1, B_2, \dots, B_N\}$ with fixed areas, a set of nets $\mathbf{h} = \{N_1, N_2, \dots, N_m\}$ specifying the block interconnections and average power dissipation values $P = \{P_1, P_2, \dots, P_N\}$ of the modules, find the set of non-dominated W+T optimized packings of Π that fit in a fixed outline specified by $W_{outline} \times H_{outline}$. A non-dominated set (also termed a Pareto-optimal set) by definition excludes solutions whose all parameters are either inferior or identical to those of another solution.

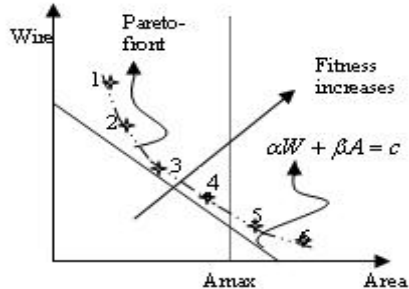


Figure 1. Dependency of the final optimal solution in a conventional floorplanner on the slope of the fitness line.

To compare MFP and CFP, recall that CFP is typically solved by minimizing the objective function:

$$f_{CFP} = \mathbf{a} * \text{Area} + \mathbf{b} * \text{Wirelength} \quad (1)$$

where \mathbf{a} and \mathbf{b} are weights determining the relative importance of the objectives. Graphically this means that the constant-fitness line (with slope $m = -\mathbf{a} / \mathbf{b}$) is shifted towards the origin as far as possible to locate an optimal solution, as illustrated in Figure 1. Now consider the problem in which the designer wants to optimize wirelength subject to the constraint $A \leq A_{\max}$.

From the Pareto-optimal set in Figure 1, solution point 4 can be quickly identified as optimal. However, the conventional, single-objective approach would require a sweep through many m values to find this solution. A large $|m|$ ensures that the optimal solution will satisfy the area constraint, but will generate a poor wirelength solution (example: points 1, 2, and 3). However, a fitness line with low $|m|$ may not satisfy the area constraint, as illustrated by points 5 and 6. In practice, the number of Pareto-optimal solution points close to $A = A_{\max}$ may be so large that even dynamically updating the weights in discrete steps might not reliably find the required solution. MFP inherits all difficulties of the CFP, in the sense that it optimizes wirelength and temperature subject to the fixed-outline constraint, but is much harder because the two optimizations are performed simultaneously.

The naïve approach to mapping out the Pareto front by varying the weights in Eqn. (1) is inefficient as it requires multiple starts of the single-objective annealer. Genetic Algorithms (GA) promise better efficiency in multiobjective optimization by tracking multiple solutions with varying fitness values, and can also employ annealing-style moves in the form of mutations. Therefore we develop such a genetic

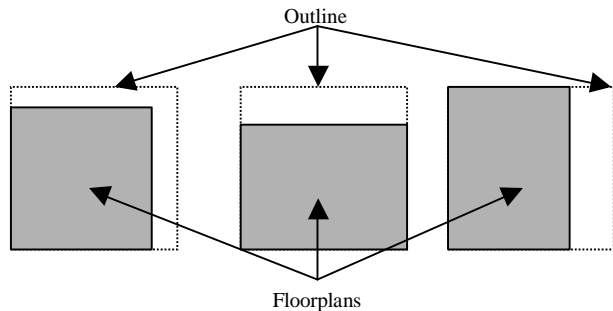


Figure 2. Possible variations in floorplan aspect ratio while satisfying the outline constraint.

algorithm and empirically compare it with a popular single-objective floorplanner based on simulated annealing (SA).

The rest of the paper is organized as follows, Section 2 covers necessary background and related work. Section 3 introduces our multiobjective floorplanning and fast thermal optimization technique for the MFP. Section 4 discusses empirical validation, and Section 5 concludes our work.

2. Background and previous work

Modern floorplanning has generated much interest among researchers since the year 2000, and several sophisticated methods have been proposed to deal with the manifold complexities of the MFP.

2.1. Previous work

Previous techniques for fixed-outline floorplanning rely on single-objective optimization. In [2], the fixed-outline constraint was achieved by invoking a specially designed subroutine from time to time in course of floorplanning. The subroutine compared the aspect ratio of the floorplan with that of the prescribed outline and initiated slack-based moves to repair aspect ratio distortions. In [6] fixed-outline floorplanning was achieved by introducing an aspect-ratio difference term in the fitness function and dynamically updating the weights. [2, 6] assume that the aspect ratios of the outline and the optimal floorplan would be the same. Figure 2 shows that the floorplan aspect ratio can be significantly different from outline aspect ratio and yet satisfy the fixed-outline constraint. This flexibility in aspect ratio can be exploited for better interconnect optimization. An evolutionary technique for fixed-outline area-packing was proposed in [7] but it does not consider interconnect optimization.

The need for thermal-aware microarchitecture design has been established in [8]. The computational cost of embedding a highly accurate temperature simulator (using FEM, FDM or Green’s function method) within a floorplanner can be prohibitively high. Thus Compact Thermal Model (CTM) is used for temperature simulations during floorplanning. Tsai *et al.* [9] proposed a CTM that can be used to estimate temperature profile from a distribution of power dissipating sources on chip. Thermal floorplanners [10] and [11] use Hotspot within SA and GA procedures respectively, but are very slow, especially on large circuits, as CTM-based temperature simulation dominates runtime. In contrast, the runtime of CFP is dominated by interconnect evaluation [12]. Alternative approaches to thermal optimization based on power analysis, including the matrix synthesis approach in [13], have failed to reduce maximum on-chip temperature. Skadron *et al.* [8] found very small correlation between the total power dissipation in a block and its temperature.

To this end, our study reveals a high correlation between the steady-state temperature at the center of a block and the gradient of power density reduction computed at its periphery. We thus introduce power-density aware thermal floorplanning and study its role in reducing the maximum on-chip temperature as well as improving the runtime of existing thermal-aware floorplanners.

2.2. Sequence pair floorplan representation

Over the years a variety of floorplan data-structures such as Slicing Tree, Sequence Pair (SP), Bounded Slicing Grid, O-tree and B* tree have been proposed. Our implementation is based on the Sequence Pair (SP) [14] floorplan representation, since this has been used by previous fixed-outline floorplanners [2]. In SP representation, a pair of permutation of $\{1,2,\dots,N\}$ is used to encode the floorplan. Depending on the relative order of elements in these pairs, certain topological constraints are imposed. For example:

$$(\dots, B_i, \dots, B_j, \dots) \& (\dots, B_i, \dots, B_j, \dots) \Rightarrow B_j \text{ is right of } B_i \quad (2)$$

$$(\dots, B_i, \dots, B_j, \dots) \& (\dots, B_j, \dots, B_i, \dots) \Rightarrow B_j \text{ is below } B_i \quad (3)$$

Given a sequence pair, block locations can be found in $O(n^2)$ time, and even faster in $O(n(\log n))$ or $O(n(\log(\log n)))$ time, using more recent algorithms. However, it has been shown in [2] that for $n < 100$, the $O(n^2)$ time algorithm by Tang *et al.* [15] outperforms more sophisticated algorithms. Therefore we adopted this algorithm in our work.

2.3. Hotspot

Hotspot is a temperature simulator. The thermal model in Hotspot is based on the well-known duality between electric and thermal quantities. The average power dissipation within a block is modeled by a current source and the lateral heat flow from a block is modeled by thermal resistors connected to another block or the ambient. The values of resistors and capacitors are calculated from the relative positioning of the blocks, assuming typical values for sink dimension, conductivity etc. For steady-state analysis, the capacitors are open-circuited. The temperature at the center of each block is then computed by solving the nodal equations of the equivalent RC circuit.

3. Multiobjective optimization

To solve the MFP via multiobjective optimization, we consider an objective function which is a 3D vector of the form:

$$F = \langle f_1, f_2, f_3 \rangle \quad (4)$$

Where $f_1 = \Phi$ and $f_2 = \Theta$ are the wirelength objective and thermal objective respectively. The function f_3 is either the outline objective (Ω) or the area objective (A), depending on the mode of operation of the floorplanner. The following definitions are useful in understanding the multiobjective approach:

DEFINITION 1: A floorplan **a** is said to dominate a floorplan **b** (also written as $a \mathbf{f} b$) if and only if all elements in the objective vector of **a** are greater than or equal to the corresponding elements in the objective vector of **b** and at least one element in the objective vector of **a** is strictly greater than the corresponding element in the objective vector of **b**. Mathematically, we say $a \mathbf{f} b$ iff:

$$\begin{aligned} f_i(a) &\geq f_i(b) \quad i=1,2,3 \\ \wedge \exists j \in \{1,2,3\} : f_j(a) &> f_j(b) \end{aligned} \quad (5)$$

DEFINITION 2: A floorplan **a** is said to cover a floorplan **b** if and only if **a** dominates **b** or their objective vectors are identical. Mathematically, we say a covers b iff:

$$a \mathbf{f} b \text{ or } F(a) = F(b) \quad (6)$$

DEFINITION 3: A floorplan **a** is called Pareto-optimal if and only if **a** is not dominated by any other floorplan. The set of all non-dominated floorplans constitutes the so-called Pareto-optimal set.

3.1. Genetic algorithm

Our multiobjective solver implements a Genetic Algorithm (GA). Figure 3 shows the basic flowchart of the multiobjective GA. In a Simple Genetic Algorithm, there is a single evolving population. To generate the Pareto-optimal solution set, an external non-dominated population has to be maintained. At each step of the GA, non-dominated members from the evolving population P are stored in the external population P' . However, members which are non-dominated in P may be *covered* or might *cover* previously stored members in P' . Thus, each time members are copied to the external population, the covered members have to be removed to ensure Pareto-optimality of the external population. Parent selection is done from the union set of the external and evolving population using the Binary Tournament Selection scheme. Next Partially Mapped Crossover and Swap Mutation are applied to the selected parents and a new population is generated. The fitness assignment has been done according to the well-known Strength Pareto Evolutionary Approach (SPEA) that ensures that members in the external population have higher probability of being selected in the mating pool.

The procedure for fitness assignment in SPEA is a two-step process. At first, each member in the external population is assigned a fitness value (strength) given by:

$$f_i = \frac{n}{N+1} \quad (7)$$

where n denotes the number of individuals in P that are covered by i and N is the size of P . Once the fitness has been assigned to the members in the external population, fitness is assigned to each member in the evolving population according to the following equation:

$$f_j = 1 + \sum_{i, i \neq j} f_i \quad (8)$$

3.2. Fixed-outline mode

For fixed-outline floorplanning, the function f_3 in the objective vector is driven by the outline-objective (Ω) defined below:

$$f_3 = \Omega = \exp[-(V_V + V_H)] \quad (9)$$

$$V_V = (H_{chip} - H_{outline})U(H_{chip} - H_{outline}) \quad (10)$$

$$V_H = (W_{chip} - W_{outline})U(W_{chip} - W_{outline}) \quad (11)$$

where V_V and V_H are measures of the violation of the outline constraint in the vertical and horizontal

directions respectively and $U(x)$ is the step function defined as: $U(x)=1$ when $x > 0$ and $U(x)=0$ when $x \leq 0$. If the maximum allowable percentage whitespace \mathbf{w} and outline aspect ratio \mathbf{r} are provided, then the width and height of the outline can be easily computed from:

$$W_{outline} = [(1 + \mathbf{w}/100)A / \mathbf{r}]^{1/2} \quad (12)$$

$$H_{outline} = [(1 + \mathbf{w}/100)A\mathbf{r}]^{1/2} \quad (13)$$

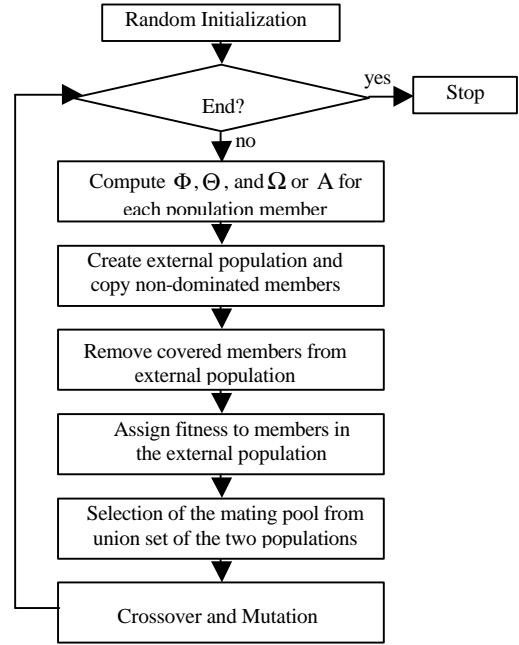


Figure 3. Multiobjective genetic algorithm.

Theorem 1: $\Omega=1$ is a necessary and sufficient condition for satisfying the fixed outline constraint.

Proof: $W_{chip} \leq W_{outline}$ and $H_{chip} \leq H_{outline}$

$$\Leftrightarrow (W_{chip} - W_{outline})U(W_{chip} - W_{outline}) = 0 \text{ and}$$

$$(H_{chip} - H_{outline})U(H_{chip} - H_{outline}) = 0 \Leftrightarrow \Omega = 1$$

Note that $\Omega=1$ does not necessitate that the floorplan and outline aspect ratios be identical. The fixed-outline floorplanner works in two phases. In the first phase, the external non-dominated population is not pruned. During this phase the GA explores the solution space uniformly. In the second phase, a selective pruning mechanism is applied on the external population to weed out solutions with low Ω , whenever the size of the external population increases a preset value (~ 5).

3.3. Outline-free mode

For outline-free floorplanning, the function f_3 in the objective vector is driven by the area-objective (A) defined below:

$$f_3 = A = \sum_{i=1}^N A(B_i) / A_{chip} \quad (14)$$

where $A(B_i)$ is the area of block i and A_{chip} is the area of the chip represented by the sequence pair.

3.4. Wirelength optimization

The half-perimeter of the bounding box of a net serves as a good approximation for the total wirelength in that net. The Half Perimeter Wire Length (HPWL) is used to define the wirelength-objective (Φ) as follows:

$$\Phi = \sum_{i \in h} [(x_{\max}^i - x_{\min}^i) + (y_{\max}^i - y_{\min}^i)] \quad (15)$$

Where (x_{\max}^i, y_{\max}^i) and (x_{\min}^i, y_{\min}^i) denote the coordinates of the top-right and bottom-left corners of the bounding box of net i .

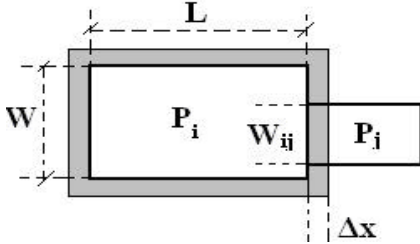


Figure 4. Imaginary rectangular boundary surrounding a floorplan module for computing gradient of power-density reduction.

3.5. Thermal objectives and optimization

In this section, we introduce power-density aware floorplanning for thermal optimization. Our intention is to model maximum on-chip temperature (T_{\max}) using a power-density based metric. Since information about the block power-density is readily available, such an approach is computationally efficient. To understand the relation between power-density and temperature, consider the equation for steady-state temperature in a 3D substrate:

$$k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) + Q(x, y, z) = 0 \quad (16)$$

Subject to the boundary condition:

$$k \frac{\partial T}{\partial n_i} + h_i T = f_i(x, y, z) \quad (17)$$

where T is the temperature as a function of position; k and ρ are thermal conductivity ($\text{W/m}^\circ\text{C}$) and density (Kg/m^3) of the material respectively, h_i is the heat transfer coefficient of the packaging components ($\text{W/m}^2\text{C}$), Q is the power-density (W/m^3), $\partial / \partial n_i$ represents the differentiation along the outward normal drawn at the boundary surface, and f_i is an arbitrary function.

If the power-density Q is uniformly distributed across the spatial coordinates and the initial and boundary conditions are identical for all points, then from symmetry of the differential terms in (16) it can be concluded that the temperature distribution will also be uniform. The assumption of uniform initial and boundary condition is justified because any thermal-aware methodology, including CTMs must be Boundary and Initial Condition Independent [8]. Because power densities are localized and it is not possible to make any changes to the circuitry within a block, perfectly uniform power density or temperature distribution cannot be achieved by redistributing the blocks. However, it is possible to identify modules with high power density as potential candidates for developing hotspots and surround them by whitespace or modules with low power density to cool down potential hotspots. In order to carry out a mathematical analysis of the problem, let us define the following terms:

H : Ordered set of high power-density modules (modules having power densities greater than the 80 percentile value) sorted in descending order.

I_i : Set of all blocks adjacent to block $i \forall i \in H$

W_{ij} : Shared boundary-length between $i \in H$ and $j \in I_i$

P_i : Surface power density of block i

W_i, L_i : Width and length of module i respectively.

Let us construct an imaginary rectangle of length $L + 2\Delta x$ and width $W + 2\Delta x$ around block i as shown in Figure 4. The effective power density within the imaginary rectangle is given by:

$$P_{eff} = \frac{P_i W_i L_i + \sum_{j \in I_i} W_{ij} \Delta x P_j + O(\Delta x^2)}{(W_i + 2\Delta x)(L_i + 2\Delta x)} \quad (18)$$

The rate at which power density diminishes with Δx can be calculated from (18) and is given by:

$$\frac{\Delta P}{\Delta x} = \frac{P_i - P_{eff}}{\Delta x} = \frac{1}{W_i L_i} \sum_{j \in I_i} (P_i - P_j) W_{ij} \quad (19)$$

```

begin
Input: Sequence Pair, Module Dimensions, Power Density
Output:  $\Theta$ 
1:  $H \leftarrow$  vector of high power-density blocks in descending order
2: for all  $i \in H$ 
3:    $I_i \leftarrow$  set of blocks adjacent to  $i$ ,  $Scale \leftarrow 1$ 
4:   for all  $j \in I_i$ 
5:      $\Theta \leftarrow \Theta + (P_i - P_j)(W_{ij} / W_i L_i * Scale)$ 
6:    $Scale \leftarrow Scale * S$ 
End

```

Figure 5. Algorithm for calculating the thermal objective.

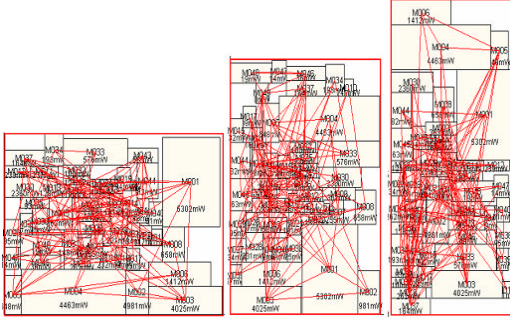


Figure 6. Fixed-outline W+T optimized layouts of ami49 with $w = 0.20$ and $r = 1,2,3$.

The idea is to determine a layout which minimizes the effective power densities around modules with high power density. This is equivalent to maximizing the gradient of power-density reduction $\forall i \in H$. In order to reduce T_{max} , $\Delta P / \Delta x$ for the first element in H (i.e. the highest power-density block) must be maximized. Next, $\Delta P / \Delta x$ for the second element in H is maximized keeping the relative location of the highest power-density block and its neighbours unaltered. Thus, it will not be correct to develop a scalar thermal-objective simply by summing up $\Delta P / \Delta x$ from different blocks in H . This is because, the block with second highest power density may have a much greater perimeter, in which case maximizing $\sum \Delta P / \Delta x$ would result in surrounding it with blocks of low power density. This will neglect the block with highest power density and hence would fail to reduce T_{max} . To overcome this problem, we define the thermal objective Θ as follows:

$$\Theta = \sum_{i \in H} \left(\frac{1}{S^i} \right) \left(\frac{1}{W_i L_i} \right) \sum_{j \in I_i} (P_i - P_j) W_{ij} \quad (20)$$

Table 1. Correlation coefficients between thermal objective Θ and maximum and mean on-chip temperature.

BENCHMARKS	Θ VS T_{MAX}	Θ VS T_{MEAN}
apte	0.94	0.67
xerox	0.92	0.91
hp	0.96	0.87
ami33	0.83	0.71
ami49	0.85	0.89

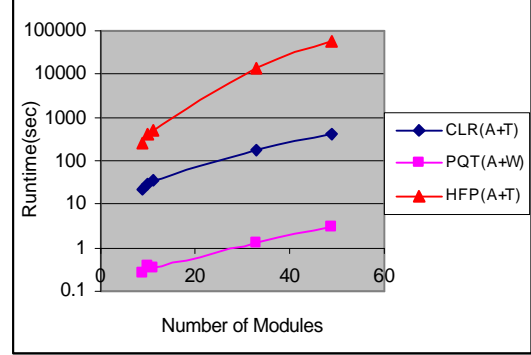


Figure 7. Comparison of runtimes for A+T optimization using Hotfloorplan (HFP), A+T optimization using COOLER (CLR) and A+W optimization using Parquet (PQT).

where S is the scaling factor (~ 10). The entire procedure is summarized in Figure 5. From experimental results we find that Θ as defined in (20) has high degree of correlation with maximum and mean temperature across the chip. The correlation coefficients are given in Table 1.

4. Simulation Results

All results were computed on an Intel Pentium IV laptop running at 1.8 GHz with a 1GB RAM. In the absence of details on power dissipation for MCNC benchmarks, power densities were randomly assigned in the range from $1mW/mm^2$ to $1W/mm^2$. At first we run Parquet in fixed-outline mode on the largest MCNC benchmark (ami49). We also run COOLER under the same conditions and compare the HPWL for different aspect ratios and percentage whitespace in Table 2. Block rotations in Parquet were disabled to make fair comparison with COOLER. It is observed that our approach improves the average HPWL by 14.39%. Figure 6 shows fixed outline W+T optimized layouts of ami49 with different outline aspect ratios.

Table 2. Comparison of average HPWL results (over 10 independent runs) produced by Parquet and COOLER for fixed-outline placements of ami49 without block rotation

maxWS (w)	HPWL (mm)								
	r = 1			r = 2			r = 3		
	Parquet	COOLER	%diff	Parquet	COOLER	%diff	Parquet	COOLER	%diff
15	1020.79	1019.11	0.16	993.13	1014.44	-2.1	1023.34	1007.39	1.58
20	1013.05	862.06	17.52	1041.21	905.51	14.98	1088.36	870.49	25.03
25	973.30	820.246	18.65	1011.6	875.99	15.48	1133.46	955.59	18.61
30	1016.12	777.62	30.67	973.60	913.97	6.52	976.47	1023.53	-4.59
35	1004.88	985.31	1.98	1024.43	845.71	21.13	1140.37	791.04	44.16
40	998.11	943.03	5.84	1085.72	861.56	26.01	1045.14	889.17	17.54

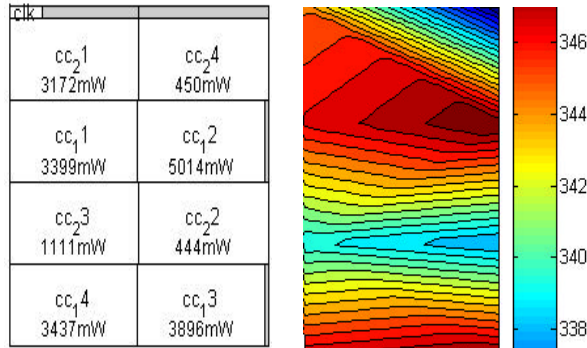


Figure 8. (a) Thermal-aware layout of apte generated by COOLER and (b) spatial distribution of temperature for the same layout.

Table 3. Percentage whitespace, maximum and mean on-chip temperatures and runtime for A+T optimization using Hotfloorplan

MCNC Benchmarks	Percentage Whitespace	T _{max} (K)	T _{mean} (K)	Runtime (sec)
apte	3.27	348.90	343.91	257
xerox	7.83	337.5	329.01	399
hp	13.14	349.6	348.01	501
ami33	44.89	327.5	327.02	13526
ami49	17.43	341.8	336.21	58838

For thermal-aware floorplanning, we use Hotfloorplan [10] to obtain area and temperature (A+T) optimized layouts for MCNC benchmarks. The percentage whitespace, mean and maximum temperatures of the layouts are given in Table 3. Next, we use Parquet for obtaining area and wirelength (A+W) optimized layouts for the same circuits. The runtime for these two cases are compared in Figure 7. It is found that for ami33 and ami49, A+T optimization using Hotfloorplan is around 10,000-20,000 times longer than A+W optimization using Parquet. This trend may become even more pronounced for larger circuits. We thus conclude that temperature simulation using CTM can become a performance bottleneck for placing large benchmarks. Results in Table 3 show clearly a cubic growth in runtime as the number of modules increase.

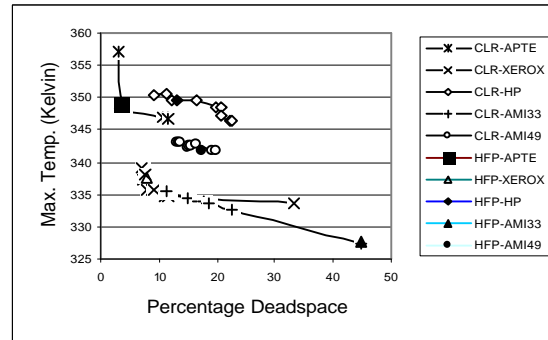


Figure 9. Pareto-optimal solutions generated by COOLER (CLR) and solutions produced by Hotfloorplan (HFP) for MCNC benchmarks.

It is also observed that reduction in peak temperature for ami33 and ami49 is accompanied by considerable increase in whitespace, which may not fit design specifications. On the other hand for apte and xerox, it is possible to bring down the temperature further by relaxing the area constraint. In short, a single run of Hotfloorplan does not give us a clear idea of the trade-off surface or the Pareto-optimal solutions at our disposal. Next we used COOLER (in outline-free mode) to generate thermal-aware layouts for all MCNC benchmarks. One such layout for apte is shown in Figure 8(a). It was found that the runtime for A+T optimization using COOLER varies from 23 sec for apte to 401 sec for ami49. Thus our method is around 11-146 times faster than Hotfloorplan for MCNC benchmarks. Figure 7 compares runtimes for COOLER and Hotfloorplan. This clearly demonstrates the superiority of our method over CTM-based floorplanners, especially when the number of modules is large.

We use Hotspot [16] to analyze the spatial temperature distribution at the block level for the layouts produced by COOLER. The temperature distribution for the apte layout in Figure 8(a) is plotted in Figure 8(b). For each MCNC benchmark, COOLER produces a Pareto-optimal solution set. T_{max} for each

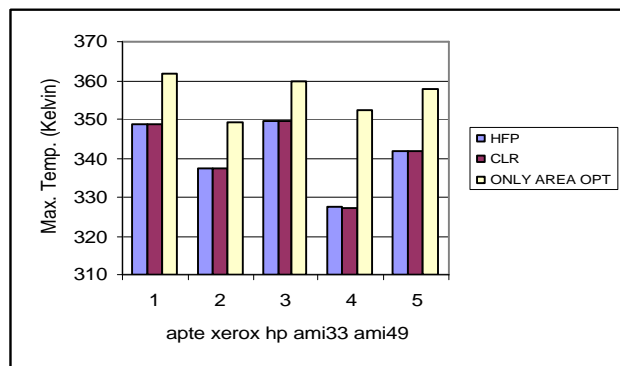


Figure 10. Reduction in T_{max} by COOLER (CLR) and Hotfloorplan (HFP) as compared to non-thermal floorplanner.

layout in the Pareto-optimal set is extracted using Hotspot and plotted in Figure 9. The maximum temperature and deadspace percentage corresponding to the layouts generated by Hotfloorplan are also plotted on the same figure. It is observed that the solutions produced by Hotfloorplan lie on the Pareto front generated by COOLER.

We thus conclude that COOLER is as effective in reducing T_{max} as CTM-based temperature-aware floorplanners but runs much faster. Moreover, our method allows the designer to explore various design possibilities without having to modify the objective function in the source code. For example, Figure 9 indicates that T_{max} in apte can be reduced from 357K to 347K by increasing the deadspace by 7%. In short, by using Multiobjective GA as opposed to conventional SA it is possible to get a complete picture of the trade-off surface. Figure 10 shows that COOLER and Hotfloorplan are capable of reducing T_{max} by 15.3°C and 15.1°C on an average over simple area optimized layouts. Finally, COOLER was run in an outline free mode and the Pareto-optimal solution set for A+W+T optimization was generated. Figure 11 shows the Pareto-optimal solutions for ami49 benchmark in the outline free mode.

5. Conclusion

In this paper, we proposed a novel multiobjective approach driven by a new outline objective function for better interconnect optimization in fixed-outline context. We also presented a maximum on-chip temperature reduction technique that improves the runtime of existing thermal floorplanners by several orders of magnitude.

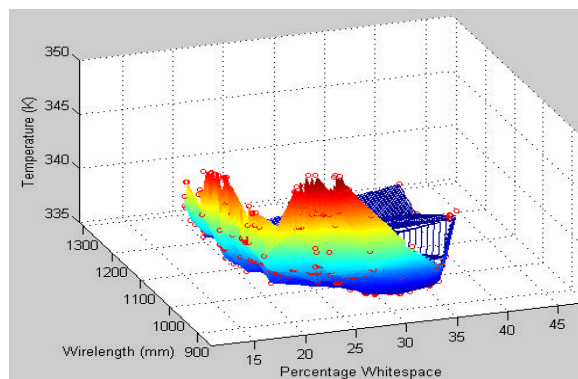


Figure 11. Non-dominated surface showing 382 Pareto-optimal solution points for outline-free A+W+T optimized floorplanning of ami49.

6. References

- [1] A. B. Kahng, "Classical Floorplanning Harmful?", *ISPD*, 2000, 207-213.
- [2] S. N. Adya, and I. L. Markov, "Fixed Outline Floorplanning: Enabling Hierarchical Design", *IEEE Trans. on VLSI*, 11, 6, 2003, 1120-1135.
- [3] <http://www.intel.com/technology/magazine/computing/platform-2015-0305.htm>
- [4] A. Vassighi, O. Semenov, M. Sachdev, A. Keshavarzi, and C. Hawkins, "CMOS IC Technology and its Impact on Burn-in", *IEEE Trans. on device and materials reliability*, 4, 2, 2004, 208-221.
- [5] S. Borkar, "Design Challenges of Technology Scaling", *IEEE Micro*, Jul.-Aug. 1999, 23-29.
- [6] T.-C. Chen, and Y.-W. Chang, "Modern Floorplanning Based on Fast Simulated Annealing", *ISPD*, 2005.
- [7] C. T. Lin, D.-S. Chen, Y. W. Wang, "Robust Fixed-outline Floorplanning through Evolutionary Search", *IEEE/ACM ASPDAC*, 2004, 42 – 44.
- [8] K. Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, "Temperature-Aware Microarchitecture", *30th Intl. Symp. on Comp. Architecture*, June 2003, 2-13.
- [9] C.-H. Tsai, and S.-M. Kang, "Cell-Level Placement for Improving Substrate Thermal Distribution", *IEEE Trans. on CAD of IC and Systems*, 19, 2, Feb. 2000, 253-266.
- [10] K. Sankaranarayanan, S. Velusamy, M. R. Stan, and K. Skadron, "A Case for Thermal-Aware Floorplanning at the Microarchitectural Level", *Journal of Instruction-Level Parallelism*, Sept. 2005.
- [11] W.-L. Hung, Y. Xie, N. Vijaykrishnan, C. Addo-Quaye, T. Theocharides, and M. J. Irwin, "Thermal Aware Floorplanning using Genetic Algorithms", *6th ISQED*, 2005, 634-639.
- [12] H. H. Chan, S. N. Adya, and I. L. Markov, "Are Floorplan Representations Important in Digital Design?" *Intl. Symposium on Physical Design (ISPD)*, San Francisco, April 2005, 129-136.
- [13] C. N. Chu, and D. F. Wong, "A Matrix Synthesis Approach to Thermal Placement", *IEEE Trans. on CAD of IC and Systems*, 17, 11, Nov. 1998, 1166-1174.
- [14] H. Murata, K. Fujiyoshi, S. Nakatake, and Y. Kajitani, "VLSI Module Placement based on Rectangle-Packing by Sequence Pair", *IEEE Transactions on Computer Aided Design*, 15, 12, 1996, 1518-1524.
- [15] X. Tang, D. F. Wong, R. Tian, "Fast Evaluation of Sequence Pair in Block Placement by Longest Common Subsequence Computation", *DATE*, 2000, 106-111.
- [16] <http://lava.cs.virginia.edu/HotSpot/index.htm>